

# A type theory for cartesian closed bicategories

Marcelo Fiore and Philip Saville\*

University of Cambridge  
Department of Computer Science and Technology

\* now at University of Edinburgh  
School of Informatics

26th June 2019

## Bicategories

Categories with axioms 'up to isomorphism'.

Arise where composition is defined by universal property.

Examples throughout mathematics + theoretical computer science:

- Semantics of computation,
- Datatype models,
- Categorical logic,
- Categorical algebra.

## Cartesian closed bicategories

Cartesian closed categories 'up to isomorphism'.

Examples:

- Generalised species and cartesian distributors  
particularly for applications in higher category theory  
(Fiore, Gambino, Hyland, Winskel), (Fiore & Joyal)
- Categorical algebra (operads)  
(Gambino & Joyal)
- Game semantics (concurrent games)  
(Yamada & Abramsky, Winskel *et al.*, Paquet)

# Internal monoids

In a category with finite products:

$$1 \xrightarrow{e} M \xleftarrow{m} M \times M$$

Unit law

$$\begin{array}{ccccc} 1 \times M & \xrightarrow{e \times M} & M \times M & \xleftarrow{M \times e} & M \times 1 \\ & \searrow & \downarrow m & \swarrow & \\ & \cong & M & \leftarrow & \cong \end{array}$$

Assoc. law

$$\begin{array}{ccccc} (M \times M) \times M & \xrightarrow{\cong} & M \times (M \times M) & \xrightarrow{M \times m} & M \times M \\ m \times M \downarrow & & & & \downarrow m \\ M \times M & \xrightarrow{\quad m \quad} & & & M \end{array}$$

# Internal monoids

In a category with finite products:

$$1 \xrightarrow{e} M \xleftarrow{m} M \times M$$

In **Set**: monoids

In **Cat**: **strict** monoidal categories

Unit law

$$\begin{array}{ccccc} 1 \times M & \xrightarrow{e \times M} & M \times M & \xleftarrow{M \times e} & M \times 1 \\ & \searrow & \downarrow m & \swarrow & \\ & \cong & M & \cong & \end{array}$$

Assoc. law

$$\begin{array}{ccccc} (M \times M) \times M & \xrightarrow{\cong} & M \times (M \times M) & \xrightarrow{M \times m} & M \times M \\ m \times M \downarrow & & & & \downarrow m \\ M \times M & \xrightarrow{\quad m \quad} & & & M \end{array}$$

# Internal pseudomonoids

In **Cat**:

$$1 \xrightarrow{e} M \xleftarrow{m} M \times M$$

Unit 2-cells

$$\begin{array}{ccccc}
 1 \times M & \xrightarrow{e \times M} & M \times M & \xleftarrow{M \times e} & M \times 1 \\
 & \searrow \lambda \cong & \downarrow m & \swarrow \rho \cong & \\
 & & M & & 
 \end{array}$$

Diagram illustrating the Unit 2-cells. The top row shows the multiplication  $m$  in the monoid  $M$  applied to the product of the unit  $1$  and  $M$ . The bottom row shows the unit  $1$  and  $M$  themselves. The 2-cells  $\lambda$  and  $\rho$  (in red) represent the coherence of the unit laws. Blue arrows from the **data** box point to the 2-cells  $\lambda$  and  $\rho$ .

Assoc. 2-cell

$$\begin{array}{ccccc}
 (M \times M) \times M & \xrightarrow{\cong} & M \times (M \times M) & \xrightarrow{M \times m} & M \times M \\
 \downarrow m \times M & & & & \downarrow m \\
 M \times M & \xrightarrow{m} & & & M
 \end{array}$$

Diagram illustrating the Associativity 2-cell. The top row shows the multiplication  $m$  in the monoid  $M$  applied to the product of  $M$  and  $M$ . The bottom row shows the multiplication  $m$  in the monoid  $M$  applied to the product of  $M$  and  $M$ . The 2-cell  $\alpha$  (in red) represents the coherence of the associativity law. A blue arrow from the **data** box points to the 2-cell  $\alpha$ .

# Internal pseudomonoids

In Cat:

$$1 \xrightarrow{e} M \xleftarrow{m} M \times M$$

Unit 2-cells

$$\begin{array}{ccccc}
 1 \times M & \xrightarrow{e \times M} & M \times M & \xleftarrow{M \times e} & M \times 1 \\
 & \searrow & \downarrow m & \swarrow & \\
 & \searrow \cong & M & \swarrow \cong & \\
 & & & & 
 \end{array}$$

Diagram illustrating Unit 2-cells. The top row shows the multiplication  $m: M \times M \rightarrow M$  and the unit  $e: 1 \rightarrow M$ . The bottom row shows the multiplication  $m: M \times M \rightarrow M$ . The 2-cells are represented by blue arrows labeled  $\lambda$  and  $\rho$ , which are isomorphisms between the two ways of multiplying  $M \times M$  using the unit  $e$ . A box labeled "data" has blue arrows pointing to the  $\lambda$  and  $\rho$  2-cells.

Assoc. 2-cell

$$\begin{array}{ccccc}
 (M \times M) \times M & \xrightarrow{\cong} & M \times (M \times M) & \xrightarrow{M \times m} & M \times M \\
 \downarrow m \times M & & & & \downarrow m \\
 M \times M & \xrightarrow{m} & & & M
 \end{array}$$

Diagram illustrating the Associativity 2-cell. The top row shows the multiplication  $m: M \times M \rightarrow M$  and the unit  $e: 1 \rightarrow M$ . The bottom row shows the multiplication  $m: M \times M \rightarrow M$ . The 2-cell is represented by a blue arrow labeled  $\alpha$ , which is an isomorphism between the two ways of multiplying  $M \times M$  using the unit  $e$ .

+ triangle and pentagon laws

$\rightsquigarrow$  monoidal category

# Internal pseudomonoids

In **Cat**:

$$1 \xrightarrow{e} M \xleftarrow{m} M \times M$$

...likewise in any fp-bicategory

Unit 2-cells

$$\begin{array}{ccccc}
 1 \times M & \xrightarrow{e \times M} & M \times M & \xleftarrow{M \times e} & M \times 1 \\
 & & \downarrow m & & \\
 & \searrow \lambda \cong & & \swarrow \rho \cong & \\
 & & M & & 
 \end{array}$$

Unit 2-cells are represented by blue arrows labeled  $\lambda$  and  $\rho$  pointing from the top row to the bottom row, and by curved arrows labeled  $\cong$  representing the unit laws. A box labeled "data" has blue arrows pointing to the  $\lambda$  and  $\rho$  2-cells.

Assoc. 2-cell

$$\begin{array}{ccccc}
 (M \times M) \times M & \xrightarrow{\cong} & M \times (M \times M) & \xrightarrow{M \times m} & M \times M \\
 m \times M \downarrow & & & & \downarrow m \\
 M \times M & \xrightarrow{m} & & & M
 \end{array}$$

Associativity is represented by a red arrow labeled  $\alpha \cong$  pointing from the top row to the bottom row. A box labeled "data" has a blue arrow pointing to the  $\alpha$  2-cell.

+ triangle and pentagon laws

$\rightsquigarrow$  monoidal category




In a CCC every  $[X \Rightarrow X]$  becomes a monoid:

$$\left( 1 \xrightarrow{\text{Id}_X} [X \Rightarrow X] \stackrel{\circ}{\leftarrow} [X \Rightarrow X] \times [X \Rightarrow X] \right)$$

? In a cc-bicategory every  $[X \Rightarrow X]$  becomes a **pseudomonoid**:

$$\left( 1 \xrightarrow{\text{Id}_X} [X \Rightarrow X] \stackrel{\circ}{\leftarrow} [X \Rightarrow X] \times [X \Rightarrow X] \right)$$

  
need to check  
coherence laws  
(i.e. triangle + pentagon)

## Programme:

1. Construct a type theory  $\Lambda_{\text{ps}}^{\times, \rightarrow}$  for cartesian closed bicategories  
**(this work)**,
2. Use NBE to prove the type theory is coherent  
bicategorical version of [Fiore2002]  
**(my thesis)**,

## Programme:

1. Construct a type theory  $\Lambda_{\text{ps}}^{\times, \rightarrow}$  for cartesian closed bicategories  
**(this work)**,
2. Use NBE to prove the type theory is coherent  
bicategorical version of [Fiore2002]  
**(my thesis)**,

## Application:

Algebraic structure definable in every CCC

$\Rightarrow$  algebraic pseudo-structure definable in every cc-bicategory

A type theory  $\Lambda_{\text{ps}}^{\times, \rightarrow}$  that:

A type theory  $\Lambda_{\text{ps}}^{\times, \rightarrow}$  that:

1. Generalises the simply-typed lambda calculus,
2. Is reasonable for calculations,
3. Is *sound* and *complete*

A type theory  $\Lambda_{\text{ps}}^{\times, \rightarrow}$  that:

1. Generalises the simply-typed lambda calculus,
2. Is reasonable for calculations,
3. Is *sound* and *complete*  
*i.e.* freeness property for the syntactic model.

# Bicategories

# Bicategories

- Objects  $X \in ob(\mathcal{B})$ ,



# Bicategories

- Objects  $X \in ob(\mathcal{B})$ ,
- *Hom-categories*  $(\mathcal{B}(X, Y), \bullet, id)$ :

# Bicategories

- Objects  $X \in ob(\mathcal{B})$ ,
- Hom-categories  $(\mathcal{B}(X, Y), \bullet, id)$ :

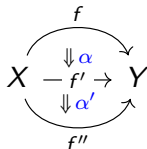
$$\begin{array}{l} \text{1-cells } X \xrightarrow{f} Y \\ \text{2-cells } X \begin{array}{c} \xrightarrow{f} \\ \Downarrow \alpha \\ \xrightarrow{f'} \end{array} Y \end{array}$$

# Bicategories

- Objects  $X \in ob(\mathcal{B})$ ,
- Hom-categories  $(\mathcal{B}(X, Y), \bullet, id)$ :

1-cells  $X \xrightarrow{f} Y$

2-cells  $X \begin{array}{c} \xrightarrow{f} \\ \Downarrow \alpha \\ \xrightarrow{f'} \end{array} Y$



# Bicategories

- Objects  $X \in ob(\mathcal{B})$ ,
- Hom-categories  $(\mathcal{B}(X, Y), \bullet, id)$ :

$$\begin{array}{c} \text{1-cells } X \xrightarrow{f} Y \\ \text{2-cells } X \begin{array}{c} \xrightarrow{f} \\ \Downarrow \alpha \\ \xrightarrow{f'} \end{array} Y \end{array}$$

- Functors

$$\begin{aligned} 1 &\xrightarrow{\text{Id}_X} \mathcal{B}(X, X) \\ \mathcal{B}(Y, Z) \times \mathcal{B}(X, Y) &\xrightarrow{\circ_{X,Y,Z}} \mathcal{B}(X, Z) \end{aligned}$$

# Bicategories

- Objects  $X \in ob(\mathcal{B})$ ,
- Hom-categories  $(\mathcal{B}(X, Y), \bullet, id)$ :

$$\begin{array}{l} \text{1-cells } X \xrightarrow{f} Y \\ \text{2-cells } X \begin{array}{c} \xrightarrow{f} \\ \Downarrow \alpha \\ \xrightarrow{f'} \end{array} Y \end{array}$$

- Functors

$$\begin{array}{l} 1 \xrightarrow{Id_X} \mathcal{B}(X, X) \\ \mathcal{B}(Y, Z) \times \mathcal{B}(X, Y) \xrightarrow{\circ_{X,Y,Z}} \mathcal{B}(X, Z) \end{array}$$

$$X \begin{array}{c} \xrightarrow{f} \\ \Downarrow \alpha \\ \xrightarrow{f'} \end{array} Y \begin{array}{c} \xrightarrow{g} \\ \Downarrow \beta \\ \xrightarrow{g'} \end{array} Z$$

# Bicategories

- Objects  $X \in ob(\mathcal{B})$ ,
- Hom-categories  $(\mathcal{B}(X, Y), \bullet, id)$ :

$$\begin{array}{l} \text{1-cells } X \xrightarrow{f} Y \\ \text{2-cells } X \begin{array}{c} \xrightarrow{f} \\ \Downarrow \alpha \\ \xrightarrow{f'} \end{array} Y \end{array}$$

- Functors

$$1 \xrightarrow{Id_X} \mathcal{B}(X, X)$$

$$\mathcal{B}(Y, Z) \times \mathcal{B}(X, Y) \xrightarrow{\circ_{X,Y,Z}} \mathcal{B}(X, Z)$$

- Invertible 2-cells

$$(h \circ g) \circ f \xRightarrow{a_{h,g,f}} h \circ (g \circ f)$$

$$Id_X \circ f \xRightarrow{l_f} f$$

$$g \circ Id_X \xRightarrow{r_g} g$$

subject to a triangle law and pentagon law.

# Cartesian closed bicategories

Bicategories  $\mathcal{B}$  equipped with *biuniversal* 1-cells

# Cartesian closed bicategories

Bicategories  $\mathcal{B}$  equipped with *biuniversal* 1-cells

$$(fp) \quad \pi_i : \Pi_n(A_1, \dots, A_n) \rightarrow A_i \quad (1 \leq i \leq n)$$

$$(cc) \quad eval : (A \Rightarrow B) \times A \rightarrow B$$

**NB:** Differ from the ‘cartesian bicategories’ of Carboni and Walters!



# Cartesian closed bicategories

Bicategories  $\mathcal{B}$  equipped with *biuniversal* 1-cells

$$(fp) \quad \pi_i : \Pi_n(A_1, \dots, A_n) \rightarrow A_i \quad (1 \leq i \leq n)$$

$$(cc) \quad eval : (A \Rightarrow B) \times A \rightarrow B$$

inducing families of equivalences

$$\mathcal{B}(X, \Pi_n(A_1, \dots, A_n)) \simeq \prod_{i=1}^n \mathcal{B}(X, A_i)$$

$$\mathcal{B}(X, A \Rightarrow B) \simeq \mathcal{B}(X \times A, B)$$

**NB:** Differ from the ‘cartesian bicategories’ of Carboni and Walters!

# Cartesian closed bicategories

Bicategories  $\mathcal{B}$  equipped with *biuniversal* 1-cells

$$(fp) \quad \pi_i : \Pi_n(A_1, \dots, A_n) \rightarrow A_i \quad (1 \leq i \leq n)$$

$$(cc) \quad eval : (A \Rightarrow B) \times A \rightarrow B$$

inducing families of equivalences

$$\begin{array}{ccc} & \xrightarrow{(\pi_1 \circ -, \dots, \pi_n \circ -)} & \\ \mathcal{B}(X, \Pi_n(A_1, \dots, A_n)) & \perp \simeq & \prod_{i=1}^n \mathcal{B}(X, A_i) \\ & \xleftarrow{\langle -, \dots, \rangle} & \\ & \text{(tupling)} & \end{array}$$

$$\begin{array}{ccc} & \xrightarrow{eval_{A,B} \circ (- \times A)} & \\ \mathcal{B}(X, A \Rightarrow B) & \perp \simeq & \mathcal{B}(X \times A, B) \\ & \xleftarrow{\lambda} & \\ & \text{(currying)} & \end{array}$$

**NB:** Differ from the 'cartesian bicategories' of Carboni and Walters!

## Substitution and composition

In any CCC:

$$\llbracket x_k[u_1/x_1, \dots, u_n/x_n] \rrbracket = \llbracket u_k \rrbracket = \pi_k \circ \langle \llbracket u_1 \rrbracket, \dots, \llbracket u_n \rrbracket \rangle$$

# Substitution and composition

In any CCC:

$$\llbracket x_k[u_1/x_1, \dots, u_n/x_n] \rrbracket = \llbracket u_k \rrbracket = \pi_k \circ \langle \llbracket u_1 \rrbracket, \dots, \llbracket u_n \rrbracket \rangle$$

In any **cc-bicategory**:

$$\llbracket x_k[u_1/x_1, \dots, u_n/x_n] \rrbracket = \llbracket u_k \rrbracket \cong \pi_k \circ \langle \llbracket u_1 \rrbracket, \dots, \llbracket u_n \rrbracket \rangle$$

# Substitution and composition

In any CCC:

$$\llbracket x_k[u_1/x_1, \dots, u_n/x_n] \rrbracket = \llbracket u_k \rrbracket = \pi_k \circ \langle \llbracket u_1 \rrbracket, \dots, \llbracket u_n \rrbracket \rangle$$

In any **cc-bicategory**:

$$\llbracket x_k[u_1/x_1, \dots, u_n/x_n] \rrbracket = \llbracket u_k \rrbracket \cong \pi_k \circ \langle \llbracket u_1 \rrbracket, \dots, \llbracket u_n \rrbracket \rangle$$

**Question:** what is **bicategorical** substitution?

An algebraic theory with substitution:

An algebraic theory with substitution:

- *Sorts*  $S$ ,

An algebraic theory with substitution:

- *Sorts*  $S$ ,
- *Constants*  $x_1 : X_1, \dots, x_n : X_n \vdash t(x_1, \dots, x_n) : Y$ ,



An algebraic theory with substitution:

- *Sorts*  $S$ ,
- *Constants*  $x_1 : X_1, \dots, x_n : X_n \vdash t(x_1, \dots, x_n) : Y$ ,
- *Variables*  $x_1 : X_1, \dots, x_n : X_n \vdash x_i : X_i$  ( $1 \leq i \leq n$ ),

An algebraic theory with substitution:

- *Sorts*  $S$ ,
- *Constants*  $x_1 : X_1, \dots, x_n : X_n \vdash t(x_1, \dots, x_n) : Y$ ,
- *Variables*  $x_1 : X_1, \dots, x_n : X_n \vdash x_i : X_i$  ( $1 \leq i \leq n$ ),
- A *substitution* rule

$$t, (u_1, \dots, u_n) \mapsto t[u_i/x_i]$$

An algebraic theory with substitution:

- *Sorts*  $S$ ,
- *Constants*  $x_1 : X_1, \dots, x_n : X_n \vdash t(x_1, \dots, x_n) : Y$ ,
- *Variables*  $x_1 : X_1, \dots, x_n : X_n \vdash x_i : X_i$  ( $1 \leq i \leq n$ ),
- *A substitution rule*

$$t, (u_1, \dots, u_n) \mapsto t[u_i/x_i]$$

such that

$$x_k[u_i/x_i] = u_k \quad (1 \leq k \leq n)$$

$$t[x_i/x_i] = t$$

$$t[u_i/x_i][v_j/y_j] = t[u_i[v_j/y_j]/x_i]$$

Abstract clone  $(S, \mathbb{C})$  = abstract theory of substitution:

Abstract clone  $(S, \mathbb{C})$  = abstract theory of substitution:

- *Sorts*  $S$ ,

Abstract clone  $(S, \mathbb{C}) =$  abstract theory of substitution:

- Sorts  $S$ ,
- Hom-sets  $\mathbb{C}(X_1, \dots, X_n; Y)$  of *operations*  $X_1, \dots, X_n \xrightarrow{t} Y$ ,

Abstract clone  $(S, \mathbb{C}) =$  abstract theory of substitution:

- *Sorts*  $S$ ,
- Hom-sets  $\mathbb{C}(X_1, \dots, X_n; Y)$  of *operations*  $X_1, \dots, X_n \xrightarrow{t} Y$ ,
- *Projections*  $X_1, \dots, X_n \xrightarrow{p_{X_1, \dots, X_n}^{(i)}} X_i \ (1 \leq i \leq n)$ ,

Abstract clone  $(S, \mathbb{C})$  = abstract theory of substitution:

- Sorts  $S$ ,
- Hom-sets  $\mathbb{C}(X_1, \dots, X_n; Y)$  of operations  $X_1, \dots, X_n \xrightarrow{t} Y$ ,
- Projections  $X_1, \dots, X_n \xrightarrow{p_{X_1, \dots, X_n}^{(i)}} X_i$  ( $1 \leq i \leq n$ ),
- *Substitution* mappings

$$\begin{aligned} \mathbb{C}(X_1, \dots, X_n; Y) \times \prod_{i=1}^n \mathbb{C}(\Gamma; X_i) &\rightarrow \mathbb{C}(\Gamma; Y) \\ t, (u_1, \dots, u_n) &\mapsto t[u_1, \dots, u_n] \end{aligned}$$



Abstract clone  $(S, \mathbb{C}) =$  abstract theory of substitution:

- Sorts  $S$ ,
- Hom-sets  $\mathbb{C}(X_1, \dots, X_n; Y)$  of operations  $X_1, \dots, X_n \xrightarrow{t} Y$ ,
- Projections  $X_1, \dots, X_n \xrightarrow{p_{X_1, \dots, X_n}^{(i)}} X_i$  ( $1 \leq i \leq n$ ),
- *Substitution* mappings

$$\begin{aligned} \mathbb{C}(X_1, \dots, X_n; Y) \times \prod_{i=1}^n \mathbb{C}(\Gamma; X_i) &\rightarrow \mathbb{C}(\Gamma; Y) \\ t, (u_1, \dots, u_n) &\mapsto t[u_1, \dots, u_n] \end{aligned}$$

such that

$$\begin{aligned} p^{(k)}[u_1, \dots, u_n] &= u_k \quad (1 \leq k \leq n) \\ t[p^{(1)}, \dots, p^{(n)}] &= t \\ t[u_{\bullet}][v_{\bullet}] &= t[v_{\bullet}[u_{\bullet}]] \end{aligned}$$

Abstract clone  $(S, \mathbb{C}) =$  abstract theory of substitution:

- Sorts  $S$ ,
- Hom-sets  $\mathbb{C}(X_1, \dots, X_n; Y)$  of operations  $X_1, \dots, X_n \xrightarrow{t} Y$ ,
- Projections  $X_1, \dots, X_n \xrightarrow{p_{X_1, \dots, X_n}^{(i)}} X_i$  ( $1 \leq i \leq n$ ),
- *Substitution* mappings

$$\begin{aligned} \mathbb{C}(X_1, \dots, X_n; Y) \times \prod_{i=1}^n \mathbb{C}(\Gamma; X_i) &\rightarrow \mathbb{C}(\Gamma; Y) \\ t, (u_1, \dots, u_n) &\mapsto t[u_1, \dots, u_n] \end{aligned}$$

such that

$$\begin{aligned} p^{(k)}[u_1, \dots, u_n] &= u_k \quad (1 \leq k \leq n) \\ t[p^{(1)}, \dots, p^{(n)}] &= t \\ t[u_\bullet][v_\bullet] &= t[v_\bullet[u_\bullet]] \end{aligned}$$

**Note:** every clone defines a category

**B**iclone  $(S, \mathbb{C})$  = abstract theory of bicategorical substitution:

**B**iclone  $(S, \mathbb{C})$  = abstract theory of bicategorical substitution:

- *Sorts*  $S$ ,

Biclone  $(S, \mathbb{C})$  = abstract theory of bicategorical substitution:

- *Sorts*  $S$ ,
- Hom-categories  $(\mathbb{C}(X_1, \dots, X_n; Y), \bullet, \text{id})$ ,

**B**iclone  $(S, \mathbb{C})$  = abstract theory of bicategorical substitution:

- *Sorts*  $S$ ,
- Hom-**categories**  $(\mathbb{C}(X_1, \dots, X_n; Y), \bullet, \text{id})$ ,
- *Projection* 1-cells  $p_{X_1, \dots, X_n}^{(i)} : X_1, \dots, X_n \rightarrow X_i$  ( $1 \leq i \leq n$ ),

**Bic**clone  $(S, \mathbb{C})$  = abstract theory of bicategorical substitution:

- *Sorts*  $S$ ,
- Hom-**categories**  $(\mathbb{C}(X_1, \dots, X_n; Y), \bullet, \text{id})$ ,
- *Projection* 1-cells  $p_{X_1, \dots, X_n}^{(i)} : X_1, \dots, X_n \rightarrow X_i$  ( $1 \leq i \leq n$ ),
- *Substitution* **functors**

$$\begin{aligned} \mathbb{C}(X_1, \dots, X_n; Y) \times \prod_{i=1}^n \mathbb{C}(\Gamma; X_i) &\rightarrow \mathbb{C}(\Gamma; Y) \\ t, (u_1, \dots, u_n) &\mapsto t[u_1, \dots, u_n] \\ \tau, (\sigma_1, \dots, \sigma_n) &\mapsto \tau[\sigma_1, \dots, \sigma_n] \end{aligned}$$

**Bic**clone  $(S, \mathbb{C})$  = abstract theory of bicategorical substitution:

- *Sorts*  $S$ ,
- Hom-**categories**  $(\mathbb{C}(X_1, \dots, X_n; Y), \bullet, \text{id})$ ,
- *Projection* 1-cells  $p_{X_1, \dots, X_n}^{(i)} : X_1, \dots, X_n \rightarrow X_i$  ( $1 \leq i \leq n$ ),
- *Substitution* **functors**

$$\begin{aligned} \mathbb{C}(X_1, \dots, X_n; Y) \times \prod_{i=1}^n \mathbb{C}(\Gamma; X_i) &\rightarrow \mathbb{C}(\Gamma; Y) \\ t, (u_1, \dots, u_n) &\mapsto t[u_1, \dots, u_n] \\ \tau, (\sigma_1, \dots, \sigma_n) &\mapsto \tau[\sigma_1, \dots, \sigma_n] \end{aligned}$$

- **Structural isomorphisms**

$$\begin{aligned} p^{(k)}[u_1, \dots, u_n] &\xrightarrow{\varrho_{u_\bullet}^{(k)}} u_k & (1 \leq k \leq n) \\ t[p^{(1)}, \dots, p^{(n)}] &\xRightarrow{\iota_t} t \\ t[u_\bullet][v_\bullet] &\xrightarrow{\text{assoc}_{t; u_\bullet; v_\bullet}} t[v_\bullet[u_\bullet]] \end{aligned}$$

subject to a triangle law and pentagon law.



**Bic**lone  $(S, \mathbb{C})$  = abstract theory of bicategorical substitution:

- *Sorts*  $S$ ,
- Hom-**categories**  $(\mathbb{C}(X_1, \dots, X_n; Y), \bullet, \text{id})$ ,
- *Projection* 1-cells  $p_{X_1, \dots, X_n}^{(i)} : X_1, \dots, X_n \rightarrow X_i$  ( $1 \leq i \leq n$ ),
- *Substitution* **functors**

$$\begin{aligned} \mathbb{C}(X_1, \dots, X_n; Y) \times \prod_{i=1}^n \mathbb{C}(\Gamma; X_i) &\rightarrow \mathbb{C}(\Gamma; Y) \\ t, (u_1, \dots, u_n) &\mapsto t[u_1, \dots, u_n] \\ \tau, (\sigma_1, \dots, \sigma_n) &\mapsto \tau[\sigma_1, \dots, \sigma_n] \end{aligned}$$

- **Structural isomorphisms**

$$\begin{aligned} p^{(k)}[u_1, \dots, u_n] &\xrightarrow{\varrho_{u_\bullet}^{(k)}} u_k \quad (1 \leq k \leq n) \\ t[p^{(1)}, \dots, p^{(n)}] &\xRightarrow{\iota_t} t \\ t[u_\bullet][v_\bullet] &\xrightarrow{\text{assoc}_{t; u_\bullet; v_\bullet}} t[v_\bullet[u_\bullet]] \end{aligned}$$

subject to a triangle law and pentagon law.

# A type theory for biclones

# A type theory for biclones

Hom-categories  $(\mathbb{C}(X_1, \dots, X_n; Y), \bullet, \text{id})$  (c.f. Hilken, Seely, Hirschowitz)

---

# A type theory for biclones

Hom-categories  $(\mathbb{C}(X_1, \dots, X_n; Y), \bullet, \text{id})$  (c.f. Hilken, Seely, Hirschowitz)

---

Judgements:

# A type theory for biclones

Hom-**categories**  $(\mathbb{C}(X_1, \dots, X_n; Y), \bullet, \text{id})$  (c.f. Hilken, Seely, Hirschowitz)

---

Judgements:

- Relating *terms*:  $\Gamma \vdash t : B$

# A type theory for biclones

Hom-**categories**  $(\mathbb{C}(X_1, \dots, X_n; Y), \bullet, \text{id})$  (c.f. Hilken, Seely, Hirschowitz)

---

Judgements:

- Relating *terms*:  $\Gamma \vdash t : B$
- Relating *rewrites*:  $\Gamma \vdash \tau : t \Rightarrow t' : B$

# A type theory for biclones

Hom-**categories**  $(\mathbb{C}(X_1, \dots, X_n; Y), \bullet, \text{id})$  (c.f. Hilken, Seely, Hirschowitz)

---

Judgements:

- Relating *terms*:  $\Gamma \vdash t : B$
- Relating *rewrites*:  $\Gamma \vdash \tau : t \Rightarrow t' : B$
- Equational theory  $\Gamma \vdash \tau \equiv \tau' : t \Rightarrow t' : B$

# A type theory for bicolones

Hom-categories  $(\mathbb{C}(X_1, \dots, X_n; Y), \bullet, \text{id})$  (c.f. Hilken, Seely, Hirschowitz)

---

Judgements:

- Relating *terms*:  $\Gamma \vdash t : B$
- Relating *rewrites*:  $\Gamma \vdash \tau : t \Rightarrow t' : B$
- Equational theory  $\Gamma \vdash \tau \equiv \tau' : t \Rightarrow t' : B$

Vertical composition: 
$$\frac{\Gamma \vdash \tau' : t' \Rightarrow t'' : B \quad \Gamma \vdash \tau : t \Rightarrow t' : B}{\Gamma \vdash \tau' \bullet \tau : t \Rightarrow t'' : B}$$

Identities: 
$$\frac{\Gamma \vdash t : B}{\Gamma \vdash \text{id}_t : t \Rightarrow t : B}$$



# A type theory for biclones

A *substitution functor*

$$\begin{aligned}\mathbb{C}(X_1, \dots, X_n; Y) \times \prod_{i=1}^n \mathbb{C}(\Gamma; X_i) &\rightarrow \mathbb{C}(\Gamma; Y) \\ t, (u_1, \dots, u_n) &\mapsto t[u_1, \dots, u_n] \\ \tau, (\sigma_1, \dots, \sigma_n) &\mapsto \tau[\sigma_1, \dots, \sigma_n]\end{aligned}$$

---

# A type theory for bicolones

A *substitution functor*

$$\begin{aligned}\mathbb{C}(X_1, \dots, X_n; Y) \times \prod_{i=1}^n \mathbb{C}(\Gamma; X_i) &\rightarrow \mathbb{C}(\Gamma; Y) \\ t, (u_1, \dots, u_n) &\mapsto t[u_1, \dots, u_n] \\ \tau, (\sigma_1, \dots, \sigma_n) &\mapsto \tau[\sigma_1, \dots, \sigma_n]\end{aligned}$$

*Explicit substitution:*

$$\frac{x_1 : A_1, \dots, x_n : A_n \vdash t : B \quad (\Delta \vdash u_i : A_i)_{i=1..n}}{\Delta \vdash t \{x_i \mapsto u_i\} : B}$$

$$\frac{x_1 : A_1, \dots, x_n : A_n \vdash \tau : t \Rightarrow t' : B \quad (\Delta \vdash \sigma_i : u_i \Rightarrow u'_i : A_i)_{i=1..n}}{\Delta \vdash \tau \{x_i \mapsto \sigma_i\} : t \{x_i \mapsto u_i\} \Rightarrow t' \{x_i \mapsto u'_i\} : B}$$

$\rightsquigarrow$  binds the variables  $x_1, \dots, x_n$

# A type theory for biclones

Structural isomorphisms  $\varrho^{(k)}, \iota, \text{assoc}$

---

# A type theory for biclones

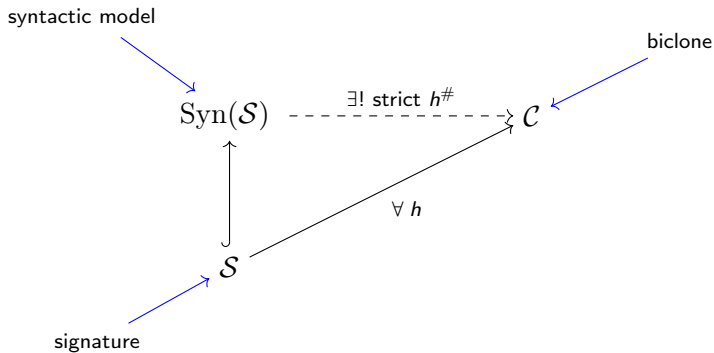
Structural isomorphisms  $\varrho^{(k)}, \iota, \text{assoc}$

Distinguished invertible rewrites e.g.:

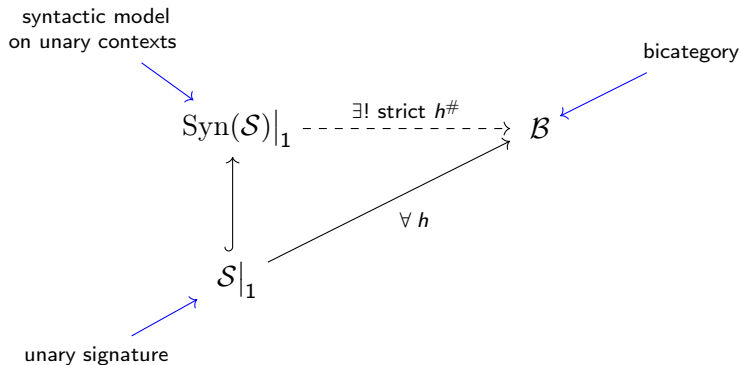
$$\frac{(\Delta \vdash u_i : A_i)_{i=1, \dots, n}}{x_1 : A_1, \dots, x_n : A_n \vdash \varrho_{u_\bullet}^{(k)} : x_k \{x_i \mapsto u_i\} \xrightarrow{\cong} u_k : A_k} \quad (1 \leq k \leq n)$$

The syntactic model is free

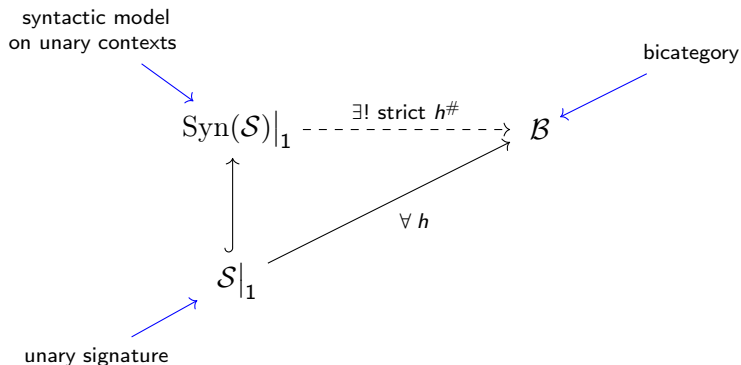
# The syntactic model is free



# The syntactic model is free



# The syntactic model is free



~~~~~> An internal language for bicategories.





1-cells

$$\pi_i : \Pi_n(A_1, \dots, A_n) \rightarrow A_i \quad (1 \leq i \leq n)$$

Adjoint equivalences

$$\begin{array}{ccc} & (\pi_1 \circ -, \dots, \pi_n \circ -) & \\ & \curvearrowright & \\ \mathcal{B}(X, \Pi_n(A_1, \dots, A_n)) & \perp \simeq & \prod_{i=1}^n \mathcal{B}(X, A_i) \\ & \curvearrowleft & \\ & \langle -, \dots, = \rangle & \end{array}$$

## A type theory for fp-bicategories

1-cells  $\pi_i : \prod_n(A_1, \dots, A_n) \rightarrow A_i \quad (1 \leq i \leq n)$

---

# A type theory for fp-bicategories

1-cells  $\pi_i : \prod_n(A_1, \dots, A_n) \rightarrow A_i \quad (1 \leq i \leq n)$

---

Projections  $\frac{}{p : \prod_n(A_1, \dots, A_n) \vdash \pi_i(p) : A_i} \quad (1 \leq i \leq n)$

## A type theory for fp-bicategories

$$\begin{array}{ccc} & (\pi_1 \circ -, \dots, \pi_n \circ -) & \\ & \curvearrowright & \\ \text{Equivalences } \mathcal{B}(X, \Pi_n(A_1, \dots, A_n)) & \perp \simeq & \prod_{i=1}^n \mathcal{B}(X, A_i) \\ & \curvearrowleft & \\ & \langle -, \dots, = \rangle & \end{array}$$

---

# A type theory for fp-bicategories

$$\text{Equivalences } \mathcal{B}(X, \Pi_n(A_1, \dots, A_n)) \perp \simeq \prod_{i=1}^n \mathcal{B}(X, A_i)$$

$$\begin{array}{c} \xrightarrow{(\pi_1 \circ -, \dots, \pi_n \circ -)} \\ \xleftarrow{\langle -, \dots, = \rangle} \end{array}$$

$$\varpi^{(i)} \bullet (\pi_i \circ (-)) \left( \frac{\pi_i \circ u \Rightarrow t_i : A_i \quad (i = 1, \dots, n)}{u \Rightarrow \langle t_1, \dots, t_n \rangle : \Pi_n(A_1, \dots, A_n)} \right) \text{p}^\dagger(-, \dots, =)$$

for a counit  $(\varpi^{(i)} : \pi_i \circ \langle t_1, \dots, t_n \rangle \Rightarrow t_i : A_i)_{i=1, \dots, n}$

# A type theory for fp-bicategories

$$\text{Equivalences } \mathcal{B}(X, \Pi_n(A_1, \dots, A_n)) \perp \simeq \prod_{i=1}^n \mathcal{B}(X, A_i)$$

$(\pi_1 \circ -, \dots, \pi_n \circ -)$   
 $\langle -, \dots, = \rangle$

syntactic sugar

$$\varpi^{(i)} \bullet \pi_i \{(-)\} \left( \frac{\pi_i \{u\} \Rightarrow t_i : A_i \quad (i = 1, \dots, n)}{u \Rightarrow \text{tup}(t_1, \dots, t_n) : \Pi_n(A_1, \dots, A_n)} \right) p^\dagger(-, \dots, =)$$

for a counit  $(\varpi^{(i)} : \pi_i \{\text{tup}(t_1, \dots, t_n)\} \Rightarrow t_i : \Pi_n(A_1, \dots, A_n))_{i=1, \dots, n}$

## A type theory for fp-bicategories

$$\varpi^{(i)} \bullet \pi_i \{(-)\} \left( \frac{\pi_i \{u\} \Rightarrow t_i : A_i \quad (i = 1, \dots, n)}{u \Rightarrow \text{tup}(t_1, \dots, t_n) : \Pi_n(A_1, \dots, A_n)} \right) \text{p}^\dagger(-, \dots, =)$$

---



## A type theory for fp-bicategories

$$\varpi^{(i)} \bullet \pi_i \{(-)\} \left( \frac{\pi_i \{u\} \Rightarrow t_i : A_i \quad (i = 1, \dots, n)}{u \Rightarrow \text{tup}(t_1, \dots, t_n) : \prod_n(A_1, \dots, A_n)} \right) p^\dagger(-, \dots, =)$$

---

Tupling map 
$$\frac{(\Gamma \vdash t_i : A_i)_{i=1, \dots, n}}{\Gamma \vdash \text{tup}(t_1, \dots, t_n) : \prod_n(A_1, \dots, A_n)}$$

# A type theory for fp-bicategories

$$\varpi^{(i)} \bullet \pi_i \{(-)\} \left( \frac{\pi_i \{u\} \Rightarrow t_i : A_i \quad (i = 1, \dots, n)}{u \Rightarrow \text{tup}(t_1, \dots, t_n) : \prod_n(A_1, \dots, A_n)} \right) p^\dagger(-, \dots, =)$$

Tupling map 
$$\frac{(\Gamma \vdash t_i : A_i)_{i=1, \dots, n}}{\Gamma \vdash \text{tup}(t_1 \dots, t_n) : \prod_n(A_1, \dots, A_n)}$$

Counit ( $\beta$ -law) 
$$\frac{(\Gamma \vdash t_i : A_i)_{i=1, \dots, n}}{\Gamma \vdash \varpi_{t_\bullet}^{(k)} : \pi_k \{\text{tup}(t_1 \dots, t_n)\} \xrightarrow{\cong} t_k : A_k} \quad (1 \leq k \leq n)$$

# A type theory for fp-bicategories

$$\varpi^{(i)} \bullet \pi_i \{(-)\} \left( \frac{\pi_i \{u\} \Rightarrow t_i : A_i \quad (i = 1, \dots, n)}{u \Rightarrow \text{tup}(t_1, \dots, t_n) : \prod_n(A_1, \dots, A_n)} \right) p^\dagger(-, \dots, =)$$

Tupling map 
$$\frac{(\Gamma \vdash t_i : A_i)_{i=1, \dots, n}}{\Gamma \vdash \text{tup}(t_1, \dots, t_n) : \prod_n(A_1, \dots, A_n)}$$

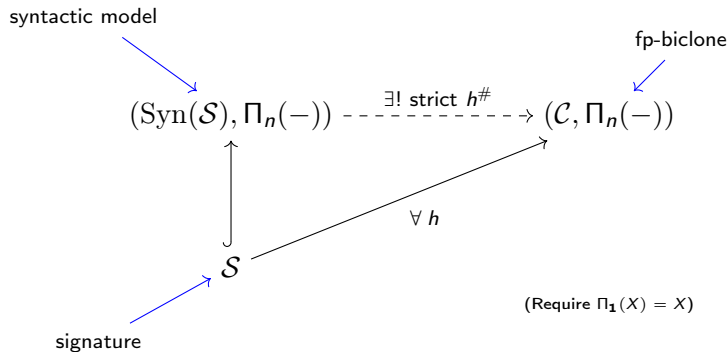
Counit ( $\beta$ -law) 
$$\frac{(\Gamma \vdash t_i : A_i)_{i=1, \dots, n}}{\Gamma \vdash \varpi_{t_\bullet}^{(k)} : \pi_k \{ \text{tup}(t_1, \dots, t_n) \} \cong t_k : A_k} \quad (1 \leq k \leq n)$$

Mediating 2-cell 
$$\frac{(\Gamma \vdash \alpha_i : \pi_i \{u\} \Rightarrow t_i : A_i)_{i=1, \dots, n}}{\Gamma \vdash p^\dagger(\alpha_1, \dots, \alpha_n) : u \Rightarrow \text{tup}(t_1, \dots, t_n) : \prod_n(A_1, \dots, A_n)}$$

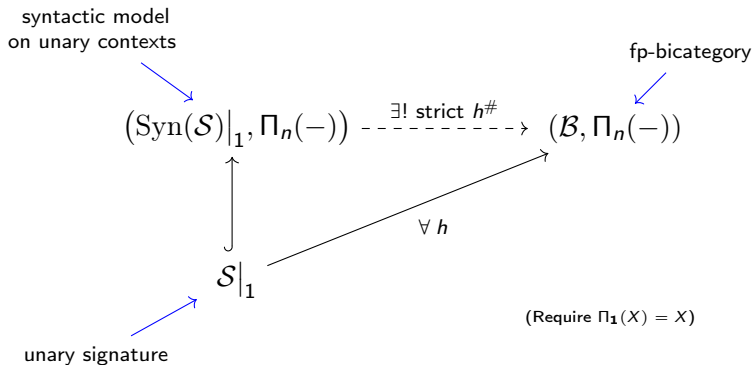
+ three equational rules.

$\rightsquigarrow$   $\eta$ -law is derivable

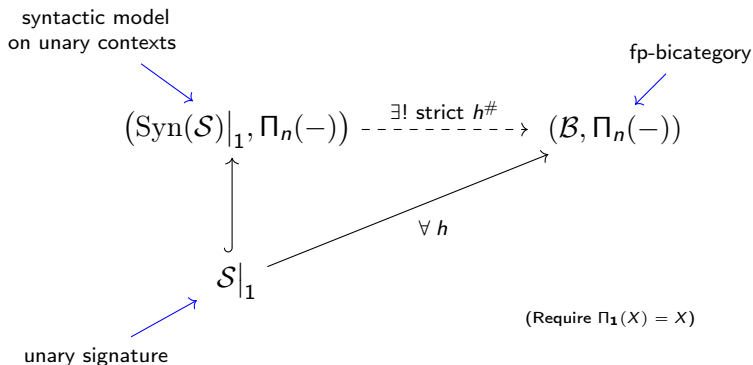
# The syntactic model is free



# The syntactic model is free

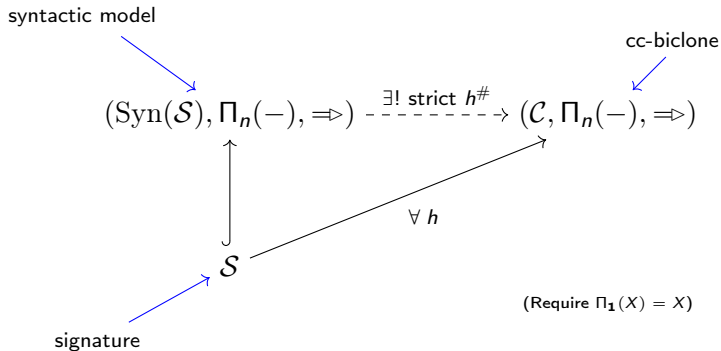


# The syntactic model is free



$\rightsquigarrow$  An internal language for fp-bicategories.  
derived from definition of biadjoint

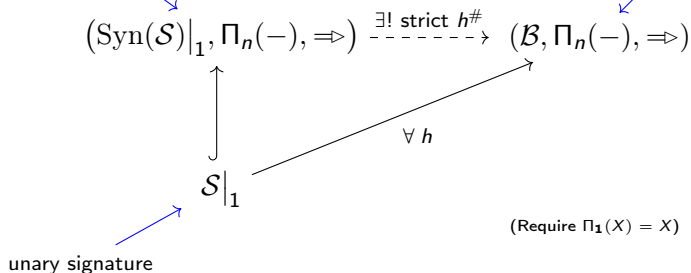
# The syntactic model is free



# The syntactic model is free

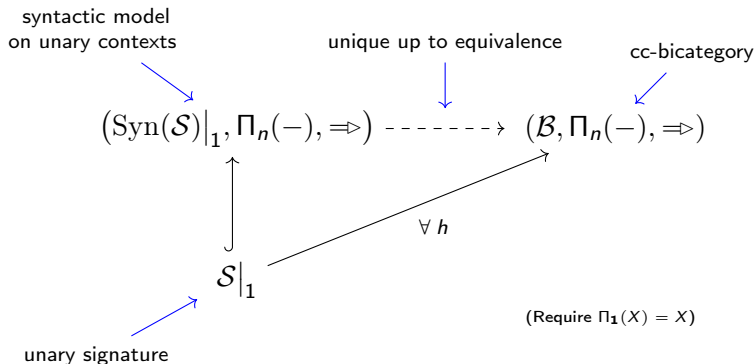
syntactic model  
on unary contexts

cc-bicategory with  
*strict* products

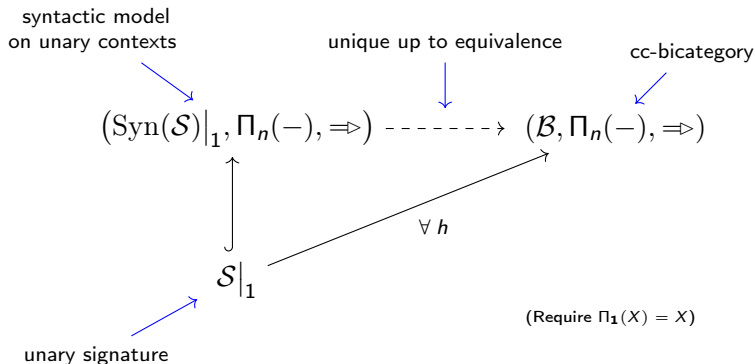




# The syntactic model is free



# The syntactic model is free



$\rightsquigarrow$  An internal language for cartesian closed bicategories.

## STLC up to isomorphism

Embedding of STLC-terms to  $\Lambda_{\text{ps}}^{\times, \rightarrow}$ -terms:

$$x_k \mapsto x_k$$

$$\pi_k(t) \mapsto \pi_k \{ \langle t \rangle \}$$

$$\langle t_1, \dots, t_n \rangle \mapsto \text{tup}(\langle t_1 \rangle, \dots, \langle t_n \rangle)$$

$$\text{app}(t, u) \mapsto \text{eval} \{ \langle t \rangle, \langle u \rangle \}$$

$$\lambda x. t \mapsto \lambda x. \langle t \rangle$$

---

# STLC up to isomorphism

Embedding of STLC-terms to  $\Lambda_{\text{ps}}^{\times, \rightarrow}$ -terms:

$$x_k \mapsto x_k$$

$$\pi_k(t) \mapsto \pi_k \{ \langle t \rangle \}$$

$$\langle t_1, \dots, t_n \rangle \mapsto \text{tup}(\langle t_1 \rangle, \dots, \langle t_n \rangle)$$

$$\text{app}(t, u) \mapsto \text{eval} \{ \langle t \rangle, \langle u \rangle \}$$

$$\lambda x. t \mapsto \lambda x. \langle t \rangle$$

---

$$(\text{STLC terms } \Gamma \vdash t : B) / \beta\eta \cong (\Lambda_{\text{ps}}^{\times, \rightarrow}\text{-terms } \Gamma \vdash t : B) / \cong_B^\Gamma$$

$$t \cong_B^\Gamma t' \Leftrightarrow \Gamma \vdash \tau : t \xRightarrow{\cong} t' : B$$

for some invertible  $\tau$

Key properties of  $\Lambda_{\text{ps}}^{\times, \rightarrow}$ :

Key properties of  $\Lambda_{\text{ps}}^{\times, \rightarrow}$ :

1. Principled development  $\Rightarrow$  few rules,
2. An internal language for cc-bicategories,
3. STLC up-to-isomorphism.



1-cells

$$\mathrm{eval}_{A,B} : (A \Rightarrow B) \times A \rightarrow B$$

Adjoint equivalences

$$\begin{array}{ccc} & \xrightarrow{\mathrm{eval}_{A,B} \circ (- \times A)} & \\ \mathcal{B}(X, A \Rightarrow B) & \perp \simeq & \mathcal{B}(X \times A, B) \\ & \xleftarrow{\lambda} & \end{array}$$



# Rules for exponentials

$$\epsilon \bullet \text{eval} \{(-) \{ \text{inc}_x \}, x\} \left( \frac{\text{eval} \{u \{ \text{inc}_x \}, x\} \Rightarrow t : B}{u \Rightarrow \lambda x. t : A \Rightarrow B} \right) e^\dagger(x. -)$$

explicit weakening by  $x$   $\curvearrowright$   $(x : A)$   $\leftarrow$  free variable in context

$$\frac{}{f : A \Rightarrow B, x : A \vdash \text{eval}(f, x) : B} \text{eval} \quad \frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x. t : A \Rightarrow B} \text{lam}$$

$$\frac{\Gamma, x : A \vdash t : B}{\Gamma, x : A \vdash \epsilon_t : \text{eval} \{(\lambda x. t) \{ \text{inc}_x \}, x\} \Rightarrow t : B} \epsilon\text{-intro } (\beta\text{-rule})$$

$$\frac{\Gamma, x : A \vdash \alpha : \text{eval} \{u \{ \text{inc}_x \}, x\} \Rightarrow t : B \quad \Gamma \vdash u : A \Rightarrow B}{\Gamma \vdash e^\dagger(x. \alpha) : u \Rightarrow \lambda x. t : A \Rightarrow B} e^\dagger(x. \alpha)\text{-intro}$$

+ three equational rules

$\rightsquigarrow$   $\eta$ -rule derivable