Formation

$$\frac{A : \mathrm{Type} \qquad B : A \to \mathrm{Type}}{W_{x:A}\, B\, x : \mathrm{Type}}$$

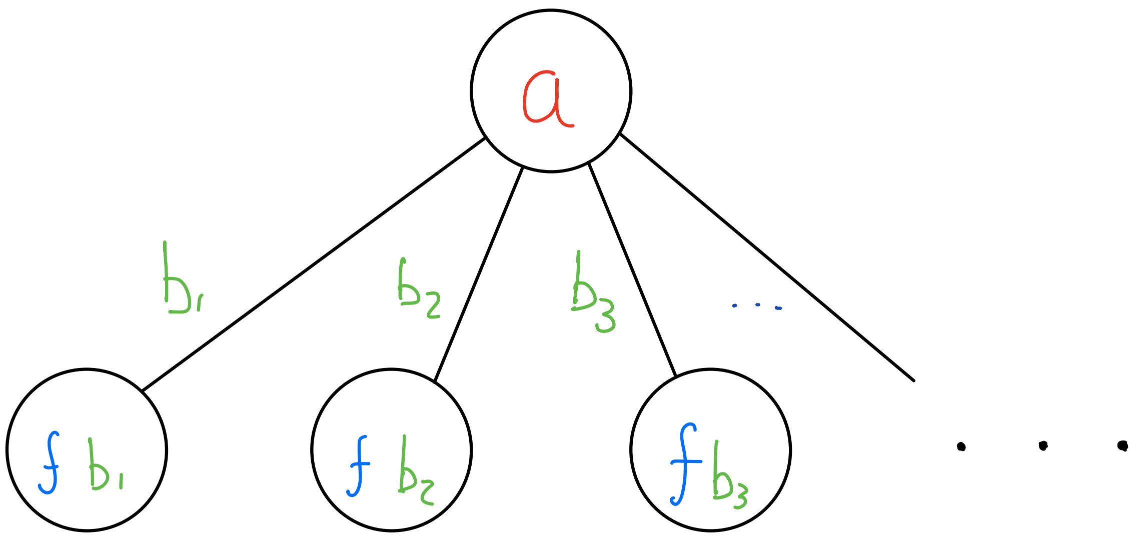
Note: in this scheme we do not have indices

Introduction

Common case
$B\,a$ is a finite type

$$\frac{\Gamma \vdash a : A \qquad \Gamma \vdash f : \overbrace{B\,a} \to W_{x:A}\, B\,x}{\Gamma \vdash \underbrace{\boxed{\mathrm{sup}_a}}_{\text{operation with label } a}\; f : W_{x:A}\, B\,x}$$

As trees:



As constructors:

for each $a : A$
we have a constructor sup_a
with $B\,a$ recursive arguments

Inductive nat

: Type

:=

| Z : nat

| S : nat → nat

Constructors :

$Z : nat$

$S : nat \to nat$

As a W-type

$A = \mathbb{1} + \mathbb{1}$

for S

for Z

$B(\mathrm{inl}\ tt) = \mathbb{0}$ no recursive arguments

$B(\mathrm{inr}\ tt) = \mathbb{1}$ one recursive argument

```
Inductive list
          (X : Type)
  : Type
  :=
| nil : list X
| cons : forall (x : X)
                (xs : list),
           list X
```

Constructors:

$\text{nil} : \text{list } X$

$\text{cons}_x : \text{list } X \to \text{list } X \quad \text{foreach } x : X$

As a W-type

$A = \mathbb{1} + X$ for cons

for nil

$B(\text{inl } tt) = 0$ no recursive arguments

$B(\text{inr } x) = \mathbb{1}$ one recursive argument