Internship report M2
Parisian Master of Research in Computer Science Level 2

Non-singleton Elimination

Soudant Léo
fourth year student at ENS Paris Saclay

2025

Supervisor
Pierre-Merie Pédrot, chargé de recherche at Inria Rennes-Bretagne-Atlantique,

Abstract

Contents

1 Introduction **2**
1.1 The outline . . . 2

2 Sheaves **2**
2.1 Topoï and sheaves in topoï . . . 2
2.2 Sheaves in type theory . . . 2

3 Models **3**

4 System T and MLTT **3**
4.1 MLTT . . . 3
4.2 System T . . . 4

5 ShTT **5**
5.1 Martin Baillon's ShTT . . . 5
5.2 ShTT . . . 5

6 Logical relations **6**
6.1 System T extension . . . 6

7 Meta-informations **6**
7.1 Time expenditure . . . 6
7.2 Difficulties . . . 6

8 Conclusion **6**

A Dummy **7**

1 Introduction

1.1 The outline

2 Sheaves

2.1 Topoï and sheaves in topoï

Proofs, results and details for this section can often be found in *Sheaves in geometry and logic: A first introduction to topos theory* by Saunders Maclane and Ieke Moerdijk [1].

Definition 2.1 (Subobject)**.** *In a category, a* subobject *of X is an equivalence class of monomorphism $m : A \rightarrowtail X$, where the equivalence comes from the preorder where $m : A \rightarrowtail X$ is smaller than $m' : A' \rightarrowtail X$ when there is a map $f : A \to A'$ with $m' \circ f = m$.*

We deduce a presheaf **Sub** where $\mathbf{Sub}(X)$ is the the set of subobjects of X, and $\mathbf{Sub}(f) : \mathbf{Sub}(Y) \to \mathbf{Sub}(X)$ sends $m : A \to Y$ to its pullback by $f : X \to Y$.

Definition 2.2 (Topos)**.** *A* topos *is a cartesian closed category with all finite limits and a suboject* classifier Ω *and an isomorphism* $\mathbf{Sub}(X) \cong \mathbf{Hom}(X, \Omega)$ *natural in X.*

We note that topoï also have finite colimits.

A topos serves to give models of intuitionistic logic in classical mathematical language. It has an internal logic which is higher order.

For example, **Set** is a topos, and given a topos $\mathcal{E}$, $\mathcal{E}/X$, the category of maps with codomain X and commuting triangles, as well as $\mathcal{E}^{\mathbf{C}^{op}}$, the category of contravariant functors from a small category $\mathbf{C}$ and natural transformation, are all topoï. In particular categories of presheaves are topoï, and correspond to Kripke models.

The subobject classifier Ω is equipped with an internal meet-semilattice structure inherited from the meet-semilattice structure on each $\mathbf{Sub}(X)$, which is natural in X.

Definition 2.3 (Lawvere-Tierney topology)**.** *A* Lawvere-Tierney topology *is a left exact idempotent monad j on the internal meet-semilattice on Ω.*

- $id_\Omega \leq j$,
- $j \circ j \leq j$
- $j \circ \wedge = \wedge \circ j \times j$

From a topology j we extract a closure operator J_X of $\mathbf{Sub}(X)$ for any X.

Definition 2.4 (Dense subobject)**.** *A suboject U of X is dense if $J_X U = X$*

A topology can be lifted to a left exact idempotent monad on the entirety of the topos, the sheafification monad.

Definition 2.5 (j-Sheaf in topos)**.** *An object F is a j-*sheaf *in a topos if for any dense subobject U of any object X, the morphism* $\mathbf{Hom}(X, F) \to \mathbf{Hom}(U, F)$ *obtained by precomposition is an isomorphism.*

A j-Sheaf is up to isomorphism the result of sheafifying an object.

j-Sheaves form a topos. The sheaves on a presheaf topos correspond to Beth semantics.

2.2 Sheaves in type theory

Consider a type theory with a notion of proof irrelevant propositions Prop, *e.g.* book-HoTT with mere propositions, or Rocq with SProp.

In this case, a Lawvere-Tierney topology may be similarily defined, as a monad:

- $\mathsf{J} : \mathsf{Prop} \to \mathsf{Prop}$
- $\eta : \Pi(P : \mathsf{Prop}).P \to \mathsf{J}\ P$
- $\mathsf{bind} : \forall (PQ : \mathsf{Prop}).\mathsf{J}\ P \to (P \to \mathsf{J}\ Q) \to \mathsf{J}\ Q$

Then a sheaf is just a type T with

- A map $\mathsf{ask}_T : \Pi(P : \mathsf{Prop}).\ \mathsf{J}\ P \to (P \to T) \to T$
- A coherence $\varepsilon_T : \Pi(P : \mathsf{Prop})\ (j : \mathsf{J}\ P)\ (x : T).\ \mathsf{ask}_T\ \ P\ j\ (\lambda p : P.x) = x$

Now, the sheafified of a type doesn't exists in general, if the theory admits quotient inductive types, it can then be defined as follow :

$$\begin{aligned}
&\mathsf{Inductive}\ \ \mathcal{S}_\mathsf{J}\ T : \mathsf{Type} := \\
&|\,\mathsf{ret} : T \to \mathcal{S}_\mathsf{J}\ T \\
&|\,\mathsf{ask} : \Pi(P : \mathsf{Prop}), \mathsf{J}\ P \to (P \to \mathcal{S}_\mathsf{J}\ T) \to \mathcal{S}_\mathsf{J}\ T \\
&|\ \varepsilon : \Pi(P : \mathsf{Prop})\ (j : J\ P)\ (x : \mathcal{S}_\mathsf{J}\ T).\,\mathsf{ask}\ \ P\ j\ (\lambda p : P.x) = x
\end{aligned}$$

We note that by taking $I := \Sigma(P : \mathsf{Prop}).\mathsf{J}\ P$ and $O\ (P, j) : P$, a sheaf is then a

- A type T
- A map $\mathsf{ask}_T : \Pi(i : I), (O\ i \to T) \to T$
- A coherence map $\varepsilon_T : \Pi(i : I)\ (x : T), \mathsf{ask}_T\ \ i\ (\lambda o : O\ i.x) = x$

and simimlarily for sheafification. This is marginally simpler, and make sheaves appear as quotient dialogue trees, hence why we will henceforth consider (I, O)-*sheaves* instead of J-*sheaves*.

3 Models

A significant part of my internship was dedicated to contructing models of type theory in Rocq.

1. A model of a variant of Baclofen TT using dialogue trees. Predicates must be linearized before eliminating an inductive into them.
2. An exceptional model, with a type of exceptions E. A special type of dialogue trees where $I = E$ and $Oi = \mathbf{0}$, the resulting theory is inconsistent (when E is inhabited), as always when $Oi \to \mathbf{0}$ for some i.
3. A model using sheaves, which requires univalence, and quotient inductive types to model positive types.
4. A incomplete model using presheaves.

4 System T and MLTT

4.1 MLTT

We sought to extend the following variant of MLTT.

We consider a type of levels containing two elements $\mathbf{s}$ and $\mathbf{l}$ for small and large, with generic ℓ

With terms :

$$M, N ::= x|\lambda x.M|MN|0|S|\mathbf{N}_{\text{rec}}|\bot_{\text{rec}}|\mathbf{N}|\bot|\Pi x : A.B|\Box_\mathbf{s}|\Box_\mathbf{l}$$

Contexts :

$$\Gamma ::= \Gamma, x : A|\cdot$$

And conversion rules :

$$\textsc{Wf-Empty}\,\frac{}{\cdot \vdash \mathsf{well\text{-}formed}} \qquad \textsc{Wf-Ext}\,\frac{\Gamma \vdash A \equiv A \qquad \Gamma \vdash \mathsf{well\text{-}formed}}{\Gamma, x : A \vdash \mathsf{well\text{-}formed}}$$

$$\textsc{Int-Typ}\,\frac{\Gamma \vdash \mathsf{well\text{-}formed}}{\Gamma \vdash \mathbf{N} \equiv \mathbf{N} : \Box_\ell} \qquad \textsc{Emp-Typ}\,\frac{\Gamma \vdash \mathsf{well\text{-}formed}}{\Gamma \vdash \bot \equiv \bot : \Box_\ell}$$

$$\textsc{Fun-Typ}\,\frac{\Gamma \vdash A \equiv A' : \Box_\ell \qquad \Gamma, x : A \vdash B \equiv B' : \Box_\ell}{\Gamma \vdash \Pi x : A, B \equiv \Pi x : A', B' : \Box_\ell} \qquad \textsc{Typ-Typ}\,\frac{\Gamma \vdash \mathsf{well\text{-}formed}}{\Gamma \vdash \Box_\mathbf{s} : \Box_\mathbf{l}}$$

$$\textsc{Fun-Intro}\,\frac{\Gamma, x : A \vdash M \equiv M' : B \qquad \Gamma \vdash A \equiv A : \Box_\ell}{\Gamma \vdash \lambda x.M \equiv \lambda x.M' : \Pi x : A, B} \qquad \textsc{Fun-Elim}\,\frac{\Gamma \vdash M \equiv M' : \Pi x : A, B \qquad \Gamma \vdash N \equiv N' : A}{\Gamma \vdash MN \equiv M'N' : B(N/x)}$$

$$\textsc{Axiom}\,\frac{\Gamma \vdash \mathsf{well\text{-}formed} \qquad x : A \in \Gamma}{\Gamma \vdash x \equiv x : A} \qquad \textsc{Beta}\,\frac{\Gamma, x : A \vdash M \equiv M' : B \qquad \Gamma \vdash N \equiv N' : A}{\Gamma \vdash (\lambda x.M)N \equiv M'(N'/x) : B(N/x)}$$

$$\textsc{Int-Zero}\,\frac{\Gamma \vdash \mathsf{well\text{-}formed}}{\Gamma \vdash 0 \equiv 0 : \mathbf{N}} \qquad \textsc{Int-Succ}\,\frac{\Gamma \vdash \mathsf{well\text{-}formed}}{\Gamma \vdash S \equiv S : \mathbf{N} \to \mathbf{N}}$$

$$\textsc{Int-Rec}\,\frac{\Gamma \vdash \mathsf{well\text{-}formed}}{\Gamma \vdash \mathbf{N}_{\text{rec}} \equiv \mathbf{N}_{\text{rec}} : \Pi A : \mathbf{N} \to \Box_\mathbf{s}, A0 \to (\Pi n : \mathbf{N}, An \to A(Sn)) \to \Pi n : \mathbf{N}, An}$$

$$\textsc{Int-Rec-Zero}\frac{\Gamma \vdash A \equiv A : \mathbf{N} \to square_{\mathbf{s}} \quad \Gamma \vdash N_0 \equiv N_0' \equiv A0 \quad \Gamma \vdash N_S \equiv N_S : \Pi n : \mathbf{N}, An \to A(Sn)}{\Gamma \vdash \mathbf{N}_{\text{rec}} A N_0 N_S 0 \equiv N_0' : A0}$$

$$\textsc{Int-Rec-Succ}\frac{\Gamma \vdash A \equiv A' : \mathbf{N} \to square_{\mathbf{s}} \quad \Gamma \vdash N_0 \equiv N_0' \equiv A0 \quad \Gamma \vdash N_S \equiv N_S' : \Pi n : \mathbf{N}, An \to A(Sn) \quad \Gamma \vdash N \equiv N' : \mathbf{N}}{\Gamma \vdash \mathbf{N}_{\text{rec}} A N_0 N_S (SN) \equiv N_S' N' (\mathbf{N}_{\text{rec}} A' N_0' N_S' N') : A(SN)}$$

$$\textsc{Emp-Rec}\frac{\Gamma \vdash \textsf{well-formed}}{\Gamma \vdash \bot_{\text{rec}} \equiv \bot_{\text{rec}} : \Pi A : \bot \to \Box_{\mathbf{s}}, \Pi e : \bot, Ae}$$

$$\textsc{Sym}\frac{\Gamma \vdash M \equiv M' : A}{\Gamma \vdash M' \equiv M : A} \qquad \textsc{Trans}\frac{\Gamma \vdash M \equiv M' : A \quad \Gamma \vdash M' \equiv M'' : A}{\Gamma \vdash M \equiv M'' : A}$$

$$\textsc{Conv}\frac{\Gamma \vdash M \equiv M' : A \quad \Gamma \vdash A \equiv A' : \Box_\ell}{\Gamma M \equiv M' : A'}$$

But it is useful to consider the extension, with new terms : $M, N ::= \ldots |\mathbf{B}|\mathbf{B}_{\text{rec}}|\overline{\mathsf{tt}}|\overline{\mathsf{ff}}$
And conversion rules

$$\textsc{Bool-True}\frac{\Gamma \vdash \textsf{well-formed}}{\Gamma \vdash \overline{\mathsf{tt}} \equiv \overline{\mathsf{tt}} : \mathbf{B}} \qquad \textsc{Bool-False}\frac{\Gamma \vdash \textsf{well-formed}}{\Gamma \vdash \overline{\mathsf{ff}} \equiv \overline{\mathsf{ff}} : \mathbf{B}}$$

$$\textsc{Bool-Rec}\frac{\Gamma \vdash \textsf{well-formed}}{\Gamma \vdash \mathbf{B}_{\text{rec}} \equiv \mathbf{B}_{\text{rec}} : \Pi A : \mathbf{B} \to \Box_{\mathbf{s}}, A\overline{\mathsf{tt}} \to A\overline{\mathsf{ff}} \to \Pi b : \mathbf{B}, Ab}$$

$$\textsc{Bool-Rec-True}\frac{\Gamma \vdash A \equiv A : \mathbf{B} \to \Box_{\mathbf{s}} \quad \Gamma \vdash M_{\overline{\mathsf{tt}}} \equiv M_{\overline{\mathsf{tt}}}' \equiv A\overline{\mathsf{tt}} \quad \Gamma \vdash M_{\overline{\mathsf{ff}}} \equiv M_{\overline{\mathsf{ff}}} : A\overline{\mathsf{ff}}}{\Gamma \vdash \mathbf{B}_{\text{rec}} A M_{\mathsf{tt}} M_{\mathsf{ff}} \overline{\mathsf{tt}} \equiv M_{\overline{\mathsf{tt}}}' : A\overline{\mathsf{tt}}}$$

$$\textsc{Bool-Rec-False}\frac{\Gamma \vdash A \equiv A : \mathbf{B} \to \Box_{\mathbf{s}} \quad \Gamma \vdash M_{\overline{\mathsf{tt}}} \equiv M_{\overline{\mathsf{tt}}} \equiv A\overline{\mathsf{tt}} \quad \Gamma \vdash M_{\overline{\mathsf{ff}}} \equiv M_{\overline{\mathsf{ff}}}' : A\overline{\mathsf{ff}}}{\Gamma \vdash \mathbf{B}_{\text{rec}} A M_{\overline{\mathsf{tt}}} M_{\overline{\mathsf{ff}}} \overline{\mathsf{ff}} \equiv M_{\overline{\mathsf{ff}}}' : A\overline{\mathsf{ff}}}$$

4.2 System T

To identify and solve problems in a simpler envirronement, we studied a modified System T before, based on the following variant.

With types :

$$A ::= A \to A|\mathbf{N}|\bot$$

Terms :

$$M, N ::= x|\lambda x.M|MN|0|S|\mathbf{N}_{\text{rec}}|\bot_{\text{rec}}$$

Contexts :

$$\Gamma ::= \Gamma, x : A|\cdot$$

And conversion rules :

$$\textsc{Fun-Intro}\frac{\Gamma, x : A \vdash M \equiv M' : B}{\Gamma \vdash \lambda x.M \equiv \lambda x.M' : A \to B} \qquad \textsc{Fun-Elim}\frac{\Gamma \vdash M \equiv M' : A \to B \quad \Gamma \vdash N \equiv N' : A}{\Gamma \vdash MN \equiv M'N' : B}$$

$$\textsc{Axiom}\frac{x : A \in \Gamma}{\Gamma \vdash x \equiv x : A} \qquad \textsc{Beta}\frac{\Gamma, x : A \vdash M \equiv M' : B \quad \Gamma \vdash N \equiv N' : A}{\Gamma \vdash (\lambda x.M)N \equiv M' : B}$$

$$\textsc{Int-Zero}\frac{\Gamma \vdash \textsf{well-formed}}{\Gamma \vdash 0 \equiv 0 : \mathbf{N}} \qquad \textsc{Int-Succ}\frac{\Gamma \vdash \textsf{well-formed}}{\Gamma \vdash S \equiv S : \mathbf{N} \to \mathbf{N}}$$

$$\textsc{Int-Rec}\frac{\Gamma \vdash \textsf{well-formed}}{\Gamma \vdash \mathbf{N}_{\text{rec}} \equiv \mathbf{N}_{\text{rec}} : A \to (\mathbf{N} \to A \to A) \to \mathbf{N} \to A}$$

$$\textsc{Int-Rec-Zero}\frac{\Gamma \vdash N_0 \equiv N_0' \equiv A \quad \Gamma \vdash N_S \equiv N_S : \mathbf{N} \to A \to A}{\Gamma \vdash \mathbf{N}_{\text{rec}} N_0 N_S 0 \equiv N_0' : A}$$

$$\textsc{Int-Rec-Succ}\frac{\Gamma \vdash N_0 \equiv N_0' \equiv A \quad \Gamma \vdash N_S \equiv N_S' : \mathbf{N} \to A \to A \quad \Gamma \vdash N \equiv N' : \mathbf{N}}{\Gamma \vdash \mathbf{N}_{\text{rec}} N_0 N_S (SN) \equiv N_S' N' (\mathbf{N}_{\text{rec}} N_0' N_S' N') : A}$$

$$\text{Emp-Rec}\ \frac{}{\Gamma \vdash \bot_{\text{rec}} \equiv \bot_{\text{rec}} : \bot \to A}$$

$$\text{Sym}\ \frac{\Gamma \vdash M \equiv M' : A}{\Gamma \vdash M' \equiv M : A} \qquad \text{Trans}\ \frac{\Gamma \vdash M \equiv M' : A \quad \Gamma \vdash M' \equiv M'' : A}{\Gamma \vdash M \equiv M'' : A}$$

5 ShTT

5.1 Martin Baillon's ShTT

My work was meant to generalise the work of my supervisor past PhD student, Martin Baillon. He worked on a roughly similar MLTT extended with boolean and a generic function α.

$$M, N ::= \ldots |\alpha$$

With *forcing contexts*, with n and b an integer and boolean respectively

$$\mathcal{L} ::= \mathcal{L}, n \mapsto b | \cdot$$

We write $n \mapsto_{\mathcal{L}} b$ when $n \mapsto b$ appears in $\mathcal{L}$, and $n \not\mapsto_{\mathcal{L}}$ when neither $n \mapsto \mathsf{tt}$ nor $n \mapsto \mathsf{ff}$ do.

And conversion rules :

$$\text{name}\ \frac{\mathcal{L}, \Gamma \vdash ?_0 \quad \cdots \quad \mathcal{L}, \Gamma \vdash ?_n}{\mathcal{L}, \Gamma \vdash ?}$$

whenever the following is a rule of MLTT (with booleans)

$$\text{name}\ \frac{\Gamma \vdash ?_0 \quad \cdots \quad \Gamma \vdash ?_n}{\Gamma \vdash ?}$$

Exception made of Wf-Empty, which becomes

$$\text{Wf-Empty}\ \frac{}{\cdot, \cdot \vdash \mathsf{well\text{-}formed}}$$

The new conversion rules are :

$$\text{Wf-Ext-Forc}\ \frac{\mathcal{L}, \Gamma \vdash \mathsf{well\text{-}formed} \qquad n \not\mapsto_{\mathcal{L}}}{\mathcal{L}, n \mapsto b, \Gamma \vdash \mathsf{well\text{-}formed}}$$

$$\text{Gen}\ \frac{\mathcal{L}, \Gamma \vdash \mathsf{well\text{-}formed}}{\mathcal{L}, \Gamma \vdash \alpha \equiv \alpha : \mathbf{N} \to \mathbf{B}} \qquad \text{Ask}\ \frac{\mathcal{L}, \Gamma \vdash \mathsf{well\text{-}formed} \qquad n \mapsto_{\mathcal{L}} b}{\mathcal{L}, \Gamma \vdash \alpha \overline{n} \equiv \overline{b}}$$

$$\text{Split}\ \frac{\mathcal{L}, n \mapsto \mathsf{tt}, \Gamma \vdash M \equiv M' : A \qquad \mathcal{L}, n \mapsto \mathsf{ff}, \Gamma \vdash M \equiv M' : A \qquad n \not\mapsto_{\mathcal{L}}}{\mathcal{L}, \Gamma \vdash M \equiv M' : A}$$

Entre autre, cette théorie permet de montrer que tout terme de MLTT $\cdot \vdash M \equiv M : (\mathbf{N} \to \mathbf{B}) \to \mathbf{N}$ est continu.

5.2 ShTT

The prototype of theory I have defined and hoped to study is as follows :

We imagine we have a set Ω of *atoms*. We first set a set I with decidable equality, and a (morally finite) set Ai for each $i \in I$ standing for arity, and finally a family $((O_{i,\alpha})_{\alpha \in Ai})_{i \in I}$ of elements of Ω. The term extend those of MLTT as follows :

$$M, N ::= \ldots | \digamma_i (M_\alpha)_{\alpha \in Ai}$$

When instanciating with finite Ai, it would rather be $\digamma_i M_1 \cdots M_n$.

The forcing contexts $\mathcal{L}$ are now subset of Ω, and never cause ill-formation.

The conversion rules also copy those from MLTT by adding a forcing context as in 5.1, without the WF-Empty exception.

The new conversion rules are as follows :

$$\text{Dig-i}\ \frac{\forall \alpha \alpha'. \mathcal{L}, O_{i,\alpha}, O_{i,\alpha'}, \Gamma \vdash M_\alpha \equiv M'_{\alpha'} : A}{\mathcal{L}, \Gamma \vdash \digamma_i (M_\alpha)_\alpha \equiv \digamma_i (M'_\alpha)_\alpha : A}$$

This rule is both the expected congruence rule and a compatibility rule.

$$\text{Ask-i}\ \frac{\forall \alpha \alpha'. \mathcal{L}, O_{i,\alpha}, O_{i,\alpha'}, \Gamma \vdash M_\alpha \equiv M'_{\alpha'} : A}{\mathcal{L}, O_{i,\alpha}, \Gamma \vdash \digamma_i (M_{\alpha'})_{\alpha'} \equiv M'_\alpha : A}$$

$$\text{Dig-Ev}\ \frac{\mathcal{L}, \Gamma \vdash N \equiv N' : A \qquad \forall \alpha \alpha', \mathcal{L}, O_{i,\alpha}, O_{i,\alpha}, \Gamma \vdash M_\alpha \equiv M_{alpha'} : \Pi x : A, B}{\mathcal{L}, \Gamma \vdash (\digamma_i (M_\alpha)_\alpha) N \equiv \digamma (M_\alpha N)_\alpha : B(N/x)}$$

$$\textsc{Int-Rec-Dig}\ \frac{\mathcal{L},\Gamma \vdash A \equiv A : \mathbf{N} \to \square_{\mathbf{s}} \quad \mathcal{L},\Gamma \vdash M_0 \equiv M_0' : \mathbf{N} \quad \mathcal{L},\Gamma \vdash M_S \equiv M_S' : \Pi n : \mathbf{N}, An \to A(Sn) \quad \forall \alpha\alpha', \mathcal{L}, O_{i,\alpha}, O_{i,\alpha}, \Gamma \vdash M_\alpha \equiv M_{\alpha'} : \mathbf{N}}{\mathcal{L},\Gamma \vdash \mathbf{N}_{\mathrm{rec}} A M_0 M_S (\digamma_i (M_\alpha)_\alpha) \equiv \digamma(\mathbf{N}_{\mathrm{rec}} M_0 M_S M_\alpha)_\alpha : A \digamma_i (M_\alpha)_\alpha}$$

$$\textsc{Emp-Rec-Dig}\ \frac{\mathcal{L},\Gamma \vdash A \equiv A : \bot \to \square_{\mathbf{s}} \quad \forall \alpha\alpha', \mathcal{L}, O_{i,\alpha}, O_{i,\alpha}, \Gamma \vdash M_\alpha \equiv M_{\alpha'} : \bot}{\mathcal{L},\Gamma \vdash \bot_{\mathrm{rec}} A (\digamma_i (M_\alpha)_\alpha) \equiv \digamma(\bot_{\mathrm{rec}} M_\alpha)_\alpha : A \digamma_i (M_\alpha)_\alpha}$$

We may encode 5.1 with $I := \mathbf{N} + \mathbf{N}$

6 Logical relations

6.1 System T extension

7 Meta-informations

7.1 Time expenditure

7.2 Difficulties

The subject is rather vast and a bit unclear.

Doing logical relations for MLTT is a large task, even before adding sheaves. My next attempt will probably start from logrel rocq insted of nothing

8 Conclusion

Appendix

A Dummy

References

[1] Saunders MacLane and Ieke Moerdijk. *Sheaves in geometry and logic: A first introduction to topos theory*. Springer Science & Business Media, 2012.