Curry - Howard

Haskell Curry

~

William Alvin Howard

Curry - Howard

Haskell

Correspondence

~

Brook

Curry

Curry ~ Howard ~ Lambek

The holy trinity

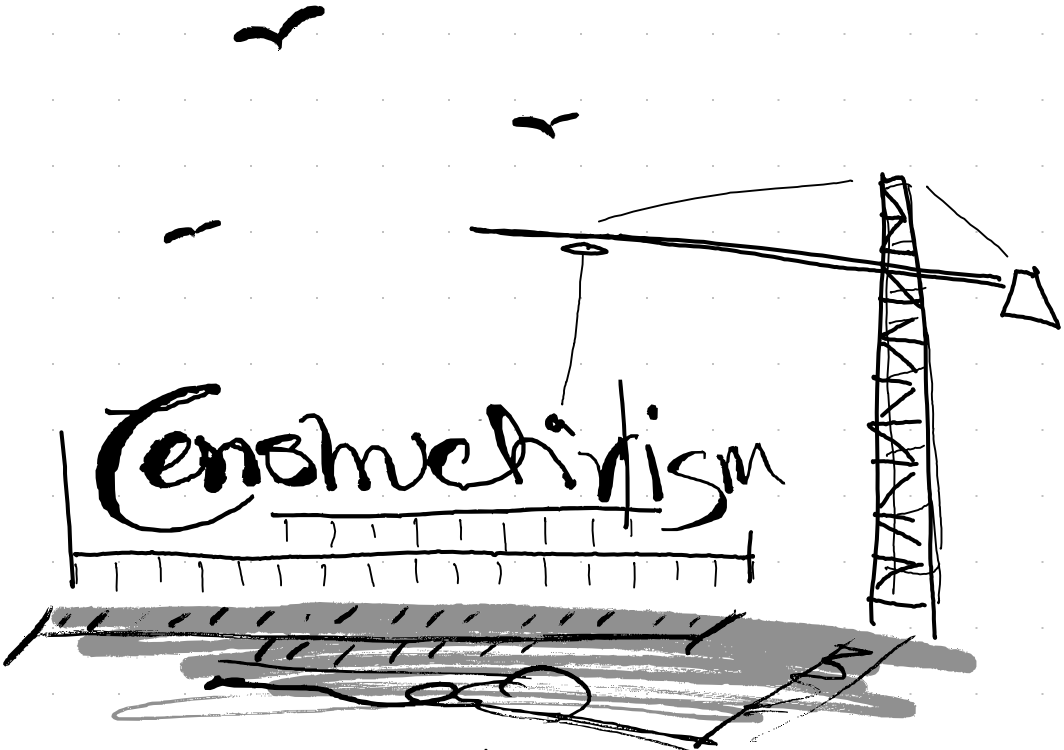
STLC

~

Intuitionistic Logic

~

CCCs



Sam Frohlich

Agenda

- What is constructivism
- Dependent types
- Curry-Howard
- ~~Lean~~ (using Curry Howard)
- HoTT

connecting STLC to constructive logic

Construct	Evidence
false	———
true	———
$A \wedge B$	eA and eB
$A \vee B$	eA or eB
$A \Rightarrow B$	eA $\rightarrow$ eB
$\neg A$	eA $\rightarrow$ false

Not constructive

LEM : $\neg A \vee A$
DNeg : $\neg\neg A \rightarrow A$
Peirce : $((A \Rightarrow B) \Rightarrow A) \Rightarrow A$

$((A \Rightarrow B) \Rightarrow A) \Rightarrow A$ $((a \rightarrow b) \rightarrow a)$

1. Assume $((A \Rightarrow B) \Rightarrow A)$
2. Proof by contra, ASS $\neg A$
3. Subproof showing $A \Rightarrow B$
4. | ASS A
5. | $\bot$ -- 4 contradicts 2
6. | B.
7. A -- 1 @ 3
8. $\bot$

Dependant types

values at the type level

```
> head :: [a] -> a
> head (x:xs) = x

ghci> head []

> data Nat = Z | S Nat

> two = S(S Z)

> data Vec (n:Nat) a where
>     VNil :: Vec 'Z a
>     VCons :: a
>           -> Vec n a
>           -> Vec ('S n) a
```

+
Plus
:+

```
> safeHead :: Vec (S'n) a -> a
> safeHead (VCons x xs) = x
```

Construct	Evidence Meaning
$\exists x \in X . A$	$e\, A^{y \in X}_{[y/x]}$ $\quad e\ y \in X$
$\forall x \in X . A$	$(y, e\ y \in X)$ $\xrightarrow[eA]{}$

Exists - Dependent pair

:: (n:Nat, Vec n Bool)

(0, VNil)
(1, VCons T VNil)

$$\sum_{(n:nat)} \text{Vec } n \text{ Bool}$$

Forall - Dependant Function

:: n:nat -> Vec n Bool

```
> data SNat :: Nat -> * where
>     SZ :: SNat 'Z
>     SS :: SNat n -> SNat (S n)

> replicate :: SNat n -> Vec n Bool
> replicate SZ = VNil
> replicate (SS n) = VCons T (replicate n)
```