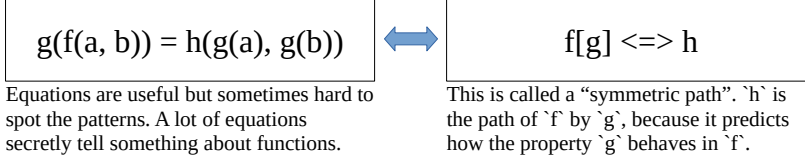


# Path Semantics Cheat Sheet

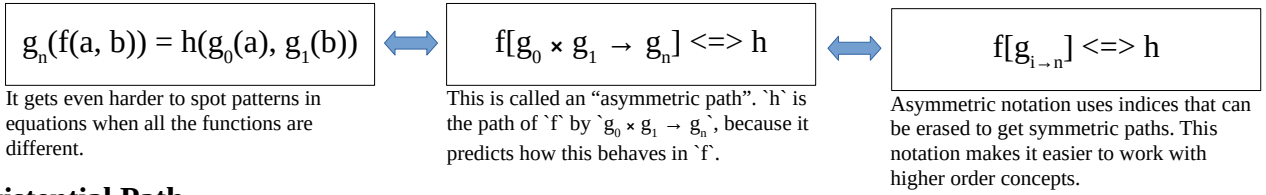
by Sven Nilsen, 2017

*Path semantics is used to study functions and how they are related to each other. The notation makes it easier to wrap one's head around high order concepts that are used to derive algorithms or do theorem proving.*

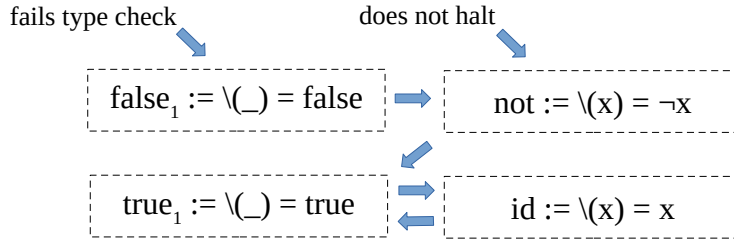
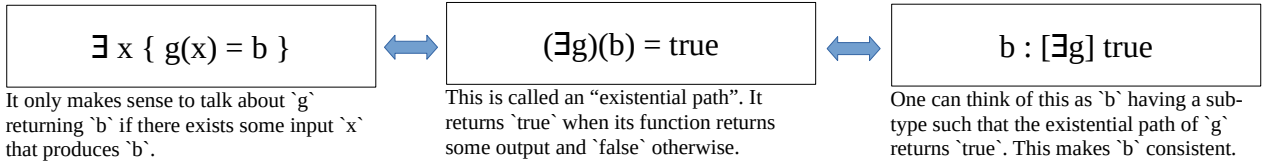
## Symmetric Path



## Asymmetric Path



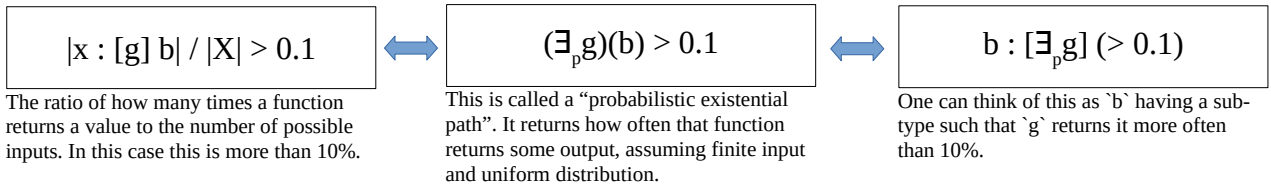
## Existential Path



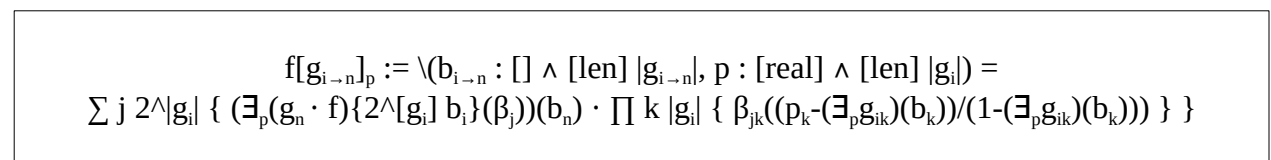
*Every double-existential path  $\exists \exists g$  is one of 4 functions. Strangely, some functions can be thought of as failing type checking or do not terminate on any input. All "normal" functions have double-existential paths `true\_1` or `id`. Those who got `id` are surjective, `true\_1` are injective.*

Every existential path returns a `bool` which is a small type, so there are many functions that share the same existential paths. For example, by rearranging the outputs of a function, its permutations all point to the same existential path. One output can also be repeated more or fewer times but not zero.

## Probabilistic Existential Path

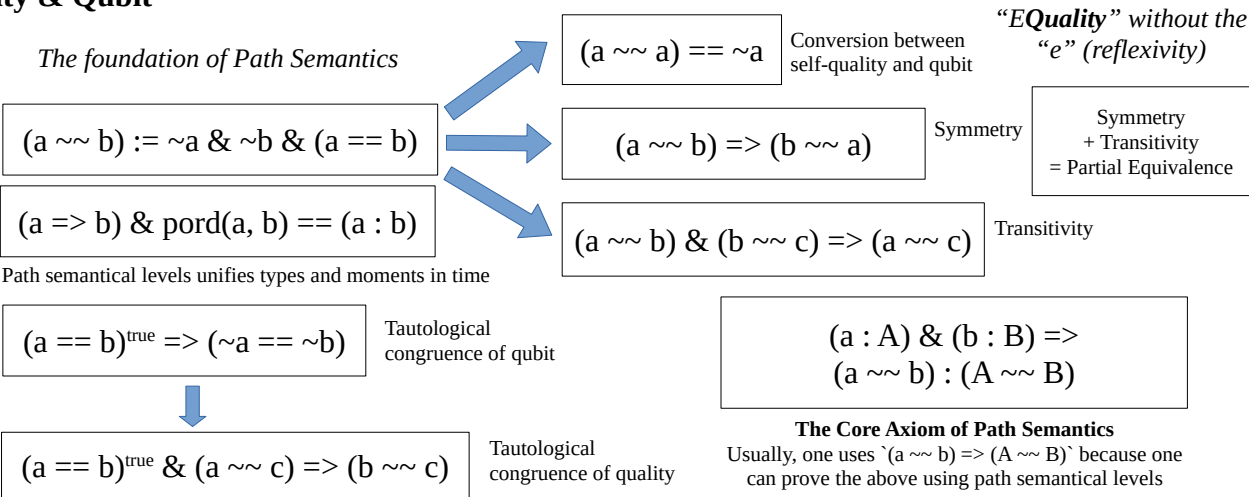


## Probabilistic Path



A probabilistic path finds sub-type probability of output from sub-type probability of input. It is an interesting function. :)

# Quality & Qubit



## Lifting equality to quality

