Mathematics

Tuesday, 18 June 2019

Type theory

A **type** is an indefinite abstraction. In contrast, a representation is a definite abstraction. There is a one to one relationship between a representation and an object of representation. The relationship is abstraction. The relationship of a type to its object of representation is one to unknown many greater than inutile, confusing, obfuscating, or distracting. That is, this quantification may remain obscure while the syntactic purpose of a type is not.

In **type analysis**, a *syntactic type* maps onto a *semantic type*, for some relationship of syntax to semantics, or representation to object of recognition. The proof of these objects and relationships as existent is a proof of semantic continuity, and the proof of these relationships as not contradictory is a proof of semantic consistency. The product is a type algebra, or a type system.

A mathematical object of syntax, space, type, class, or set, has no properties of extent or existence when null, void, or empty. The first objects of construction need to be able to represent themselves in the analysis of their own theory. Therefore the type is the first object of representation of an object of recognition. The type type is equivalent to the type. And, the empty type is nonexistent. When we distinguish an empty type from an empty class or set, we maintain a pedantic adherence to the algebra of reason despite there being a sort of natural equivalence between any representation of void and another which remains unidentified by language.

By analysis, the assignment of definition to the objects of syntax: "null", "void", and "empty".

> *κύμα*$_{\varnothing}$ = *"null", "a type having a wave node association".*
> *κενός*$_{\varnothing}$ = *"void", "a type having a space infinitude association".*
> *σειρά*$_{\varnothing}$ = *"empty", "a type having a finite, enumerable set association".*

These are type objects as representational but not collecting or accumulating or mutable. They share no relationships and are not mutually contradictory or exclusive. These representations identify objects of recognition. Therefore, this type algebra (in three assignments) is continuous and consistent.

Notes

[TREE] Ahrens, Capriotti, Spadotti, “Non-wellfounded trees in Homotopy Type Theory” https://arxiv.org/abs/1504.02949v1
[SELF] Escardo, “A self-contained, brief and complete formulation of Voevodsky’s univalence axiom” https://homotopytypetheory.org/2018/03/07/a-self-contained-brief-and-complete-formulation-of-voevodskys-univalence-axiom/ https://arxiv.org/abs/1803.02294
[HTOY] Homotopy Type Theory
https://homotopytypetheory.org/book/
https://github.com/hott
https://www.ams.org/notices/201309/rnoti-p1164.pdf
[VOEY] Vladimir Voevodsky, Univalent foundations project
http://www.math.ias.edu/vladimir/Univalent_Foundations

Mathematics: type theory

John Pritchard, @syntelos

https://docs.google.com/document/d/1XNKDHWhxO1RulN-frS5QON2UjKkFrxDjDPM-Xwy4uNI/edit?usp=sharing