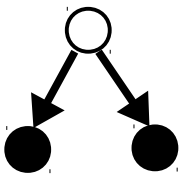
Witness Duality

by Sven Nilsen, 2020

In this paper I introduce Witness Duality for Avatar Extensions with the analogue of univalence.

Witness Duality might be thought of as two different models of equivalence (Avatar Graph[1] notation):



Product Witness

Loop Witness

The Product Witness might be thought of as the global perspective of equivalence.
The Loop Witness might be thought of as the local perspective of equivalence.

When reasoning about Witness Duality, the symbols `=` and `~=` are used to distinguish them:

A = B	`A` and `B` related by a product witness
A ~= B	`A` and `B` related by a loop witness

Normally, `=` is used to describe equality, while `~=` is used for the existence of an isomorphism.
Witness Duality does not make this distinction, because both might describe e.g. isomorphisms,
depending on the interpretation. The distinction is more fundamental and deeper.

The Product Witness has its origin in the treatment of equivalence as a binary relation[2].
This witness allows expressing the existence of an isomorphism,
but not distinguish between isomorphisms when there exists multiple ones.

The Loop Witness has its origin in the treatment of equivalence as an identity morphism[3].
This witness allows distinguishing between isomorphisms,
but not express the global constraint of an equivalence relation.

The analogue of univalence[4] says that Witness Duality can only be witnessed with a Loop Witness:

(A = B) ~= (A ~= B)

The Symmetry Forcing[5] property is related to the Product Witness.

This result was produced through helpful discussions with Tupshin Harper and Adam Nemecek.

References:

[1] "Avatar Graphs"
Sven Nilsen, 2020
https://github.com/advancedresearch/path_semantics/blob/master/papers-wip/avatar-graphs.pdf

[2] "Equivalence relation"
Wikipedia
https://en.wikipedia.org/wiki/Equivalence_relation

[3] "Category theory"
Wikipedia
https://en.wikipedia.org/wiki/Category_theory

[4] "Homotopy type theory – The univalence axiom"
Wikipedia
https://en.wikipedia.org/wiki/Homotopy_type_theory#The_univalence_axiom

[5] "Symmetry Forcing"
Sven Nilsen, 2020
https://github.com/advancedresearch/path_semantics/blob/master/papers-wip/symmetry-forcing.pdf