Maclane pentagone is some comonadic descent. Christopher Mary

1 Introduction

Blabla . Here $\sim$ Vila. $\wedge$ Encore.

2 Assoc

Blablabla

```
Print objects.
```

```
Inductive objects : Set :=
    letter : letters → objects | up_0 : objects → objects → objects
```

```
Print nodes.
```

```
Inductive nodes : objects → Set :=
    self : ∀ A : objects, A
  | at_left : ∀ A : objects, A → ∀ B : objects, A ∧0 B
  | at_right : ∀ A B : objects, B → A ∧0 B
```

```
Print lt_right.
```

```
Inductive lt_right : ∀ A : objects, A → A → Set :=
    lt_right_cons1 : ∀ (B : objects) (z : B) (C : objects),
                       self (C ∧0 B) <r at_right C z
  | lt_right_cons2 : ∀ (B C : objects) (x y : B),
                       x <r y → at_left x C <r at_left y C
  | lt_right_cons3 : ∀ (B C : objects) (x y : B),
                       x <r y → at_right C x <r at_right C y
```

```
Print arrows.
```

```
Inductive arrows : objects → objects → Set :=
    unitt : ∀ A : objects, arrows A A
  | bracket_left : ∀ A B C : objects,
                     arrows (A ∧0 B ∧0 C) ((A ∧0 B) ∧0 C)
  | up_1 : ∀ A B A0 B0 : objects,
             arrows A B → arrows A0 B0 → arrows (A ∧0 A0) (B ∧0 B0)
  | com : ∀ A B C : objects, arrows A B → arrows B C → arrows A C
```

```
Print arrows_assoc.
```

```
Inductive arrows_assoc : objects → objects → Set :=
    unitt_assoc : ∀ A : objects, arrows_assoc A A
  | bracket_left_assoc : ∀ A B C : objects,
                         arrows_assoc (A ∧0 B ∧0 C) ((A ∧0 B) ∧0 C)
  | bracket_right_assoc : ∀ A B C : objects,
                          arrows_assoc ((A ∧0 B) ∧0 C) (A ∧0 B ∧0 C)
  | up_1_assoc : ∀ A B A0 B0 : objects,
                 arrows_assoc A B →
                 arrows_assoc A0 B0 → arrows_assoc (A ∧0 A0) (B ∧0 B0)
  | com_assoc : ∀ A B C : objects,
                arrows_assoc A B → arrows_assoc B C → arrows_assoc A C

About arrows_coerce.


arrows_coerce : ∀ A B : objects, arrows A B → arrows_assoc A B

Print same.


Inductive same : ∀ A B : objects, arrows A B → arrows A B → Set :=
    same_refl : ∀ (A B : objects) (f : arrows A B), f ∼s f
  | same_trans : ∀ (A B : objects) (f g h : arrows A B),
                 f ∼s g → g ∼s h → f ∼s h
  | same_sym : ∀ (A B : objects) (f g : arrows A B), f ∼s g → g ∼s f
  | same_cong_com : ∀ (A B C : objects) (f f0 : arrows A B)
                      (g g0 : arrows B C),
                    f ∼s f0 → g ∼s g0 → (g <o f) ∼s (g0 <o f0)
  | same_cong_up_1 : ∀ (A B A0 B0 : objects) (f f0 : arrows A B)
                       (g g0 : arrows A0 B0),
                     f ∼s f0 → g ∼s g0 → (f ∧1 g) ∼s (f0 ∧1 g0)
  | same_cat_left : ∀ (A B : objects) (f : arrows A B),
                    (unitt B <o f) ∼s f
  | same_cat_right : ∀ (A B : objects) (f : arrows A B),
                     (f <o unitt A) ∼s f
  | same_cat_assoc : ∀ (A B C D : objects) (f : arrows A B)
                       (g : arrows B C) (h : arrows C D),
                     (h <o g <o f) ∼s ((h <o g) <o f)
  | same_bif_up_unit : ∀ A B : objects,
                       (unitt A ∧1 unitt B) ∼s unitt (A ∧0 B)
  | same_bif_up_com : ∀ (A B C A0 B0 C0 : objects)
                        (f : arrows A B) (g : arrows B C)
                        (f0 : arrows A0 B0) (g0 : arrows B0 C0),
                      ((g <o f) ∧1 (g0 <o f0)) ∼s (g ∧1 g0 <o f ∧1 f0)
  | same_bracket_left_5 : ∀ A B C D : objects,
```

$$
\begin{array}{l}
(\mathit{bracket\_left}\ (A \wedge 0\ B)\ C\ D <_o \\
\quad \mathit{bracket\_left}\ A\ B\ (C \wedge 0\ D)) \sim_s \\
(\mathit{bracket\_left}\ A\ B\ C \wedge 1\ \mathit{unitt}\ D <_o \\
\quad \mathit{bracket\_left}\ A\ (B \wedge 0\ C)\ D <_o \\
\quad \mathit{unitt}\ A \wedge 1\ \mathit{bracket\_left}\ B\ C\ D)
\end{array}
$$

Voila la fin