



## 2.5 The higher groupoid structure of type formers

Canonical name	25TheHigherGroupoidStructureOfTypeFormers
Date of creation	2013-11-14 18:33:26
Last modified on	2013-11-14 18:33:26
Owner	PMBookProject (1000683)
Last modified by	rspuzio (6075)
Numerical id	3
Author	PMBookProject (6075)
Entry type	Feature
Classification	msc 03B15

In <http://planetmath.org/node/87533> Chapter 1, we introduced many ways to form new types: cartesian products, disjoint unions, dependent products, dependent sums, etc. In <http://planetmath.org/21typesarehighergroupoids§2.1>, <http://planetmath.org/node/87533> Chapter 1, we saw that *all* types in homotopy type theory behave like spaces or higher groupoids. Our goal in the rest of the chapter is to make explicit how this higher structure behaves in the case of the particular types defined in <http://planetmath.org/node/87533> Chapter 1.

It turns out that for many types  $A$ , the equality types  $x =_A y$  can be characterized, up to equivalence, in terms of whatever data was used to construct  $A$ . For example, if  $A$  is a cartesian product  $B \times C$ , and  $x \equiv (b, c)$  and  $y \equiv (b', c')$ , then we have an equivalence

$$((b, c) = (b', c')) \simeq ((b = b') \times (c = c')). \quad (2.5.1)$$

In more traditional language, two ordered pairs are equal just when their components are equal (but the equivalence (??) says rather more than this). The higher structure of the identity types can also be expressed in terms of these equivalences; for instance, concatenating two equalities between pairs corresponds to pairwise concatenation.

Similarly, when a type family  $P : A \rightarrow \mathcal{U}$  is built up fiberwise using the type forming rules from <http://planetmath.org/node/87533> Chapter 1, the operation  $\text{transport}^P(p, -)$  can be characterized, up to homotopy, in terms of the corresponding operations on the data that went into  $P$ . For instance, if  $P(x) \equiv B(x) \times C(x)$ , then we have

$$\text{transport}^P(p, (b, c)) = (\text{transport}^B(p, b), \text{transport}^C(p, c)).$$

Finally, the type forming rules are also functorial, and if a function  $f$  is built from this functoriality, then the operations  $\text{ap}_f$  and  $\text{apd}_f$  can be computed based on the corresponding ones on the data going into  $f$ . For instance, if  $g : B \rightarrow B'$  and  $h : C \rightarrow C'$  and we define  $f : B \times C \rightarrow B' \times C'$  by  $f(b, c) \equiv (g(b), h(c))$ , then modulo the equivalence (??), we can identify  $\text{ap}_f$  with “ $(\text{ap}_g, \text{ap}_h)$ ”.

The next few sections (<http://planetmath.org/26cartesianproducttypes§2.6> to <http://planetmath.org/213naturalnumbers§2.13>) will be devoted to stating and proving theorems of this sort for all the basic type forming rules, with one section for each basic type former. Here we encounter a certain apparent deficiency in currently available type theories; as will become clear in

later chapters, it would seem to be more convenient and intuitive if these characterizations of identity types, transport, and so on were *judgmental* equalities. However, in the theory presented in <http://planetmath.org/node/87533> Chapter 1, the identity types are defined uniformly for all types by their induction principle, so we cannot “redefine” them to be different things at different types. Thus, the characterizations for particular types to be discussed in this chapter are, for the most part, *theorems* which we have to discover and prove, if possible.

Actually, the type theory of <http://planetmath.org/node/87533> Chapter 1 is insufficient to prove the desired theorems for two of the type formers:  $\Pi$ -types and universes. For this reason, we are forced to introduce axioms into our type theory, in order to make those “theorems” true. Type-theoretically, an *axiom* (c.f. <http://planetmath.org/11typetheoryversussettheory§1.1>) is an “atomic” element that is declared to inhabit some specified type, without there being any rules governing its behavior other than those pertaining to the type it inhabits.

The axiom for  $\Pi$ -types (<http://planetmath.org/29pitypesandthefunctionextensionality>) is familiar to type theorists: it is called *function extensionality*, and states (roughly) that if two functions are homotopic in the sense of <http://planetmath.org/24homotopies> then they are equal. The axiom for universes (<http://planetmath.org/210universesandtheuniv>) however, is a new contribution of homotopy type theory due to Voevodsky: it is called the *univalence axiom*, and states (roughly) that if two types are equivalent in the sense of <http://planetmath.org/24homotopiesandequivalences§2.4>, then they are equal. We have already remarked on this axiom in the introduction; it will play a very important role in this book.<sup>1</sup>

It is important to note that not *all* identity types can be “determined” by induction over the construction of types. Counterexamples include most non-trivial higher inductive types (see <http://planetmath.org/node/87579> Chapter 6, <http://planetmath.org/node/87582> Chapter 8). For instance, calculating the identity types of the types  $\mathbb{S}^n$  (see <http://planetmath.org/64circlesandspheres§6.4>) is equivalent to calculating the higher homotopy groups of spheres, a deep and important field of research in algebraic topology.

---

<sup>1</sup>We have chosen to introduce these principles as axioms, but there are potentially other ways to formulate a type theory in which they hold. See the Notes to this chapter.