The Truth about Types

Bartosz Milewski

Truth

Logic	Types	Categories
Truth intro	Unit type	Terminal object
$\dfrac{}{\top \ \text{true}}\top_I$	$() : ()$	

Proofs

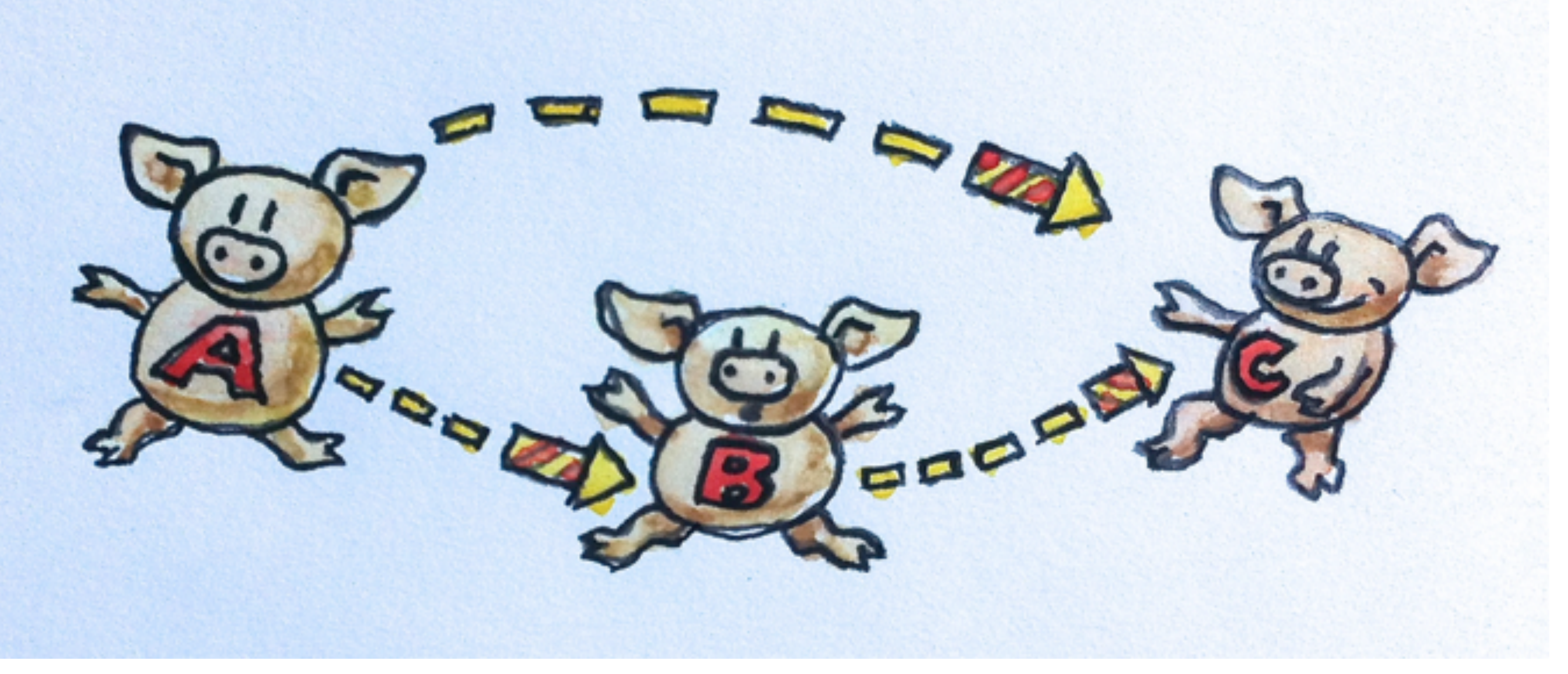
Logic	Types	Categories
Proof of proposition **A**	Type **A** is inhabited	Morphism from terminal object
$\dfrac{[\ldots]}{A}$	$\Gamma \vdash x : A$	$1 \to A$

Category Theory

Category

- A generalisation of a graph (transitive closure)
- Nodes are called objects: a, b, c…
- Arrows between objects are called morphisms:
  - `f :: a -> b`

- Arrows are composable:
  - **`f :: a -> b`**,
  - **`g :: b -> c`**,
  - **`g ○ f :: a -> c`** (always exists!)
- Composition is associative



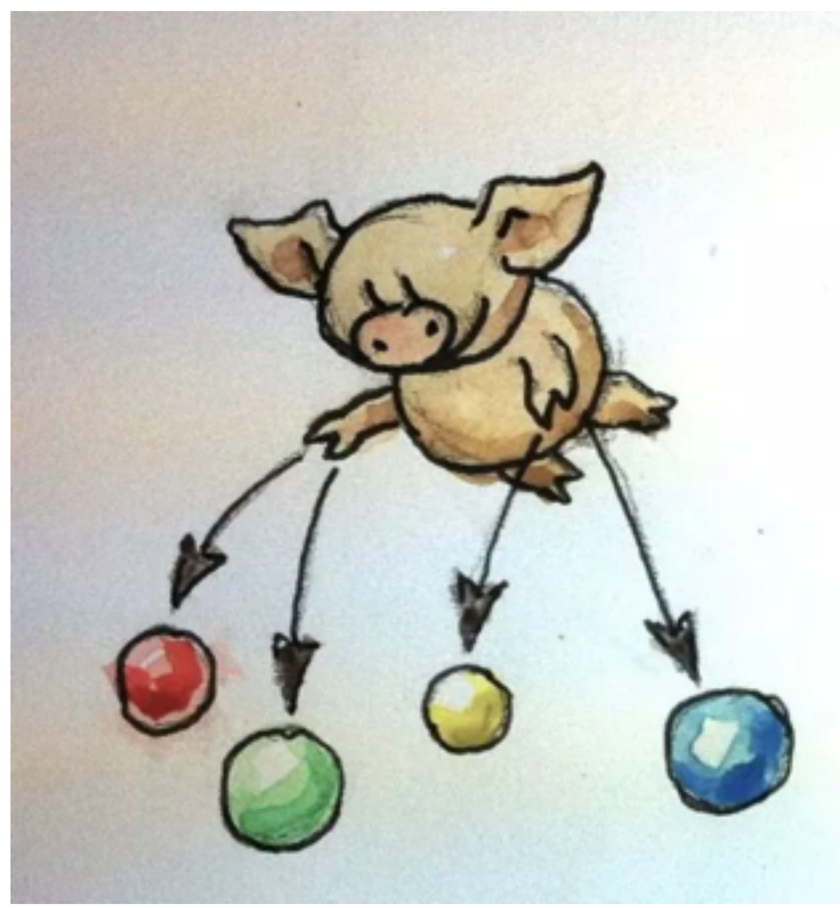
- Identity arrows (always exist!):
  - id_a :: a -> a,
  - id ∘ f = f,
  - g ∘ id = g

Set

- Category in which:
    - Objects are sets
    - Arrows are functions

Initial Object

- There is a unique arrow from initial object to any other object



Initial Object in Set

- Empty set $\varnothing$
- Unique function from $\varnothing$ `-> a`
- **`absurd :: Void -> a`**
- **`Void`** is the uninhabited type

Terminal Object

- There is a unique arrow from any object to terminal object

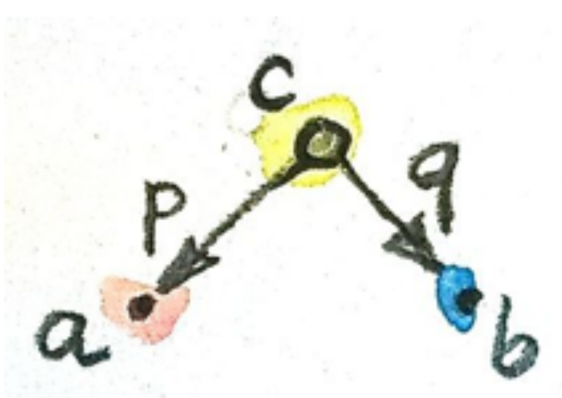
Terminal Object in Set

- Singleton set
- Unique function: for every element of set **a** return the single element of the singleton set
- Unit type **`()`** with one element **`()`**
- **`unit :: a -> ()`**
- **`unit _ -> ()`**

Universal Constructions

Product (Elimination)

- c is a product of a and b
- Two arrows p and q (projections)
- In Set: cartesian product, pairs of elements
- In logic: and (conjunction elimination)

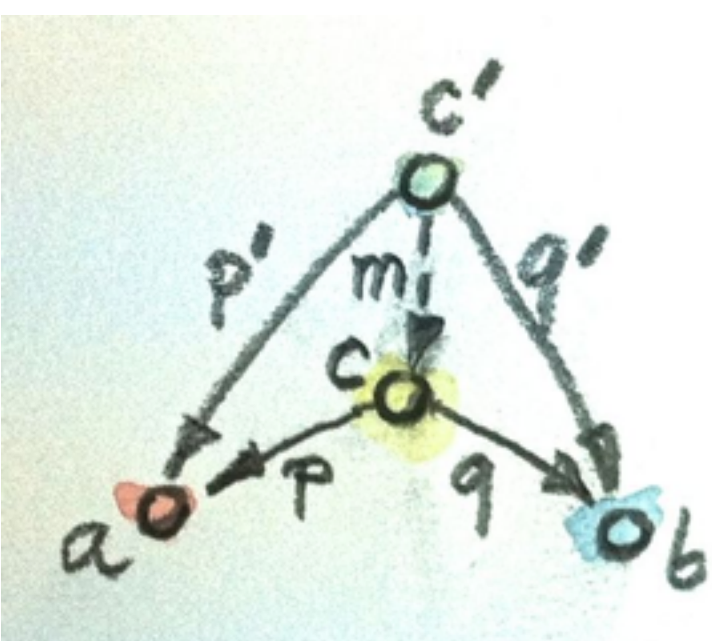


c :: (a, b)
p (a, b) = a
q (a, b) = b

$$\frac{a \wedge b}{a} \qquad \frac{a \wedge b}{b}$$

Product (Universality)

- Universal construction
- Product is the "best" candidate
- Any other candidate (c', p', q') uniquely factorizes through (c, p, q).

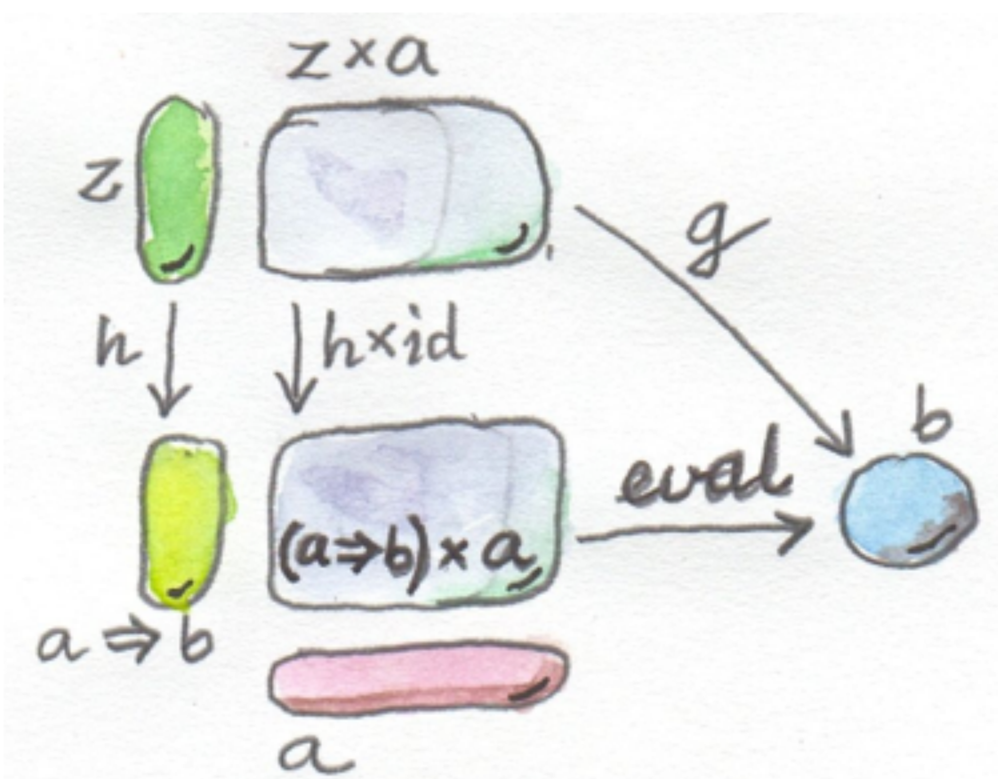


m :: c' -> c
p' = p ∘ m
q' = q ∘ m

Conjunction intro:
if a follows from c' (p')
and b follows from c' (q')
a ^ b (c) follows from c' (m)

Function Object

- Universal construction
- Logical implication

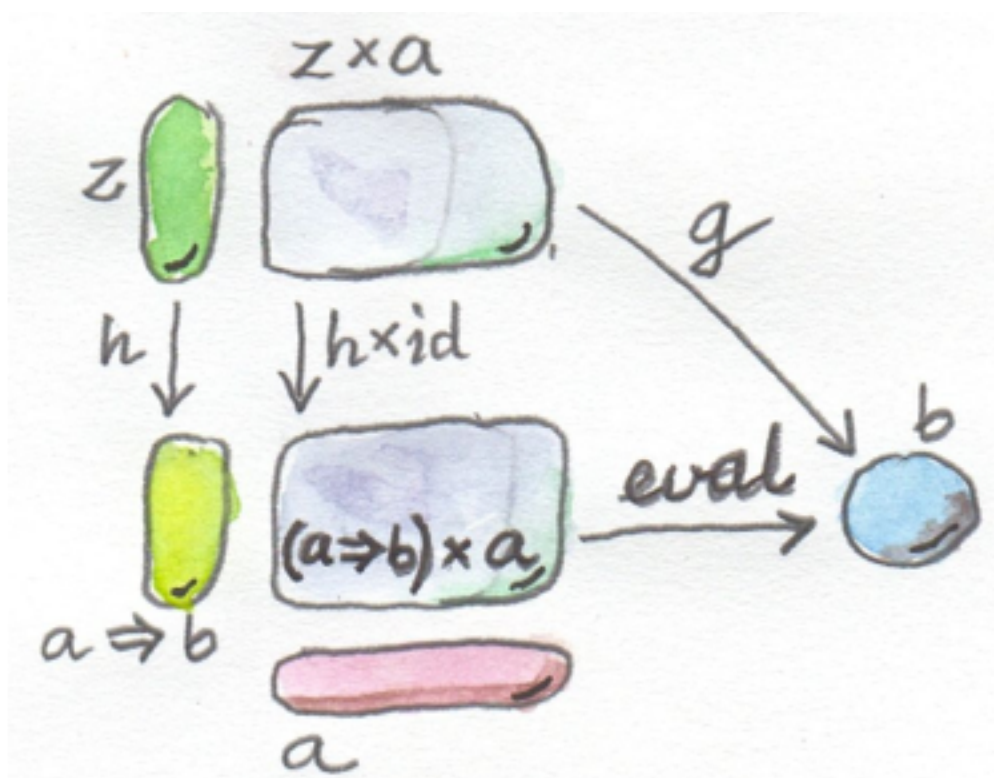


Modus ponens

$$\frac{(a{=}{>}b)\ \wedge\ a}{b}$$

Currying

- **`g`** as a function of two arguments **`z`** and **`a`**
- **`h`** is the curried version **`z->(a->b)`**



Negation

- Not A corresponds to A -> Void
- If A inhabited, A->Void not inhabited
- If A not inhabited (is Void), Void->Void is id_{Void}

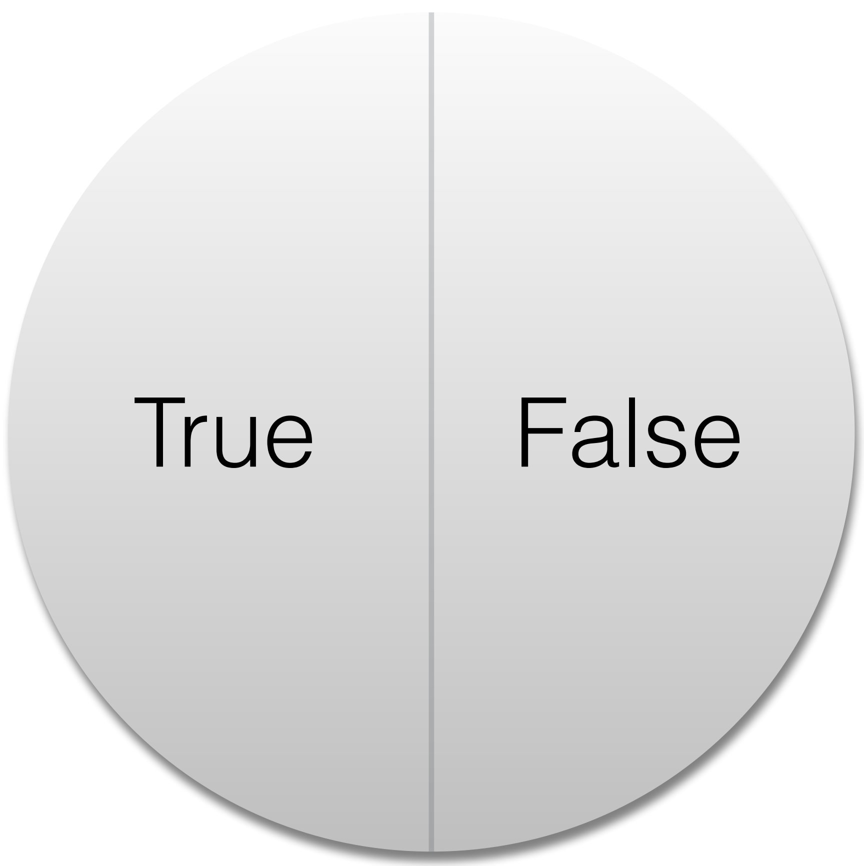
Cartesian Closed Category

- CCC
  - Has all products (cartesian)
  - Has all function objects (exponentials) (closed)
  - Has terminal object (nullary product)

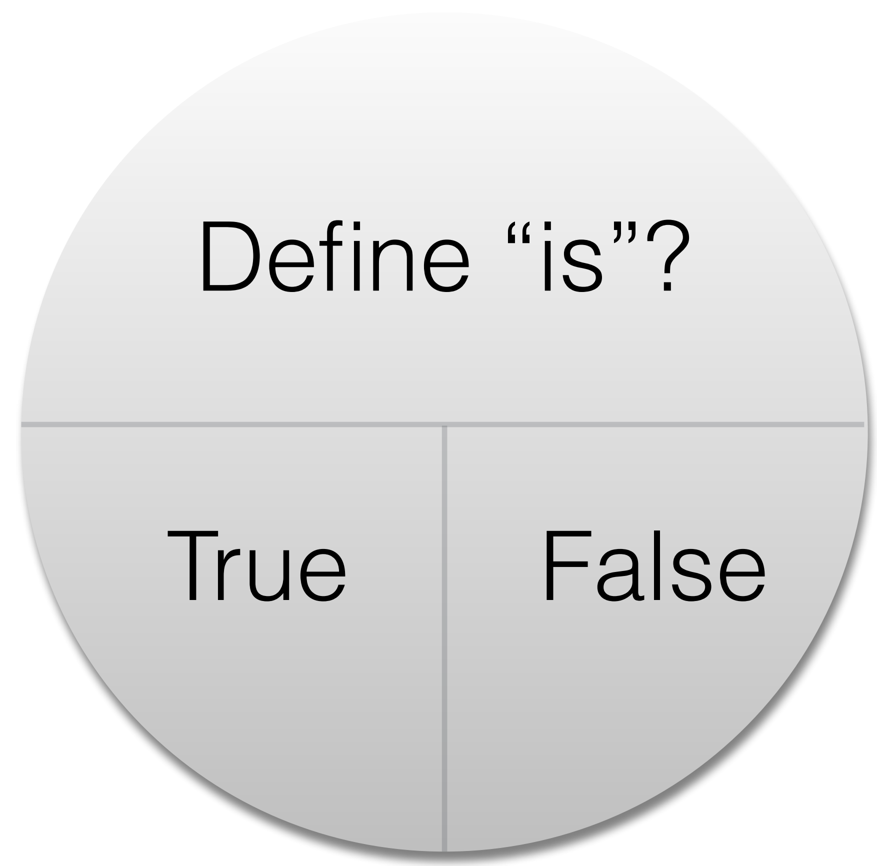
Curry-Howard-Lambek

- Lambek: CCC is a model for simply typed lambda calculus
    - Objects are types
    - Morphisms are terms
    - Environment Γ is a product of judgments **`a:A`**
    - Empty environment is **`():()`**

Logical Universes



Classical



Intuitionistic

Intuitionistic Logic

- No LEM
  - `A | (A->Void)` not provable
- No double negation elimination
  - `(A->Void)->Void` not the same as `A`
- Curry-Howard equivalence: simply typed lambda calculus equivalent to intuitionistic logic

Goedel Gentzen

- Classical logic can be embedded in intuitionistic logic
  - Classical logic = Intuitionistic + double negation elimination (or LEM)
- Map every classical formula to its double negation

Continuations

- Double negation: **`(a->Void)->Void`**
- More general: **`(a->r)->r`**
- **`a->r`** is a continuation
- CPS transform: **`a`** is identified with **`(a->r)->r`**
- Classical logic!

Yoneda's Lemma

- F is a functor from category C to Set
- Hom-set is the set of morphisms between object a and b, C(a, b)
- Fix a and C(a, _) is a functor from C to Set
- Yoneda: Nat(C(a, x), F x) ~ F a
- **`forall x . (a -> x) -> F x ~ F a`**
- Pick F = identity functor and you get CPS

Conclusions

- It's all the same:
  - Type theory (typed lambda calculus)
  - Category theory (Cartesian Closed)
  - Logic
- Lots of cross-pollination
- Grand Unified Theory? HoTT?