Université Denis Diderot-Paris 7

# Le calcul dans les topos relatifs

# et

# l'interprétation du langage $W$

Méven CADET

Mémoire de Master 2$^{\text{ème}}$ année
Logique Mathématique et Fondements de l'Informatique

réalisé sous la direction d'Alain Prouté
Équipe de Logique Mathématique CNRS UMR 7056

*Septembre 2010*

# Remerciements

Parmi ceux qui m'ont permis de mener à terme ce mémoire, j'adresse des remerciements particuliers à Alain Prouté qui en a idéalement assuré la direction via un investissement et une disponibilité sans borne. Ses conseils avisés ont conduit ma progression et alimenté mon intérêt pour un domaine que je souhaitais depuis longtemps découvrir.

Je remercie également Arnaud Durand et René Guitart qui ont accepté de faire partie du jury de ce mémoire.

Je tiens d'autre part à remercier l'Equipe de Logique dans son ensemble pour l'accueil généreux qu'elle réserve aux étudiants, notamment ceux dont les mathématiques ne constituent pourtant pas le cursus principal.

Enfin, je me dois de signaler que ce travail ne s'est pas fait sans dommages collatéraux. Je tiens donc à m'excuser auprès des innocents impliqués dans la bataille (et ainsi leur témoigner ma gratitude) : Julien Ross, pour avoir courageusement soigné mon incapacité chronique à utiliser LaTeX, et plus généralement mon proche entourage qui a su modérer mes impatiences lors des longues heures passées à faucher les brousses du calcul.

# Introduction

La théorie des topos nous vient de la géométrie algébrique, précisément des travaux d'A. Grothendieck sur les faisceaux vers 1957. W. Lawvere et M. Tierney généralisent la définition des topos vers 1970, créant ainsi la théorie des topos élémentaires, avec la constatation qu'un topos est un univers mathématique à part entière. Cette discipline s'incorpora alors légitimement à la logique mathématique, induisant de nombreux travaux qui en font aujourd'hui, sur le plan des fondations, la principale concurrente de la théorie des ensembles. Un topos élémentaire est défini comme une catégorie possédant un objet final (un singleton pour le topos des ensembles), des produits (cartésiens), des produits fibrés (des ensembles de la forme $\{(x, y) \in X \times Y \mid f(x) = g(y)\}$) et une notion qui généralise celle d'ensemble des parties. On demande en général aussi un objet des entiers naturels. Par ailleurs, le langage de Bénabou-Mitchell (dit « langage interne »), d'aspect ensembliste, est interprétable dans tous les topos, aussi différents soient-ils. Ceci révèle la proximité du comportement *formel* de chaque topos avec celui des ensembles.

Cette théorie s'écarte de celle des ensembles en deux endroits. D'une part, les raisonnements réalisés dans le langage interne ne sont valides dans tous les topos que s'ils sont démontrés intuitionnistiquement. Par exemple, pour la catégorie des graphes, qui est un topos, le principe du tiers exclu n'est pas valide, ce dont on peut se douter dès qu'on s'aperçoit que le graphe singleton (objet final de la catégorie des graphes), qui a une seule arête et un seul sommet a trois sous-graphes, alors qu'un singleton ensembliste n'a que deux sous-ensembles. La logique intuitionniste a été introduite par J. Brouwer au cours du siècle dernier. Le mathématicien mêlait à ses travaux des recherches d'ordre philosophique qui rebutèrent bon nombre de ses contemporains. Néanmoins, les preuves intuitionnistes possèdent la qualité d'être toutes constructives, fournissant systématiquement une méthode explicite de construction des objets dont elles démontrent l'existence. En contrepartie, bien sûr, certains énoncés classiquement prouvables ne sont pas des théorèmes intuitionnistes. Il n'est donc pas toujours souhaitable d'évoluer dans ce cadre, cela dépend de nos objectifs. Notons seulement que la théorie des topos offre une sémantique particulièrement riche à cette logique, et qu'il est par ailleurs possible d'y raisonner classiquement en ajoutant, par exemple, l'axiome du choix, et en se restreignant aux topos pour lesquels il est valide.

D'autre part, contrairement à la théorie des ensembles, la théorie des topos induit naturellement une notion de type. Tous les ensembles ne sont donc plus comparables comme en théorie des ensembles. De plus, ce typage se révèle extrêmement proche de l'intuition fournie par la pratique mathématique. Par ce biais, il indique de façon extrêmement efficace quel rôle est censé jouer tel objet dans telle preuve. Ainsi, d'un point de vue pragmatique (sans introduire ici aucune considération d'ordre ontologique ou épistémique), cette qualité rend la théorie des topos tout à fait exploitable informatiquement, et extrêmement fertile pour l'assistanat de preuves. Par exemple, l'interprétation d'une matrice et celle d'un polynôme ne seront jamais identiques, car ces objets mathématiques sont évidemment de types différents. C'est pourtant parfois le cas en théorie des ensembles.

Ainsi, la théorie des topos offre une fondation alternative des mathématiques et possède certains avantages sur sa concurrente. Pour cette raison notamment (cette théorie débordant largement du cadre des fondements, il en existe d'autres), elle a fait et fait encore l'objet de nombreuses recherches. Son développement permit l'émergence de plusieurs théorèmes importants parmi lesquels nous distinguerons celui de

P. Freyd (1972 [3])([1]) qui montre que toute catégorie relative $\mathcal{T}/A$ d'un topos $\mathcal{T}$ est aussi un topos. Le prime objet de ce travail s'appuie sur ce théorème et consiste à traduire les opérations du topos relatif $\mathcal{T}/A$ en opérations du topos originel $\mathcal{T}$. En effet, le premier est entièrement construit à partir du second, ses opérations, par exemple la construction de l'objet des parties ou encore la curryfication, sont donc entièrement exprimables dans celui-ci. Cette remarque s'étend à deux foncteurs agissant sur des topos relatifs. L'exercice possède un intérêt propre, celui d'expliciter en quoi consiste réellement un topos relatif, la façon dont il opère et les objets qui le constituent. De plus, il nous permettra, et ce sera notre second travail, d'interpréter le langage $W$, introduit par A. Prouté dans [4].

Le langage $W$ s'interprète aussi dans un topos $\mathcal{T}$ quelconque. Comme le langage interne, il introduit le symbole $\in$, manifestant la proximité du comportement formel de tout topos avec celui du topos des ensembles, et une notion naturelle de type qui ne contraste en rien avec celle qui est sous-jacente aux mathématiques usuelles.

En outre, sur le plan de la traduction du vernaculaire mathématique, il remédie aux principaux défauts du langage interne.([2]) En effet, il autorise la *dépendance*, c'est-à-dire que le type d'une expression peut dépendre de la valeur d'une variable. Les différentes déclarations de variables, qui constituent le contexte, ne commutent plus entre elles, car si la valeur du type de $y$ dépend de $x$, nous ne pouvons plus le déclarer avant d'avoir déclaré $x$ lui-même. Cette caractéristique lui permet de représenter les preuves, ce dont le langage interne est incapable, sauf bien sûr au prix d'une simulation nécessairement non naturelle.

En effet, un énoncé $E$ dans un contexte $\Gamma$ s'interprète comme une flèche $\Gamma \to \Omega$.([3]) Si $\top$ représente le sous-ensemble « singleton vrai » de $\Omega$, on peut considérer son pullback $j$ le long de $E$ :

$$\begin{tikzcd} \bullet \arrow[r] \arrow[d, "j"'] & \mathbf{1} \arrow[d, "\top"] \\ \Gamma \arrow[r, "E"'] & \Omega \end{tikzcd}$$

Ensemblistement, il existe une section de ce pullback si et seulement si celui-ci est surjectif, c'est-à-dire si et seulement si $E$ est vrai quelque soit la valeur des variables déclarées. Or, nous l'expliquerons, un terme s'interprète précisément comme une section de l'interprétation de son type. Ainsi, nous notons $W(E)$ le type dont l'interprétation est $j$. Tout terme de ce type démontre que $E$ est vrai quelque soit la valeur des variables déclarées dans le contexte $\Gamma$. C'est précisément ce que nous attendons d'une preuve de $E$ dans ce contexte. Les termes de type $W(E)$ sont donc les preuves de $E$.

Remarquons d'une part que ce type est clairement dépendant de $E$, c'est donc parce que, contrairement au langage interne, le langage $W$ autorise la dépendance qu'il peut représenter les preuves. D'autre part, notons que la flèche $\top$ étant injective (plus généralement un monomorphisme), il en est de même de $j$ dont il ne peut exister qu'une seule section. Ainsi, toutes les preuves d'un même énoncé, aussi éloignés soient les chemins qu'elles empruntent, ont une interprétation identique que nous nommerons leur *garant.*([4]) Le langage $W$ respecte donc le principe d'unicité du garant, qui est une version sémantique du principe de l'indiscernabilité des preuves. On notera qu'il peut exister dans certains topos des garants non représentables par des expressions du langage $W$, ce qui est une expression du théorème d'incomplétude de Gödel. Un garant non représentable correspond à un énoncé vrai (dans le topos considéré) mais non démontrable dans le langage $W$. Par contre, il est vrai qu'un énoncé quelconque a un garant si et seulement si il est vrai (dans le topos considéré).

Il nous incombe naturellement d'exprimer dans ce langage les déclarations inhérentes à toute démons-

1. Ce théorème était connu de Grothendieck dans le cas des topos de Grothendieck et joue même un rôle central dans son travail.
2. Précisons qu'il ne s'agit de défauts que relativement à cet objectif précis de traduction. Les intentions de Bénabou et Mitchell étaient toutes différentes lorsqu'ils l'ont construit.
3. $\Omega$ est la notation usuelle pour $\mathcal{P}(\mathbf{1})$, l'objet des parties de $\mathbf{1}$, notation que nous emploierons désormais.
4. Ce terme a été introduit pour la première fois par A. Prouté dans [5].

tration mathématique. Il se trouve que, compte tenu de la structure de $W$, les connecteurs logiques se définissent tous à partir de l'un d'entre eux, à savoir le quantificateur universel $\forall$. Il suffit donc d'introduire les symboles et règles propres aux démonstrations qui traitent d'énoncés universels. Deux seulement sont requis, le premier réfère à la déclaration « Soit $x \in X$. » qui permet de prouver un énoncé universellement quantifié, le second particularise (ou spécialise) une preuve d'un énoncé universellement quantifié $\forall_{x\in X}\ E$ en une preuve de l'énoncé $E[a/x]$, c'est-à-dire $E$ où $a$ prend la place de $x$.

Fournissons à présent un exemple du genre d'énoncés courants en mathématiques usuelles qui ne s'écrivent pas directement dans le langage interne mais s'expriment dans $W$. Considérons la borne supérieure d'une partie non vide de l'intervalle $]-\infty, 1]$. Pour $A \subset ]-\infty, 1]$, on pourra écrire la conjonction :

$$A \neq \emptyset \wedge \sup(A) > 0$$

qui est correcte, qu'elle soit vraie ou fausse. L'énoncé obtenu en permutant les opérandes de cette conjonction est bien sûr incorrect (car $\sup(A)$ n'est alors pas bien défini). Le second opérande n'a de sens que si le premier est vrai, c'est-à-dire s'il en existe un garant. La définition même de $\sup(A)$ dépend donc de l'existence de ce garant. Il est ainsi légitime de supposer qu'il existe une occurrence « invisible » d'une variable([5]) dans l'expression $\sup(A)$, représentant un garant de $A \neq \emptyset$. Le langage $W$ rend cette occurrence explicite et nous écrirons notre conjonction :

$$\exists_{p\in W(A\neq\emptyset)}\ \sup[p](A) > 0$$

ce qui se lit textuellement : « il existe un garant de $A \neq \emptyset$ tel que $\sup(A) > 0$ » et qui a bien le sens de l'énoncé $A \neq \emptyset \wedge \sup(A) > 0$ des mathématiques usuelles. Le fait que les types du langage $W$ dépendent des variables déclarées dans un contexte $\Gamma$ nous contraint à les *paramétrer* par $\Gamma$. Contrairement au langage interne, il n'est donc plus pertinent de les interpréter comme des objets du topos $\mathcal{T}$, mais comme des objets du topos relatif $\mathcal{T}/\Gamma$, c'est-à-dire comme des flèches de $\mathcal{T}$ de cible $\Gamma$. À cette différence près, les constructions de leurs sémantiques respectives sont assez similaires, c'est-à-dire que l'interprétation du langage $W$ *regardée dans les topos relatifs* procède d'une façon analogue à celle du langage interne *dans le topos originel.* Néanmoins, l'interprétation s'effectue dans le topos $\mathcal{T}$ originel. La réalisation de ce travail nécessite donc impérativement les traductions, effectuées en première partie, des opérations des topos relatifs (incluant celles des foncteurs qui permettent de circuler entre les contextes). Ajoutons que pour *compiler* le langage $W$, avec le langage de $\mathcal{T}$ comme langage cible, il est nécessaire d'exprimer ces traductions dans un langage suffisamment non ambigu. Il nous faudra donc en premier lieu définir ce langage dit « natif » qui répondra à cette contrainte et permettra d'effectuer des traductions parfaitement adéquates à l'usage que nous en ferons ensuite.

Ajoutons qu'il ne s'agit pas ici de redémontrer le théorème de Freyd, mais seulement, dans une première partie, de décrire les objets dont il prouve l'existence afin de construire, dans une seconde, l'interprétation du langage $W$ dans un topos $\mathcal{T}$.

5. Une variable cachée !

# Table des matières

**1 Topos et topos relatifs** **7**
- 1.1 Langage . . . 7
  - 1.1.1 Notations non ambiguës . . . 7
  - 1.1.2 Le langage « natif » . . . 8
  - 1.1.3 Abréviations (extensions du langage) . . . 9
  - 1.1.4 Le langage interne . . . 11
  - 1.1.5 Produits fibrés . . . 12
- 1.2 Topos relatif . . . 13
  - 1.2.1 Préliminaires . . . 13
  - 1.2.2 Objet final, produits, produits fibrés. . . . 13
  - 1.2.3 Le théorème fondamental de la théorie des topos. . . . 13
  - 1.2.4 Le classifiant de $\langle f\rangle_X \mapsto \mathbf{Sub}(\langle f\rangle_X \times \langle g\rangle_Y)$ . . . 14
  - 1.2.5 La bijection $\Sigma_{\langle g\rangle_Y}$ et la flèche $\ni$ . . . 16
  - 1.2.6 Les foncteurs $f^*$ et $\Pi_f$ . . . 17
  - 1.2.7 Résumé . . . 20

**2 Le langage $W$** **22**
- 2.1 Syntaxe du langage $W$ . . . 22
- 2.2 Interprétation du langage $W$ . . . 25
  - 2.2.1 Préliminaires . . . 25
  - 2.2.2 Les contextes . . . 26
  - 2.2.3 Les types . . . 27
  - 2.2.4 Les variables . . . 29
  - 2.2.5 Les termes . . . 30

# Chapitre 1

# Topos et topos relatifs

Un topos (élémentaire) est une catégorie qui a toutes les limites finies et telle que pour tout objet $Y$, le foncteur $X \mapsto \mathrm{Sub}(X \times Y)$ soit représentable.([1]) C'est donc quelque chose d'assez simple.([2]) De plus, il est possible d'exprimer tout cela sous forme d'opérations agissant sur les objets et les flèches du topos. Éventuellement, comme dans le cas de la division dans les corps (problème de la division par 0), une opération peut n'être appliquable que sous certaines conditions. Mais ceci n'est pas plus gênant pour calculer dans un topos que ce ne l'est pour calculer dans un corps.([3]) On va donc introduire des notations « non ambiguës » pour les objets et les flèches d'un topos, qui permettent de calculer d'une manière mécanique. Ces notations forment un langage qu'on peut appeler « natif », par opposition au « langage interne ».

## 1.1 Langage

### 1.1.1 Notations non ambiguës

Nous utiliserons des « notations non ambiguës », notations qui devront avoir les propriétés suivantes :

- Deux catégories différentes n'ont jamais la même notation.
- Deux objets différents d'une même catégorie n'ont jamais la même notation.
- Si deux flèches différentes d'une même catégorie ont la même notation, alors elles sont discernables par la donnée syntaxique de leurs sources.
- Les notations d'une flèche et de sa source permettent de déterminer une notation de son but.

1. On omet donc ici d'introduire un objet des entiers naturels.
2. Simplicité apparente, car elle se réfère aux notions de limite, de foncteur des sous-objets (Sub) et de classifiant, qui elles-mêmes ont besoin d'être définies. À l'extrême limite, une définition encore plus simple des topos consiste à dire qu'un topos est un topos pour qui sait déjà ce qu'est un topos. Toutefois, pour le lecteur non familier avec ces notions, on peut signaler les points suivants. On a des produits fibrés puisqu'il s'agit de limites finies, donc on peut parler du pullback d'une flèche le long d'une autre de même cible. Un sous-objet d'un objet $X$ est une classe d'équivalence de monomorphismes de cible $X$. Si $f : X \longrightarrow Y$ est une flèche quelconque et $m : \bullet \longrightarrow Y$ un monomorphisme, le pullback de $m$ le long de $f$ est un monomorphisme de cible $X$, et cette opération de pullback induit une application $\mathrm{Sub}(f) : \mathrm{Sub}(Y) \longrightarrow \mathrm{Sub}(X)$ entre les ensembles de sous-objets correspondants, faisant de Sub un foncteur (contravariant) à valeurs dans **Ens**. Ensemblistement, se donner une application de $X$ vers $\mathcal{P}(Y)$ (l'ensemble des parties de $Y$) revient à se donner une partie de $X \times Y$. C'est ce fait qui est généralisé sous la forme d'une bijection naturelle en $X$ entre $\mathcal{T}(X, \mathcal{P}(Y))$ et $\mathrm{Sub}(X \times Y)$ ($X \mapsto \mathrm{Sub}(X \times Y)$ est alors « représenté » par $\mathcal{P}(Y)$), ce qui suffit à caractériser $\mathcal{P}(Y)$.
3. Il se trouve que la structure de topos est « algébrique » sur la structure de graphe dans un sens très précis. Il s'agit du fait que le foncteur d'oubli de la catégorie des topos vers celle des graphes est « monadique », résultat établi par Burroni ([1]) avec une correction de C. Lair, correction publiée par Dubuc et Kelly ([2]). Nous n'utilisons pas ce résultat ici pour deux raisons. D'abord le fait que l'existence de conditions (de type diagrammes commutatifs) pour pouvoir appliquer une opération ne gêne en rien. D'autre part parce qu'une présentation équationnelle sur les graphes à la Burroni est fort peu intuitive et d'une très grande complexité par rapport à celle qui est adoptée ici.

Ces principes sont conformes à ceux de la plupart des langages de programmation dans lesquels un terme n'a de sens que relativement à un contexte donné. Il est par ailleurs possible, connaissant ce contexte, de déterminer le type du terme. Ici, au terme correspond la flèche, au contexte sa source et au type sa cible. Si la notation d'une flèche $f$ requiert effectivement une notation de sa source $X$, on écrira : $f \;\; (\text{rel. } X)$. De même, une égalité entre deux flèches $f$ et $g$ (nécessairement de même source $X$) pourra être notée $f = g \;\; (\text{rel. } X)$. Cette méthode nous permettra donc, dans notre seconde partie, de décrire l'interprétation des termes du langage $W$ par des formulations facilement transformables en programmes (d'une machine réelle).

### 1.1.2 Le langage « natif »

Pour présenter la structure de topos à l'aide d'opérations, nous utilisons une définition des topos légèrement différente de celle proposée ci-dessus, mais bien entendu équivalente. En fait, on remplace l'exigence d'avoir des limites finies par celle d'avoir un objet final, noté $\mathbf{1}$, et des produits fibrés. Il est bien connu que cela est équivalent au fait d'avoir toutes les limites finies. Quant au classifiant du foncteur $X \mapsto \mathrm{Sub}(X \times Y)$, on scinde le problème en deux parties. On demande d'abord un classifiant $\Omega$ du foncteur $X \mapsto \mathrm{Sub}(X)$, et on traduit cette exigence opérationnellement, et il reste à introduire les opérations qui établissent une bijection naturelle entre $\mathcal{T}(X, \mathcal{P}(Y))$ et $\mathcal{T}(X \times Y, \Omega)$. Ces opérations sont la « compréhension » $\Sigma$ et l'« appartenance » $\ni$. Par ailleurs, en présence de ces opérations, il n'est plus nécessaire de demander des produits fibrés, il suffit de demander des produits binaires ordinaires, les produits fibrés pouvant être reconstruits à partir de l'axiomatisation de $\Omega$. Nous introduisons donc les notations suivantes pour les objets :

| Objet | Interprétation |
|---|---|
| $\mathbf{1}$ | objet final |
| $X \times Y$ | produit des objets $X$ et $Y$ |
| $\mathcal{P}(X)$ | objet des parties de $X$ |
| $X \vert f$ | source d'un pullback de $\top : \mathbf{1} \longrightarrow \Omega$ le long de $f : X \longrightarrow \Omega$. |

où $\Omega$ est une abréviation pour $\mathcal{P}(\mathbf{1})$. Intuitivement, il faut comprendre la flèche $f : X \longrightarrow \Omega$ comme un prédicat sur $X$, $\Omega$ jouant le rôle d'« objet des valeurs de vérité », et il faut comprendre la flèche $\top : \mathbf{1} \longrightarrow \Omega$ comme la valeur de vérité « vrai ». Dès lors, le « sous-objet » $X|f$ de $X$ est intuitivement formé des « éléments » de $X$ qui satisfont le prédicat $f$. L'expression $X|f$ se lira « $X$ tel que $f$ ». C'est une forme combinatoire([4]) de la syntaxe de compréhension $\{x \in X \mid f\}$.

On n'impose aucune égalité entre des notations d'objets. Il se peut par exemple que les objets $X \times \mathbf{1}$ et $X$ soient égaux dans certains topos et distincts dans d'autres topos.([5]) On demande tout de même que l'égalité se propage comme d'habitude, c'est-à-dire que les opérations soient « déterministes ». Par exemple, si $X = X'$, alors $X \times Y = X' \times Y$, etc. sauf dans un cas. On ne demande pas en effet que $X|f$ et $X|g$ soient égaux quand $f$ et $g$ sont des flèches égales. L'opération $(X, f) \mapsto X|f$ peut donc très bien être « non déterministe » par rapport à $f$, les objets $X|f$ et $X|g$ n'étant éventuellement égaux que si $f$ et $g$ sont des notations identiques sur le plan syntaxique. Toutefois, si $f = g$, les objets $X|f$ et $X|g$ sont bien entendu isomorphes d'une manière canonique, et représentent donc le même sous-objet de $X$.([6])

4. Au sens où ces notations sont exemptes de variables.
5. Ils sont bien sûr toujours canoniquement isomorphes.
6. Cette possibilité que $X|f$ et $X|g$ ne soient pas égaux alors que $f$ et $g$ le sont pourra paraître étrange. Elle est toutefois naturelle et par ailleurs indispensable pour préserver l'utilité des preuves telles qu'elles ont été définies dans l'introduction, c'est-à-dire comme des notations pour les garants. En effet, si les deux énoncés $f$ et $\top$ sont équivalents, autrement-dit si $f$ est vrai, ou si l'on préfère si $f = \top$, on aurait $X|f = X|\top$ si on exigeait une propagation stricte de l'égalité. Mais alors tout garant de $\top$ serait un garant de n'importe quel énoncé démontrable, et par conséquent la notation triviale de ce garant serait une preuve de tout énoncé démontrable. Dans ces conditions, les preuves n'auraient plus aucun intérêt. Cet argument sera repris au début du second chapitre à propos du langage $W$. Noter que la possibilité pour une catégorie de ne pas être squelettale joue ici un rôle essentiel, et tout cela est en définitive bien dans l'esprit des catégories.

Les notations pour les flèches sont données dans la première colonne du tableau suivant. La deuxième colonne donne la forme que doit avoir une notation de l'objet source de la flèche pour que la notation de la flèche soit « adaptée » à la notation de son objet source, c'est-à-dire dans la forme qui convient pour que nos principes de non ambiguïté soient applicables. La troisième colonne indique la cible de la flèche et la quatrième son interprétation dans tout topos.

| Flèche | Source | Cible | Interprétation |
|---|---|---|---|
| $1$ | $X$ | $X$ | flèche identité de $X$ |
| $g \circ f$ | $X$ | $Z$ | composition des flèches $f : X \longrightarrow Y$ et $g : Y \longrightarrow Z$ |
| $\langle\rangle$ | $X$ | $\mathbf{1}$ | unique flèche de $X$ vers $\mathbf{1}$ |
| $\langle f, g\rangle$ | $X$ | $Y \times Z$ | flèche de composantes $f : X \longrightarrow Y$ et $g : X \longrightarrow Z$ |
| $\pi_1$ | $X \times Y$ | $X$ | première projection |
| $\pi_2$ | $X \times Y$ | $Y$ | seconde projection |
| $\top$ | $X$ | $\Omega$ | flèche « vrai » |
| $j$ | $X\vert f$ | $X$ | pullback de $\top$ le long de $f : X \longrightarrow \Omega$ |
| $u\vert f$ | $Y$ | $X\vert f$ | relèvement de $u : Y \longrightarrow X$ le long de $j : X\vert f \longrightarrow X$ |
| $\asymp$ | $X \times X$ | $\Omega$ | égalité interne (flèche caractéristique de $\Delta$) |
| $\sigma$ | $\mathcal{P}(X)$ | $\Omega$ | flèche caractéristique de $\Sigma_X(\asymp)$ |
| $\delta$ | $\mathcal{P}(X)\vert\sigma$ | $X$ | relèvement de $j : \mathcal{P}(X)\vert\sigma \longrightarrow \mathcal{P}(X)$ le long de $\Sigma_X(\asymp)$ |
| $\Sigma_Y(f)$ | $X$ | $\mathcal{P}(Y)$ | curryfiée de $f : X \times Y \longrightarrow \Omega$ |
| $\ni$ | $\mathcal{P}(X) \times X$ | $\Omega$ | appartenance (évaluateur) |

Les opérations décrites ci-dessus sont applicables à des flèches ayant des source et cibles convenables, comme indiquées ci-dessus, et l'opération $(u, f) \mapsto u|f$ requiert de plus que $f \circ u = \top$. On ne prétend pas que ces opérations sont suffisantes pour représenter toutes les flèches communes à tous les topos, et par conséquent, on ne prétend pas non plus que même accompagnées des équations convenables elle forment une définition de la notion de topos, même si on est très proche d'une telle définition.

Voici quelques considérations intuitives concernant ces flèches. $j : X|f \longrightarrow X$ est intuitivement l'« inclusion canonique » de $X|f$ dans $X$. Si la flèche $u : Y \longrightarrow X$ est telle que $f \circ u = \top$, alors cela signifie intuitivement que pour tout élément $y \in Y$, l'élément $u(y) \in X$ vérifie le prédicat $f$. C'est pourquoi l'image de $u$ est incluse dans $X|f$, d'où le relèvement $u|f$ de $u$ le long de $j : X|f \longrightarrow X$. La flèche $\asymp : X \times X \longrightarrow \Omega$ est bien évidemment le prédicat d'égalité $(x, y) \mapsto x = y$. La flèche $\sigma : \mathcal{P}(X) \longrightarrow \Omega$ est le prédicat sur $\mathcal{P}(X)$ « être un singleton », et $\delta : \mathcal{P}(X)|\sigma \longrightarrow X$ est l'« opérateur de description » qui extrait d'un singleton son unique élément, autrement-dit $\delta(\{x\}) = x$. $\Sigma_Y(f) : X \longrightarrow \mathcal{P}(Y)$ est intuitivement la fonction $x \mapsto \{y \in Y \mid f(x, y)\}$. On dit que $\Sigma_Y(f)$ est la « curryfiée » de $f$. L'opération $\Sigma_Y$ est aussi appelée « compréhension ». Noter que $\Sigma_X(\asymp) : X \longrightarrow \mathcal{P}(X)$ est la « flèche singleton » envoyant $x$ sur $\{x\}$. Enfin, $\ni: \mathcal{P}(X) \times X \longrightarrow \Omega$ est l'« appartenance ». Intuitivement, $\ni (A, x)$ signifie que $x$ appartient à $A$.

### 1.1.3 Abréviations (extensions du langage)

Nous introduisons un certain nombre d'abréviations qui nous éviteront d'avoir à écrire des expressions trop lourdes ou pire incompréhensibles.( [7]) Ce sont les suivantes :

7. Comme on va le voir plus loin, on en écrit quand même quelques unes pour le fun.

| Abréviation | Signification |
|---|---|
| $\top$ | $\top \circ \langle\rangle$ |
| $f \times g$ | $\langle f \circ \pi_1, g \circ \pi_2 \rangle$ |
| $\Delta$ | $\langle 1, 1 \rangle$ |
| $f \asymp g$ | $\asymp \circ \langle g, f \rangle$ |
| $f \in g$ | $\ni \circ \langle g, f \rangle$ |
| $f \wedge g$ | $\langle f, g \rangle \asymp \langle \top, \top \rangle$ (pour $f, g : X \longrightarrow \Omega$) |
| $f \Rightarrow g$ | $f \asymp (f \wedge g)$ (pour $f, g : X \longrightarrow \Omega$) |
| $f \Leftrightarrow g$ | $(f \Rightarrow g) \wedge (g \Rightarrow f)$ (pour $f, g : X \longrightarrow \Omega$) |
| $\forall_X(f)$ | $\Sigma_X(f) \asymp \Sigma_X(\top)$ (pour $f : Y \times X \longrightarrow \Omega$) |
| $\exists_X(f)$ | $\forall_\Omega((\forall_X(f \circ (\pi_1 \times 1) \Rightarrow \pi_2 \circ \pi_1)) \Rightarrow \pi_2)$ (pour $f : Y \times X \longrightarrow \Omega$) |
| $f^\bullet_X$ | $\Sigma_X( \asymp \circ (1 \times f))$ (pour $f : X \longrightarrow Y$) |
| $\chi^X_f$ | $\exists_X( \asymp \circ (1 \times f))$ (pour $f : X \longrightarrow Y$) |
| $\sigma$ | $\chi^X_{\Sigma_X(\asymp)}$ |
| $\exists!_X(f)$ | $\sigma \circ \Sigma_X(f)$ (pour $f : Y \times X \longrightarrow \Omega$) |
| $f \vec{\wedge} g$ | $\chi^{(X\|f)\|g}_{j \circ j}$ (pour $f : X \longrightarrow \Omega$ et $g : X\|f \longrightarrow \Omega$) |
| $\subset$ | $\forall_X \ ((\pi_2 \in \pi_1 \circ \pi_1) \Rightarrow (\pi_2 \in \pi_2 \circ \pi_1))$ |

La plupart de ces abréviations sont classiques en théorie des topos. Voici quelques indications intuitives (ensemblistes). Pour $f : X \longrightarrow Y$ et $g : Z \longrightarrow T$, $f \times g : X \times Z \longrightarrow Y \times T$ envoie $(x, z)$ sur $(f(x), g(z))$. $\Delta : X \longrightarrow X \times X$ est bien sûr la « diagonale » de $X$, c'est-à-dire $x \mapsto (x, x)$. $f \wedge g$ est la conjonction des deux « prédicats » $f$ et $g$. L'abréviation ci-dessus la définit comme l'égalité du couple $(f, g)$ avec le couple $(\top, \top)$. $f \Rightarrow g$ est défini comme l'égalité entre $f$ et $f \wedge g$. L'énoncé $\forall_{x \in X} \ E$ est défini comme l'égalité de l'ensemble $\{x \in X \mid E\}$ et de la partie pleine de $X$. La définition ci-dessus du quantificateur existentiel est la traduction de la définition bien connue des intuitionnistes $\exists_{x \in X} \ f := \forall_{q \in \Omega} \ ((\forall_{x \in X} \ (f \Rightarrow q)) \Rightarrow q$, qui devient d'ailleurs $\exists_{x \in X} \ f := \neg \forall_{x \in X} \ \neg f$ dans le cas classique. Pour une flèche $f : X \longrightarrow Y$, la flèche $f^\bullet_X : Y \longrightarrow \mathcal{P}(X)$([8]) envoie tout $y \in Y$ sur l'ensemble de ses antécédents par $f$. Pour $f : X \longrightarrow Y$, $\chi^X_f : Y \longrightarrow \Omega$ est la flèche caractéristique de l'image de $f$. L'opération $\sigma$ est la flèche caractéristique de la flèche « singleton » $\Sigma_X( \asymp ) : X \longrightarrow \mathcal{P}(X)$. C'est donc le prédicat sur $\mathcal{P}(X)$ qui caractérise les singletons. Bien qu'on puisse définir $\exists!_X$ directement à partir de $\exists_X$, il est préférable de le définir comme on l'a fait dans le tableau ci-dessus, c'est-à-dire par la formule qui signifie intuitivement : « l'ensemble des éléments de $X$ qui satisfont $f$ est un singleton ». Les deux définitions sont bien sûr clairement équivalentes, puisque $\chi$ est défini à partir de $\exists$. L'opération $(f, g) \mapsto f \vec{\wedge} g$ est la « conjonction dépendante » de $f$ et $g$. On se donne un prédicat $f$ sur $X$, puis un prédicat $g$ qui n'est défini que sur les éléments de $X$ pour lesquels $f$ est vrai. On a donc le monomorphisme composé $(X|f)|g \xrightarrow{j} X|f \xrightarrow{j} X$ qui représente les éléments de $X$ qui satisfont à la fois $f$ et $g$. Sa flèche caractéristique $f \vec{\wedge} g : X \longrightarrow \Omega$ est donc une conjonction (dépendante) de $f$ et $g$. Enfin, la définition de l'inclusion $\subset \ : \mathcal{P}(X) \times \mathcal{P}(X) \longrightarrow \Omega$ est la traduction de $A \subset B := \forall_{x \in X} \ x \in A \Rightarrow x \in B$. Comme l'expansion de ces « macros » peut se faire syntaxiquement (à condition de connaître la source de la flèche dans le cas de $\subset$), ces nouvelles notations respectent les principes de non ambiguïté énoncés plus haut.

Pour toute flèche $f : X \longrightarrow \Omega$ dans un topos $\mathcal{T}$, on a une correspondance bijective entre $\mathcal{T}(Y, X|f)$ et le sous-ensemble de $\mathcal{T}(Y, X)$ des $u : Y \longrightarrow X$ telles que $f \circ u = \top$. Cette correspondance, qui est induite par $j_* : \mathcal{T}(Y, X|f) \longrightarrow \mathcal{T}(Y, X)$, est bien sûr naturelle en $Y$. Remarquer également que $j$ étant un monomorphisme, on a $(u|f) \circ v = (u \circ v)|f$, pour toutes flèches $Z \xrightarrow{v} Y \xrightarrow{u} X \xrightarrow{f} \Omega$. Il est clair par ailleurs qu'on a $(f \wedge g) \circ \varphi = (f \circ \varphi) \wedge (g \circ \varphi)$, et de même avec $\asymp$ ou $\Rightarrow$ à la place de $\wedge$, et il est facile d'établir que $(f \vec{\wedge} g) \circ \varphi = (f \circ \varphi) \vec{\wedge} (g \circ ((\varphi \circ j)|f))$.

8. Que nous évitons de noter $f^{-1}$, cette notation étant réservée pour les inverses des isomorphismes.

### 1.1.4 Le langage interne

Nous adoptons les définitions et notations utilisées par A. Prouté dans son cours ([5]) concernant le langage interne des topos. Voici un résumé de ce qu'il est utile de savoir pour l'usage qui en est fait ici. Un « type » du langage interne de $\mathcal{T}$ est juste un objet de $\mathcal{T}$. Un « contexte » est une suite de « déclarations de variables » de la forme :

$$(x_1 \in X_1) \dots (x_n \in X_n)$$

où les symboles (variables) $x_1, \dots, x_n$ sont distincts et où $X_1, \dots, X_n$ sont des types. On utilisera la lettre $\Gamma$ pour représenter un contexte quelconque. On note $\emptyset$ le contexte vide (aucune déclaration). On note $\Gamma(x \in X)$ le contexte obtenu en ajoutant la déclaration $(x \in X)$ à $\Gamma$. L'interprétation du contexte $\Gamma$ est l'objet de $\mathcal{T}$ défini inductivement par :

- $\overline{\emptyset} = \mathbf{1}$
- $\overline{(x \in X)} = X$
- $\overline{\Gamma(x \in X)} = \overline{\Gamma} \times X$ (si $\Gamma$ est non vide)

On fera attention au fait que le produit n'est pas supposé associatif de façon à ce que les opérations $\pi_1$ et $\pi_2$ restent non ambiguës. Les « termes » du langage interne de $\mathcal{T}$, valides relativement à un contexte donné, sont définis comme suit :

- Les variables relativement à tout contexte $\Gamma$ dans lequel elles sont déclarées. Leur type est donné par la déclaration les concernant dans $\Gamma$.
- $*$ relativement à tout contexte. Son type est $\mathbf{1}$.
- $(a, b)$ (« paire ») relativement au contexte $\Gamma$, pourvu que $a$ et $b$ soient des termes relativement à $\Gamma$. Son type est $X \times Y$, où $X$ est le type de $a$ et $Y$ le type de $b$.
- $p_1(a)$ et $p_2(a)$ (« projections ») relativement au contexte $\Gamma$, pourvu que $a$ soit un terme relativement au contexte $\Gamma$ dont le type est un produit $X \times Y$. Ces deux termes sont alors de types respectifs $X$ et $Y$.
- $\{x \in X \mid E\}$ (« ensemble ») dans le contexte $\Gamma$, pourvu que $E$ soit un terme de type $\Omega$ dans le contexte $\Gamma(x \in X)$. Son type est $\mathcal{P}(X)$.
- $a \in A$ (« appartenance ») dans le contexte $\Gamma$, pourvu que $A$ soit un terme de type $\mathcal{P}(X)$ dans le contexte $\Gamma$, et $a$ un terme de type $X$ dans le contexte $\Gamma$. Son type est $\Omega$.
- $f[u]$ relativement au contexte $\Gamma$, pourvu que $u$ soit un terme de type $X$ dans le contexte $\Gamma$ et $f$ la notation en langage natif d'une flèche de $X$ vers $Y$. Le type de ce terme est $Y$ ( $f[u]$ est simplifié en $f$ dans le cas où la source de $f$ est $\mathbf{1}$).
- $a = b$ (« égalité ») relativement à $\Gamma$, pourvu que $a$ et $b$ soient deux termes de même type relativement à $\Gamma$. Son type est $\Omega$.

La « sémantique » (ou « compilation » ou « interprétation ») d'un terme $t$ dans le contexte $\Gamma$ est notée $\lfloor t \rfloor_\Gamma$. Il s'agit d'une flèche de $\overline{\Gamma}$ vers $Y$, où $\overline{\Gamma}$ est l'objet qui est l'interprétation du contexte $\Gamma$ et $Y$ le type de $t$. Elle est définie inductivement par :

- $\lfloor x \rfloor_{(x \in X)} = 1_X$
- $\lfloor x \rfloor_{\Gamma(x \in X)} = \pi_2$ (si $\Gamma$ n'est pas vide),
- $\lfloor x \rfloor_{\Gamma(y \in Y)} = \lfloor x \rfloor_\Gamma \circ \pi_1$ (si $y$ est distinct de $x$).
- $\lfloor * \rfloor_\Gamma = \langle\rangle$
- $\lfloor (a, b) \rfloor_\Gamma = \langle \lfloor a \rfloor_\Gamma, \lfloor b \rfloor_\Gamma \rangle$
- $\lfloor p_1(a) \rfloor_\Gamma = \pi_1 \circ \lfloor a \rfloor_\Gamma$
- $\lfloor p_2(a) \rfloor_\Gamma = \pi_2 \circ \lfloor a \rfloor_\Gamma$
- $\lfloor \{x \in X \mid E\} \rfloor_\Gamma = \Sigma_X(\lfloor E \rfloor_{\Gamma(x \in X)})$
- $\lfloor a \in A \rfloor_\Gamma = \ni \circ \langle \lfloor A \rfloor_\Gamma, \lfloor a \rfloor_\Gamma \rangle$

- $\lfloor f[u] \rfloor_\Gamma = f \circ \lfloor u \rfloor_\Gamma$ ($\lfloor f \rfloor_\Gamma = f \circ \langle\rangle$ dans le cas où $\mathbf{s}(f) = \mathbf{1}$)
- $\lfloor a = b \rfloor_\Gamma = \lfloor a \rfloor_\Gamma \asymp \lfloor b \rfloor_\Gamma = \asymp \circ \langle \lfloor a \rfloor_\Gamma, \lfloor b \rfloor_\Gamma \rangle$

Un certain nombre d'abréviations (macros) étendent le langage interne. Ce sont les suivantes :

- $\top$ représente $* = *$
- $\forall_{x \in X}\ E$ représente $\{x \in X \mid E\} = \{x \in X \mid \top\}$
- $E \wedge F$ représente $(E, F) = (\top, \top)$
- $E \Rightarrow F$ représente $E = (E \wedge F)$
- $\perp$ représente $\forall_{q \in \Omega}\ q$
- $E \vee F$ représente $\forall_{q \in \Omega}\ ((E \Rightarrow q) \wedge (F \Rightarrow q)) \Rightarrow q$
- $\exists_{x \in X}\ E$ représente $\forall_{q \in \Omega}\ (\forall_{x \in X}\ (E \Rightarrow q)) \Rightarrow q$

Dans la suite, nous utiliserons les définitions ci-dessus pour « compiler » des termes, sans autre forme de procès. On retiendra les faits suivants que nous rappelons sans démonstration :

- $\lfloor t \rfloor_{(x \in X)} \circ \lfloor s \rfloor_\Gamma = \lfloor t[s/x] \rfloor_\Gamma$ (pour $s$ de type $X$).
- $\lfloor t \rfloor_{\Gamma(x \in X)} = \lfloor t \rfloor_\Gamma \circ \pi_1$ (si $t$ relatif à $\Gamma$).
- Tout énoncé (terme de type $\Omega$) du langage interne démontré intuitionnistiquement dans le contexte $\Gamma$, représente la flèche $\top : \overline{\Gamma} \longrightarrow \Omega$ (théorème de robustesse).
- Une flèche $f : X \longrightarrow Y$ est un monomorphisme (resp. épimorphisme) si et seulement si $\lfloor \forall_{x \in X}\ \forall_{y \in X}\ f[x] = f[y] \Rightarrow x = y \rfloor_\emptyset = \top$ (resp. $\lfloor \forall_{y \in Y}\ \exists_{x \in X}\ y = f[x] \rfloor_\emptyset = \top$).

### 1.1.5 Produits fibrés

Remarquons que le langage natif ne fournit pas de notation pour le produit fibré. Néanmoins, il est possible de le construire. En effet, soient $f : X \longrightarrow Z$ et $g : Y \longrightarrow Z$ deux flèches de $\mathcal{T}$. Il s'agit de construire la limite du diagramme (gnomon) formé par $f$ et $g$. Pour cela, considérons un pullback de $\top$ le long de la flèche $\asymp \circ (f \times g) : X \times Y \longrightarrow \Omega$. On obtient une flèche $j : X \times Y | \asymp \circ (f \times g) \longrightarrow X \times Y$. La source de $j$ est alors notre produit fibré, qu'on notera désormais $X\ {}_f{\times}_g\ Y$, et les deux projections de ce produit fibré sont les deux composantes de $j$. Autrement-dit, on a le carré cartésien :

$$\begin{tikzcd} X\ {}_f{\times}_g\ Y \arrow[r, "\pi_2 \circ j"] \arrow[d, "\pi_1 \circ j"'] & Y \arrow[d, "g"] \\ X \arrow[r, "f"'] & Z \end{tikzcd}$$

En effet, donnons-nous deux flèches $u : T \longrightarrow X$ et $v : T \longrightarrow Y$ telles que $f \circ u = g \circ v$. On a alors $\asymp \circ (f \times g) \circ \langle u, v \rangle = \top$. Il en résulte que $\langle u, v \rangle$ se relève le long de $j : X \times Y | \asymp \circ (f \times g) \longrightarrow X \times Y$ en la flèche $\langle u, v \rangle | \asymp \circ (f \times g)$, qu'on notera désormais $\langle u, v \rangle_{(f,g)}$.([9])

$$\begin{tikzcd} T \arrow[dr, "\langle u,v \rangle_{(f,g)}"] \arrow[drr, bend left, "v"] \arrow[ddr, bend right, "u"'] & & \\ & X\ {}_f{\times}_g\ Y \arrow[r, "\pi_2 \circ j"] \arrow[d, "\pi_1 \circ j"'] & Y \arrow[d, "g"] \\ & X \arrow[r, "f"'] & Z \end{tikzcd}$$

9. L'indice $(f, g)$ est nécessaire pour préserver nos propriétés de non ambiguïté.

## 1.2 Topos relatif

### 1.2.1 Préliminaires

Si $\mathcal{C}$ est une catégorie et $A$ un objet de $\mathcal{C}$, on a une « catégorie relative » notée $\mathcal{C}/A$, dont les objets sont les clones des flèches de $\mathcal{C}$ de cible $A$. La flèche $f : X \longrightarrow A$ de $\mathcal{C}$ est donc un objet qu'on notera $\langle f\rangle_X$ dans $\mathcal{C}/A$. Une flèche de $\mathcal{C}/A$ de $\langle f\rangle_X$ vers $\langle g\rangle_Y$ est un clone d'une flèche $\varphi$ de $\mathcal{C}$ de $X$ vers $Y$ telle que $g \circ \varphi = f$. Cette flèche de $\mathcal{C}/A$ sera notée $[\varphi]_g$. On dira que $f$ et $\varphi$ *représentent* respectivement $\langle f\rangle_X$ et $[\varphi]_g$. Ces notations, qui pourront paraître un peu lourdes, sont non ambiguës au sens défini plus haut, pour la catégorie $\mathcal{C}/A$. En effet, une notation d'objet $\langle f\rangle_X$ donne les notations $X$ et $f$, et détermine donc sans ambiguïté la flèche $f$ de $\mathcal{C}$, donc l'objet correspondant de $\mathcal{C}/A$. De même, les notations $\langle f\rangle_X$ et $[\varphi]_g$ étant données, on dispose de $X$, $f$, $\varphi$ et $g$ comme notations de $\mathcal{C}$. La flèche $\varphi$, qui est de source $X$, est donc bien déterminée et on peut calculer la cible $Y$ de $\varphi$, ce qui fait que $g$ est bien déterminée et qu'aucune ambiguïté ne subsiste.

### 1.2.2 Objet final, produits, produits fibrés.

L'objet final $\mathbf{1}$ de $\mathcal{T}/A$ est $\langle 1\rangle_A$. On a donc $\mathbf{1} = \langle 1\rangle_A$ dans $\mathcal{T}/A$. De plus l'unique flèche de $\langle f\rangle_X$ vers $\langle 1\rangle_A$ est clairement $[f]_1$. On a donc $\langle\rangle = [f]_1$ (rel. $\langle f\rangle_X$).

On sait que si $\mathcal{C}$ a des produits fibrés, il en est de même de $\mathcal{C}/A$. Un gnomon dans $\mathcal{T}/A$ est formé de deux flèches $[\varphi]_h : \langle f\rangle_X \longrightarrow \langle h\rangle_Z$ et $[\psi]_h : \langle g\rangle_Y \longrightarrow \langle h\rangle_Z$. Essentiellement, le produit fibré est « le même » dans $\mathcal{T}$ et dans $\mathcal{T}/A$, ce qui signifie qu'on a le carré cartésien suivant dans $\mathcal{T}/A$ :([10])

$$\begin{tikzcd}
\langle h\circ\varphi\circ\pi_1\circ j\rangle_{X\times Y|\asymp\circ(\varphi\times\psi)} \arrow[r, "{[\pi_2\circ j]_g}"] \arrow[d, "{[\pi_1\circ j]_f}"'] & \langle g\rangle_Y \arrow[d, "{[\psi]_h}"] \\
\langle f\rangle_X \arrow[r, "{[\varphi]_h}"'] & \langle h\rangle_Z
\end{tikzcd}$$

De plus, le produit dans la catégorie relative n'est autre que le produit fibré dans la catégorie initiale. En effet, soient $\langle f\rangle_X$ et $\langle g\rangle_Y$ deux objets de $\mathcal{T}/A$, alors $\langle f\rangle_X \times \langle g\rangle_Y = \langle f\circ\pi_1\circ j\rangle_{(X\times Y)|(\asymp\circ(f\times g))}$ et les première et seconde projections sont respectivement $[\pi_1\circ j]_f$ et $[\pi_2\circ j]_g$.

$$\begin{tikzcd}
X\times Y|\asymp\circ(f\times g) \arrow[r, "\pi_2\circ j"] \arrow[d, "\pi_1\circ j"'] \arrow[dr, "f\circ\pi_1\circ j"] & Y \arrow[d, "g"] \\
X \arrow[r, "f"'] & A
\end{tikzcd}$$

### 1.2.3 Le théorème fondamental de la théorie des topos.

Le « théorème fondamental de la théorie des topos », dû à P. Freyd ([3]), dit que si $\mathcal{T}$ est un topos et $A$ un objet de $\mathcal{T}$, alors la catégorie relative $\mathcal{T}/A$ est un topos, et que si $f : A \longrightarrow B$ est une flèche de $\mathcal{T}$, le foncteur de changement de base $f^* : \mathcal{T}/B \longrightarrow \mathcal{T}/A$ est un foncteur (logique) avec un adjoint à gauche $\amalg_f$ et un adjoint à droite $\Pi_f$. L'objet de ce chapitre n'est pas de redémontrer ce théorème, mais d'en revisiter certains aspects, en utilisant nos notations non ambiguës pour rendre effectives (programmables) les constructions proposées par le théorème. Il s'agit en particulier d'exprimer le classifiant $\mathcal{P}(\langle g\rangle_Y)$ du

10. Noter que $\varphi\circ\pi_1\circ j = \psi\circ\pi_2\circ j$.

foncteur $\langle f\rangle_X \mapsto \mathrm{Sub}(\langle f\rangle_X \times \langle g\rangle_Y)$ dans le langage natif de $\mathcal{T}$, et d'obtenir une formule explicite pour le foncteur $\Pi_f$.

Concernant ce dernier, le cas particulier où $\mathcal{T}$ est la catégorie <u>**Ens**</u> des ensembles est particulièrement important pour deux raisons. D'abord, il est essentiel pour soutenir l'intuition. Autant les constructions du théorème sont assez évidentes dans ce cas, autant elles risquent d'être assez obscures dans le cas général. La deuxième raison est que ce cas particulier est en fait assez générique, puisque si on le traite intuitionnistiquement, le langage interne permettra d'en déduire le cas général. La méthode utilisée ici consiste donc à traiter la question ensemblistement et intuitionnistiquement, et à laisser le langage interne faire son travail.

Développons un peu ce dernier point. Le langage interne s'interprète dans n'importe quel topos : quelque soit le topos $\mathcal{T}$ fixé, il existe une interprétation des termes de ce langage dans $\mathcal{T}$. Notons qu'il s'agit d'un langage d'aspect ensembliste (essentiellement parce qu'il réintroduit la notion d'élément). De plus, les règles de démonstration intuitionnistes valent pour ce langage, c'est-à-dire qu'il y a un théorème de robustesse qui prouve que toutes ces règles sont valides, et ce quelque soit le topos $\mathcal{T}$ dans lequel nous interprétons le langage interne.

Pratiquement, ceci nous autorise à construire des objets via le langage interne et à raisonner sur eux dans un cadre ensembliste (mais bien sûr intuitionniste) tout en étant assurés que les conclusions de ces raisonnements demeureront valides pour ces objets interprétés dans un topos $\mathcal{T}$ quelconque. Cette arme est d'autant plus puissante que nous définissons la plupart de nos objets pour leurs propriétés ensemblistes (c'est pourquoi nous faisons principalement appel à cette intuition), tout en souhaitant conserver la vérité de ces propriétés dans le cadre général. Ainsi, cette méthode simplifie souvent les choses.

En particulier, on a les deux foncteurs $f^*$ et $\Pi_f$, le premier adjoint à gauche du second. L'action ensembliste de ces foncteurs est évidente. Cette partie gagnera donc beaucoup en clarté et en simplicité par l'utilisation du langage interne.

### 1.2.4 Le classifiant de $\langle f\rangle_X \mapsto \mathbf{Sub}(\langle f\rangle_X \times \langle g\rangle_Y)$

Pour tout objet $\langle g\rangle_Y$ de $\mathcal{T}/A$, nous souhaitons exprimer l'objet $\mathcal{P}(\langle g\rangle_Y)$, « objet des parties de $\langle g\rangle_Y$ », classifiant du foncteur $\langle f\rangle_X \mapsto \mathbf{Sub}(\langle f\rangle_X \times \langle g\rangle_Y)$, dans le langage natif de $\mathcal{T}$. Ensemblistement, $\langle g\rangle_Y$ est un fibré sur $A$ dont la fibre au dessus de $a \in A$ est $g^\bullet_Y(a)$ (l'ensemble des antécédents de $a$ par $g$) et $\mathcal{P}(\langle g\rangle_Y)$ est un fibré sur $A$ tel que pour chaque $a \in A$, la fibre au dessus de $a$ soit l'ensemble des parties de $g^\bullet_Y(a)$. Le représentant $\psi_g$ de $\mathcal{P}(\langle g\rangle_Y)$ dans $\mathcal{T}$ est donc défini sur le sous-ensemble $\Gamma_g$ de $\mathcal{P}(Y) \times A$ qui contient tous les couples $(S, a)$ tels que $S \subset g^\bullet_Y(a)$, autrement-dit on a $\Gamma_g = \{(S, a) \in \mathcal{P}(Y) \times A \mid S \subset g^\bullet_Y(a)\}$, et $\psi_g$ envoie chacun de ces couples sur sa deuxième composante : $a$. On construit dans le cas général l'objet $\Gamma_g$ et la flèche $\psi_g$ en demandant que les deux carrés :

$$\begin{tikzcd}
\Gamma_g \arrow[r] \arrow[d, "\langle \rho_g, \psi_g\rangle"'] & (\mathcal{P}(Y) \times \mathcal{P}(Y))|\subset \arrow[r] \arrow[d] & 1 \arrow[d, "\top"] \\
\mathcal{P}(Y) \times A \arrow[r, "1 \times g^\bullet_Y"'] & \mathcal{P}(Y) \times \mathcal{P}(Y) \arrow[r, "\subset"'] & \Omega
\end{tikzcd}$$

soient cartésiens, c'est-à-dire que $\Gamma_g = (\mathcal{P}(Y) \times A)| \subset \circ (1 \times g^\bullet_Y)$, et $\psi_g = \pi_2 \circ j$, ce qui donne finalement :

$$\mathcal{P}(\langle g\rangle_Y) = \langle \psi_g\rangle_{\Gamma_g} = \langle \pi_2 \circ j\rangle_{(\mathcal{P}(Y)\times A)|(\subset \circ (1 \times g^\bullet_Y))}$$

Comme pour le topos $\mathcal{T}$, nous noterons $\Omega$ l'objet $\mathcal{P}(\mathbf{1})$ du topos relatif $\mathcal{T}/A$. Il s'avère que la source de la flèche correspondante de $\mathcal{T}$, c'est-à-dire $\Gamma_{\mathbf{1}}$, est isomorphe à $A \times \Omega$ (autrement-dit, $\Omega$ comme objet de $\mathcal{T}/A$, est un fibré trivial au dessus de $A$ dont la fibre est l'$\Omega$ de $\mathcal{T}$). Quitte à composer avec des isomorphismes canoniques convenables, il est donc possible de regarder cette flèche comme la première projection $\pi_1 : A \times \Omega \longrightarrow A$. Cette option sera tout à fait pertinente lors de l'interprétation du langage $W$. Nous allons donc exhiber dès à présent l'isomorphisme en question et son inverse.

L'intuition derrière cet isomorphisme est que dans le fibré $1 : A \longrightarrow A$, qui est l'objet final $\mathbf{1} = \langle 1 \rangle_A$ de $\mathcal{T}/A$, la fibre au dessus de $a \in A$ est le singleton $\{a\}$. En conséquence, la fibre de $\langle \mathcal{P}(\mathbf{1}) \rangle_{\Gamma_1}$ est isomorphe au $\mathcal{P}(\mathbf{1})$ de $\mathcal{T}$, c'est-à-dire à $\Omega$ de $\mathcal{T}$. La flèche (de $\mathcal{T}$) de $A \times \Omega$ vers $\Gamma_1$ qui va réaliser cet isomorphisme envoie donc intuitivement le couple $(a, v) \in A \times \Omega$ vers le couple $(S(v), a) \in \Gamma_1 \subset \mathcal{P}(A) \times A$ tel que $S(v)$, comme partie de $\{a\}$, représente la même valeur de vérité que $v$ comme partie de $\mathbf{1}$. Autrement-dit $S(v) = \{x \in \{a\} \mid v\}$, ou encore $S(v) = \{x \in A \mid (x = a) \wedge v\}$, par définition des singletons. On considère donc la flèche $h : A \times \Omega \longrightarrow \mathcal{P}(A) \times A$ définie intuitivement par $h(a, v) = (\{x \in A \mid (x = a) \wedge v\}, a)$ ce qui peut s'écrire comme ceci à l'aide du langage interne :

$$h = \lfloor (\{x \in A \mid (x = a) \wedge v\}, a) \rfloor_{(a \in A)(v \in \Omega)}$$

On remarque que $h$ est un monomorphisme. En effet, de $(\{x \in A \mid (x = a) \wedge v\}, a) = (\{x \in A \mid (x = a') \wedge v'\}, a')$, on déduit $a = a'$, puis, pour tout $x \in A$, $((x = a) \wedge v) \Leftrightarrow ((x = a') \wedge v')$, d'où $v \Rightarrow v'$ en faisant $x = a$, puis $v \Leftrightarrow v'$. On peut également compiler cette expression pour obtenir $h = \langle \Sigma_A((\asymp \circ \langle \pi_2, \pi_1 \circ \pi_1 \rangle) \wedge (\pi_2 \circ \pi_1)), \pi_1 \rangle$ (sans faire l'expansion de $\wedge$).

Dans le cas qui nous occupe ici (c'est-à-dire le cas où $g = 1$), la flèche $\subset \circ (1 \times g_Y^\bullet)$, qui forme la ligne inférieure du diagramme précédent, est $\subset \circ (1 \times 1_A^\bullet)$, et comme $1_A^\bullet : A \longrightarrow \mathcal{P}(A)$ n'est autre que la flèche singleton, on a :

$$\subset \circ (1 \times 1_A^\bullet) = \lfloor S \subset \{a\} \rfloor_{(S \in \mathcal{P}(A))(a \in A)}$$

Il en résulte que :

$$\subset \circ (1 \times 1_A^\bullet) \circ h = \lfloor \{x \in A \mid (x = a) \wedge v\} \subset \{a\} \rfloor_{(a \in A)(v \in \Omega)}$$

L'énoncé $\{x \in A \mid (x = a) \wedge v\} \subset \{a\}$ étant clairement vrai, ce composé est $\top$ et $h$ se relève en un monomorphisme $i$ (puisque $h$ en est un), le long de $j = \langle \rho_1, \psi_1 \rangle$ :

$$\begin{tikzcd}
 & \Gamma_1 \arrow[r] \arrow[d, "\langle \rho_1, \psi_1 \rangle"] & (\mathcal{P}(A) \times \mathcal{P}(A))|\subset \arrow[r] \arrow[d] & 1 \arrow[d, "\top"] \\
A \times \Omega \arrow[ur, "i"] \arrow[r, "h"'] & \mathcal{P}(A) \times A \arrow[r, "1 \times 1_A^\bullet"'] & \mathcal{P}(A) \times \mathcal{P}(A) \arrow[r, "\subset"'] & \Omega
\end{tikzcd}$$

avec bien sûr $i = h|(\subset \circ (1 \times 1_A^\bullet))$. Exhibons maintenant (toujours en utilisant le langage interne) un inverse $k$ de cette flèche. À un $z \in \Gamma_1$ on associe le couple $(\psi_1(z), \psi_1(z) \in \rho_1(z))$, ce qui s'écrit :

$$k = \lfloor (\psi_1[z], \psi_1[z] \in \rho_1(z)) \rfloor_{(z \in \Gamma_1)}$$

Pour montrer que $i$ et $k$ sont des isomorphismes réciproques l'un de l'autre, il suffit de vérifier que $\langle \rho_1, \psi_1 \rangle \circ i = h$ et $h \circ k = \langle \rho_1, \psi_1 \rangle$. La première égalité est vraie par définition de $i$. Pour la seconde, on a :

$$\begin{array}{rcl}
h \circ k & = & \lfloor (\{x \in A \mid (x = a) \wedge v\}, a) \rfloor_{(a \in A)(v \in \Omega)} \circ \lfloor (\psi_1[z], \psi_1[z] \in \rho_1(z)) \rfloor_{(z \in \Gamma_1)} \\
 & = & \lfloor (\{x \in A \mid (x = \psi_1[z]) \wedge \psi_1[z] \in \rho_1[z]\}, \psi_1[z]) \rfloor_{(z \in \Gamma_1)} \\
 & = & \lfloor (\rho_1[z], \psi_1[z]) \rfloor_{(z \in \Gamma_1)} \qquad (\text{car } \rho_1[z] \subset \{\psi_1[z]\}) \\
 & = & \langle \rho_1, \psi_1 \rangle
\end{array}$$

Pour un usage futur, nous exprimons $i$ et $k$ dans les notations non ambiguës de $\mathcal{T}$ :

$$\begin{array}{rcll}
i & = & (\langle \Sigma_A((\asymp \circ \langle \pi_2, \pi_1 \circ \pi_1 \rangle) \wedge (\pi_2 \circ \pi_1)), \pi_1 \rangle)|(\subset \circ (1 \times 1_A^\bullet)) & (\text{rel. } A \times \Omega) \\
k & = & \langle \pi_2 \circ j, \ni \circ \langle \pi_1 \circ j, \pi_2 \circ j \rangle \rangle & (\text{rel. } (\mathcal{P}(A) \times A)|(\subset \circ (1 \times 1_A^\bullet)))
\end{array}$$

### 1.2.5 La bijection $\Sigma_{\langle g \rangle_Y}$ et la flèche $\ni$

Soit $\langle g \rangle_Y$ un objet de $\mathcal{T}/A$. Nous cherchons à exprimer l'application $\Sigma_{\langle g \rangle_Y} : (\mathcal{T}/A)(\langle f \rangle_X \times \langle g \rangle_Y, \Omega)) \to (\mathcal{T}/A)(\langle f \rangle_X, \mathcal{P}(\langle g \rangle_Y))$ (bijective naturelle en $\langle f \rangle_X$) dans le langage natif de $\mathcal{T}$. Ainsi, considérons un objet $\langle f \rangle_X$, une flèche $[\varphi]_g : \langle f \rangle_X \times \langle g \rangle_Y \longrightarrow \Omega$. Notons $\varphi'$ la flèche de $\mathcal{T}$ représentant la flèche $\Sigma_{\langle g \rangle_Y}([\varphi]_g)$ de $\mathcal{T}/A$. Il s'agit de construire $\varphi'$. Dans $\mathcal{T}$, on devra avoir les deux triangles commutatifs ci-dessous :

$$\begin{tikzcd}
X \,{}_f\!\times_g Y \arrow[rr, "\varphi"] \arrow[dr, "f \circ \pi_1 \circ j"'] & & \Gamma_1 \arrow[dl, "\psi_1"] \\
& A &
\end{tikzcd}
\qquad
\begin{tikzcd}
X \arrow[rr, "\varphi'"] \arrow[dr, "f"'] & & \Gamma_g \arrow[dl, "\psi_g"] \\
& A &
\end{tikzcd}$$

$\varphi$ s'applique (ensemblistement) à des couples $(x, y)$ tels que $f(x) = g(y)$, et envoie un tel couple sur un élément qui est dans la fibre de $\psi_1$ au dessus de $f(x)$, c'est-à-dire sur un couple de la forme $(S, f(x))$ avec $S \subset 1^\bullet_A(f(x))$. On peut donc définir $\varphi'(x)$ comme le couple $(S'(x), f(x))$ où $S'(x)$ est l'ensemble des $y \in Y$ tels que $\varphi(x, y)$ soit « vrai », c'est-à-dire $\varphi(x, y) = (\{f(x)\}, f(x))$. On aura donc :

$$\varphi'(x) = (\{y \in Y \mid f(x) = g(y) \wedge \varphi(x, y) = (\{f(x)\}, f(x))\}, f(x))$$

Notez que la conjonction ci-dessus est une conjonction dépendante, car $\varphi$ n'est définie que sur les couples $(x, y)$ qui vérifient $f(x) = g(y)$. De plus, le type de l'expression ci-dessus n'est pas $\Gamma_g$, mais $\mathcal{P}(Y) \times A$. Il faut donc encore relever cette flèche le long de $j : \Gamma_g \longrightarrow \mathcal{P}(Y) \times A$. En définitive, l'expression de $\varphi'$ est :

$$\varphi' = \lfloor \langle \Sigma_Y(f \circ \pi_1 \asymp g \circ \pi_2 \vec{\wedge} \varphi \asymp \langle \Sigma_A(\asymp) \circ f \circ \pi_1 \circ j, f \circ \pi_1 \circ j \rangle), f \rangle \rfloor_{(x \in X)} | \subset \circ (1 \times g^\bullet_Y)$$

On a donc :

$$\Sigma_{\langle g \rangle_Y}([\varphi]_{\pi_2 \circ j}) = [\langle \Sigma_Y(\asymp \circ (f \times g) \vec{\wedge} \varphi \asymp \langle \Sigma_A(\asymp) \circ f \circ \pi_1 \circ j, f \circ \pi_1 \circ j \rangle), f \rangle | \subset \circ (1 \times g^\bullet_Y)]_{\pi_2 \circ j} \quad (\text{rel. } \langle f \rangle_X)$$

La réciproque $\Theta$ de la correspondance $\varphi \mapsto \varphi'$ est (ensemblistement) donnée par la condition que la « valeur de vérité » $\varphi(x, y)$ signifie $y \in S'(x)$. On va donc poser $\varphi(x, y) = (y \in (\pi_1 \circ j)(\varphi'(x)))$, mais cette expression dans le langage interne est de type $\Omega$ et non de type $\Gamma_1$. On va donc plutôt considérer l'application envoyant $(x, y)$ sur le couple $(f(x), y \in (\pi_1 \circ j)(\varphi'(x)))$ appartenant à $A \times \Omega$, et composer avec l'isomorphisme $i : A \times \Omega \longrightarrow \Gamma_1$. on posera donc :

$$\begin{array}{rcl}
\Theta(\varphi') & = & i \circ \lfloor (f \circ \pi_1 \circ j[z], \pi_2 \circ j[z] \in (\pi_1 \circ j \circ \varphi' \circ \pi_1 \circ j[z])) \rfloor_{(z \in X \,{}_f\!\times_g Y)} \\
& = & i \circ \langle f \circ \pi_1 \circ j, \ni \circ \langle \pi_1 \circ j \circ \varphi' \circ \pi_1 \circ j, \pi_2 \circ j \rangle \rangle
\end{array}$$

La démonstration, en raisonnant intuitionnistiquement dans le langage interne, du fait que $\Theta$ est bien l'inverse de la correspondance $\varphi \mapsto \varphi'$ est immédiate et n'est rien d'autre que l'expression de notre intuition ensembliste. On a donc :

$$\Sigma^{-1}_{\langle g \rangle_Y}([\varphi']_{\pi_2 \circ j}) = [i \circ \langle f \circ \pi_1 \circ j, \ni \circ \langle \pi_1 \circ j \circ \varphi' \circ \pi_1 \circ j, \pi_2 \circ j \rangle \rangle]_{\pi_2 \circ j} \quad (\text{rel. } \langle f \rangle_X \times \langle g \rangle_Y)$$

La flèche $\ni : \mathcal{P}(\langle g \rangle_Y) \times \langle g \rangle_Y \longrightarrow \Omega$ n'est autre que $\Sigma^{-1}_{\langle g \rangle_Y}(1)$, et on a :

$$\ni = [i \circ \langle \pi_2 \circ j \circ \pi_1 \circ j, \ni \circ \langle \pi_1 \circ j \circ \pi_1 \circ j, \pi_2 \rangle \rangle \circ j]_{\pi_2 \circ j} \quad (\text{rel. } \langle \pi_2 \circ j \rangle_{(\mathcal{P}(Y) \times A) | \subset \circ (1 \times g^\bullet_Y)} \times \langle g \rangle_Y)$$

La naturalité de la bijection $\Sigma_{\langle g \rangle_Y}$ par rapport à $\langle f \rangle_X$ signifie qu'on a l'égalité suivante dans $\mathcal{T}/A$ pour toute flèche $[u]_f : \langle h \rangle_Z \longrightarrow \langle f \rangle_X$ :

$$\Sigma_{\langle g \rangle_Y}([\varphi]_{\pi_2 \circ j} \circ ([u]_f \times 1)) = \Sigma_{\langle g \rangle_Y}([\varphi]_{\pi_2 \circ j}) \circ [u]_f$$

Un calcul un peu fastidieux démontre cette égalité :

$$
\begin{aligned}
&\Sigma_{\langle g\rangle_Y}([\varphi]_{\pi_2\circ j})\circ[u]_f \\
&\quad= [\langle\Sigma_Y(\asymp\circ(f\times g)\ \vec{\barwedge}\ \varphi\asymp\langle\Sigma_A(\asymp)\circ f\circ\pi_1\circ j, f\circ\pi_1\circ j\rangle), f\rangle|\subset\circ(1\times g^\bullet_Y)]_{\pi_2\circ j}\circ[u]_f \\
&\quad= [\langle\Sigma_Y(\asymp\circ(f\times g)\ \vec{\barwedge}\ \varphi\asymp\langle\Sigma_A(\asymp)\circ f\circ\pi_1\circ j, f\circ\pi_1\circ j\rangle), f\rangle\circ u|\subset\circ(1\times g^\bullet_Y)]_{\pi_2\circ j} \\
&\quad= [\langle\Sigma_Y(\asymp\circ(f\times g)\circ(u\times 1)\ \vec{\barwedge} \\
&\qquad\qquad \varphi\circ((u\times 1)\circ j)|(\asymp\circ(f\times g)) \\
&\qquad\qquad\qquad \asymp\langle\Sigma_A(\asymp)\circ f\circ\pi_1\circ j\circ((u\times 1)\circ j)|(\asymp\circ(f\times g)), \\
&\qquad\qquad\qquad\qquad f\circ\pi_1\circ j\circ((u\times 1)\circ j)|(\asymp\circ(f\times g))\rangle), f\circ u\rangle|\subset\circ(1\times g^\bullet_Y)]_{\pi_2\circ j}
\end{aligned}
$$

En remarquant que $[u]_f\times 1=[((u\times 1)\circ j)|(\asymp\circ(f\times g))]_{f\circ\pi_1\circ j}$ (rel. $\langle h\rangle_Z\times\langle g\rangle_Y$), on a :

$$
\begin{aligned}
&\Sigma_{\langle g\rangle_Y}([\varphi]_{\pi_2\circ j})\circ[u]_f \\
&\quad= [\langle\Sigma_Y(\asymp\circ((f\circ u)\times g)\ \vec{\barwedge} \\
&\qquad\qquad \varphi\circ((u\times 1)\circ j)|(\asymp\circ(f\times g)) \\
&\qquad\qquad\qquad \asymp\langle\Sigma_A(\asymp)\circ f\circ u\circ\pi_1\circ j, \\
&\qquad\qquad\qquad\qquad f\circ u\circ\pi_1\circ j\rangle), f\circ u\rangle|\subset\circ(1\times g^\bullet_Y)]_{\pi_2\circ j} \\
&\quad= \Sigma_{\langle g\rangle_Y}([\varphi]_{\pi_2\circ j}\circ([u]_f\times 1))
\end{aligned}
$$

### 1.2.6 Les foncteurs $f^*$ et $\Pi_f$

Soit $f : A\longrightarrow B$ une flèche de $\mathcal{T}$. D'après le théorème fondamental, on a la paire $f^*\dashv\Pi_f$ de foncteurs adjoints :

$$
\begin{tikzcd}
\mathcal{T}/A \arrow[r, bend right, "\Pi_f"'] & \mathcal{T}/B \arrow[l, bend right, "f^*"']
\end{tikzcd}
$$

On va expliciter ces deux foncteurs dans notre langage non ambigu, comme résultat de la compilation d'expressions du langage interne de $\mathcal{T}$. Soit $\langle\varphi\rangle_Y$ un objet de $\mathcal{T}/B$, autrement-dit, on a la flèche $\varphi : Y\longrightarrow B$ dans $\mathcal{T}$. On peut considérer le pullback de $\varphi$ le long de $f$. On a donc le carré cartésien :

$$
\begin{tikzcd}
A\ {}_f\!\times_\varphi Y \arrow[r, "\pi_2\circ j"] \arrow[d, "\pi_1\circ j"'] & Y \arrow[d, "\varphi"] \\
A \arrow[r, "f"'] & B
\end{tikzcd}
$$

Par définition, l'objet image de $\langle\varphi\rangle_Y$ par le foncteur $f^*$ est $\langle\pi_1\rangle_{A\ {}_f\times_\varphi Y}$. Autrement-dit, le foncteur de changement de base opère par pullback. Si $\langle\varphi'\rangle_{Y'}$ est un autre objet de $\mathcal{C}/B$. Notons respectivement $X$ et $X'$ les objets $A\ {}_f\!\times_\varphi Y$ et $A\ {}_f\!\times_{\varphi'} Y'$ de $\mathcal{T}$. Soit $[\lambda]_{\varphi'} : \langle\varphi\rangle_Y\longrightarrow\langle\varphi'\rangle_{Y'}$ une flèche de $\mathcal{C}/B$, on a $f^*([\lambda]_{\varphi'})=[\mu]_{\pi_1}$, où $\mu : X\longrightarrow X'$ est, par définition, l'unique flèche telle que le diagramme suivant soit commutatif :

$$
\begin{tikzcd}
& X' \arrow[rr, "\pi_2\circ j"] \arrow[dd, "\pi_1\circ j"' near end] & & Y' \arrow[dd, "\varphi'"] \\
X \arrow[ur, "\mu"] \arrow[rr, "\pi_2\circ j" near end] \arrow[dr, "\pi_1\circ j"'] & & Y \arrow[ur, "\lambda"] \arrow[dr, "\varphi"'] & \\
& A \arrow[rr, "f"'] & & B
\end{tikzcd}
$$

L'existence et l'unicité de $\mu$ résultent bien sûr des propriétés des carrés cartésiens. Ensemblistement, la flèche $\mu$ s'écrit $(a,y)\mapsto(a,\lambda(y))$. En fait, $\mu$ ne fait que reproduire au dessus de $a$ ce que fait $\lambda$ au dessus de $f(a)$. C'est l'essence même de la notion de pullback.

On peut donc maintenant écrire les formules qui expriment explicitement le foncteur $f^*$. On a :

- $f^*(\langle\varphi\rangle_Y)=\langle\pi_1\circ j\rangle_{A\ {}_f\times_\varphi Y}$

- $f^*([\lambda]_{\varphi'}) = [\langle \pi_1 \circ j, \lambda \circ \pi_2 \circ j\rangle_{(f,\varphi')}]_{\pi_1 \circ j}$ (rel. $\langle \pi_1 \circ j\rangle_X$)

Nous allons maintenant exprimer l'adjoint à droite $\Pi_f$ du foncteur $f^*$, et d'abord essayer de le comprendre ensemblistement. À tout objet $\langle\varphi\rangle_X$ de $\mathcal{T}/A$ on doit associer un objet $\Pi_f(\langle\varphi\rangle_X)$ de $\mathcal{T}/B$, objet que nous noterons $\langle\psi\rangle_Y$. On a donc les flèches suivantes dans $\mathcal{T}$ :

$$\begin{tikzcd} X \arrow[d, "\varphi"'] & Y \arrow[d, "\psi"] \\ A \arrow[r, "f"'] & B \end{tikzcd}$$

Il y a donc juste à définir $\psi$ en fonction de $\varphi$. On connait la cible de $\psi$ qui est $B$, mais on ne connait pas sa source $Y$, qu'on doit donc aussi construire. On va voir $\psi$ comme un fibré, ce qui veut dire qu'il suffit de construire, pour tout élément $b \in B$ l'image réciproque $\psi^\bullet_Y(b)$ de $b$ par $\psi$. Cette image réciproque sera le produit d'ensembles suivant (ce qui justifie la notation $\Pi_f$ pour notre foncteur) :

$$\psi^\bullet_Y(b) = \prod_{f(a)=b} \varphi^\bullet_X(a)$$

Toutefois, une façon plus intuitive (et plus utile dans cette situation) de voir cet ensemble est de le voir comme l'ensemble des sections partielles de $\varphi$ au dessus de $f^\bullet_A(b)$. En effet, une telle section associe un élément de $\varphi^\bullet_X(a)$ à tout $a \in A$ tel que $f(a) = b$. Ceci permet de voir chaque élément de $\psi^\bullet_Y(b)$ comme une relation entre $A$ et $X$, donc comme un élément de $\mathcal{P}(A \times X)$. Évidemment, la donnée d'un tel ensemble de sections partielles ne détermine pas $b$. On doit donc construire $Y$ comme un ensemble de couples de la forme $(b, s)$ où $s$ est une telle section partielle. On a donc ensemblistement :

$$Y = \{(b, s) \in B \times \mathcal{P}(A \times X) \mid s \text{ est une section de } \varphi \text{ au dessus de } f^\bullet_A(b)\}$$

Bien sûr $\psi$ sera juste la première projection. La condition que $s \in \mathcal{P}(A \times X)$ soit une section de $\varphi$ au dessus de $f^\bullet_A(b)$ s'exprime comme ceci :

$$(\forall_{(a,x)\in s}\ f(a) = b \wedge \varphi(x) = a) \wedge (\forall_{a\in A}\ f(a) = b \Rightarrow \exists!_{x\in X}\ (a, x) \in s)$$

Autrement-dit, la relation $s$ est définie sur $f^\bullet_A(b)$ (car $f(a) = b$) et c'est bien une section (car $\varphi(x) = a$). De plus elle est fonctionnelle sur $f^\bullet_A(b)$ (car $f(a) = b \Rightarrow \exists!_{x\in X}\ (a, x) \in s$).

Formalisons maintenant ceci dans le langage natif de $\mathcal{T}$. On se sert du langage interne pour construire le monomorphisme $j : Y \longrightarrow B \times \mathcal{P}(A \times X)$ comme un représentant du sous-objet de $B \times \mathcal{P}(A \times X)$ de flèche caractéristique :([11])

$$\lfloor(\forall_{a\in A}\ \forall_{x\in X}\ (a, x) \in s \Rightarrow (f(a) = b \wedge \varphi(x) = a)) \wedge (\forall_{a\in A}\ f(a) = b \Rightarrow \exists!_{x\in X}\ (a, x) \in s)\rfloor_{(b\in B)(s\in\mathcal{P}(A\times X))}$$

flèche que nous noterons $m_{\varphi,f}$. On a donc :

$$\Pi_f(\langle\varphi\rangle_X) = \langle\pi_1 \circ j\rangle_{(B\times\mathcal{P}(A\times X))|m_{\varphi,f}}$$

Soit maintenant $\langle\xi\rangle_Z$ un objet quelconque de $\mathcal{T}/B$ et $[\lambda]_\psi : \langle\xi\rangle_Z \longrightarrow \Pi_f(\langle\varphi\rangle_X)$ une flèche de $\mathcal{T}/B$. La flèche $\lambda$ (de $\mathcal{T}$) associe à tout $z \in Z$ une section de $\varphi$ au dessus de $f^\bullet_A(\xi(z))$. On peut décurryfier $\lambda$ (et permuter les facteurs du produit), ce qui donne une application $\mu$ qui envoie le couple $(a, z) \in A \times Z$ sur l'élément $\lambda(z)(a)$ de $X$, mais qui n'est définie que pour les $a$ tels que $f(a) = \xi(z)$. L'application $\mu$ est

11. Un début de compilation de cette formule donne :

$$\begin{aligned}&\forall_A(\forall_X(\pi_2 \circ \pi_1 \in \pi_2 \Rightarrow (f \circ \pi_2 \circ \pi_1 \asymp \pi_1 \circ \pi_1 \circ \pi_1 \wedge \varphi \circ \pi_2 \asymp \pi_2 \circ \pi_1)))\\ &\qquad\wedge\forall_A(f \circ \pi_2 \asymp \pi_2 \circ \pi_1 \Rightarrow \exists!_X(\langle\pi_2 \circ \pi_1, \pi_2\rangle \in \pi_2 \circ \pi_1 \circ \pi_1))\end{aligned}$$

donc définie sur le produit fibré $A\,{}_f\!\times_\xi Z$. De plus l'élément $\mu(a,z)$ est projeté sur $a$ par $\varphi$, puisque $\lambda(z)$ est une section (partielle) de $\varphi$. On a donc $\varphi(\mu(a,z)) = a$, c'est-à-dire le diagramme commutatif :

$$\begin{tikzcd}
A\,{}_f\!\times_\xi Z \arrow[rr, "\mu"] \arrow[dr, "\pi_1\circ j"'] & & X \arrow[dl, "\varphi"] \\
& A &
\end{tikzcd}$$

et on voit qu'on a une flèche $[\mu]_\varphi : f^*(\langle\xi\rangle_Z) \longrightarrow \langle\varphi\rangle_X$ dans $\mathcal{T}/A$.

Catégoriquement, on se sert à nouveau du langage interne. On obtient la flèche $k : A\,{}_f\!\times_\xi Z \longrightarrow \mathcal{P}(X)$ en envoyant $(a,z)$ sur la partie de $X$ dont les éléments vérifient la condition souhaitée. Mais nous savons qu'il n'existe qu'un tel élément, cette partie est donc toujours un singleton et nous sommes en mesure de composer avec $\delta$ :

$$\begin{tikzcd}
X & \mathcal{P}(X) \mid \sigma \arrow[l, "\delta"'] \arrow[d, "j"] & \mathbf{1} \arrow[d, "\top"] \\
A\,{}_f\!\times_\xi Z \arrow[r, "k"'] \arrow[ur, "k|\sigma"] & \mathcal{P}(X) \arrow[r, "\sigma"'] & \Omega
\end{tikzcd}$$

$$k = \lfloor\{x \in X \mid (\pi_1\circ j(c), x) \in \pi_2\circ j\circ\lambda\circ\pi_2\circ j(c)\}\rfloor_{c\in A\,{}_f\!\times_\xi Z}$$

$$\Theta_{\langle\xi\rangle_Z}([\lambda]_{\pi_1\circ j}) = [\delta\circ\lfloor\{x \in X \mid (\pi_1\circ j(c), x) \in \pi_2\circ j\circ\lambda\circ\pi_2\circ j(c)\}\rfloor_{c\in A\,{}_f\!\times_\xi Z}|\sigma]_\varphi \ (\text{rel. } \langle\pi_1\circ j\rangle_{A\,{}_f\!\times_\xi Z})$$

Pour voir que la correspondance $\Theta_{\langle\xi\rangle_Z}$ :

$$\begin{tikzcd}
{[\lambda]_\psi} \arrow[r, mapsto] & {[\mu]_\varphi} \\
(\mathcal{T}/B)(\langle\xi\rangle_Z, \Pi_f(\langle\varphi\rangle_X)) \arrow[r, "\Theta_{\langle\xi\rangle_Z}"'] & (\mathcal{T}/A)(f^*(\langle\xi\rangle_Z), \langle\varphi\rangle_X)
\end{tikzcd}$$

est bijective, on va exhiber son inverse. Donnons-nous donc $[\mu]_\varphi : f^*(\langle\xi\rangle_Z) \longrightarrow \langle\varphi\rangle_X$, c'est-à-dire $\mu : A\,{}_f\!\times_\xi Z \longrightarrow X$ telle que $\varphi\circ\mu = \pi_1\circ j$. Ensemblistement, $\mu$ associe à toute paire $(a,z)$ telle que $f(a) = \xi(z)$ un élément $\mu(a,z)$ de $X$ tel que $\varphi(\mu(a,z)) = a$. En curryfiant $\mu$ (à la permutation près des facteurs du produit), on obtient une application $\lambda$ qui à tout $z \in Z$ associe la fonction $a \mapsto \mu(a,z)$, mais définie seulement sur les $a$ tels que $f(a) = \xi(z)$, autrement-dit, définie sur $f_A^\bullet(\xi(z))$. Comme $\varphi(\mu(a,z)) = a$ on voit que cette fonction est une section de $\varphi$ au dessus de $f_A^\bullet(\xi(z))$. Il est clair qu'on vient de construire l'inverse de $\Theta_{\langle\xi\rangle_Z}$.

Comme précédemment, nous allons nous servir du langage interne pour obtenir la flèche $q : X \longrightarrow B\times\mathcal{P}(A\times X)$ agissant exactement comme $\lambda$ (il suffit d'expliciter l'écriture ensembliste de $\lambda$) puis nous restreignons son ensemble d'arrivée $(B\times\mathcal{P}(A\times X))|m_{\varphi,f}$ grâce au diagramme suivant :

$$\begin{tikzcd}
& (B\times\mathcal{P}(A\times X))|m_{\varphi,f} \arrow[d, "j"] & \mathbf{1} \arrow[d, "\top"] \\
Z \arrow[r, "q"'] \arrow[ur, "q|m_{\varphi,f}"] & B\times\mathcal{P}(A\times X) \arrow[r, "m_{\varphi,f}"'] & \Omega
\end{tikzcd}$$

$$q = \lfloor(\xi(z), \{s \in A\times X \mid f\circ\pi_1(s) = \xi(z) \wedge \exists_{c\in A\,{}_f\!\times_\xi Z}(\pi_1\circ j[c] = \pi_1(s) \wedge \pi_2\circ j[c] = z \wedge \pi_2(s) = \mu[c])\})\rfloor_{(z\in Z)}$$

$$\Theta^{-1}_{\langle\xi\rangle_Z}([\mu]_\psi) = [\lfloor(\xi(z), \{s \in A\times X \mid f\circ\pi_1(s) = \xi(z) \wedge \exists_{c\in A\,{}_f\!\times_\xi Z}(\pi_1\circ j[c] = \pi_1(s) \wedge \pi_2\circ j[c] = z \wedge \pi_2(s) = \mu[c])\})\rfloor_{(z\in Z)}|m_{\varphi,f}]_{\pi_1\circ j}$$

$$(\text{rel. } \langle\xi\rangle_Z)$$

Voyons maintenant pourquoi $\Theta_{\langle\xi\rangle_Z}$ est naturelle en $\langle\xi\rangle_Z$. Donnons-nous une flèche $[u]_\xi : \langle\zeta\rangle_U \longrightarrow \langle\xi\rangle_Z$. Il s'agit de montrer que $\Theta_{\langle\zeta\rangle_U}([\lambda]_{\pi_1\circ j}\circ[u]_\xi) = \Theta_{\langle\xi\rangle_Z}([\lambda]_{\pi_1\circ j})\circ(1\times[u]_\xi)$. Remarquons ici une légère erreur : le représentant de $1\times[u]_\xi$ a pour but $A\times Z$ et non $A\;{}_f{\times}_\xi\;Z$. Nous nous permettons cette approximation car nous allons comprendre la naturalité du point du vue ensembliste (qui suffit ici). Le côté droit de l'égalité envoie un couple $(a,c)$ de $A\times U$ sur l'unique $x$ de $X$ tel que $(a,x)\in\pi_2(\lambda(u(c))$, c'est précisément, par définition de l'action de $\Theta$, ce que fait le côté gauche de l'égalité.

$\Pi_f$ ainsi défini sur les objets se prolonge automatiquement en un foncteur adjoint à droite de $f^*$. En effet, son action sur une flèche $[u]_\varphi : \langle\xi\rangle_Z \longrightarrow \langle\varphi\rangle_X$ s'exprime à l'aide de $\Theta^{-1}_{\Pi_f(\langle\xi\rangle_Z)}$ et de la co-unité $\varepsilon$ de l'adjonction que nous sommes en train de construire par la formule $\Pi_f([u]_\varphi) = \Theta^{-1}_{\Pi_f(\langle\xi\rangle_Z)}([u]_\varphi\circ\varepsilon_{\langle\xi\rangle_Z})$. Mais la co-unité $\varepsilon_{\langle\xi\rangle_Z}$ est elle-même $\Theta_{\Pi_f(\langle\xi\rangle_Z)}(1)$, où 1 est la flèche identité de $\Pi_f(\langle\xi\rangle_Z)$. On a donc :

$$\Pi_f([u]_\varphi) = \Theta^{-1}_{\Pi_f(\langle\xi\rangle_Z)}([u]_\varphi\circ\Theta_{\Pi_f(\langle\xi\rangle_Z)}(1))$$

### 1.2.7 Résumé

Les opérations sont à présent traduites dans le langage natif. Notons qu'elles agissent évidemment sur des flèches et objets des topos *relatifs*. Or, s'il est pertinent de comprendre le fonctionnement du langage $W$ à travers ces derniers, son interprétation se déroule *dans le topos initial*. Ainsi, ce ne sont pas directement ces opérations qui nous intéressent, mais les applications qui agissent *identiquement* sur les *représentants* des flèches et objets de $\mathcal{T}/A$ dans $\mathcal{T}$. Par exemple, pour toute flèche $g : Y \longrightarrow A$, nous souhaitons définir $\overline{\mathcal{P}}(g)$ qui envoie $g$ sur le représentant de $\mathcal{P}(\langle g\rangle_Y)$.

Nous avons déjà effectué le plus gros du travail en traduisant les opérations des topos relatifs, il nous faut néanmoins faire très attention, en définissant leurs « représentantes » dans $\mathcal{T}$ à conserver des notations non ambiguës. En effet, s'il y a clairement une bijection entre les objets de $\mathcal{T}/A$ et les flèches de $\mathcal{T}$ de cible $A$, ce n'est plus le cas pour les flèches de $\mathcal{T}/A$ (deux flèches peuvent avoir le même représentant et certaines flèches de $\mathcal{T}$ ne représentent aucune flèche de $\mathcal{T}/A$). Lorsque l'argument d'une application qui agit dans un topos relatif est $[\varphi]_g$, nous lui fournissons $g$ en information. Il sera donc peut-être nécessaire que sa représentante du topos $\mathcal{T}$ prenne deux arguments : $g$ et $\varphi$.

Il existe deux façons différentes de construire ces applications. Nous pouvons les définir en fonction des applications des topos relatifs en se servant d'une première fonction (appelons-la $H$) qui envoie les flèches et objets sur leurs représentants et une autre ($H'$) qui envoie une ou deux flèches de $\mathcal{T}$ (selon ce qu'il est nécessaire d'indiquer) sur l'objet ou la flèche qu'elle représente. Ainsi nous aurions par exemple : $\overline{\mathcal{P}}(g) = H(\mathcal{P}(H'(g))$. Il faudrait cependant définir clairement ces fonctions, établir notamment ce qu'elles doivent prendre en argument. L'autre méthode, que nous allons utiliser ici, consiste à ne faire aucun détour par les topos relatifs. Nous nous servirons bien entendu des réécritures effectuées, *mais sans jamais solliciter un objet ou une flèche des topos relatifs*. Nous écrirons donc directement $\overline{\mathcal{P}}(g) = \pi_2\circ j$ en vérifiant, là aussi, que l'écriture est non ambiguë (les deux méthodes sont donc équivalentes, exigeant la même prudence).

Informatiquement, ces opérations ne sont que des macros. Pour conserver le caractère non ambigu du langage, il est donc primordial (et suffisant) que toute variable d'objet ou de flèche présente dans le développement de la macro soit retrouvable à partir de la source et de la macro elle-même. Il n'est pas nécessaire de demander de pouvoir retrouver le but. Si la macro respecte ces conditions, elle est correctement construite et le langage se charge de faire le reste. C'est aux termes primitifs qu'il fallait imposer cette condition, notre macro étant écrite à partir de ceux-ci, elle la respecte aussi. Le tableau page suivante les présente toutes :

$$\overline{\mathcal{P}}(g,Y) = \pi_2 \circ j$$
$$(\text{rel. } (\mathcal{P}(Y) \times A) | ( \subset \circ (1 \times g_Y^{\bullet}))),\ g : Y \longrightarrow A$$

---

$$\overline{\Sigma}_{(g,Y)}(\varphi) = \langle \Sigma_Y( \asymp \circ (f \times g) \vec{\wedge}\, \varphi \asymp \langle \Sigma_A( \asymp ) \circ f \circ \pi_1 \circ j, f \circ \pi_1 \circ j \rangle), f \rangle | \subset \circ (1 \times g_Y^{\bullet})$$
$$(\text{rel. } Y),\ g : Y \longrightarrow A,\ \varphi : X \longrightarrow Y \ (\text{et } f = g \circ \varphi)$$

---

$$\overline{\ni} = i \circ \langle \pi_2 \circ j \circ \pi_1 \circ j,\ \ni \circ \langle \pi_1 \circ j \circ \pi_1 \circ j, \pi_2 \rangle \rangle \circ j$$
$$(\text{rel. } (\mathcal{P}(Y) \times A) | \subset \circ (1 \times g_Y^{\bullet})\,{}_{\pi_2 \circ j}{\times}_g\, Y)$$

---

$$\overline{f}^*(\varphi, Y) = \pi_1 \circ j$$
$$(\text{rel. } A\,{}_f{\times}_\varphi\, Y),\ \varphi : Y \longrightarrow A,\ f : A \longrightarrow B$$

---

$$\overline{f}^*(\lambda, \varphi) = \langle \pi_1 \circ j, \lambda \circ \pi_2 \circ j \rangle_{(\varphi \circ \lambda, \varphi)}$$
$$(\text{rel. } A\,{}_f{\times}_{\varphi\circ\lambda}\, Y),\ f : A \longrightarrow B,\ \varphi : Y' \longrightarrow A,\ \lambda : Y \longrightarrow Y'$$

---

$$\overline{\Pi}_f(\varphi, X) = \pi_1 \circ j$$
$$(\text{rel. } (B \times \mathcal{P}(A \times X)) | \lfloor (\forall_{a\in A}\ \forall_{x \in X}\ (a,x) \in s \Rightarrow (f(a) = b \wedge \varphi(x) = a))$$
$$\wedge (\forall_{a \in A}\ f(a) = b \Rightarrow \exists!_{x\in X}\ (a,x) \in s) \rfloor_{(b\in B)(s \in \mathcal{P}(A\times X))})$$
$$\varphi : X \longrightarrow A,\ f : A \longrightarrow B$$

---

$$\overline{\varepsilon}(\,^{12}) = \delta \circ \lfloor \{ z \in Z \mid (\pi_1 \circ j(c), z) \in \pi_2 \circ j \circ \pi_2 \circ j(c) \} \rfloor$$
$${}_{c \in A\,{}_f{\times}_{\pi_1\circ j}(B \times \mathcal{P}(A\times X))\ |\ \lfloor (\forall_{a\in A}\ \forall_{x\in X}\ (a,x)\in s \Rightarrow (f(a)=b \wedge \xi(x)=a)) \wedge (\forall_{a\in A}\ f(a)=b \Rightarrow \exists!_{x\in X}\ (a,x)\in s) \rfloor_{(b\in B)(s\in\mathcal{P}(A\times X))}} \mid \sigma$$
$$(\text{rel. } A\,{}_f{\times}_{\pi_1\circ j}\ ((B \times \mathcal{P}(A \times X))$$
$$| \lfloor (\forall_{a\in A}\ \forall_{x\in X}\ (a,x) \in s \Rightarrow (f(a) = b \wedge \xi(x) = a)) \wedge (\forall_{a\in A}\ f(a) = b \Rightarrow \exists!_{x\in X}\ (a,x) \in s) \rfloor_{(b\in B)(s\in\mathcal{P}(A\times X))}))$$

---

$$\overline{\Pi}_f(u, \varphi) = \lfloor (\pi_1 \circ j(m), \{ s \in A \times X \mid f \circ \pi_1(s) = \pi \circ j(m) \wedge \exists_{c \in A\,{}_f{\times}_{\pi_1\circ j}(B\times\mathcal{P}(A\times X))}$$
$${}_{|\ \lfloor (\forall_{a\in A}\ \forall_{x\in X}\ (a,x)\in s \Rightarrow (f(a)=b\wedge\xi(x)=a)) \wedge (\forall_{a\in A}\ f(a)=b\Rightarrow\exists!_{x\in X}\ (a,x)\in s) \rfloor_{(b\in B)(s\in\mathcal{P}(A\times X))}} (\pi_1 \circ j[c] =$$
$$\pi_1(s) \wedge \pi_2 \circ j[c] = m \wedge \pi_2(s) = u \circ (\delta \circ \lfloor \{ z \in Z \mid (\pi_1 \circ j(c), z) \in$$
$$\pi_2 \circ j \circ \pi_2 \circ j(c) \} \rfloor_{c \in A\,{}_f{\times}_{\pi_1\circ j}(B\times\mathcal{P}(A\times X))\ |\ \lfloor (\forall_{a\in A}\ \forall_{x\in X}\ (a,x)\in s\Rightarrow(f(a)=b\wedge\xi(x)=a)) \rfloor}$$
$${}_{\wedge (\forall_{a\in A}\ f(a)=b\Rightarrow\exists!_{x\in X}\ (a,x)\in s)_{(b\in B)(s\in\mathcal{P}(A\times X))}} \mid \sigma)[c]) \}) \rfloor$$
$${}_{(m \in (B\times\mathcal{P}(A\times X))\ |\ \lfloor (\forall_{a\in A}\ \forall_{x\in X}\ (a,x)\in s\Rightarrow(f(a)=b\wedge\xi(x)=a)) \wedge (\forall_{a\in A}\ f(a)=b\Rightarrow\exists!_{x\in X}\ (a,x)\in s) \rfloor_{(b\in B)(s\in\mathcal{P}(A\times X))}})$$
$$| \lfloor (\forall_{a\in A}\ \forall_{x\in X}\ (a,x) \in s \Rightarrow (f(a) = b \wedge \varphi(x) = a)) \wedge (\forall_{a\in A}\ f(a) = b \Rightarrow \exists!_{x\in X}\ (a,x) \in s) \rfloor_{(b\in B)(s\in\mathcal{P}(A\times X))}$$
$$(\text{rel. } ((B \times \mathcal{P}(A \times X)) \mid \lfloor (\forall_{a\in A}\ \forall_{x\in X}\ (a,x) \in s \Rightarrow (f(a) = b \wedge \xi(x) = a)) \wedge (\forall_{a\in A}\ f(a) = b \Rightarrow$$
$$\exists!_{x\in X}\ (a,x) \in s) \rfloor_{(b\in B)(s\in\mathcal{P}(A\times X))}))$$
$$f : A \longrightarrow B,\ u : Z \longrightarrow X,\ \varphi : X \longrightarrow A\ (\text{et } \xi = \phi \circ u)$$

---

12. $\overline{\varepsilon}$ représente la co-unité de l'adjonction entre le foncteur de changement de base et son adjoint à droite. Nous en aurons besoin pour interpréter $W$.

# Chapitre 2

# Le langage $W$

## 2.1 Syntaxe du langage $W$

Le langage $W$([1]) introduit les notions de « contexte », de « type » et de « terme », plus une notion d'égalité s'appliquant aux types et aux termes. Toutes ces notions sont définies simultanément par une récursion croisée, la définition du langage $W$ étant donnée sous forme d'un ensemble de règles permettant d'émettre des « jugements » valides à propos des éléments du langage. Les jugements, c'est-à-dire les méta-énoncés qui nous permettent d'exprimer des propriétés des éléments du langage, sont des sortes suivantes :

| Notation du jugement | Façon de le lire |
|---|---|
| $/\Gamma$ | « $\Gamma$ est un contexte » |
| $T/\Gamma$ | « $T$ est un type relativement à $\Gamma$ » |
| $t : T/\Gamma$ | « $t$ est un terme de type $T$ relativement à $\Gamma$ » |
| $T = U/\Gamma$ | « les types $T$ et $U$ sont égaux relativement à $\Gamma$ » |
| $t = u : T/\Gamma$ | « les termes $t$ et $u$, de type $T$, sont égaux relativement à $\Gamma$ » |

Un certain nombre de choses sont sous-entendues, mais sont de toute façon des conséquences des règles qui vont être donnés ci-dessous. Par exemple, si le jugement $t : T/\Gamma$ est valide, alors a fortiori, le jugement $T/\Gamma$ est valide. Autrement-dit, si les règles permettent de juger que $t$ est un terme de type $T$ dans le contexte $\Gamma$, alors elle permettent aussi de juger que $T$ est un type dans le contexte $\Gamma$.

On notera que les règles qui définissent le langage $W$ définissent à la fois ce qu'il est convenu d'appeler la « syntaxe » du langage, le « typage » des termes et la « sémantique équationnelle » aussi bien des types que des termes.

Dans un langage qui ne gère pas la dépendance, comme le langage interne d'un topos, les expressions représentant des types ne contiennent aucune sous-expression qui soit un terme, et donc en particulier n'ont pas d'occurrence libres de variables du langage lui-même.([2]) En conséquence, si le langage ne gère pas la dépendance, il n'y a pas d'inconvénient à considérer que deux types sont égaux si et seulement si leurs écritures sont identiques. Au contraire, si le langage gère la dépendance, la question de l'égalité entre types est moins évidente. Par exemple, si $E$ et $F$ sont deux énoncés équivalents (démontrables l'un à partir de l'autre) dans le contexte $\Gamma$, doit-on considérer que les deux types de garants $W(E)$ et $W(F)$ sont égaux ? La réponse est certainement « non », car si elle était « oui », on devrait considérer que les types

1. L'appellation $W$ provient de la présence des types de la forme $W(E)$, dits « types de garants ». Le choix de la lettre $W$ vient de ce que « garant » se dit « warranter » en Anglais. D'après A. Prouté, la motivation principale qui a mené à la conception de ce langage est le fait, discuté dans l'introduction, que pour tout énoncé $E$ dans le contexte $\Gamma$, les notations des flèches qui sont des sections du pullback $j$ de $\top$ le long de $E$ jouent manifestement le rôle de preuves de $E$.

2. Les variables d'un langage font généralement partie des termes de ce langage.

$W(\top)$ et $W(E)$ sont égaux pour tout énoncé démontrable $E$. Par ailleurs, tout terme de type $W(\top)$, par exemple, une preuve triviale de $\top$, serait aussi un terme de type $W(E)$, puisqu'il s'agit du même type. Mais alors, la preuve triviale de $\top$ serait une preuve de tout énoncé démontrable, ce qui retirerait tout leur intérêt aux preuves. Par ailleurs, une condition importante pour que le système $W$ soit viable sur le plan opératoire([3]) est que l'égalité entre types soit décidable par algorithme. Elle ne doit donc pas dépendre de l'égalité entre termes, sauf peut-être d'une forme faible de l'égalité entre termes testable par algorithme. Une telle forme faible de l'égalité entre termes existe. Elle résulte d'un théorème analogue au théorème de normalisation forte de W. W. Tait pour le $\lambda$-calcul typé, dont la démonstration dépasse bien sûr le cadre de ce mémoire. On appelera cette égalité l'« égalité calculatoire ». On admettra donc que deux types de la forme $W(E)$ et $W(F)$ sont égaux si et seulement si les énoncés $E$ et $F$ sont calculatoirement égaux.

Enfin, on peut facilement prouver par induction à partir des règles définissant $W$, que si $t$ est un terme de type $T$ dans le contexte $\Gamma$, autrement-dit si le jugement $t : T/\Gamma$ est valide, alors le type $T$ peut être retrouvé algorithmiquement à partir de $\Gamma$ et de $t$.([4])

Noter qu'on utilise la notation $t : T$ et non pas $t \in T$ pour signifier que $t$ est de type $T$. La notation $\in$ fait par ailleurs partie du langage $W$ lui-même, comme on va le voir, et dans un terme de la forme $a \in A$ (qui est un énoncé) $A$ n'est pas un type mais un « ensemble », notion qui sera définie plus loin.

Chaque règle de la définition du langage $W$ est constituée d'une conclusion qui est un jugement et d'une liste (éventuellement vide) d'hypothèses qui sont des jugements. De plus, les jugements qui figurent dans les règles contiennent des méta-symboles pouvant représenter des contextes, des types, des termes ou des variables quelconques du langage $W$. Ces méta-symboles sont facilement reconnaissables car ce sont toujours les mêmes, à savoir : $\Gamma$ pour les contextes, $T$ et $U$ pour les types, $E$, $a$, $b$, $c$, $A$ et $p$ pour les termes, $x$ et $y$ pour les variables. Il est sous-entendu que les méta-symboles $x$ et $y$ représentent des variables distinctes du langage $W$.

L'interprétation d'une règle (c'est-à-dire la façon de l'utiliser pour établir des jugements valides) est la suivante : la conclusion de la règle est un jugement valide dès que toutes les hypothèses de la règle sont des jugements valides, et ceci quels que soient les contextes, types, termes et variables représentés par les méta-symboles.([5])

Pour pouvoir écrire les règles qui définissent le langage $W$ sans avoir à écrire des expressions trop compliquées, on utilise une série d'abréviations (définies en cascade), qui sont les suivantes (ces abréviations seront commentées plus loin) :

- $\Omega$ est une abréviation pour $\mathcal{P}(\mathbf{1})$,
- $p \vdash E$ est une abréviation pour $p : W(E)$,
- $E \Rightarrow F$ est une abréviation pour $\forall_{p \vdash E}\ F$,
- $\exists_{x:T}\ E$ est une abréviation pour $\forall_{q:\Omega}\ (\forall_{x:T}\ (E \Rightarrow q)) \Rightarrow q$,
- $E \wedge F$ est une abréviation pour $\exists_{p \vdash E}\ F$.
- $a = b$ est une abréviation pour $\forall_{A:\mathcal{P}(T)}\ a \in A \Rightarrow b \in A$,
- $\exists!_{x:T}\ E$ est une abréviation pour $\exists_{x:T}\ E \wedge \forall_{y:T}\ E[y/x] \Rightarrow x = y$.

où le méta-terme $E[y/x]$ représente le terme obtenu en remplaçant dans $E$ toutes les occurrences libres de $x$ par $y$. Bien entendu, cette notion de remplacement est relativement délicate, mais fait partie des standards de la logique. On supposera que le lecteur n'a pas besoin de plus d'explication.

3. Ceci est un point essentiel pour une réalisation informatique.

4. Autrement-dit, un compilateur du langage $W$ peut inférer le type d'un terme correct dans un contexte donné. En fait, un compilateur réel fera même mieux que cela, car il prendra aussi en charge la surcharge des symboles (polysémie ou polymorphisme simple) et la possibilité d'utiliser des paramètres de types (polymorphisme paramétrique). L'algorithme capable de faire cela est une extension de celui qui est suggéré ici, utilisant les techniques usuelles de l'inférence de types, dont en particulier la technique de l'unification.

5. Ce genre de règle est aussi connue sous le nom de « clause de Horn », et est bien connue des programmeurs Prolog, puisqu'un programme Prolog est essentiellement un ensemble de clauses de Horn.

Remarquer que l'abréviation $a = b$ pour $\forall_{A:\mathcal{P}(T)}\ a \in A \Rightarrow b \in A$([6]) fait disparaître le type $T$, commun aux termes $a$ et $b$, de la notation. Ceci n'est pas grave, puisque comme on l'a dit plus haut, le type d'un terme peut être inféré à partir du contexte. On évitera évidemment de confondre le terme $a = b$ introduit ici avec un jugement de la forme $a = b : T/\Gamma$ dans lequel le signe $=$ figure une méta-égalité (égalité externe) et dans lequel $a = b$ n'est pas un terme.

Le tableau ci-dessous donne les règles concernant les jugements autres que les jugements d'égalité, qui ne seront pas traités ici. On remarquera que « $x$ non déclaré dans $\Gamma$ » est un jugment supplémentaire dont il n'a pas été question plus haut (on peut l'appeler un « jugement primitif »). Il est bien sûr décidable par algorithme. Chacune de ces règles (une par ligne du tableau) est commentée plus loin.

| | Conclusion | Hypothèses |
|---|---|---|
| | $/\emptyset$ | (pas d'hypothèse) |
| | $/\Gamma(x : T)$ | $T/\Gamma$ et $x$ non déclaré dans $\Gamma$ |
| | $x : T/\Gamma(x : T)$ | $T/\Gamma$ |
| | $x : T/\Gamma(y : U)$ | $x : T/\Gamma$ |
| | $\mathbf{1}/\Gamma$ | $/\Gamma$ |
| | $(x : T) \times U/\Gamma$ | $T/\Gamma$ et $U/\Gamma(x : T)$ |
| | $\mathcal{P}(T)/\Gamma$ | $T/\Gamma$ |
| | $W(E)/\Gamma$ | $E : \Omega/\Gamma$ |
| | $* : \mathbf{1}/\Gamma$ | si $/\Gamma$ |
| | $(a, b)_{(x,U)} : (x : T) \times U/\Gamma$ | $a : T/\Gamma$ et $b : U[a/x]/\Gamma$ |
| | $\pi_1(c) : T/\Gamma$ | $c : (x : T) \times U/\Gamma$ |
| | $\pi_2(c) : U[\pi_1(c)/x]/\Gamma$ | $c : (x : T) \times U/\Gamma$ |
| | $\{x : T \mid E\} : \mathcal{P}(T)/\Gamma$ | $E : \Omega/\Gamma(x : T)$ |
| | $a \in A : \Omega/\Gamma$ | $a : T/\Gamma$ et $A : \mathcal{P}(T)/\Gamma$ |
| | $\forall_{x:T}\ E : \Omega/\Gamma$ | $E : \Omega/\Gamma(x : T)$ |
| | $L_{x:T}\ p \vdash \forall_{x:T}\ E/\Gamma$ | $p \vdash E/\Gamma(x : T)$ |
| | $p.a \vdash E[a/x]/\Gamma$ | $p \vdash \forall_{x:T}\ E/\Gamma$ et $a : T/\Gamma$ |
| | $\delta(p) : T/\Gamma$ | $p \vdash \exists!_{x:T}\ E/\Gamma$ |
| | $\rho(p) \vdash E[\delta(p)/x]/\Gamma$ | $p \vdash \exists!_{x:T}\ E/\Gamma$ |

Comme on le voit, cet ensemble de règles est assez restreint, mais bien entendu, il manque des règles introduisant le type $\mathbb{N}$ des entiers naturels et les moyens d'utiliser les entiers, ou mieux encore des règles permettant de définir des type récursifs (entiers, listes, arbres, etc...), et il manque également les règles définissant les égalités entre types et entre termes. Il manque enfin, et ceci est important par exemple pour faire en langage $W$ de la théorie des catégories, des « univers » ou des « type inaccessibles » analogues aux cardinaux inaccessibles. Les questions des types récursifs et des types inaccessibles ne seront pas traitées ici.

Le premier groupe de règles définit les contextes. La première règle dit que $\emptyset$ est un contexte. On l'appelle le « contexte vide ». Il ne contient aucune déclaration. La seconde règle dit comment agrandir un contexte $\Gamma$ en lui ajoutant une « déclaration » $(x : T)$. Les conditions sont que $T$ soit un type relativement à $\Gamma$ et que la variable $x$ n'ait pas déjà été déclarée dans $\Gamma$.

Le second groupe de règles définit les types. La première règle dit que dans tout contexte, $\mathbf{1}$ est un type. Ce type doit être compris comme un singleton. La seconde règle introduit les produits dépendants. Si $T$ est un type dans le contexte $\Gamma$, et $U$ un type dans le contexte $\Gamma(x : T)$, alors on a un « type produit dépendant » noté $(x : T) \times U$. Intuitivement, on doit comprendre ce type comme l'espace total d'un fibré de base $T$. Un « élément » de $(x : T) \times U$ est intuitivement un couple $(x, y)$ avec $x \in T$ et $y \in U$. Mais comme $U$ dépend de $x$ (puisque c'est un type dans le contexte $\Gamma(x : T)$), il ne s'agit pas d'un produit ordinaire. Ce type est celui des couples $(a, b)_{(x,U)}$ où $a$ est de type $T$ et $b$ de type $U[a/x]$ dans le contexte

6. Cette définition de l'égalité est dûe à Leibniz.

$\Gamma$ (le produit étant dépendant, il faut spécifier quel terme joue le rôle de $x$ dans $b$, il s'agit précisément de $a$). Nous avons les projections qui permettent de retrouver chaque composante de ce couple.

Le type des « parties » $\mathcal{P}(T)$ permet de construire les termes ensemblistes de notre langage : $\{x : T \mid E\}$ de type $\mathcal{P}(T)$ dans le contexte $\Gamma$ et $a \in A$ de type $\Omega$ si $a$ est de type $T$ et $A$ de type $\mathcal{P}(T)$ dans le contexte $\Gamma$. Une fois encore, le langage étant dépendant, il faut impérativement que $\Gamma$ paramètre ce type. Finalement, les choses se passent dans le topos $\mathcal{T}/\overline{\Gamma}$ pour ce langage comme dans le topos $\mathcal{T}$ pour le langage interne, nous userons donc de nos traductions pour interpréter ce type et ces termes.

Le type $W(E)$ est celui du garant de $E$. S'il existe un terme de ce type dans le contexte $\Gamma$, alors $E$ est vrai dans ce contexte. Lors de son interprétation, nous expliquerons en détail son fonctionnement et justifierons cette affirmation. Il y a plusieurs termes relatifs aux démonstrations. $L_{x:T}p$ symbolise le procédé « soit $x$ de type $T$ » qui consiste à poser en début de preuve un terme non spécifié d'un type précis pour prouver que l'énoncé est vrai pour n'importe quel terme de ce type. Ainsi, si $p$ est une preuve de $E$ dans $\Gamma(x : T)$ alors $L_{x:T}p$ est bien sûr une preuve de $\forall_{x:T}E$. $p.a$ est la spécification en $a$ d'une preuve $p$ d'un énoncé de type $\forall_{x:T}E$ dans le cas où $a$ est un terme de type $T$. Si $p$ est une preuve de $\exists!_{x:T}E$ dans $\Gamma$, alors $\delta(p)$ symbolise cet unique terme de type $T$ tel que $E$ (précisément $E[\delta(p)/x]$) soit vrai dans $\Gamma$ et $\rho(p)$ est la preuve que $E$ est vrai précisément pour ce terme $\delta(p)$ dans le contexte $\Gamma$.

Parmi la présentation de cette syntaxe, nous avons laissé filtrer quelques éléments de sémantique qui permettent, bien entendu, de comprendre son intérêt et son fonctionnement. Tentons à présent de l'expliquer en détail.

## 2.2 Interprétation du langage $W$

### 2.2.1 Préliminaires

L'interprétation d'un contexte $\Gamma$ sera notée $\overline{\Gamma}$, celle d'un type $T$ dans $\Gamma$ : $[\![ T ]\!]_\Gamma$ et celle d'un terme $a$ de type $T$ dans $\Gamma$ : $[\![ a ]\!]_\Gamma$

Dans le langage interne, un type, totalement indépendant du contexte, s'interprétait comme un objet. Ensemblistement, le contexte s'interprétait comme le produit cartésien des types des variables déclarées. Dès l'instant où le type d'une variable dépend des types des variables précédemment déclarées, ce procédé n'est plus valable. Il faut trouver un moyen de *paramétrer* le contexte dans lequel la nouvelle variable est déclarée par celui dans lequel elle ne l'est pas encore, c'est à dire que $\overline{\Gamma}$ doit paramétrer $\overline{\Gamma(x : T)}$.

Plaçons nous dans le topos des ensembles et donnons-nous une instance de contexte $i$ (c'est à dire une valeur à chaque variable déclarée dans $\Gamma$). $i$ définit un ensemble $A_i$ de valeurs que peut prendre la variable $x$ de type $T$ dans $\Gamma$ lorsque les autres variables prennent les valeurs indiquées par $i$ ; l'ensemble des objets de type $T$ n'est autre que la réunion de tous ces ensembles pour toutes les instances de contexte possibles : $\bigcup\limits_i A_i$. Ici, les choses se font « à l'envers », c'est à dire que nous n'avons pas $[\![ T ]\!]_\Gamma(i) = A_i$ mais pour tout $y$ de $A_i$ : $[\![ T ]\!]_\Gamma(y) = i$. Ainsi, on retrouve l'ensemble des valeurs de $x$ autorisées par $i$ en considérant la fibre de $[\![ T ]\!]_\Gamma$ au dessus de $i$. L'ensemble de définition de $[\![ T ]\!]_\Gamma$ est donc l'ensemble de tous les éléments de type $T$.

Nous avons volontairement omis une précision. Rigoureusement, il faut répéter avant chaque élément de type $T$ l'instance de contexte qui les permet. C'est à dire que, contrairement à ce que nous venons de dire, ce n'est pas exactement $[\![ T ]\!]_\Gamma(y) = i$ mais $[\![ T ]\!]_\Gamma(i, y) = i$. Cette différence peut paraître dérisoire étant connue la bijection entre les deux ensembles de définition. Elle permet néanmoins une définition par induction tout à fait naturelle et un paramétrage immédiat. En effet, elle nous permet d'obtenir : $[\![ T ]\!]^{-1}(i) = \{(i, x) \mid x \text{ est un élément de type } T \text{ compatible avec l'instance de contexte } i\}$. Ainsi, l'ensemble de définition de $[\![ T ]\!]_\Gamma$ n'est autre que $\overline{\Gamma(x : T)}$ et nous pouvons définir chaque nouveau contexte par la source du type de la variable ajoutée : $\overline{\Gamma(x:T)} = \mathbf{s}(\llbracket T \rrbracket_\Gamma)$. Bien sûr, l'interprétation d'un terme $a$ de type $T$ dans $\Gamma$ procède précisément dans l'autre sens et associe à chaque instance $i$ du contexte $\Gamma$ un élément $(i, x)$ de $\overline{\Gamma(x:T)}$, où, donc, $x$ est *compatible avec* $i$. Ainsi, $\llbracket T \rrbracket_\Gamma \circ \llbracket a \rrbracket_\Gamma(i) = i$. Catégoriquement, l'interprétation du terme est une section de l'interprétation de son type.

$$\begin{tikzcd} \overline{\Gamma(x:T)} \arrow[r, bend left, "\llbracket T \rrbracket_\Gamma"] & \overline{\Gamma} \arrow[l, bend left, "\llbracket a \rrbracket_\Gamma"] \end{tikzcd}$$

C'est ici qu'apparait le rôle central que jouent les topos relatifs dans l'interprétation de ce langage. Nous pouvons remarquer que $\llbracket T \rrbracket_\Gamma$ est bien un objet de $\mathcal{T}/\overline{\Gamma}$ et $\llbracket a \rrbracket_\Gamma$ l'une de ses flèches de but $\llbracket T \rrbracket_\Gamma$ : c'est exactement ce qui se passait pour le langage interne dans le topos initial. Finalement, le paramétrage par $\overline{\Gamma}$ est contenu dans le passage au topos relatif, une fois dans ce dernier, les choses se passent de façon sensiblement similaire au langage interne, tout le travail était donc de traduire les opérations de ce topos dans le topos $\mathcal{T}$ dans lequel, en définitive, nous interprétons notre langage. Les foncteurs $\llbracket a \rrbracket_\Gamma^*$ et $\Pi_{\llbracket T \rrbracket_\Gamma}$ permettent de passer de $\mathcal{T}/\overline{\Gamma}$ à $\mathcal{T}/\overline{\Gamma(x:T)}$ et réciproquement, à travers l'action d'un terme ou celle d'un type. Notons néanmoins que nous n'utiliserons pas directement ces opérations là, mais évidemment leurs représentantes, qui agissent directement sur $\mathcal{T}$.

Notre interprétation, bien entendu, se construit par induction (croisée car les types sont dépendants). Maintenant que nous avons expliqué le mécanisme de paramétrage, nous tenterons de donner l'intuition de l'interprétation de chaque terme via les catégories relatives. Tout le reste n'est qu'affaire de traductions auxquelles nous procéderons en nous servant des résultats de la première partie. Nous utiliserons d'ailleurs les macros non ambigües définies dans la section « résumé » pour éviter de développer les formules jusqu'à les rendre absolument incompréhensibles. Construisons à présent récursivement cette interprétation.

### 2.2.2 Les contextes

**Contexte $\emptyset$**

L'interprétation de $\emptyset$ dans $\mathcal{T}$ est $\mathbf{1}$.

Ensemblistement, $1 = \{\emptyset\}$ (où « $\emptyset$ » désigne l'ensemble vide).

Dans le contexte vide, nos termes et formules n'ont aucune variable libre. Ensemblistement, un terme dans le contexte vide associe donc à $\emptyset$ l'élément visé, on peut mentalement identifier ce terme et cet élément. Pour cela, il sera appelé un *élément global*.

$\overline{\emptyset} = \mathbf{1}$

**Contexte $\Gamma(x:T)$**

Si $T$ est un type dans le contexte $\Gamma$ et si $x$ n'est pas déclaré dans $\Gamma$, alors $\Gamma(x:T)$ est un contexte et comme nous l'avons expliqué en préliminaire : $\overline{\Gamma(x:T)} = \mathbf{s}(\llbracket T \rrbracket_\Gamma)$

Néanmoins, « $\mathbf{s}$ » n'est pas un symbole du langage natif. Afin de nous en débarrasser, nous sommes contraints de considérer les différentes formes possibles de $T$ et $\Gamma$ et de procéder par induction (on connait toutes les façons de construire un type, il suffit donc de les parcourir une par une et donner, dans chaque cas, la source de leur interprétation). Néanmoins, n'ayant pas encore présenté l'interprétation des types, nous nous contentons ici de fournir sans justification des résultats vérifiés plus bas.

- Si $T = 1$, $\overline{\Gamma(x:T)} = \overline{\Gamma}$
- Si $T = [(x:T') \times U]_\Gamma$, $\overline{\Gamma(x:T)} = \overline{\Gamma(x:T')(y:U)}$

– Si $T = \mathcal{P}(T')$, $\overline{\Gamma(x:T)} = (\overline{\Gamma(x:T')} \times \overline{\Gamma}) \mid (\subset \circ (1 \times (\llbracket T' \rrbracket_{\Gamma})^{\bullet}_{\overline{\Gamma(x:T')}})$
– Si $T = W(E)$, alors $\overline{\Gamma(x:T)} = \overline{\Gamma} \mid (\pi_2 \circ k \circ \llbracket E \rrbracket_{\Gamma})$
– Si $\Gamma = \Gamma'(y:U)$, alors $\overline{\Gamma(x:T)} = (\overline{\Gamma'(y:U)} \times \overline{\Gamma'(x:T)}) \mid (\asymp \circ (\llbracket U \rrbracket_{\Gamma'} \times \llbracket T \rrbracket_{\Gamma'}))$

### 2.2.3 Les types

**Type** $1$

Si $\Gamma$ est un contexte, $1$ est un type dans le contexte $\Gamma$

$\llbracket 1 \rrbracket_{\Gamma} = 1$

$$\begin{tikzcd} \overline{\Gamma} \arrow[loop right, "\mathbf{1}_{\overline{\Gamma}}"] \end{tikzcd}$$

Le type $1$ « n'ajoute rien » aux déclarations, il associe à chaque valeur possible pour nos variables ces mêmes valeurs. Ainsi, $\overline{\Gamma(x:1)} = \overline{\Gamma}$, déclarer une variable supplémentaire de type $1$ revient à se taire, ce type n'a aucun autre intérêt que celui, important, d'initialiser notre construction récursive des types.

**Type** $(x:T) \times U$

Si $T/\Gamma$ et $U/\Gamma(x:T)$, alors $(x:T) \times U$ est un type dans le contexte $\Gamma$. Cette notation explicite le fait qu'il s'agit d'un *produit dépendant*, le type $U$ dépend du terme $x$ de type $T$. Ainsi, catégoriquement, sans surprise, nous obtenons le diagramme commutatif suivant :

$$\begin{tikzcd} \overline{\Gamma(x:T)(y:U)} \arrow[r, "\llbracket U \rrbracket_{\Gamma(x:T)}"] \arrow[rr, bend right, "\llbracket (x:T) \times U \rrbracket_{\Gamma}"'] & \overline{\Gamma(x:T)} \arrow[r, "\llbracket T \rrbracket_{\Gamma}"] & \overline{\Gamma} \end{tikzcd}$$

$\llbracket (x:T) \times U \rrbracket_{\Gamma} = \llbracket T \rrbracket_{\Gamma} \circ \llbracket U \rrbracket_{\Gamma(x:T)}$

Ceci confirme ce qui a été affirmé lors de notre précédente récurrence :

$$\overline{\Gamma(x:T)(y:U)} = \overline{\Gamma(y:(x:T) \times U)}$$

**Type** $\mathcal{P}(T)$

Si $\Gamma$ est un contexte et que $T$ est un type dans $\Gamma$, alors $\mathcal{P}(T)$ est un type dans $\Gamma$. Il s'interprète par le représentant dans $\mathcal{T}$ de l'objet $\mathcal{P}(\langle \llbracket T \rrbracket_{\Gamma} \rangle_{\overline{\Gamma(x:T)}})$ du topos dépendant $\mathcal{T}/\Gamma$. Ainsi, via notre étude en première partie, nous savons que $\llbracket \mathcal{P}(T) \rrbracket_{\Gamma} = \pi_2 \circ j$.

$$\begin{tikzcd} \bullet \arrow[d, "\langle \rho, [\mathcal{P}(T)]_{\Gamma} \rangle"'] & (\mathcal{P}(\overline{\Gamma(x:T)}) \times \mathcal{P}(\overline{\Gamma(x:T)})) \mid \subset \arrow[d, "j"] & 1 \arrow[d, "\top"] \\ \mathcal{P}(\overline{\Gamma(x:T)}) \times \overline{\Gamma} \arrow[r, "1 \times (\llbracket T \rrbracket_{\Gamma})^{\bullet}_{\Gamma(x;T)}"'] & \mathcal{P}(\overline{\Gamma(x:T)}) \times \mathcal{P}(\overline{\Gamma(x:T)}) \arrow[r, "\subset"'] & \Omega \end{tikzcd}$$

Remarquons que $\mathbf{s}([\mathcal{P}(T)]_{\Gamma} = (\overline{\Gamma(x:T)} \times \overline{\Gamma}) \mid (\subset \circ (1 \times (\llbracket T \rrbracket_{\Gamma})^{\bullet}_{\Gamma(x;T)}))$, ce qui justifie notre récurrence ci-dessus.

**Type** $W(E)$

Si $\Gamma$ est un contexte, alors $\Omega$ est le type $\mathcal{P}(1)$ dans le contexte $\Gamma$. Ce type est central au sein de la construction du langage $W$ car c'est lui qui rend ce dernier *dépendant* et, ce faisant, permet l'interprétation des preuves. Si $E$ est un terme de type $\Omega$ dans le contexte $\Gamma$, il s'interprète dans la catégorie relative par une flèche de but $\Omega$, c'est donc précisément un *énoncé* de notre langage.

Par commodité, notons provisoirement $B = (\mathcal{P}(\overline{\Gamma}) \times \overline{\Gamma})|(\subset \circ (1 \times 1^{\bullet}_{\overline{\Gamma}}))$.

$$\begin{tikzcd}
\overline{\Gamma} \arrow[rr, "\llbracket E \rrbracket_\Gamma"] \arrow[dr, "1"'] & & B \arrow[dl, "\pi_2 \circ j"] \\
& \overline{\Gamma} &
\end{tikzcd}$$

En utilisant notre isomorphisme $i$ et son inverse $k$ exhibés en première partie, il est possible de concevoir $\llbracket E \rrbracket_\Gamma$ comme une flèche de but $\Omega$ dans le topos initial, et d'ainsi récupérer notre intuition ensembliste (un énoncé associant à chaque élément de l'ensemble une valeur de vérité). Il suffit de poser $f' = \pi_2 \circ k \circ f$, et réciproquement $f = i \circ \langle 1, f' \rangle$. Il sera souvent plus clair de considérer $[E]_\Gamma$ comme une flèche $\overline{\Gamma} \to \Omega$, nous n'oublierons pas cependant, dans les calculs, d'effectuer les compositions nécessaires.

$$\begin{tikzcd}
& \overline{\Gamma} \arrow[dl, "f'"'] \arrow[dd, "\langle 1, f' \rangle"] \arrow[dr, "f"] & \\
\Omega & & B \arrow[dl, bend left, "k"] \\
& \overline{\Gamma} \times \Omega \arrow[ul, "\pi_2"] \arrow[ur, "i"] &
\end{tikzcd}$$

La construction du type $W(E)$, par exemple, est simplifiée par cette conception, car ce type constitue celui du *garant* de $E$. En effet, une preuve de $E$ dans le contexte $\overline{\Gamma}$ doit précisément nous assurer de sa vérité dans ce contexte, vérité qui se traduit ensemblistement par le fait que son interprétation envoie tous les éléments de sa source sur l'élément « vrai » de $\Omega$, dont le singleton est représenté par le monomorphisme $\top$. Catégoriquement, ceci signifie que la flèche $\llbracket E \rrbracket_\Gamma$ se relève le long de $\top$ (que $\llbracket E \rrbracket_\Gamma = \top \circ \langle\rangle$), donc que $j$ possède une section : $1 \mid \llbracket E \rrbracket_\Gamma$.

$$\begin{tikzcd}
\overline{\Gamma} \mid \llbracket E \rrbracket_\Gamma \arrow[d, "j"] & & \mathbf{1} \arrow[d, "\top"] \\
\overline{\Gamma} \arrow[u, bend left, "1 \mid \llbracket E \rrbracket_\Gamma"] \arrow[urr, "\langle\rangle"] \arrow[rr, "\llbracket E \rrbracket_\Gamma"'] & & \Omega
\end{tikzcd}$$

Remarquons, d'une part, que $j$ est une flèche de but $\overline{\Gamma}$ déterminée via $\llbracket E \rrbracket_\Gamma$, et d'autre part que l'interprétation d'un terme d'un type quelconque dans un contexte donné constitue justement une section de l'interprétation de ce type. Tout ceci fait de $j$ le candidat idéal pour la sémantique de $W(E)$. En effet, en posant $\llbracket W(E) \rrbracket_\Gamma = j$, un terme de type $W(E)$ dans le contexte $\Gamma$ sera nécessairement interprété par une section de $j$, prouvant ainsi que $\llbracket E \rrbracket_\Gamma = \top \circ \langle\rangle$, donc que $E$ est vrai dans le contexte $\Gamma$ ([7]). Remarquons

7. En revanche, certains énoncés sont vrais dans un contexte donné bien qu'il n'en existe aucune preuve. C'est le cas lorsque ces énoncés possèdent bien un garant mais que celui-ci n'est pas représenté.

enfin que s'il existe une telle section de $[\![\, W(E) \,]\!]_\Gamma$, elle est unique. En effet, $\top$ étant un monomorphisme, son pullback $[\![\, W(E) \,]\!]_\Gamma$ l'est aussi donc ce dernier ne peut avoir qu'une section. Ceci ne signifie pas qu'il n'existe qu'une seule preuve de $E$ (qu'un seul terme de type $W(E)$) dans le contexte $\Gamma$, mais seulement que toutes ces preuves, aussi différentes soient-elles, ont la même interprétation, que nous nommerons leur *garant*. Le principe d'unicité du garant évoqué plus haut est respecté.

Ainsi : $[\![\, W(E) \,]\!]_\Gamma = j$

Sa source est $\overline{\Gamma} \mid (\pi_2 \circ k \circ [\![\, E \,]\!]_\Gamma)$ (car, comme nous l'avons expliqué, il faut en vérité composer correctement $[E]_\Gamma$ pour en faire une flèche $\overline{\Gamma} \longrightarrow \Omega$). Notre récurrence sur les contextes est donc une nouvelle fois justifiée.

**Type $T$ dans** $\Gamma(y : U)$

Nous avons ici un premier aperçu de l'intérêt du foncteur de changement de base et de son adjoint. Comme nous l'avons expliqué, si $U$ est un type dans $\Gamma$, ils permettent de passer de la catégorie $\mathcal{T}/\overline{\Gamma}$ à $\mathcal{T}/\overline{\Gamma(y:U)}$ et inversement.

Ainsi, sans surprise, si $T/\Gamma$ et $U/\Gamma$ , alors $T/\Gamma(y:U)$ et son interprétation est l'image de $[\![\, T \,]\!]_\Gamma$ par le foncteur de changement de base $[\![\, \overline{U} \,]\!]_\Gamma^*$, c'est à dire son pullpack le long de $[\![\, U \,]\!]_\Gamma$.

$$\begin{tikzcd}
{[\![\, U \,]\!]_\Gamma}\, {}_{\overline{\Gamma(y:U)}}\!\times_{\overline{\Gamma(x:T)}} {[\![\, T \,]\!]_\Gamma} \arrow[d, "{[\![\, T \,]\!]_{\Gamma(y:U)}}"'] & \overline{\Gamma(x:T)} \arrow[d, "{[\![\, T \,]\!]_\Gamma}"] \\
\overline{\Gamma(y:U)} \arrow[r, "{[\![\, U \,]\!]_\Gamma}"'] & \overline{\Gamma}
\end{tikzcd}$$

Donc $[\![\, T \,]\!]_{\Gamma(y:U)} = \pi_1 \circ j$

source : $(\overline{\Gamma(y:U)} \times \overline{\Gamma(x:T)}) \mid (\asymp \circ ([\![\, U \,]\!]_\Gamma \times [\![\, T \,]\!]_\Gamma))$ (ce qui valide définitivement notre récurrence sur les contextes, toutes les types possibles ayant été parcourus par ces constructions).

### 2.2.4 Les variables

**Variable $x$ dans** $\Gamma(x : T)$

Si $T/\Gamma$, alors $x$ est un terme de type $T$ dans le contexte $\Gamma(x:T)$, interprété par le représentant de la flèche diagonale $\langle \langle [\![\, T \,]\!]_\Gamma \rangle_{\overline{\Gamma(x:T)}}, \langle [\![\, T \,]\!]_\Gamma \rangle_{\overline{\Gamma(x:T)}} \rangle$ du topos $\mathcal{T}/\overline{\Gamma}$. Introduire ainsi la variable « $x$ » permet en effet de passer au contexte $\Gamma(x:T)$ « sans rien toucher » (car l'objet qui représente cette flèche dans $\mathcal{T}$ est bien de but $\overline{\Gamma(x:T)}$). Justement, dans le topos initial, il s'agit bien sûr de la section canonique du pullback de $[\![\, T \,]\!]_\Gamma$ le long de lui même :

$$\begin{tikzcd}
\overline{\Gamma(x:T)(y:T)} \arrow[d, "{[\![\, T \,]\!]_{\Gamma(x:T)}}"] & \overline{\Gamma(x:T)} \arrow[d, "{[\![\, T \,]\!]_\Gamma}"] \\
\overline{\Gamma(x:T)} \arrow[u, bend left, "{\langle 1,1 \rangle_{([\![\, T \,]\!]_\Gamma, [\![\, T \,]\!]_\Gamma)}}"] \arrow[ur, "\mathbf{1}"] \arrow[r, "{[\![\, T \,]\!]_\Gamma}"'] & \overline{\Gamma}
\end{tikzcd}$$

$[\![\, x \,]\!]_{\Gamma(x:T)} = \langle 1, 1 \rangle_{([\![\, T \,]\!]_\Gamma, [\![\, T \,]\!]_\Gamma)}$.

Remarquons que $\overline{\Gamma(x:T)(y:T)} = \overline{\Gamma(x:T)}\ {}_{\llbracket T \rrbracket_\Gamma}\!\times_{\llbracket T \rrbracket_\Gamma} \overline{\Gamma(x:T)}$, ce qui n'est pas surprenant, la variable $x$ se contentant ensemblistement d'« ajouter » un représentant du type $T$.

**Variable $x$ dans** $\Gamma(y:U)$

Si $x : T/\Gamma$ et $U/\Gamma$ alors $x : T/\Gamma(y:U)$, et son interprétation est bien sûr l'image de $\llbracket x \rrbracket_\Gamma$ par le foncteur de changement de base. Notons qu'encore une fois, grâce à ce foncteur, les choses se passent correctement. En effet, $\llbracket T \rrbracket_{\Gamma(y:U)}$ étant lui-même l'image de $\llbracket T \rrbracket_\Gamma$ par ce foncteur, et $\llbracket x \rrbracket_\Gamma$ étant une section de $\llbracket T \rrbracket_\Gamma$, $\llbracket x \rrbracket_{\Gamma(y:U)}$ est bien une section de $\llbracket T \rrbracket_{\Gamma(y:U)}$.

$$\begin{tikzcd}
\overline{\Gamma(y:U)(z:T)} \arrow[r, "g"] \arrow[d, shift left, "\llbracket T \rrbracket_{\Gamma(y:U)}"] & \overline{\Gamma(z:T)} \arrow[d, shift left, "\llbracket T \rrbracket_\Gamma"'] \\
\overline{\Gamma(y:U)} \arrow[r, "\llbracket U \rrbracket_\Gamma"'] \arrow[u, bend left, "\llbracket x \rrbracket_{\Gamma(y:U)}"] \arrow[ur, "\llbracket x \rrbracket_\Gamma \circ \llbracket U \rrbracket_\Gamma"'] & \overline{\Gamma} \arrow[u, bend right, "\llbracket x \rrbracket_\Gamma"']
\end{tikzcd}$$

$\llbracket x \rrbracket_{\Gamma(y:U)} = \langle 1, \llbracket x \rrbracket_\Gamma \circ \llbracket U \rrbracket_\Gamma \rangle_{(\llbracket U \rrbracket_\Gamma, \llbracket T \rrbracket_\Gamma)}$.

### 2.2.5 Les termes

**Terme $*$**

Dans n'importe quel contexte $\Gamma$, $*$ est un terme de type 1. Nous avions remarqué que le type 1 « n'ajoutait rien aux déclarations ». $*$ est un terme de type 1, il envoie donc ensemblistement chaque objet sur lui-même.

$$\begin{tikzcd}
\overline{\Gamma} \arrow[loop right, "\mathbf{1}_{\overline{\Gamma}}"]
\end{tikzcd}$$

$\llbracket * \rrbracket_\Gamma = 1$.

**Terme** $(a;b)_{(x,U)}$

Si $a : T/\Gamma$, $U/\Gamma$ et $b : U[a/x]/\Gamma$, alors $(a;b)_{(x;U)} : ((x:T) \times U)/\Gamma$.

Il s'agit intuitivement (mais nous formellement) de la « composition » des deux termes. En effet, les types étant dépendants, composer un terme de type $U$ dans le contexte $\Gamma(x:T)$ avec un terme de type $T$ dans le contexte $\Gamma$ requiert que le second « convienne » au premier. Ensemblistement, ceci signifie que le terme de type $T$ en jeu dans $b$ soit précisément $a$. Ceci nous est assuré par le fait que $b$ est justement de type $U[a/x]$. Considérons le carré cartésien suivant :

$$\begin{tikzcd}
\overline{\Gamma(y:U[a/x])} \arrow[r, "g"] \arrow[d, shift left, "\llbracket U[a/x] \rrbracket_\Gamma"] & \overline{\Gamma(x:T)(y:U)} \arrow[d, "\llbracket U \rrbracket_{\Gamma(x:T)}"] \\
\overline{\Gamma} \arrow[r, "\llbracket a \rrbracket_\Gamma"'] \arrow[u, bend left, "\llbracket b \rrbracket_\Gamma"] & \overline{\Gamma(x:T)}
\end{tikzcd}$$

Bien que le type $U(a/x)$ dans le contexte $\Gamma$ soit issu du type $U$ dans $\Gamma(x : T)$, étant donné que nous savons quel terme joue le rôle de $x$ (il s'agit de $a$), celui-ci a été « effacé ». $g$ permet de le récupérer et, composé avec $\llbracket b \rrbracket_\Gamma$, fournit notre flèche $\llbracket (a; b)_{(x,U)} \rrbracket_\Gamma$.

Considérons un exemple simple. Posons $\gamma = \emptyset$, $T$ est le type des entiers naturels dans le contexte vide et $U$ est le type des tableaux de longueur $n$ (il dépend donc du type $T$) dans le contexte $(x : T)$. Considérons le terme 3 de type $T$. Alors $U[3/x]$ sera le type des tableaux de longueurs 3. Dans un tel cas, $\overline{(x : T)(y : U)}$ est l'ensemble des couples $(n, t)$ où $n$ est un entier naturel et $t$ un tableau de longueur $n$. Néanmoins, $\overline{y : U[3/x]}$ n'est pas constitué des couples $(3, t)$, mais seulement des tableaux $t$ de longueur 3. La fonction $g$ du diagramme associe à chaque élément $t$ le couple $(3, t)$, elle « récupère » le représentant de $T$ et nous renvoie donc sur $\overline{(x : T)(y : U)}$, faisant du terme $g \circ \llbracket b \rrbracket_\emptyset$ un terme de type $(x : T) \times U$ dans $\emptyset$.

Nous avons donc : $\llbracket (a; b)_{(x;U)} \rrbracket_\Gamma = \pi_2 \circ j \circ \llbracket b \rrbracket_\Gamma$.

#### Terme $\pi_1(c)$

Décrivons à présent le processus inverse.

Si $c : ((x : T) \times U)/\Gamma$ alors nous allons récupérer les deux composantes $\pi_1(c)$ et $\pi_2(c)$ telles que $\llbracket c \rrbracket_\Gamma = \llbracket (\pi_1(c), \pi_2(c))_{(x;U)} \rrbracket_\Gamma$

Si $T/\Gamma$, $U/\Gamma(x : T)$ et $c : ((x : T) \times U)/\Gamma$, alors $\pi_1(c) : T/\Gamma$. Le diagramme commutatif suivant nous permet donc de retrouver la première composante :

$$\begin{tikzcd}
 & \overline{\Gamma(x : T)(y : U)} \arrow[d, "\llbracket U \rrbracket_{\Gamma(x:T)}"] \\
\overline{\Gamma} \arrow[ur, "\llbracket c \rrbracket_\Gamma"] \arrow[r, "\llbracket \pi_1(c) \rrbracket_\Gamma"'] & \overline{\Gamma(x : T)}
\end{tikzcd}$$

Il s'agit bien d'une section de $\llbracket T \rrbracket_\Gamma$ car :

$\llbracket T \rrbracket_\Gamma \circ \llbracket \pi_1(c) \rrbracket_\Gamma = \llbracket T \rrbracket_\Gamma \circ \llbracket U \rrbracket_{\Gamma(x:T)} \circ \llbracket c \rrbracket_\Gamma = \llbracket (x : T) \times U \rrbracket_\Gamma \circ \llbracket c \rrbracket_\Gamma = 1$

$\llbracket \pi_1(c) \rrbracket_\Gamma = \llbracket U \rrbracket_{\Gamma(x:T)} \circ \llbracket c \rrbracket_\Gamma$.

#### Terme $\pi_2(c)$

Si $T/\Gamma$, $U/\Gamma(x : T)$ et $c : ((x : T) \times U)/\Gamma$, alors $\pi_2(c) : T/\Gamma$, le diagramme suivant fournit la seconde composante :

$$\begin{tikzcd}
\overline{\Gamma(y : U[a/x])} \arrow[r, "g"] \arrow[d, "\llbracket U[a/x] \rrbracket_\Gamma"] & \overline{\Gamma(x : T)(y : U)} \arrow[d, "\llbracket U \rrbracket_{\Gamma(x:T)}"] \\
\overline{\Gamma} \arrow[u, bend left, "\llbracket \pi_2(c) \rrbracket_\Gamma"] \arrow[ur, "\llbracket c \rrbracket_\Gamma"'] \arrow[r, "\llbracket \pi_1(c) \rrbracket_\Gamma"'] & \overline{\Gamma(x : T)}
\end{tikzcd}$$

avec, $\llbracket \pi_2(c) \rrbracket_\Gamma = \langle 1, \llbracket c \rrbracket_\Gamma \rangle_{(\llbracket \pi_1(c) \rrbracket_\Gamma, \llbracket U \rrbracket_{\Gamma(x:T)})}$ (une telle section de $\llbracket U[a/x] \rrbracket_\Gamma$ existe évidemment car $\llbracket U \rrbracket_{\Gamma(x:T)} \circ \llbracket c \rrbracket_\Gamma = \llbracket \pi_1(c) \rrbracket_\Gamma$).

Par définition, nous savons que $g \circ \llbracket \pi_2(c) \rrbracket_\Gamma = \llbracket c \rrbracket_\Gamma$, donc $\llbracket c \rrbracket_\Gamma = \llbracket (\pi_1(c); \pi_2(c))_{(x;U)} \rrbracket_\Gamma$, nous sommes bien parvenus à récupérer notre couple.

**Le terme** $\{x : T \mid E\}$

Si le langage interne permet de concevoir tout topos comme celui des ensembles, il nous prive en revanche de la dépendance. Le langage $W$ pallie ce dernier défaut tout en conservant la première caractéristique. En effet, construisons à présent les deux termes « ensemblistes » de ce langage : la « compréhension » qui permet de choisir dans un ensemble particulier les éléments qui valident une formule, et le prédicat « appartenance », qui s'applique à deux termes $a$ et $b$ tel que $a$ soit d'un type $T$ quelconque et $b$ de type $\mathcal{P}(T)$.

Si $E : \Omega/\Gamma(x : T)$ alors $\{x \in T | E\}$ est un terme de type $P(T)$ dans $\Gamma$. Nous reconnaissons là l'écriture ensembliste standard.

Pour tous objets $X$ et $Y$, la fonction $\Sigma_Y$ de notre topos $\mathcal{T}$ initial « curryfie » les flèches $f : X \times Y \longrightarrow \Omega$. Ensemblistement, elle envoie un élément $x$ de $X$ sur l'ensemble des $y \in Y$ tels que $f(x, y)$ soit vrai. Ainsi, si $F$ est un terme *du langage interne* de type $\Omega$ dans le contexte $\Gamma(y : Y)$, le terme $\{y \in Y \mid F\}$ est justement celui qui associe aux éléments $(x_1, ..., x_n)$ de $\overline{\Gamma}$, l'ensemble des éléments de $Y$ tels que « $F(x_1, ..., x_n, y)$ » soit vrai. C'est donc la curryfiée de $F$ et légitimement : $\lfloor\{y \in Y | F\}\rfloor_\Gamma = \Sigma_Y(\lfloor F\rfloor_{\Gamma(x:T)})$.

Dans le langage $W$, le fonctionnement est parfaitement identique à ceci près, encore une fois, que nous nous plaçons dans le topos relatif $\mathcal{T}/\overline{\Gamma}$. Notons d'ailleurs que l'interprétation d'un type est une flèche et non un objet de $\mathcal{T}$, nous ne sommes donc plus en mesure d'écrire $\Sigma_{\llbracket T \rrbracket_\Gamma}$, sauf s'il s'agit du $\Sigma$ du topos relatif. C'est donc ainsi que nous devons procéder. Remarquons une fois de plus combien l'interprétation de ce langage est liée au travail de traduction effectué en premier partie. En effet, les deux termes « ensemblistes » du langage $W$ constituent un moment difficile de son interprétation. Néanmoins, maintenant que nous possédons l'écriture de la fonction $\Sigma$ du topos $\mathcal{T}/\overline{\Gamma}$ dans le langage natif, nous n'avons plus qu'à écrire :

$\llbracket \{x \in T \mid E\} \rrbracket_\Gamma = \overline{\Sigma}_{(\llbracket T \rrbracket_\Gamma, \overline{\Gamma(x:T)})}(\llbracket E \rrbracket_{\overline{\Gamma(x:T)}})$.

**Le terme** $a \in A$

Cette remarque s'applique de façon totalement identique à la flèche $\ni$ (que nous avons aussi traduite dans le langage natif). Ainsi, si $a : T/\Gamma$ et $A : P(T)/\Gamma$, alors $a \in A$ est un terme de type $\Omega$ dans le contexte $\Gamma$. C'est bien sûr l'énoncé qui dit si un élément de $a$ « appartient » à un élément de $\mathcal{P}(a)$ :

$\llbracket a \in A \rrbracket_\Gamma = \overline{\ni} \circ \langle \llbracket a \rrbracket_\Gamma, \llbracket A \rrbracket_\Gamma \rangle$.

**Le terme** $\forall_{x:T} E$

Si $E$ est un terme de type $\Omega$ dans le contexte $\Gamma(x : T)$, alors $\forall_{x:T} E$ est un terme de type $\Omega$ dans le contexte $\Gamma$. L'interprétation de $W(\forall_{x:T} E)$ dans $\Gamma$ n'est autre, bien sûr, que l'image de celle $W(E)$ dans $\Gamma(x : T)$ par le foncteur $\overline{\Pi}_{\llbracket T \rrbracket_\Gamma}$. Ayant explicité l'écriture de ce foncteur et celle de l'interprétation de $W(E)$ en fonction de celle de $E$, nous connaissons donc $\llbracket W(\forall_{x:T} E) \rrbracket_\Gamma = \overline{\Pi}_{\llbracket T \rrbracket_\Gamma}(\llbracket W(E) \rrbracket_{\Gamma(x:T)}, \overline{\Gamma(x : T)} \mid (\pi_2 \circ k \circ \llbracket E \rrbracket_{\Gamma(x:T)}))$.

$$
\begin{tikzcd}
\bullet \arrow[d, "\llbracket W(E) \rrbracket_{\Gamma(x:T)}"] & \bullet \arrow[d, "\llbracket W(\forall_{x:T} E) \rrbracket_\Gamma = \overline{\Pi}_{\llbracket T \rrbracket_\Gamma}(\llbracket W(E) \rrbracket_{\Gamma(x:T)}, \overline{\Gamma(x:T)} \mid (\pi_2 \circ k \circ \llbracket E \rrbracket_{\Gamma(x:T)}))"] \\
\overline{\Gamma(x : T)} \arrow[r, "\llbracket T \rrbracket_\Gamma"'] & \overline{\Gamma}
\end{tikzcd}
$$

Il nous suffit donc de retrouver $\llbracket \forall_{x:T} E \rrbracket_\Gamma$ à partir de $\llbracket W(\forall_{x:T} E) \rrbracket_\Gamma$. Ceci ne pose aucun problème, il s'agit simplement de sa fonction caractéristique composée avec les isomorphismes adéquats :

$[\![ \forall_{x:T} E ]\!]_\Gamma = \mathbf{i} \circ \langle 1, \chi^h_{[\![ W(\forall_{x:T}E) ]\!]_\Gamma} \rangle$ donc $[\![ \forall_{x:T} E ]\!]_\Gamma = \mathbf{i} \circ \langle 1, \chi^h_{\overline{\Pi}_{[\![ T ]\!]_\Gamma([\![ W(E) ]\!]_\Gamma, \overline{\Gamma(x:T)} \ | \ (\pi_2 \circ k \circ [\![ E ]\!]_{\Gamma(x:T)}))}} \rangle$ (8)

**Le terme $L_{x:T}p$**

Si $p \vdash E/\Gamma(x:T)$ alors $L_{x:T}p \vdash \forall_{x:T}E/\Gamma$. En effet, s'il existe une preuve $p$ de $E$ dans le contexte $\Gamma(x:T)$, alors $E$ est vrai « quelque soit la valeur de $x$ », il doit donc exister une preuve de $\forall_{x:T}E$ dans le contexte $\Gamma$, notée $L_{x:T}p$. Son interprétation, sans surprise, est l'image de $[\![ p ]\!]_{\Gamma(x:T)}$ par l'adjoint à droite du foncteur de changement de base.

$$\begin{tikzcd}
\overline{\Gamma(x:T)} \arrow[r, "{[\![ p ]\!]_{\Gamma(x:T)}}"] \arrow[dr, "1"'] & \bullet \arrow[d, "{[\![ W(E) ]\!]_{\Gamma(x:T)}}"] & \overline{\Gamma} \arrow[r, "{[\![ L_{x:T}p ]\!]_\Gamma}"] \arrow[dr, "1"'] & \bullet \arrow[d, "{[\![ W(\forall_{x:T}E) ]\!]_\Gamma}"] \\
& \overline{\Gamma(x:T)} \arrow[rr, "{[\![ T ]\!]_\Gamma}"'] & & \overline{\Gamma}
\end{tikzcd}$$

En effet, $\Pi_{[\![ T ]\!]_\Gamma}$ étant un foncteur, l'image d'une section est une section de l'image. $[\![ L_{x:T}p ]\!]_\Gamma$ est donc bien le garant de $\forall_{x:T}E$ dans le contexte $\Gamma$.

$[\![ L_{x:T}p ]\!]_\Gamma = \overline{\Pi}_{[T]_\Gamma}([p]_{\Gamma(x:T)}, [\![ W(E) ]\!]_\Gamma)$.

**Le terme $p.a$**

Si $p \vdash \forall_{x:T}E/\Gamma$ et $a : T/\Gamma$ alors $p.a \vdash E[a/x]/\Gamma$ En effet, s'il existe une preuve $p$ de $\forall_{x:T}E$ dans $\Gamma$, il doit exister une preuve de $E[a/x]$ quelque soit le $a$ choisi. Sémantiquement, l'interprétation de cette preuve doit être une section de l'interprétation de $W(E[a/x])$ dans le contexte $\Gamma$.

Afin d'alléger des formules déjà complexes, nous ne ferons prendre qu'un seul argument (le premier) aux applications $\overline{\Pi}_f$ et $\overline{f}^*$ durant le développement de cette interprétation. Bien entendu, ces notations sont ambigües, nous reviendront donc aux multiples arguments au moment de fournir le résultat. Nous allons nous servir ici de la flèche $\overline{[\![ T ]\!]}^*_\Gamma([\![ \forall_{x:T}W(E) ]\!]_\Gamma) \xrightarrow{\overline{\varepsilon}} [\![ W(E) ]\!]_{\Gamma(x:T)}$.

Considérons le diagramme suivant :

$$\begin{tikzcd}
\bullet \arrow[d, bend left, "{[\![ W(E[a/x]) ]\!]_\Gamma}"] & & \bullet \arrow[rr, "\overline{\varepsilon}"] \arrow[dr, "{\overline{[\![ T ]\!]}^*_\Gamma([\![ \forall_{x:T}W(E) ]\!]_\Gamma)}"'] & & \bullet \arrow[dl, "{[\![ W(E) ]\!]_{\Gamma(x:T)}}"] & & \bullet \arrow[d, bend left, "{[\![ W(\forall_{x:T}E) ]\!]_\Gamma}"] \\
\overline{\Gamma} \arrow[u, bend left, "{\overline{[\![ a ]\!]}^*_\Gamma(\overline{\varepsilon} \circ \overline{[\![ T ]\!]}^*_\Gamma([\![ p ]\!]_\Gamma))}"] \arrow[rrr, "{[\![ a ]\!]_\Gamma}"'] & & & \Gamma(x:T) \arrow[ul, bend right, "{\overline{[\![ T ]\!]}^*_\Gamma([\![ p ]\!]_\Gamma)}"'] \arrow[ur, bend left, "{\overline{\varepsilon} \circ \overline{[\![ T ]\!]}^*_\Gamma([\![ p ]\!]_\Gamma)}"] \arrow[rrr, "{[\![ T ]\!]_\Gamma}"'] & & & \overline{\Gamma} \arrow[u, bend left, "{[\![ p ]\!]_\Gamma}"]
\end{tikzcd}$$

$[\![ p ]\!]_\Gamma$ étant une section de $[\![ \forall_{x:T}E ]\!]_\Gamma$, $\overline{[\![ T ]\!]}^*_\Gamma([\![ p ]\!]_\Gamma)$ est une section de $\overline{[\![ T ]\!]}^*_\Gamma([\![ \forall_{x:T}W(E) ]\!]_\Gamma)$. Or, par définition de $\overline{\varepsilon}$, le triangle du milieu est commutatif. Donc $\overline{\varepsilon} \circ \overline{[\![ T ]\!]}^*_\Gamma([\![ p ]\!]_\Gamma)$ est une section de $[\![ W(E) ]\!]_{\Gamma(x:T)}$ (on peut remarquer, même si ce n'est pas ce que nous recherchons ici, qu'il s'agit du garant de $E$ dans le

8. Nous ne développons pas $h$ directement dans la formule pour éviter de la surcharger. Il s'agit bien sûr de la source de $[\![ \forall_{x:T}E ]\!]_\Gamma$ retrouvable via la section « résumé » : $(\overline{\Gamma} \times \mathcal{P}(\overline{\Gamma(x:T)} \times \overline{\Gamma(x:T)} \mid (\pi_2 \circ k \circ [\![ E ]\!]_{\Gamma(x:T)}))\|$
$\lfloor (\forall_{a \in \overline{\Gamma(x:T)}} \forall_{x \in \overline{\Gamma(x:T)} \ | \ (\pi_2 \circ k \circ [\![ E ]\!]_{\Gamma(x:T)}))} (a, x) \in s \Rightarrow ([\![ T ]\!]_\Gamma(a) = b \wedge [\![ E ]\!]_{\Gamma(x:T)}(x) = a))$
$\wedge (\forall_{a \in A} \ [\![ T ]\!]_\Gamma(a) = b \Rightarrow \exists!_{x \in X} \ (a, x) \in s) \rfloor_{(b \in \overline{\Gamma})(s \in \mathcal{P}(\overline{\Gamma(x:T)} \times \overline{\Gamma(x:T)} \ | \ (\pi_2 \circ k \circ [\![ E ]\!]_{\Gamma(x:T)}))))})$

contexte $\Gamma(x : T)$). De plus, $[\![ W(E[a/x]) ]\!]_\Gamma = \overline{[\![ a ]\!]}^*_\Gamma([\![ W(E) ]\!]_{\Gamma(x:T)})$, ainsi $\overline{[\![ a ]\!]}^*_\Gamma(\overline{\varepsilon} \circ \overline{[\![ T ]\!]}^*_\Gamma([\![ p ]\!]_\Gamma))$ est une section de $[\![ W(E[a/x]) ]\!]_\Gamma$ et, par conséquent, le garant de $W(E[a/x])$ dans le contexte $\Gamma$, c'est donc la flèche recherchée. En écrivant correctement les applications, nous obtenons :

$$[\![ p.a ]\!]_\Gamma = \overline{[\![ a ]\!]}^*_\Gamma(\overline{\varepsilon} \circ [T]^*_\Gamma([p]_\Gamma, [\![ W(\forall_{x:T} E) ]\!]_\Gamma), [\![ W(E) ]\!]_{\Gamma(x:T)}).$$

# Bibliographie


[1] **A. Burroni** *Algèbres Graphiques.* Cahiers de Topologie et Géométrie Différentielle, vol. XXII, 3, 1981, pages 249-265.

[2] **E.J. Dubuc, G.M. Kelly** *A presentation of Topoi as Algebraic Relative to Categories or Graphs* J. of Algebra **81**, pages 420-433, 1983.

[3] **P. Freyd** : *Aspects of Topoi.* Bull. Austral. Math. Soc. **7** (1972), 1-76.

[4] **A. Prouté** : *Sur quelques liens entre théorie des topos et théorie de la démonstration.* `http://people.math.jussieu.fr/~alp/luminy_05_2007.pdf`

[5] **A. Prouté** : *Cours de Logique Catégorique.* `http://people.math.jussieu.fr/~alp/cours_2009.pdf`