# A remark on singleton types

March 4, 2014

## Identity types

In 1973, P. Martin-Löf introduced the identity type, which is obtained by refining the usual rules of equality in natural deduction, in the same way that one can refine the elimination rules for existential quantification and disjunction [2]. Another new refinement which appears in this paper is for the elimination rule for the empty type. In all cases, the new feature is that the elimination rule mentions the proof object.

The introduction rule is

$$\mathsf{Refl}(a) : \mathsf{Id}_A(a, a)$$

but the elimination rule will state

$$\mathsf{J}(d) : (\Pi x\ y : A)(\Pi p : \mathsf{Id}_A(x, y))C(x, y, p)$$

if we have $d : (\Pi\ x : A)C(x, x, \mathsf{Refl}(x))$ and $C(x, y, p)$ is a family of types depending on $x\ y : A$ and $p : \mathsf{Id}_A(x, y)$.

Compare this with the elimination rule

$$((\Pi z : A)C(z, z)) \rightarrow (\Pi x\ y : A)\mathsf{Id}_A(x, y) \rightarrow C(x, y)$$

which expresses that identity is the *least* reflexive relation. The new feature here is that the elimination rule

$$((\Pi z : A)C(z, z, \mathsf{Refl}(z))) \rightarrow (\Pi x\ y : A)(\Pi p : \mathsf{Id}_A(x, y))C(x, y, p)$$

mentions explicitely the proof $p : \mathsf{Id}(A, x, y)$.

The computation rule is

$$\mathsf{J}(d, x, x, \mathsf{Refl}(x)) = d(x) : C(x, x, \mathsf{Refl}(x))$$

In 1990, C. Paulin formulated a variation of this rule [1]. The introduction rule is the same but the elimination rule now states

$$\mathsf{H}_x(d) : (\Pi y : A)(\Pi p : \mathsf{Id}_A(x, y))D(y, p)$$

if we have $d : D(x, \mathsf{Refl}(x))$ and $D(y, p)$ is a family of types depending on $y : A$ and $p : \mathsf{Id}_A(x, y)$.

The computation rule is

$$\mathsf{H}_x(d, x, \mathsf{Refl}(x)) = d : D(x, \mathsf{Refl}(x))$$

This corresponds to the non dependent elimination

$$D(x) \rightarrow \mathsf{Id}_A(x, y) \rightarrow D(y)$$

while the dependent elimination can be formulated as

$$D(x, \mathsf{Refl}(x)) \rightarrow (\Pi p : \mathsf{Id}_A(x, y))D(y, p)$$

## Equivalence of the two formulations

One natural question is to ask if these two formulations are equivalent. While it is direct to define $\mathsf{J}(d)$ as $\mathsf{H}_x(d(x))$ the converse, i.e. how to define $\mathsf{H}_x$ from $\mathsf{J}$ may not seem so easy. This equivalence was first proved by M. Hofmann. However, it was traditionally considered to be something quite complex. The goal of this note is to present a natural way to understand this equivalence.

The main Lemma is the fact that in the following type

$$\mathsf{singl}(A, x) = (\Sigma y : A)\mathsf{Id}_A(x, y)$$

all elements $y, p$ are equal of the special element $x, \mathsf{Refl}(x)$. This is stated formally

$$(\Pi x\ y : A)(\Pi p : \mathsf{Id}(A, x, y))\mathsf{Id}_{\mathsf{singl}(A,x)}((x, \mathsf{Refl}(x)), (y, p))$$

and the proof is $\mathsf{J}(d)$ where

$$d : (\Pi x : A)\mathsf{Id}_{\mathsf{singl}(A,x)}((x, \mathsf{Refl}(x)), (x, \mathsf{Refl}(x))) = \mathsf{Refl}((x, \mathsf{Refl}(x)))$$

It follows from this that we have

$$D(x, \mathsf{Refl}(x)) \to D(y, p)$$

which is precisely what the elimination rule $\mathsf{H}_x$ expresses.

## Connection with algebraic topolgy

For showing this equivalence, a crucial role was played by the fact that the singleton type $\mathsf{singl}(A, a)$ is *contractible*, i.e. that it is inhabited and all its elements are equal. It might be interesting to connect this with the following extract from an interview of J.P. Serre [4]

"Yes, of course this happens quite often. For instance, when I was working on homotopy groups (around 1950), I convinced myself that, for a given space $X$, there should exist a fibre space $E$, with base $X$, which is contractible; such a space would indeed allow me (using Leray's methods) to do lots of computations on homotopy groups and Eilenberg-Maclane cohomology. But how to find it? It took me several weeks (a very long time, at the age I was then ...) to realize that the space of "paths" on $X$ had all the necessary properties - if only I dared call it a "fibre space", which I did. This was the starting point of the loop-space method in algebraic topology, many results followed quickly."

The following citation is even more explicit.

"Indeed, to apply Lerays theory I needed to construct fibre spaces which did not exist if one used the standard definition. Namely, for every space $X$, I needed a fibre space $E$ with base $X$ and with trivial homotopy (for instance contractible). But how to get such a space? One night in 1950, on the train bringing me back from our summer vacation, I saw it in a flash: just take for $E$ the space of paths on $X$ (with fixed origin $a$), the projection $E \to X$ being the evaluation map: path $\to$ extremity of the path. The fibre is then the loop space of $(X, a)$. I had no doubt: this was it! So much so that I even waked up my wife to tell her (Of course, I still had to show that $E \to X$ deserves to be called a fibration, and that Lerays theory applies to it.) ... It is strange that such a simple construction had so many consequences."

In type theoretic terms, the space $E$ is represented by $(\Sigma y : X)\mathsf{Id}_X(a, y)$ and $E \to X$ is the first projection (which assigns to a pair $y, p$ the extremity $y$ of the path $p$).

## References


[1] C. Paulin and F. Pfenning. Inductively defined types in the Calculus of Constructions. Mathematical Foundations of Programming Semantics, 1990.

[2] W.Howard. The formulae-as-types notion of construction. in Seldin, Jonathan P.; Hindley, J. Roger, *To H.B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*, Boston, MA: Academic Press, pp. 479490.

[3] P. Martin-Löf. An intuitionistic theory of types: predicative part. Logic Colloquium, 1973.

[4] An Interview with J.P. Serre. The Mathematical Intelligencer, December 1986, Volume 8, Issue 4, pp 8-13.