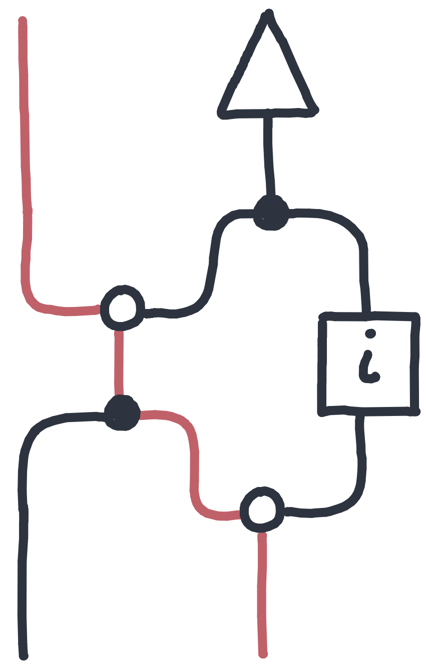
String Diagrams for Premonoidal Categories

Mario Román j.w.w. Paweł Sobociński
University of Oxford and Tallinn University of Technology

Università di Pisa, 30 maggio 2024



one-time-pad() :
unif() $\to u$
modify $(u \oplus \bullet)$
get() $\to a$
modify $(u^{-1} \oplus \bullet)$
return (a)

Promonads and String Diagrams for Effectful Categories, (Román, 2022).

ArXiv: 2205.07664

String Diagrams for Premonoidal Categories,

ArXiv: 2305.06075 (Román, Sobociński, 2023).

OUTLINE

1. String Diagrams
2. Premonoidals
3. Runtime as a Resource
4. Programs

Part 1. String Diagrams

Monoids and Lists

Proofs about monoids use lists: the inverse of a multiplication is the reverse multiplication of the inverses.

PROOF. $x \cdot y \cdot y^{-1} \cdot x^{-1} = x \cdot x^{-1} = e.$

However, if we are pedantic, we could say the following proof is more formal.

PROOF. $(x \cdot y) \cdot (y^{-1} \cdot x^{-1}) = (x \cdot (y \cdot y^{-1})) \cdot x^{-1} = (x \cdot e) \cdot x^{-1} = x \cdot x^{-1} = e.$

The first and the second are both formal, thanks to the following result.

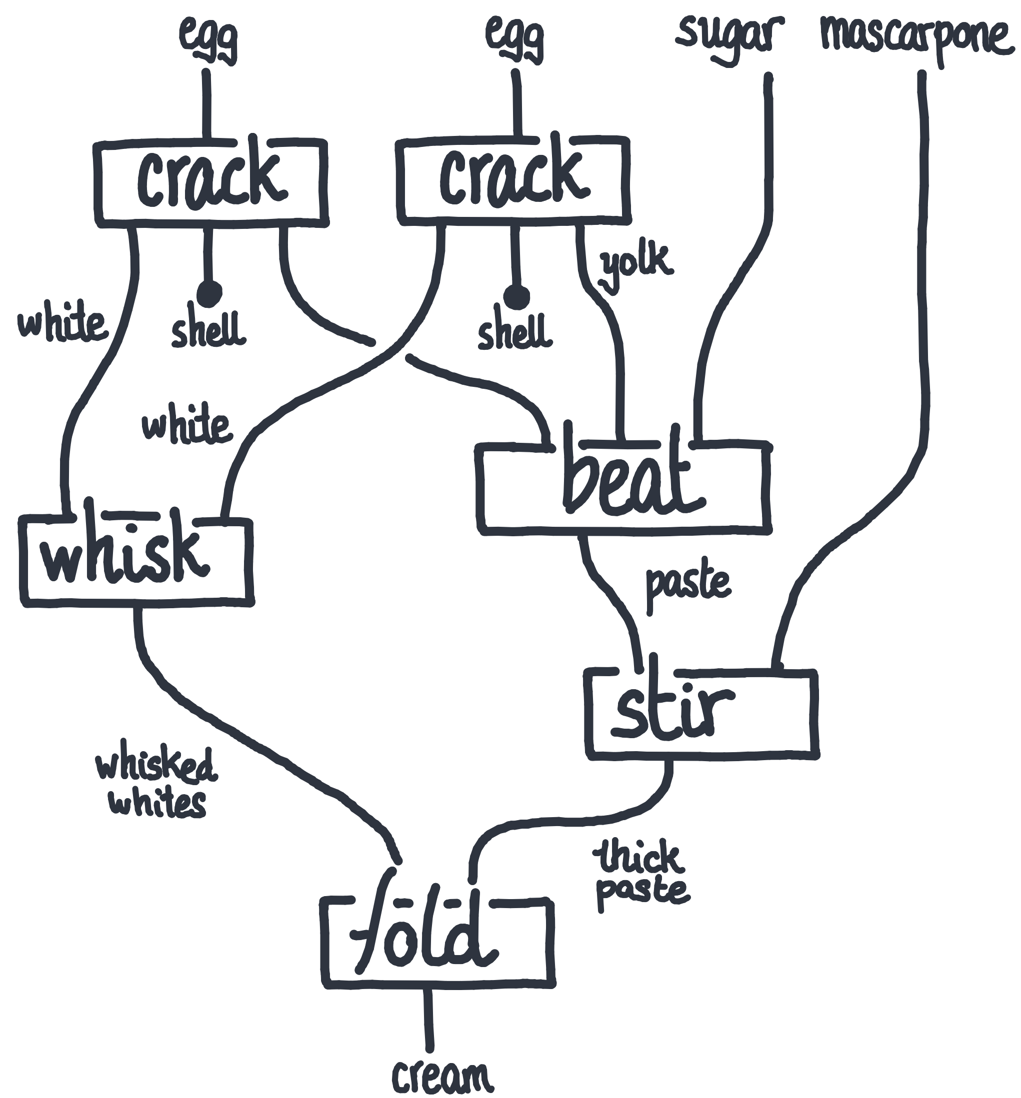
THEOREM. Lists with concatenation form the free monoid over some generators.

Lists simplify monoids. This talk is about string diagrams simplifying processes.

Permutative String Diagrams

DEFINITION (Joyal, Street). A permutative **string diagram** is an acyclic hypergraph

- such that every vertex appears once as input and once as output;
- labelled by a typed signature on hyperedges; and
- endowed with distinguished input/output nodes.



Strict Symmetric Monoidal Category

DEFINITION. A (strict symmetric) monoidal category $\mathbb{C}$ consists of a monoid of objects, or resources, $(\mathbb{C}_{obj}, \otimes, I)$ and a collection of morphisms, or processes, $\mathbb{C}(X;Y)$, indexed by $X \in \mathbb{C}_{obj}$ and $Y \in \mathbb{C}_{obj}$, with operations of

sequential composition, $(⨾) : \mathbb{C}(X;Y) \times \mathbb{C}(Y;Z) \longrightarrow \mathbb{C}(X;Z)$;
parallel composition, $(\otimes) : \mathbb{C}(X;Y) \times \mathbb{C}(X';Y') \longrightarrow \mathbb{C}(X \otimes X'; Y \otimes Y')$;
identities, $id_X : \mathbb{C}(X;X)$;
and symmetries, $\sigma_{X,Y} : \mathbb{C}(X \otimes Y; Y \otimes X)$.

Satisfying all the following axioms.

Strict Symmetric Monoidal Category

Axioms.

(1) $f \mathbin{;} id_Y = f = id_X \mathbin{;} f$;

(2) $f \mathbin{;} (g \mathbin{;} h) = (f \mathbin{;} g) \mathbin{;} h$;

(3) $f \otimes id_I = f = id_I \otimes f$;

(4) $f \otimes (g \otimes h) = (f \otimes g) \otimes h$;

(5) $id_A \otimes id_B = id_{A \otimes B}$;

(6) $(f \mathbin{;} g) \otimes (f' \mathbin{;} g') = (f \otimes f') \mathbin{;} (g \otimes g')$;

(7) $\sigma_{I,X} = id_X = \sigma_{X,I}$;

(8) $\sigma_{X,Y \otimes Z} = (\sigma_{X,Y} \otimes id_Z) \mathbin{;} (id_Y \otimes \sigma_{X,Z})$;

(9) $\sigma_{X \otimes Y,Z} = (id_X \otimes \sigma_{Y,Z}) \mathbin{;} (\sigma_{X,Z} \otimes id_Y)$;

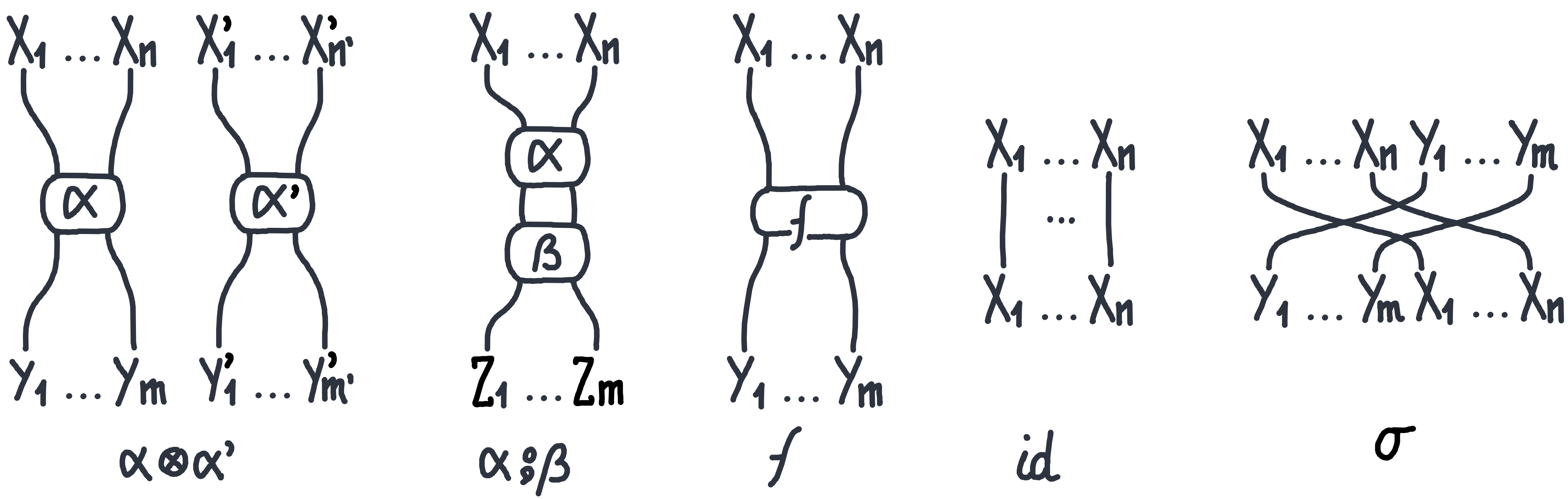
(10) $\sigma_{X,Y} \mathbin{;} \sigma_{Y,X} = id_X \otimes id_Y$;

(11) $\sigma_{X,Y} \mathbin{;} (f \otimes g) = (g \otimes f) \mathbin{;} \sigma_{X',Y'}$;

(12) $(\sigma_{X,Y} \otimes id) \mathbin{;} (id \otimes \sigma_{X,Z}) \mathbin{;} (\sigma_{Y,Z} \otimes id) =$
$(id \otimes \sigma_{Y,Z}) \mathbin{;} (\sigma_{X,Z} \otimes id) \mathbin{;} (id \otimes \sigma_{X,Y})$.

String Diagrams for Monoidal Categories

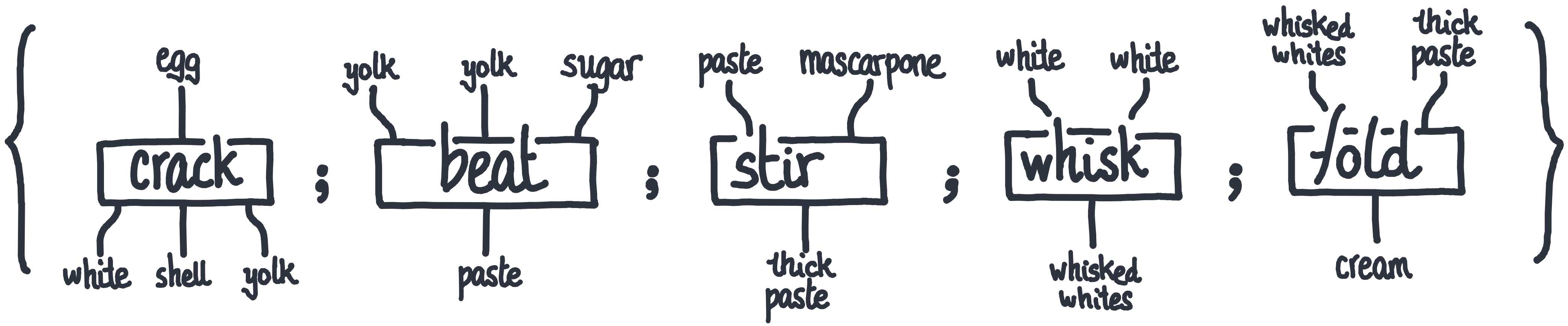
THEOREM (Joyal, Street). String diagrams form a symmetric monoidal category.



String Diagrams for Monoidal Categories

THEOREM (Joyal, Street). Permutative string diagrams form the free strict symmetric monoidal category over a monoidal signature. They are sound and complete for symmetric monoidal categories.

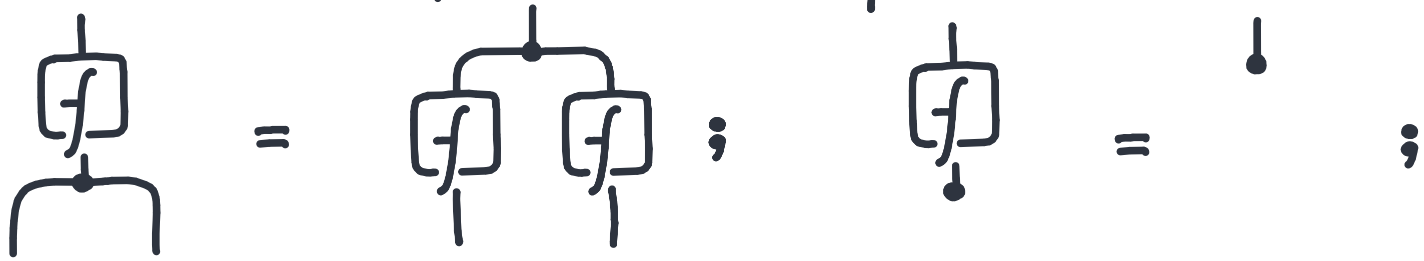
REMARK. A monoidal signature (or, polygraph) is a set of objects and a set of generators with an input and output lists of objects.



Monoidal Categories can Do Probability

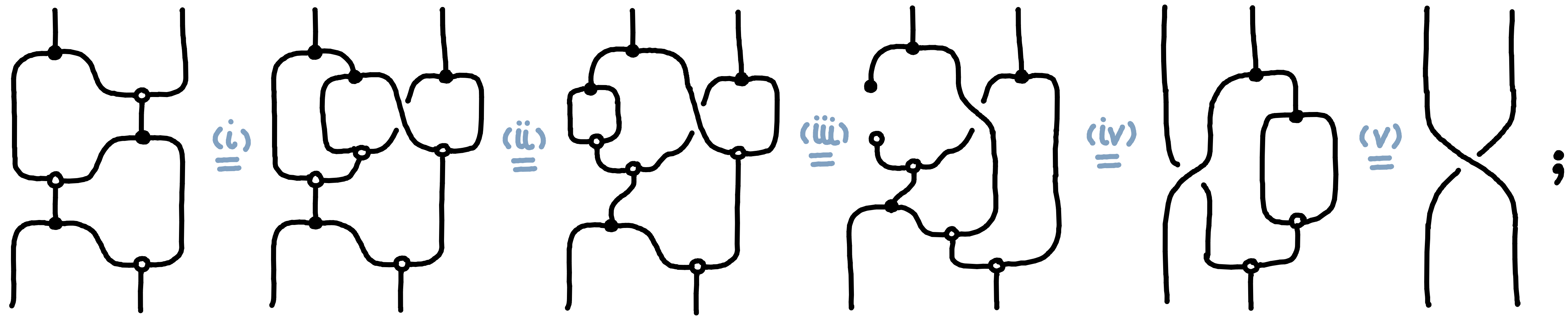
Consider two special copy/discard generators: { , }. Synthetic probability theory studies the interaction with these generators.

A morphism is carlesian if it can be copied and discarded.



Gaducci, Corradini (1999), Cho, Jacobs (2019), Fritz (2020).

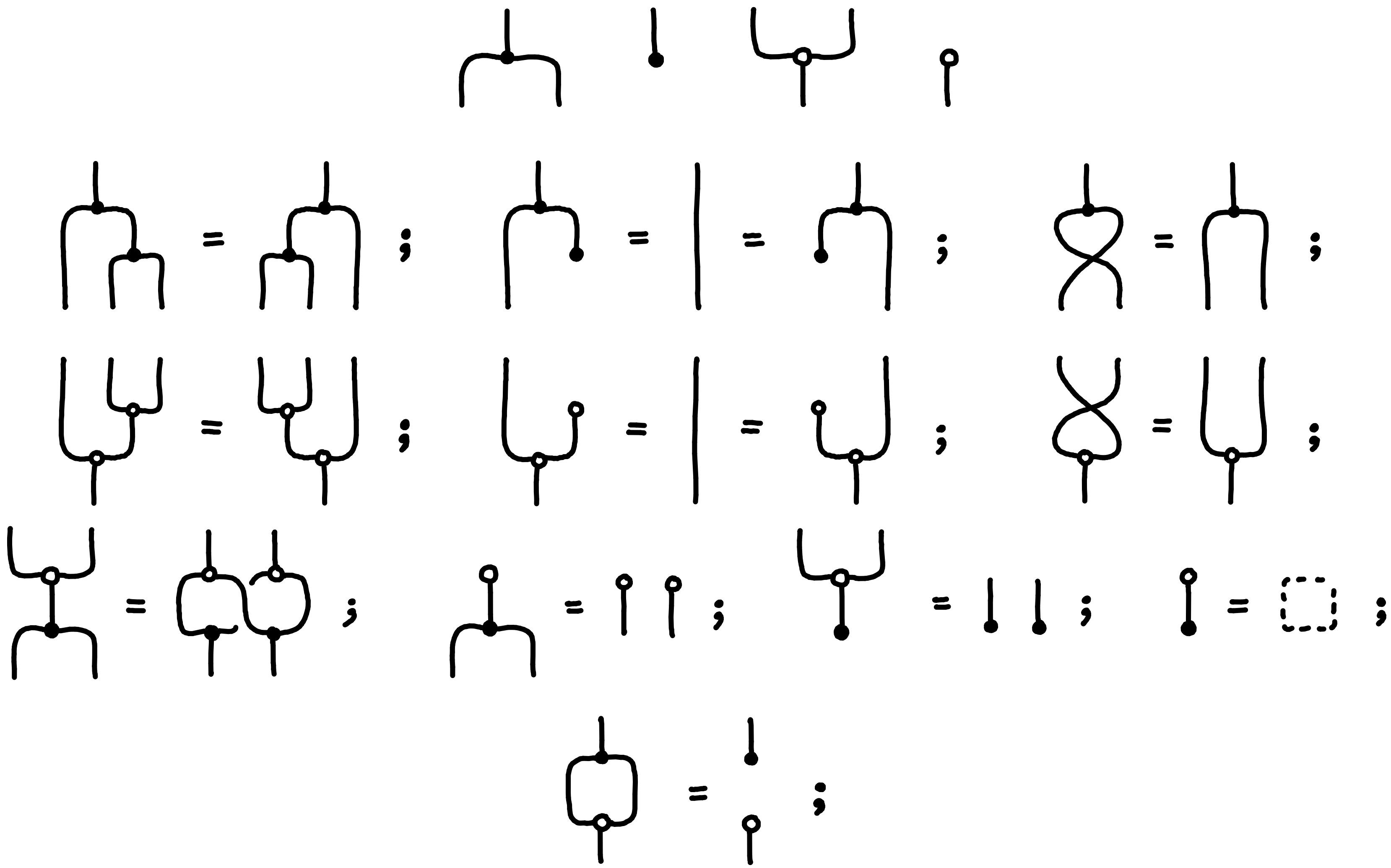
Monoidal Categories can Do Programs



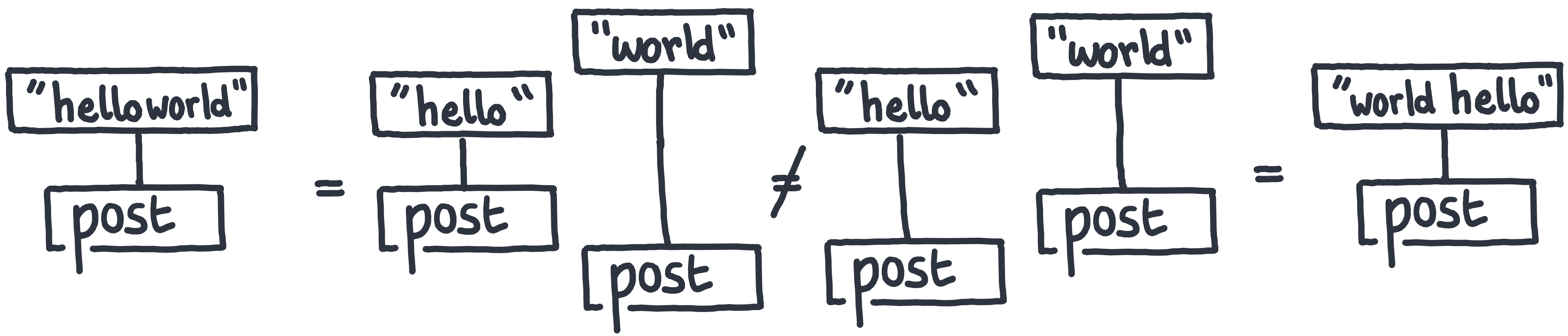

```
do (x,y):                   
   x⊕y → y                  
   x⊕y → x       }   ≡   { do (x,y):
   x⊕y → y                     return (y,x)
   return (x,y)
```

Erbele, Sobociński

Monoidal Categories can Do Programs



Monoidal Categories cannot Do Effects



Monoidal categories cannot capture global effects: we should not interchange effects.

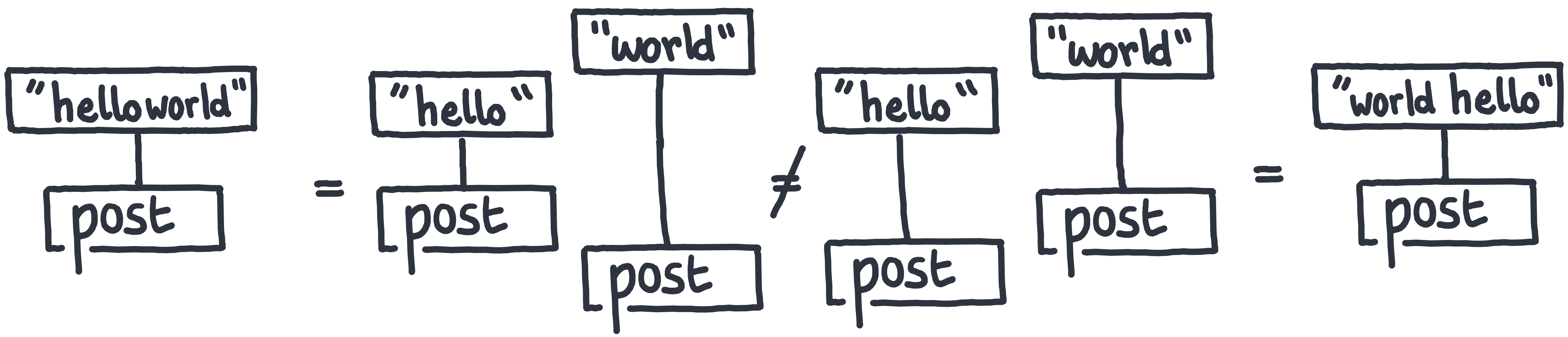
Axiom (6) $(f ; g) \otimes (f' ; g') = (f \otimes f') ; (g \otimes g')$ implies interchange, can we drop it?

$$(id \otimes g) ; (f \otimes id) = f \otimes g = (f \otimes id) ; (id \otimes g).$$

PART 2. PREMONOIDAL CATEGORIES

Premonoidal Categories

Premonoidal categories are monoidal categories without the interchange law.

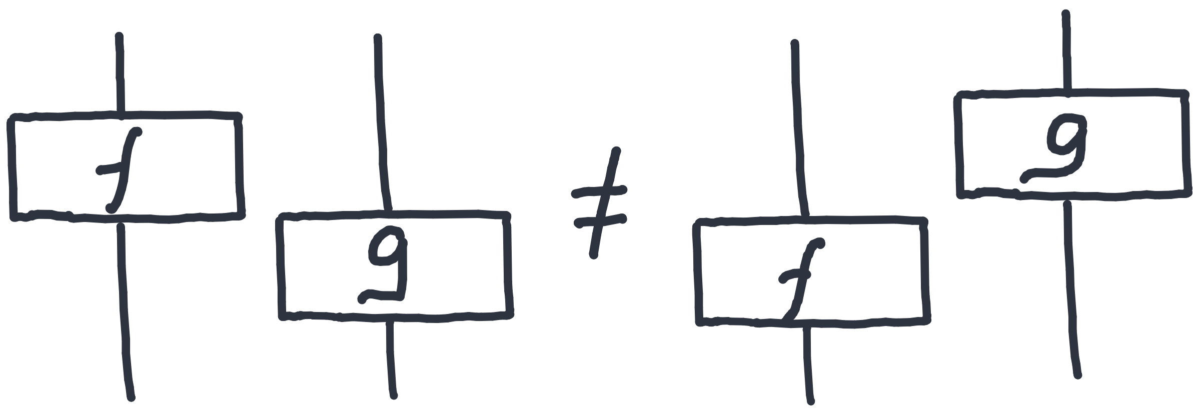


PROPOSITION. The free algebras of a strong monad form a premonoidal category.
EXAMPLES. Processes with global state, e.g. $(\Sigma^* \times \bullet) : \mathrm{SET} \to \mathrm{SET}$.

Power, Robinson, 97 ; Power, Thielecke, 99.

Premonoidal Categories

Premonoidal categories are monoidal categories without the interchange law.



Proposition. The free algebras of a strong monad form a premonoidal category.

Examples. Processes with global state, e.g. $(\Sigma^* \times \bullet) : \mathrm{Set} \to \mathrm{Set}$.

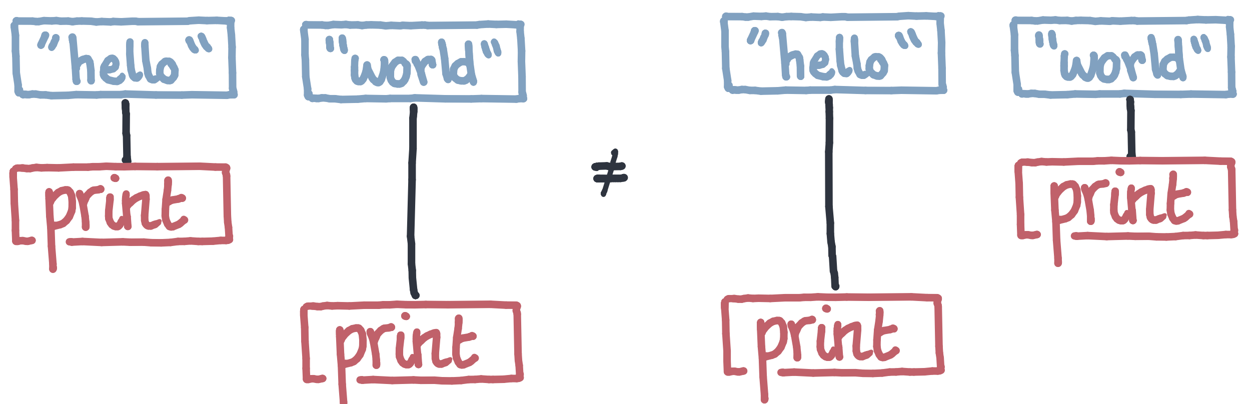
Power, Robinson, 97 ; Power, Thielecke, 99.

EFFECTFUL CATEGORIES

PROBLEM. Some premonoidal morphisms do interchange. We need premonoidals "with a chosen center"

DEFINITION. An effectful category is an identity-on-objects functor from a symmetric monoidal category $\mathbb{V}$ ("the values") to a symmetric premonoidal category $\mathbb{C}$ ("the computations"), strictly preserving the premonoidal structure and centrality.

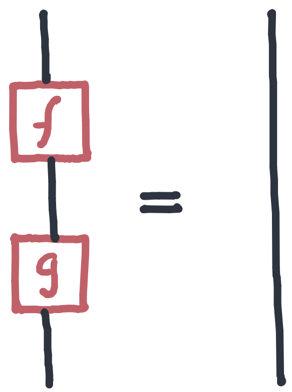
$$\underset{\text{values}}{\mathbb{V}} \xrightarrow[\text{central}]{\text{id-on-objs.}} \underset{\text{computations}}{\mathbb{C}}$$



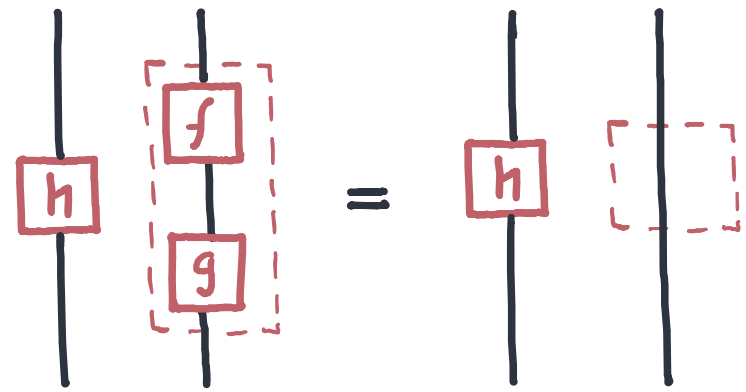
- This is the motivation for Freyd categories. Staton, Levy

EFFECTFUL CATEGORIES

Even when every effectful can be strictified, and even if we keep track of central and non-central morphisms,



does NOT imply



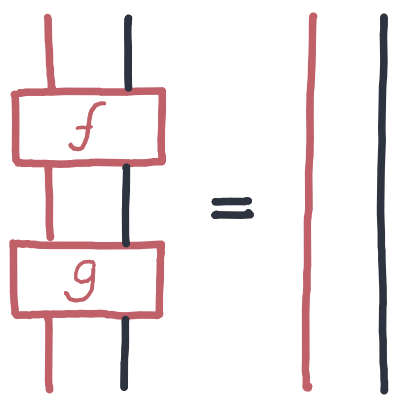
.

we lose 'locality' of equational reasoning.

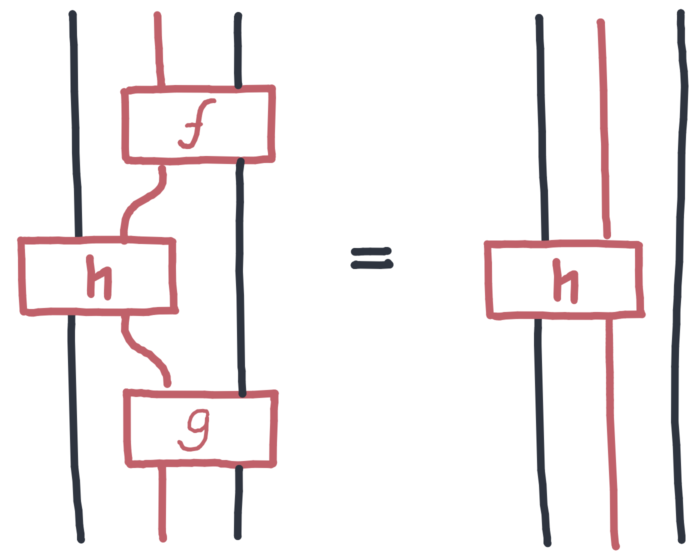
- What if there were a better solution?

EFFECTFUL CATEGORIES

A solution was proposed in the 90s: add an extra wire. Jeffrey, 97.



does NOT imply



Interpretation: Runtime, needed for computation, is a resource of your resource theory.

Runtime

Jeffrey, 97
Power, Thielecke, 99
Staton, Levy, 13

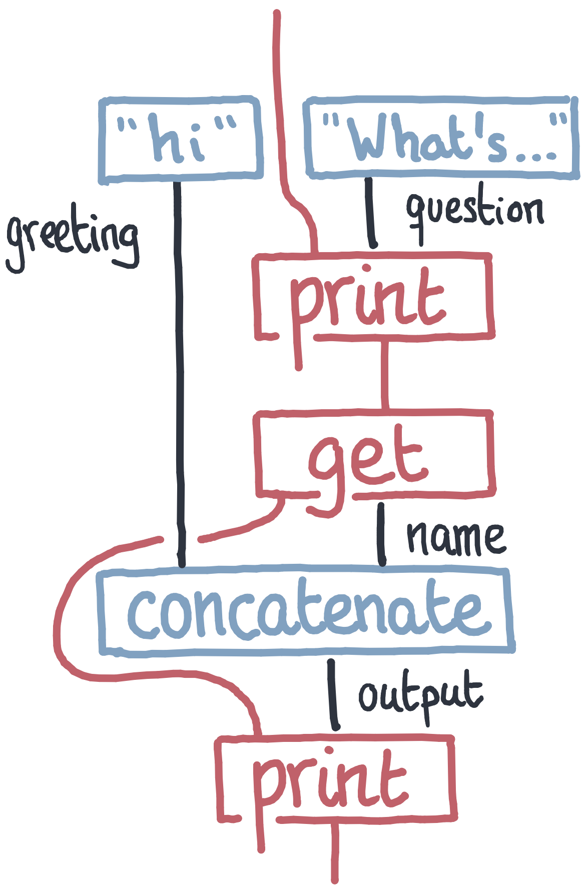


FIG 1. Basic IO.

- An extra wire represents runtime: global state of the system and control over it.
- JEFFREY'S runtime is flexible: we can choose a class of pure morphisms. It works for effectful categories.

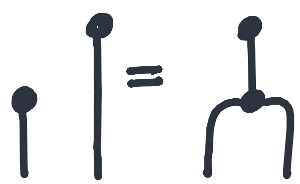
$$\underset{\text{values}}{\mathbb{V}} \xrightarrow[\text{central}]{\text{id-on-objs.}} \underset{\text{computations}}{\mathbb{C}}$$

- We can always choose $Z\mathbb{C} \to \mathbb{C}$ for premonoidals.

From now on, without loss of generality, we work with effectful categories.

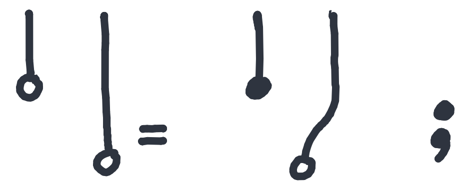
WHY EFFECTFUL CATEGORIES (GET/PUT EXAMPLE)

Many structures are clear once the runtime is present: consider the state monad $T_S X = S \to S \times X$ with get $: 1 \to S$ and put $: S \to 1$.

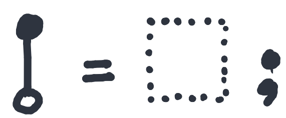


get-get

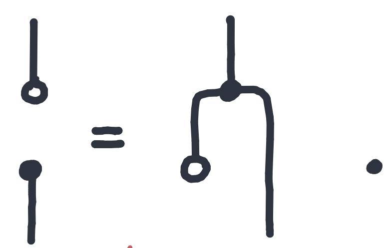
get-discard



put-put



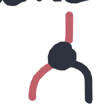
get-put

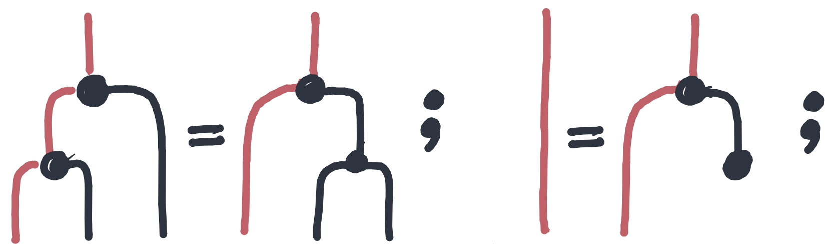


put-get

Consider some lens laws, drawn naïvely.

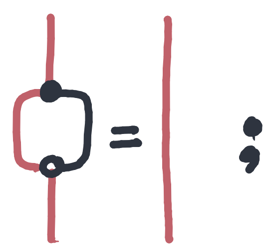
Why Effectful Categories (get/put example)

Many structures are clear once the runtime is present: consider the state monad $T_S X = S \to S \times X$ with get  $: 1 \to S$ and put  $: S \to 1$.

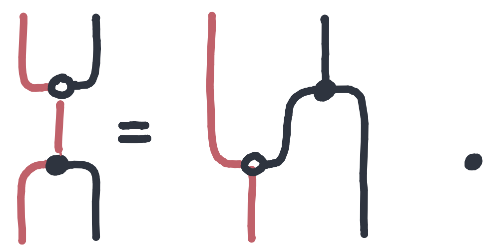


comonoid (runtime coaction)

semimonoid (runtime semiaction)



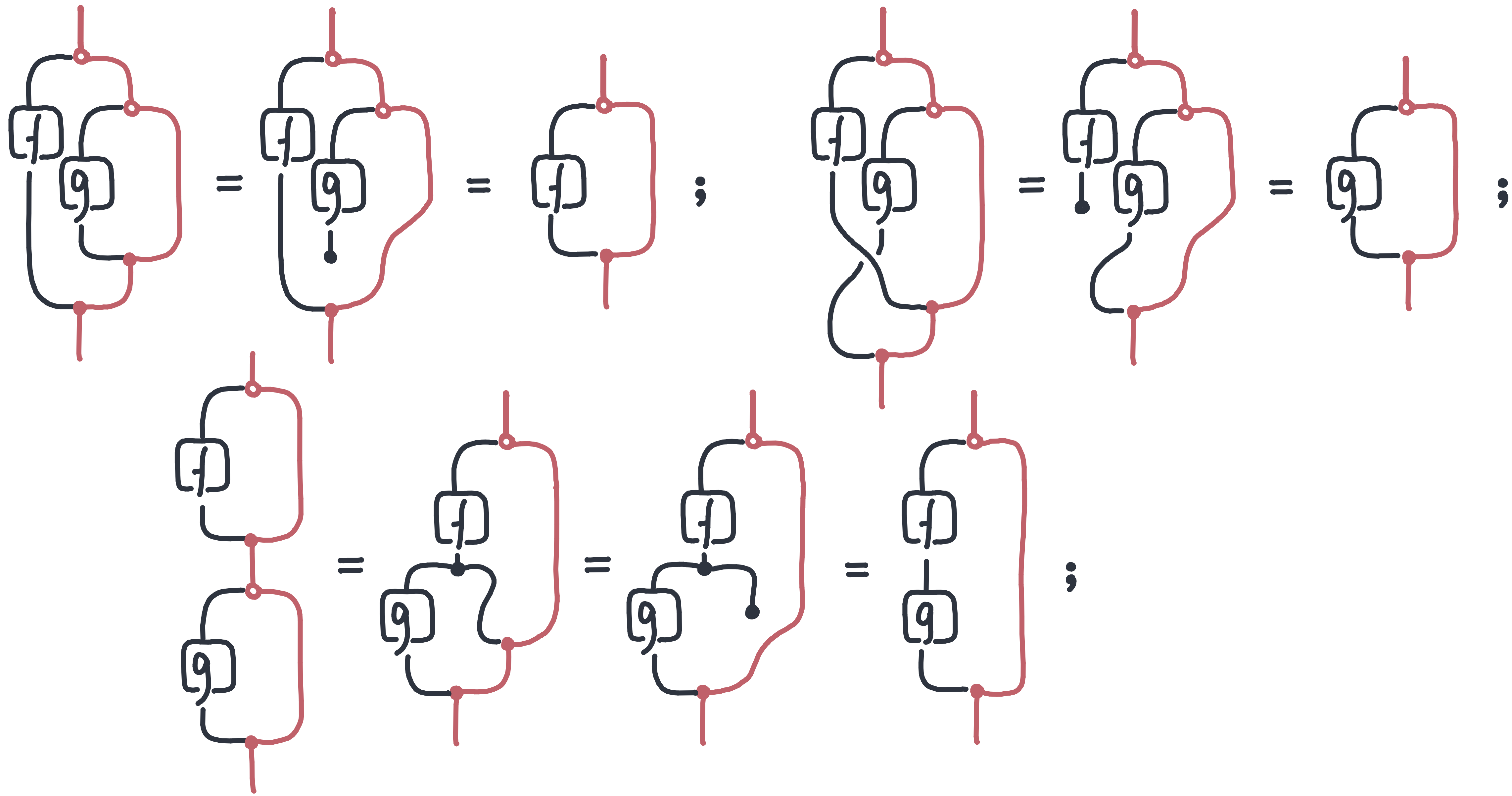
special



semifrobenius action

We can study a two-colour PROP instead.

Why Effectful Categories (Get/Put example)



CONTRIBUTIONS

1. THEOREM. The free effectful category in some generators is the free monoidal category over the same generators endowed with runtime.

$$\begin{tikzcd} \mathrm{PolyGraph} \arrow[r, bend left, "\mathrm{String}"] \arrow[r, phantom, "\perp"] & \mathrm{StrictEffectful} \arrow[l, bend left, "\mathrm{Forget}"] \end{tikzcd}$$

2. THEOREM. Effectful categories are pseudomonoids in a monoidal bicategory of promonads; in the same way that monoidal categories are pseudomonoids in a monoidal bicategory of categories.

Pseudomonoid →

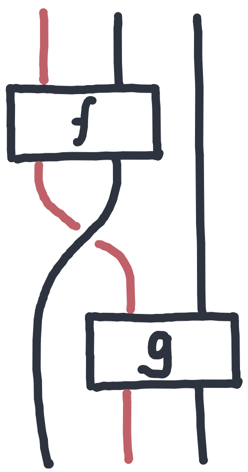
Effects ↓

Categories	Monoidal categories
Arrows	Effectful categories (Freyd categories)

Part 3. Runtime

Adding Runtime

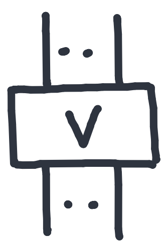
Jeffrey's Notation. Avoid interchange by adding an extra wire: the runtime.

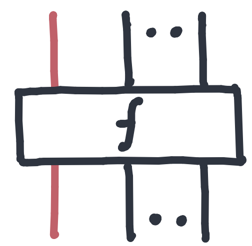


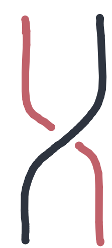
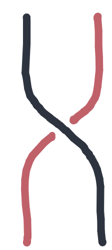
THEOREM. The morphisms $A_0 \otimes \cdots \otimes A_n \to B_0 \otimes \cdots \otimes B_m$ of the free effectful category over a 'signature' $\mathcal{H} \to \mathcal{G}$ are morphisms

$$R \otimes A_0 \otimes \cdots \otimes A_n \to R \otimes B_0 \otimes \cdots \otimes B_m$$

of a monoidal category with generators



for $v \in \mathcal{V}$;


for $f \in \mathcal{G}$;

;

.

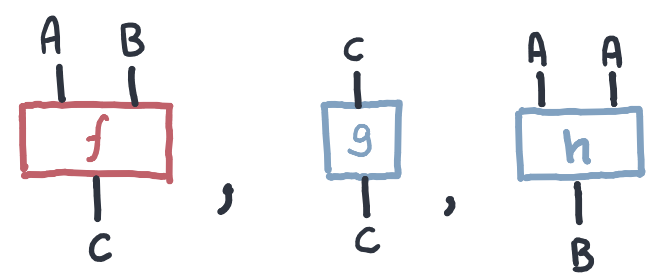
and the expected axioms.

Signatures with Runtime

DEFINITION. An *effectful polygraph* $(\mathcal{H}, \mathcal{G})$ is a pair of polygraphs sharing the same objects, $\mathcal{H}_{obj} = \mathcal{G}_{obj}$.

EXAMPLE.

$$(\mathcal{H}, \mathcal{G}) = \left\{ A, B, C \;\middle|\; \right.$$



$$\left. \right\}$$

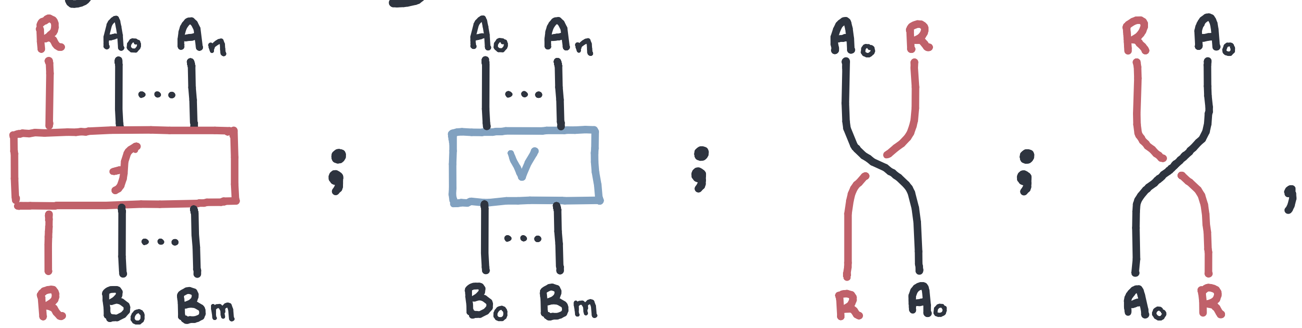
Effectful polygraphs give signatures for effectful categories.

THEOREM. String diagrams with runtime construct an adjunction

$$\begin{tikzcd} \mathbf{EffPolygraph} \arrow[r, bend left, "\mathrm{Strings}"] \arrow[r, phantom, "\perp"] & \mathbf{StrictEffectful} \arrow[l, bend left, "\mathrm{Forget}"] \end{tikzcd}.$$

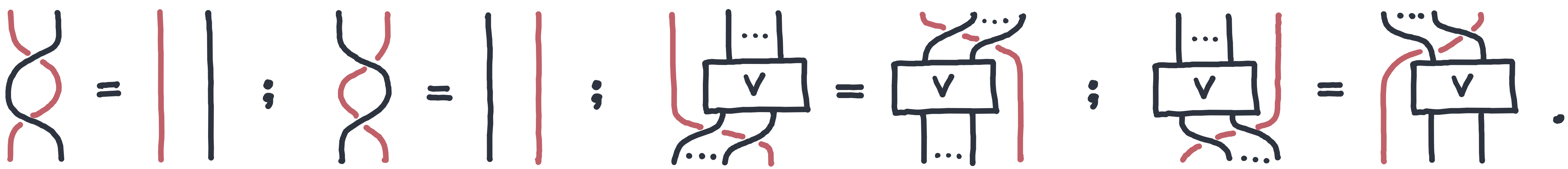
String Diagrams with Runtime

String diagrams generated by



for $f \in \mathcal{G}(A_0, \dots, A_n ; B_0, \dots, B_m)$, and $v \in \mathcal{H}(A_0, \dots, A_n ; B_0, \dots, B_m)$.

and quotiented by braiding axioms, asking R to be on the Drinfeld centre:



We prove that diagrams $R \otimes A_0 \otimes \dots \otimes A_n \to R \otimes B_0 \otimes \dots \otimes B_m$ form an effectful category.

CLIQUES

Technical problem: $R \otimes A_0 \otimes \cdots \otimes A_n \to R \otimes B_0 \otimes \cdots \otimes B_m$ always assumes that the runtime is on the left. This breaks term structural induction.

DEFINITION. A *clique* is a collection of objects, $\{A_i\}_{i \in I}$, together with an isomorphism $\gamma_{ij} : A_i \to A_j$, such that $\gamma_{ii} = id_i$, and $\gamma_{ij} \mathbin{⨟} \gamma_{jk} = \gamma_{ik}$.

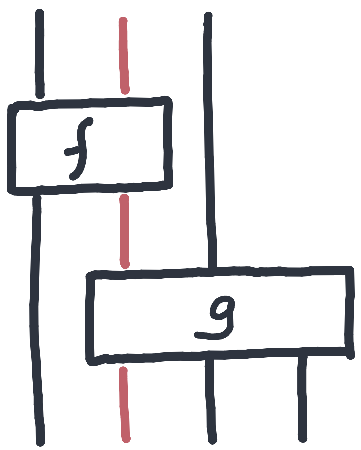
CONSTRUCTION. The *braid clique* on a list of objects $[A_0, \ldots, A_n]$ has R inserted at each position.

$$\begin{tikzcd}
 & A_0 \otimes R \otimes \cdots \otimes A_n \arrow[d, "\sigma"] \\
R \otimes A_0 \otimes \cdots \otimes A_n \arrow[ur, "\sigma"] \arrow[r, "\sigma"'] & A_0 \otimes \cdots \otimes R \otimes A_n
\end{tikzcd}$$

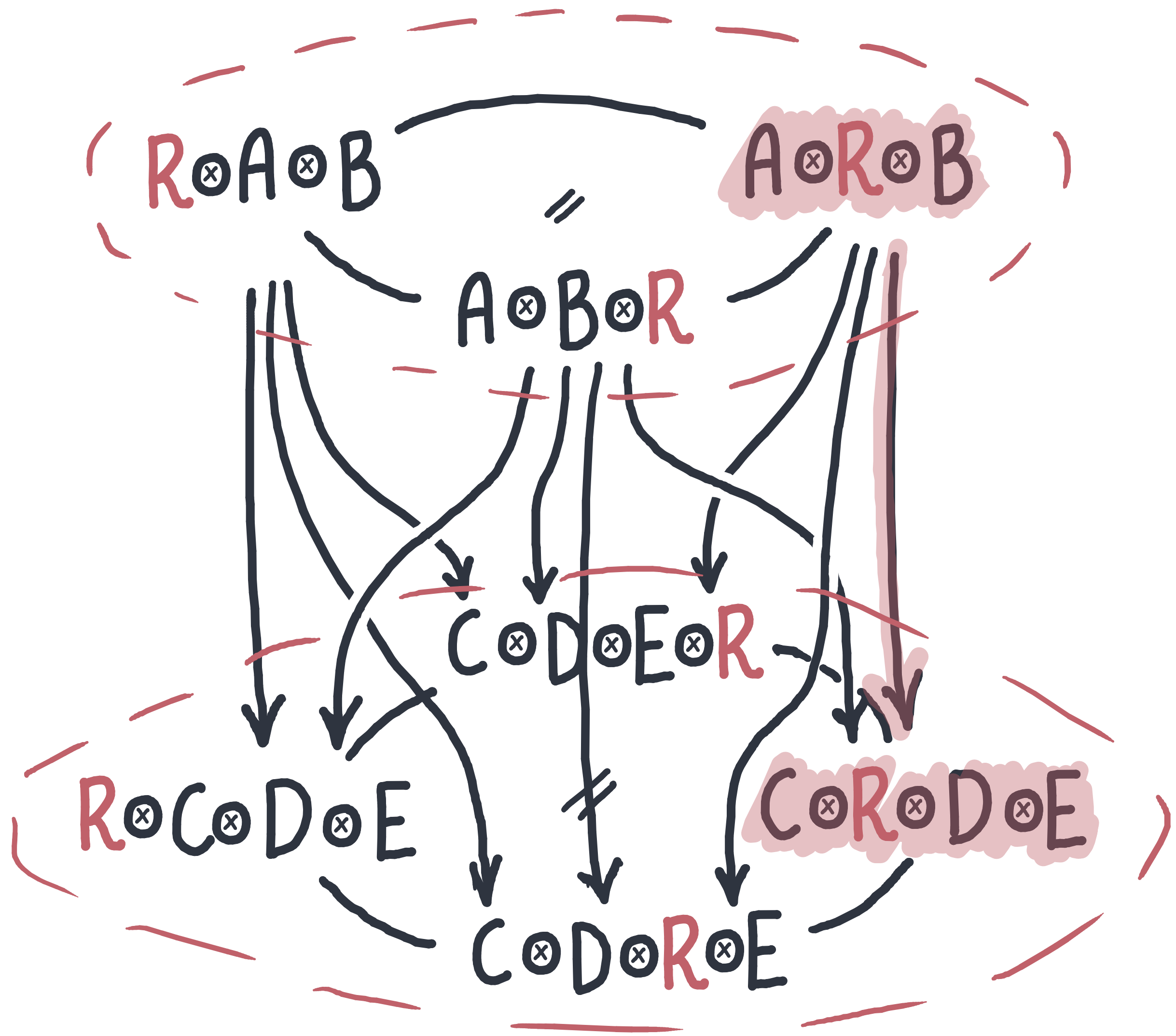
~ Trimble's intuition of monoidal coherence, on the nlab.

Cliques

$A \otimes R \otimes B$

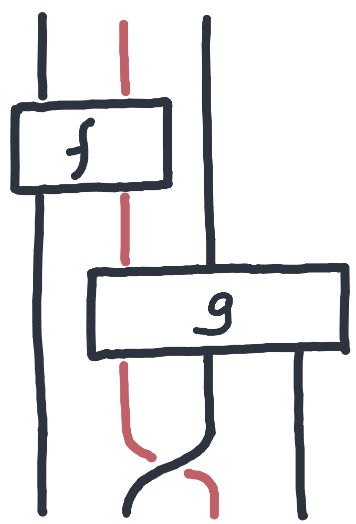


$C \otimes R \otimes D \otimes E$

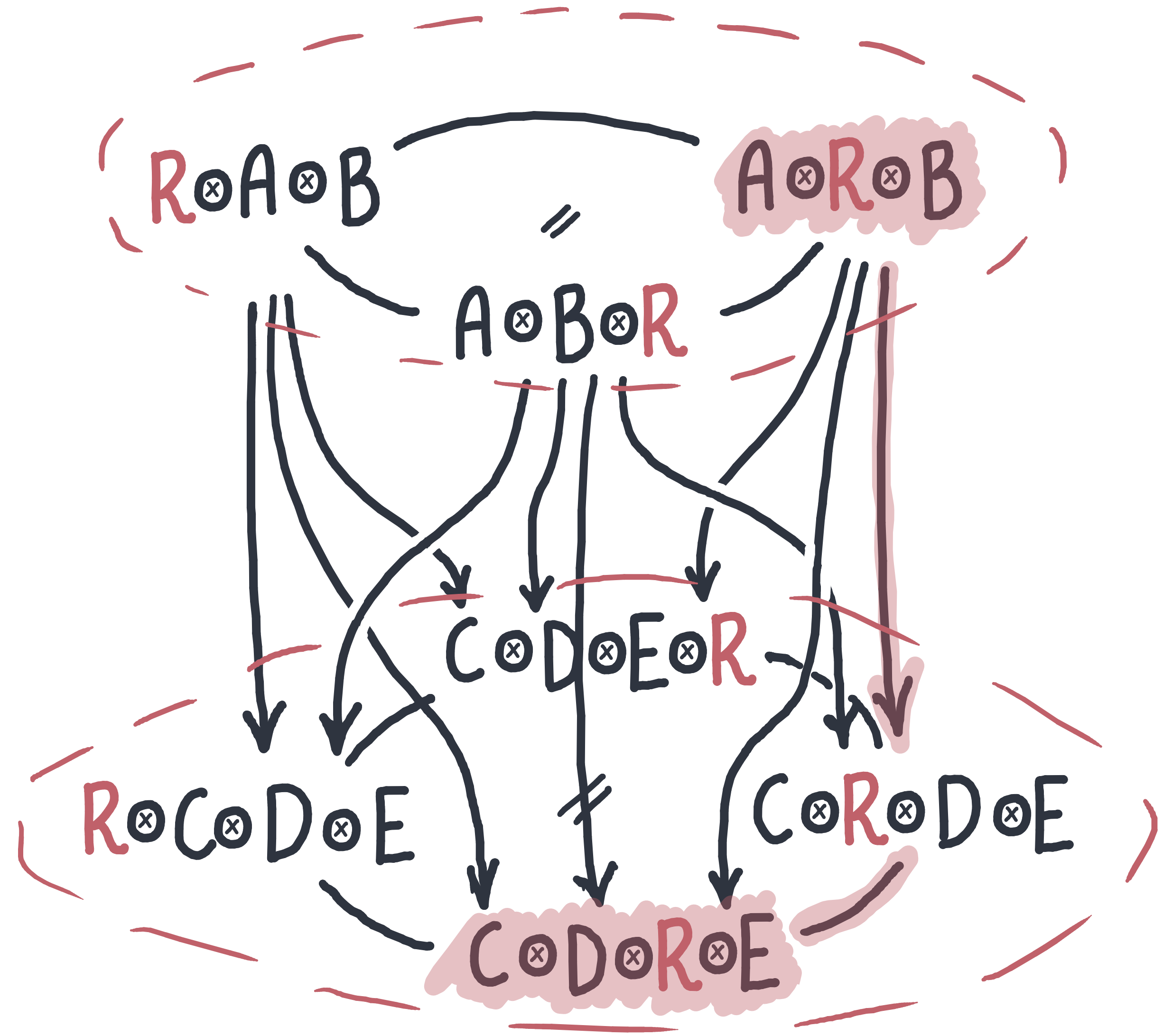


Cliques

$A \otimes R \otimes B$

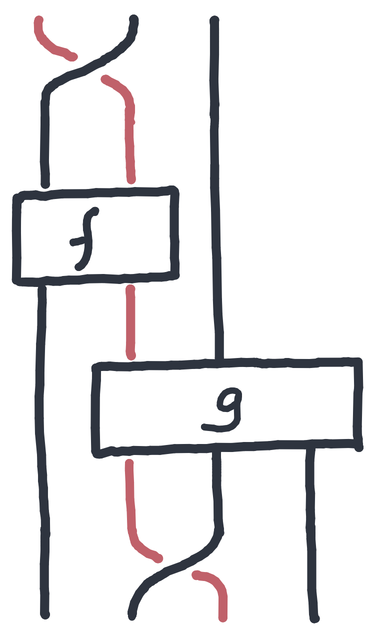


$C \otimes D \otimes R \otimes E$

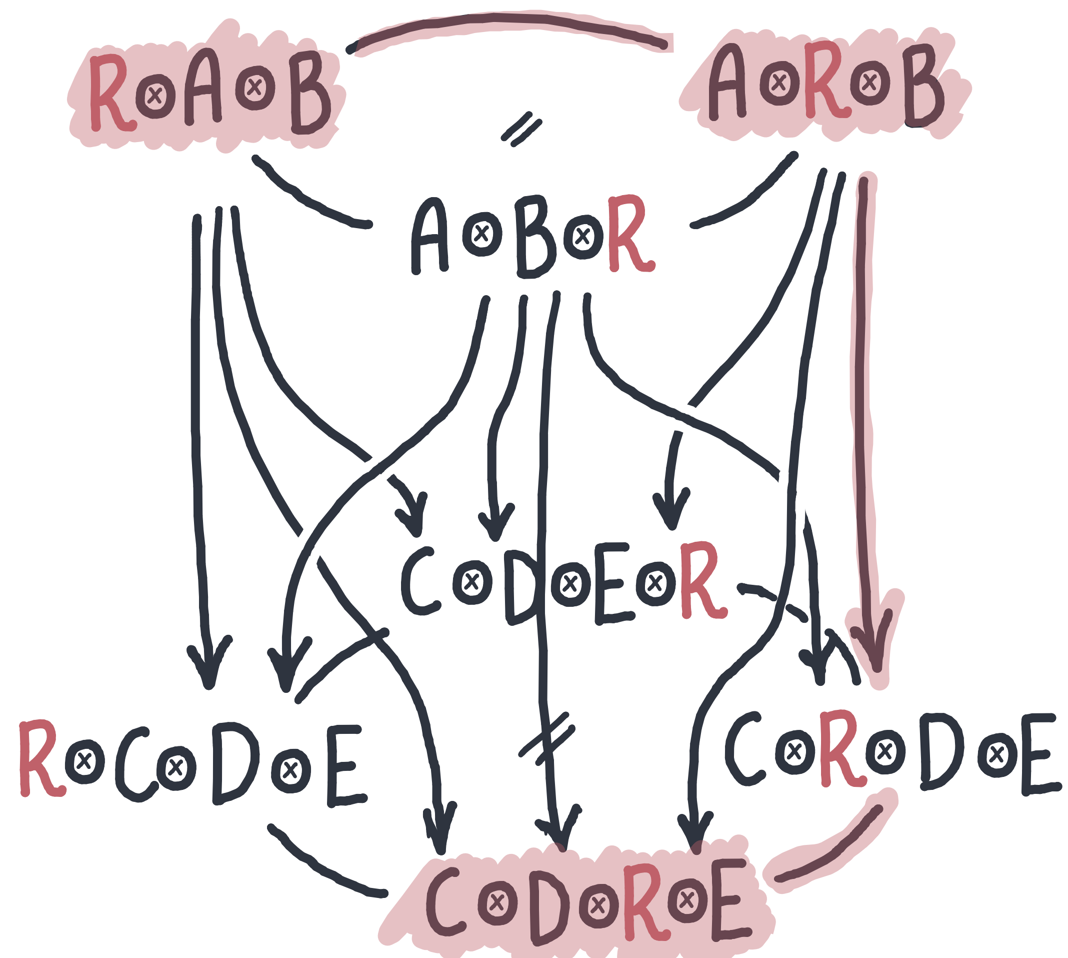


CLIQUES

$A \otimes R \otimes B$

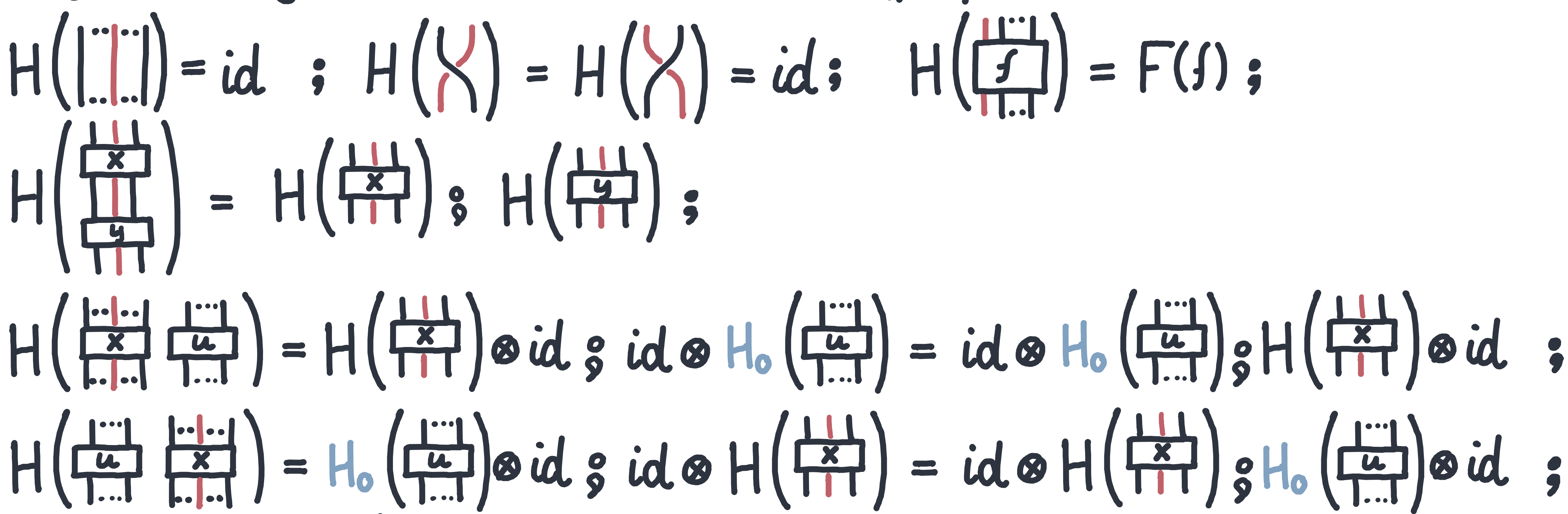


$C \otimes D \otimes R \otimes E$



String Diagrams with Runtime

Finally, the assignment determining the universal effectful functor is



THEOREM. String diagrams with runtime construct an adjunction

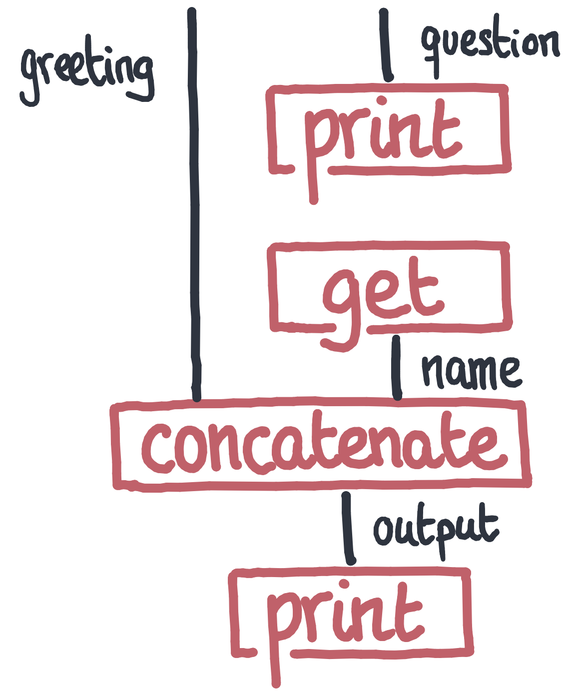
$$\begin{tikzcd} \textsc{PolygraphCouple} \arrow[r, bend left, "\text{Strings}"] \arrow[r, phantom, "\perp"] & \textsc{StrictEffectful}. \arrow[l, bend left, "\text{Forget}"] \end{tikzcd}$$

Part 4 : Diagrams are programs

Arrow-Do Notation

EXAMPLE. HelloProgram

```
proc (question, greeting) -> do
    () <- print -< question
    name <- get -< ()
    output <- concatenate -< greeting, name
    () <- print -< output
return ()
```



Arrow-Do Notation

Morphisms $A_1 \otimes \cdots \otimes A_n \to B_1 \otimes \cdots \otimes B_m$ of the free sym. premonoidal cat. over a multigraph $\mathcal{G}$ are Arrow **do-notation** blocks.

$$
\begin{array}{l}
\textbf{do } (x_0, \dots, x_n): \\
\quad g_1(y^1_0, \dots, y^1_{m_1}) \to x^1_1, \dots, x^1_{n_1} \\
\quad \vdots \\
\quad g_k(y^k_0, \dots, y^k_{m_k}) \to x^k_1, \dots, x^k_{n_k} \\
\textbf{return } (y_0, \dots, y_m)
\end{array}
$$

Where each y must be an x that appeared before. Each x must appear exactly once.

$g_1, \dots, g_k \in \mathcal{G}$ are generators.

And the types must match.

Heunen, Hasuo, Jacobs ; Hudak

COPY-DISCARD EFFECTFUL CATEGORIES

DEFINITION. A c.d.e.-category is two identity-on-objects functors from a cartesian category $\mathbb{V}$ ("the values") to a monoidal category $\mathbb{P}$ ("the local, or pure computations"), and to a symmetric premonoidal category $\mathbb{C}$ ("the computations"), strictly preserving the premonoidal structure and centrality.

$$\underset{\text{values}}{\mathbb{V}} \xrightarrow[\text{central}]{\text{id-on-objs.}} \underset{\text{local, pure}}{\mathbb{P}} \xrightarrow[\text{central}]{\text{id-on-objs.}} \underset{\text{computations}}{\mathbb{C}}$$

Jeffrey (1994)

Copy-Discard Effectful Do-notation

$$\frac{}{(x_i : X_i) \in \Gamma \vdash x_i : X_i} \qquad \frac{\Gamma \vdash v_1 : X_1 \quad \dots \quad \Gamma \vdash v_n : X_n}{\Gamma \Vdash \mathrm{return}(v_1, \dots, v_n) : X_1 \otimes \dots \otimes X_n}$$

$$\frac{\Gamma \vdash v_1 : X_1 \quad \dots \quad \Gamma \vdash v_n : X_n}{\Gamma \vdash f(v_1, \dots, v_n) : Y}$$

$$\frac{\Gamma \vdash v_1 : X_1 \quad \dots \quad \Gamma \vdash v_n : X_n \qquad y_1, \dots, y_m, \Gamma \Vdash p : \Delta}{\Gamma \vdash g(v_1, \dots, v_n) \to y_1, \dots, y_m ; p : \Delta}$$

$$\frac{\Gamma \vdash v_1 : X_1 \quad \dots \quad \Gamma \vdash v_n : X_n \qquad y_1, \dots, y_m, \Gamma \Vdash p : \Delta}{\Gamma \vdash h(v_1, \dots, v_n) \rightsquigarrow y_1, \dots, y_m ; p : \Delta}$$

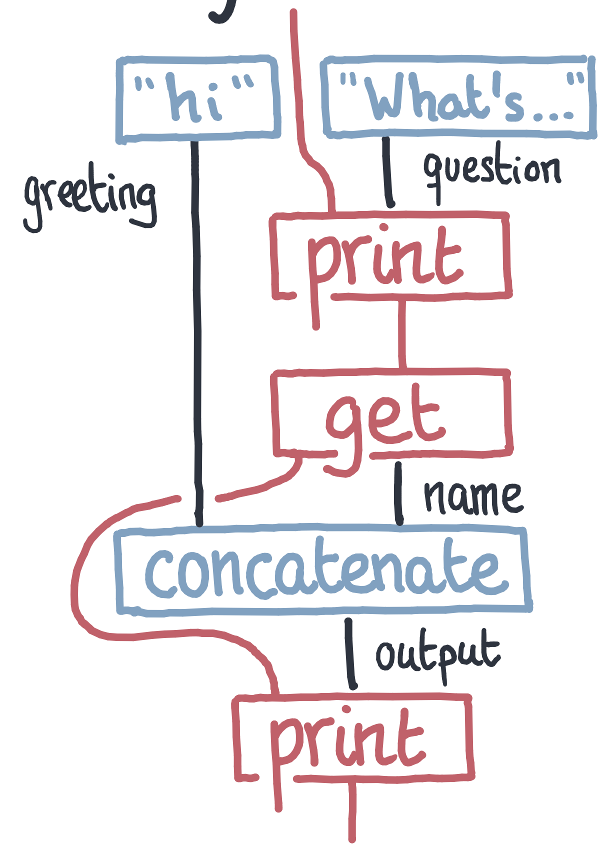
Bonchi, Di Lavore, Román (2023).

Copy-Discard Effectful Do-notation

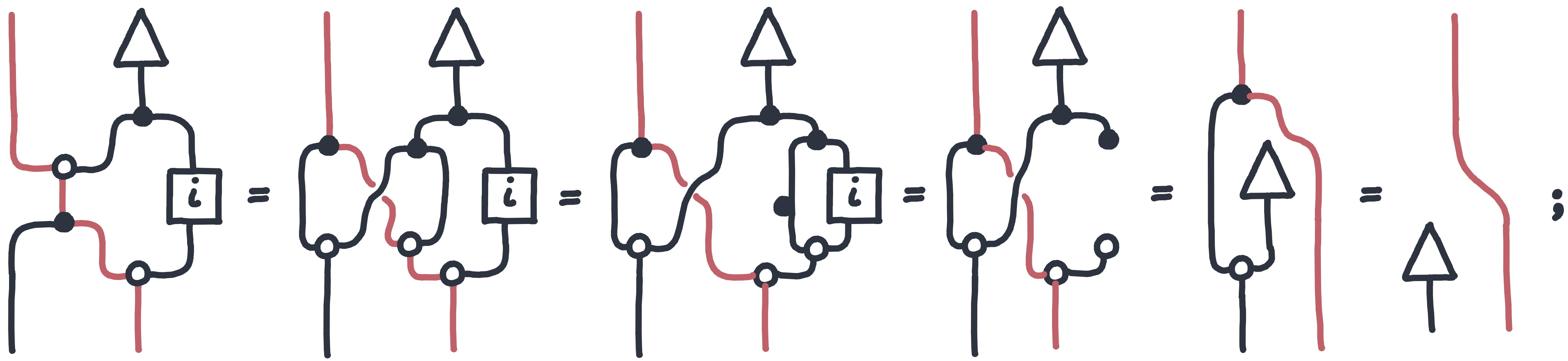
THEOREM. Effectful do-notation derivations form the free copy discard effectful category over a copy-discard effectful signature

EXAMPLE. HelloProgram

```
do
    print("What's your name?")
    get() → name
    print(concatenate("hi, ", name))
    return ()
```



Example: One-Time Pad



```
one-time-pad():
  unif() → u
  modify(u⊕•)
  get() → a
  modify(u⁻¹⊕•)
  return(a)
```

$$\equiv$$

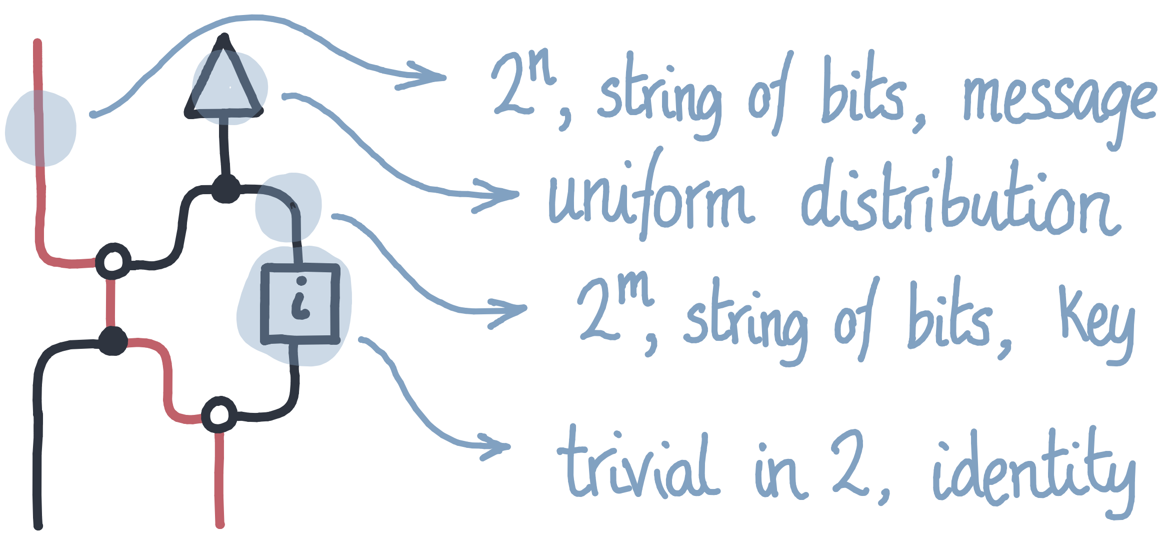
```
one-time-pad():
  unif() → a
  return(a)
```

PROPOSITION. The one-time pad protocol is secure.

PART 5: FURTHER

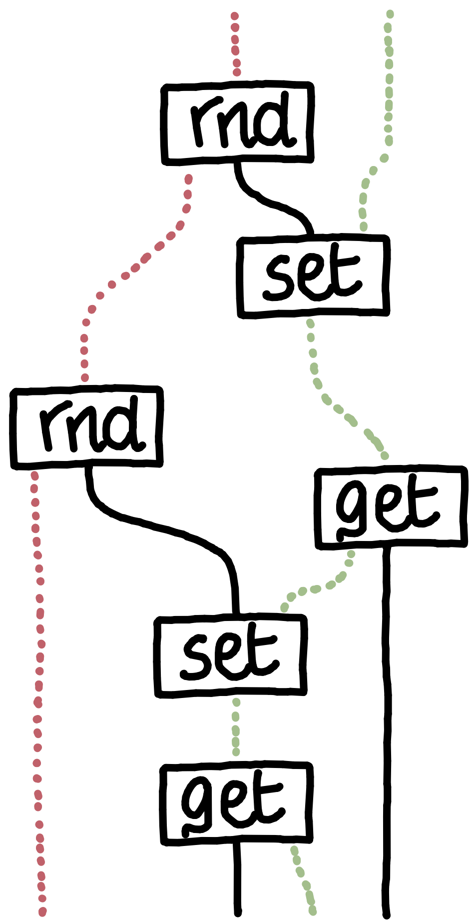
Effectful Algebra

- Lawvere theories: models are product-preserving functors into Set.
- Monoidal theories: models are monoidal functors into Set.
- Premonoidal theories: models are runtime-picking monoidal functors into Set.



Rajesh, Román, Saville

Multiple Devices



Any premonoidal can be presented by a single device; but other presentations may combine effects in exotic ways.

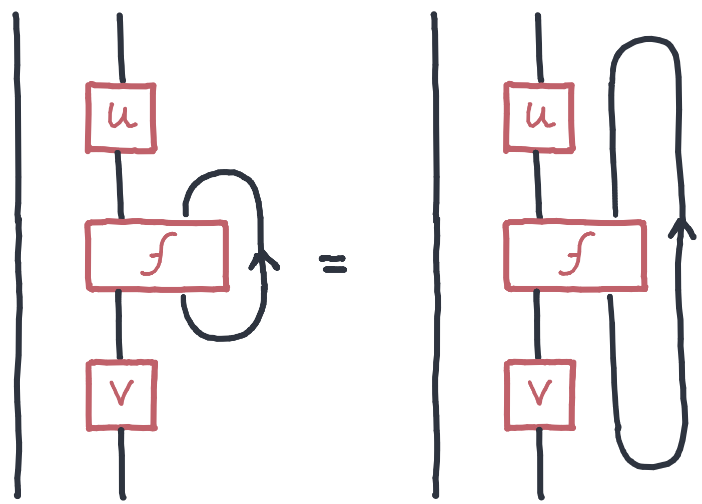
- Barrett, Heijltjes, McCusker.
- Earnshaw, Nester, Román.

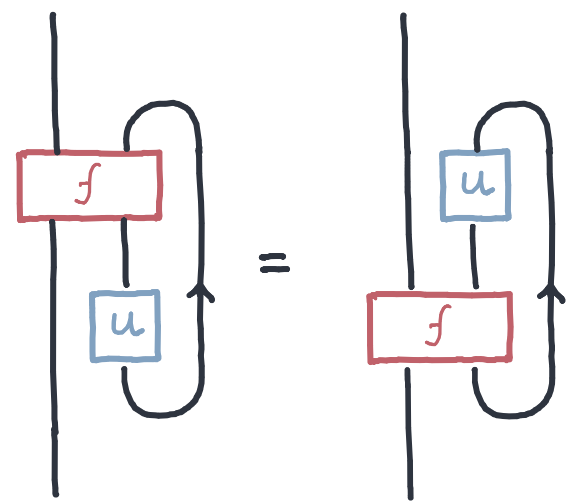
FEEDBACK PREMONOIDAL CATEGORY

DEFINITION. A feedback premonoidal category is a freyd category $\mathcal{V} \to \mathcal{C}$ endowed with an operator

$$\mathrm{fbk}^S : \hom(S \otimes A, S \otimes B) \longrightarrow \hom(A, B)$$

that satisfies the following axioms.

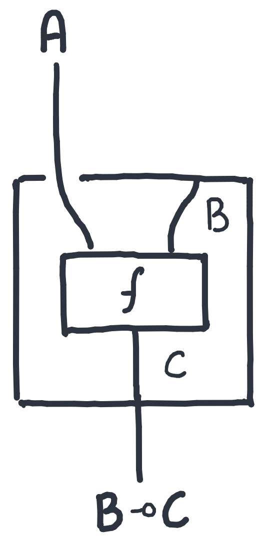
1. 

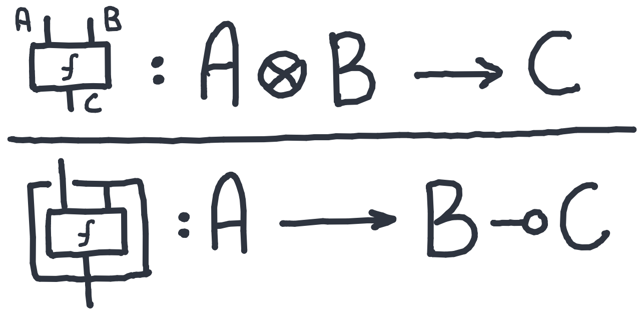
2. 

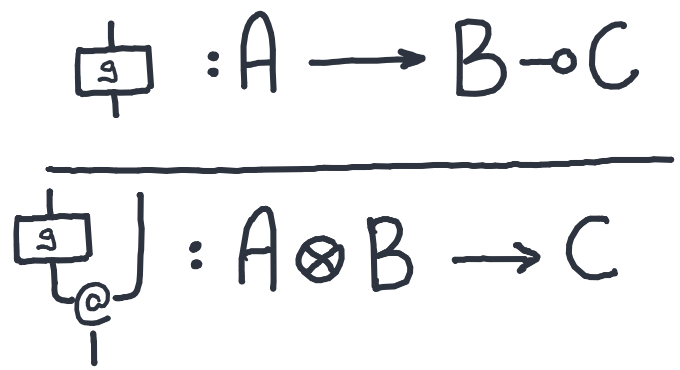
3. $\mathrm{fbk}^{S_1}(\dots \mathrm{fbk}^{S_n}(f) \dots) = \mathrm{fbk}^{S_1 \otimes \dots \otimes S_n}(f)$.

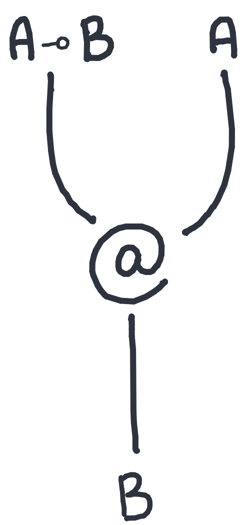
Bonchi, Di Lavore, Román

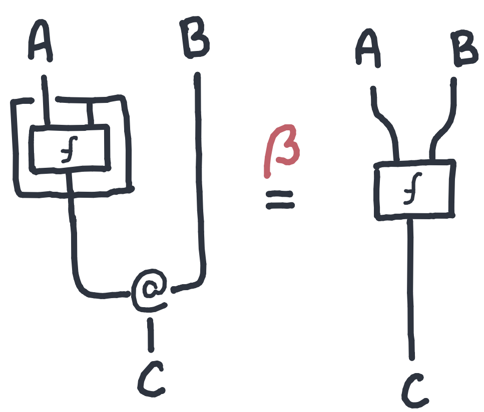
Closed Categories

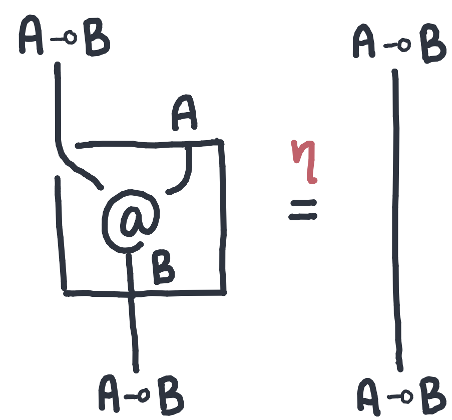










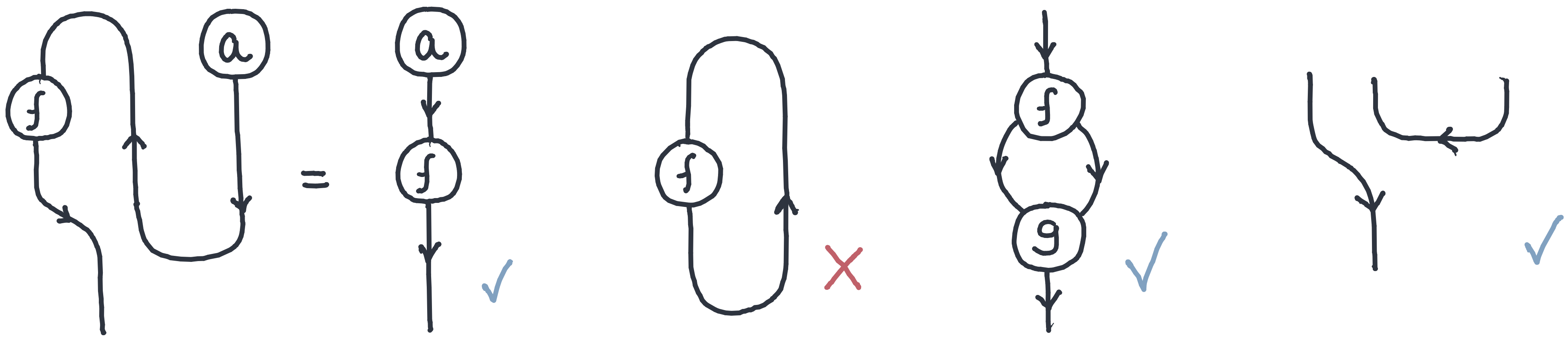


Jeffrey.

Closed Categories, $*$-Autonomous Way

Symmetric monoidal closed categories fully-faithfully embed into $*$-autonomous categories, so we can use their graphical calculus.

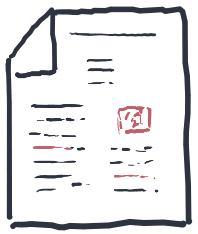
- Every object has a formal dual $A\downarrow$ $A^*\uparrow$.
- Some diagrams are valid; some are not.
- $(A \multimap B) = A^* \parr B$.



Shulman

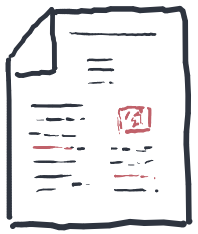
PROMONADS AND STRING DIAGRAMS FOR EFFECTFUL CATEGORIES, (ROMÁN, 2022).

ArXiv: 2205.07664



STRING DIAGRAMS FOR PREMONOIDAL CATEGORIES,

ArXiv: 2305.06075 (ROMÁN, SOBOCIŃSKI, 2023).



END

Effectfuls as Pseudomonoids

Pseudomonoids are 2-dimensional monoids.

THEOREM (Street, Day). **Monoidal categories** are pseudomonoids in the monoidal bicategory of categories.

THEOREM. **Effectful categories** (and thus premonoidal categories with their centre) are pseudomonoids in the monoidal bicategory of promonads with the pure tensor.

PURE TENSOR

We already have a bicategory: promonads, morphisms and transformations. There exists a notion of free product with commuting subgroups.

DEFINITION. The pure tensor of promonads, $\mathbb{C} : \mathbb{V} \nrightarrow \mathbb{V}$ and $\mathbb{D} : \mathbb{W} \nrightarrow \mathbb{W}$, is a promonad, $\mathbb{C} * \mathbb{D} : \mathbb{V} \times \mathbb{W} \nrightarrow \mathbb{V} \times \mathbb{W}$, whose elements are generated by

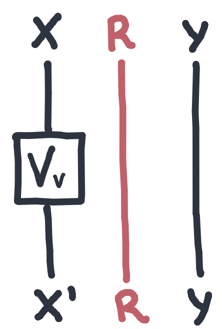
$$p_{\mathbb{C}} \in \mathbb{C} * \mathbb{D}(X, Y; X', Y) \text{ for } p \in \mathbb{C}(X, X'), \text{ or}$$
$$q_{\mathbb{D}} \in \mathbb{C} * \mathbb{D}(X, Y; X, Y') \text{ for } q \in \mathbb{D}(Y, Y').$$

quotiented by $f_{\mathbb{C}} \mathbin{;} w^{\circ}_{\mathbb{D}} = w^{\circ}_{\mathbb{D}} \mathbin{;} f_{\mathbb{C}}$ and $g_{\mathbb{D}} \mathbin{;} v^{\circ}_{\mathbb{C}} = v^{\circ}_{\mathbb{C}} \mathbin{;} g_{\mathbb{D}}$, plus other axioms.

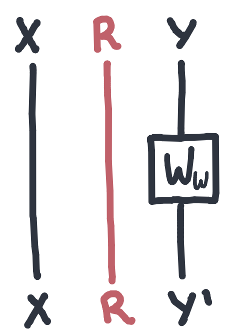
THEOREM. There exist a pair of promonad homomorphisms
$L : \mathbb{C} \times \mathbb{W} \to \mathbb{C} * \mathbb{D}$ and $R : \mathbb{V} \times \mathbb{D} \to \mathbb{C} * \mathbb{D}$
and the pure tensor is universal with respect to them.

Pure Tensor

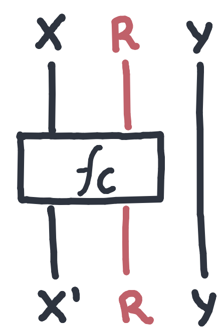
We already have a bicategory: promonads, morphisms and transformations.
There exists a notion of free product with commuting subgroups.



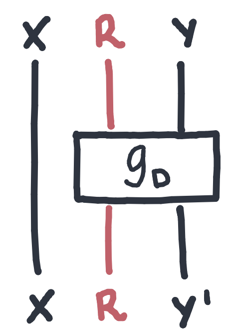
for $v \in \mathbb{V}(X,X')$



for $w \in \mathbb{W}(Y,Y')$



for $f \in \mathbb{C}(X,X')$



for $g \in \mathbb{D}(X,X')$

All these commutativity restrictions are better encoded by string diagrams.

- Thanks to an extra wire.

Premonoidal Categories

DEFINITION (Power, Robinson). A binoidal category $(\mathbb{C} \otimes I)$, is a category with an object $I \in \mathbb{C}$ and an assignment on objects that is separately functorial on each component, $(A \otimes \bullet) : \mathbb{C} \to \mathbb{C}$ and $(\bullet \otimes B) : \mathbb{C} \to \mathbb{C}$.

- That is, $(\otimes)$ is a sesquifunctor.
- A morphism $f : A \to B$ is central if $(f \otimes id) ⨟ (id \otimes g) = (id \otimes g) ⨟ (f \otimes id)$ for any $g : A' \to B'$.

DEFINITION (Power, Robinson). A premonoidal category $(\mathbb{C}, \otimes, I)$ is a binoidal category with natural central isomorphisms

$$\alpha_{A,B,C} : (A \otimes B) \otimes C \to A \otimes (B \otimes C), \quad \rho_A : A \otimes I \to A, \quad \lambda_A : I \otimes A \to A,$$

satisfying pentagon and triangle equations. It is strict if these are identities.

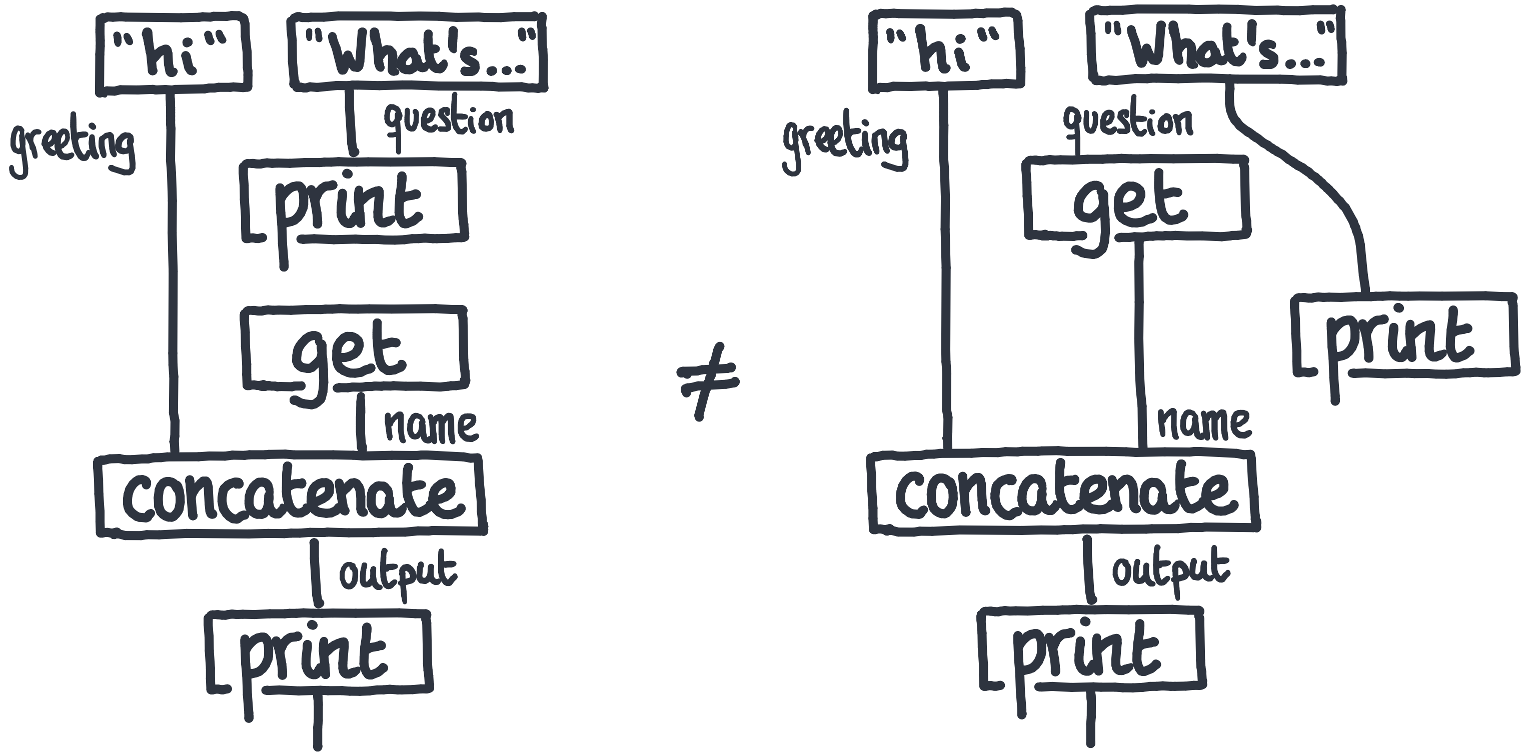
Type Theory of Symmetric Monoidal Cat.

This was a language for computations. What about values?
Morphisms $A_1 \otimes \cdots \otimes A_n \to B_1 \otimes \cdots \otimes B_m$ of the free sym. monoidal cat. over a multigraph $\mathcal{G}$ are terms of the following type theory.

$$\dfrac{}{x : A \vdash x : A}\ \text{VAR} \qquad \dfrac{f \in G(A_1, \ldots, A_n ; B) \quad \Gamma_1 \vdash x_1 : A_1 \quad \ldots \quad \Gamma_n \vdash x_n : A_n}{\Gamma_1, \ldots, \Gamma_n \vdash f(x_1, \ldots, x_n) : B}\ f$$

$$\dfrac{\Gamma_1 \vdash x_1 : A_1 \quad \ldots \quad \Gamma_n \vdash x_n : A_n}{\mathrm{Shuf}(\Gamma_1, \ldots, \Gamma_n) \vdash [x_1, \ldots, x_n] : A_1 \otimes \cdots \otimes A_n}\ \text{TUPLE}$$

$$\dfrac{\Delta \vdash m : A_1 \otimes \cdots \otimes A_n \quad \Gamma, x_1 : A_1, \ldots, x_n : A_n \vdash z : C}{\mathrm{Shuf}(\Gamma, \Delta) \vdash [x_1, \ldots, x_n] \leftarrow m ; z : C}\ \text{SPLIT}$$

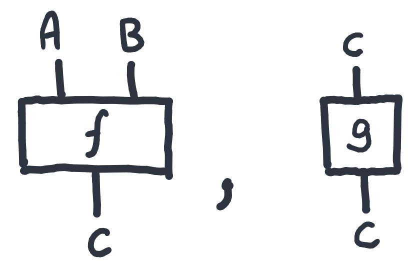

"hi"
"What's..."
greeting
question
print
get
name
concatenate
output
print
≠
"hi"
"What's..."
greeting
question
get
print
name
concatenate
output
print

POLYGRAPHS

DEFINITION. A polygraph $\mathcal{G}$ is given by a set of objects, $\mathcal{G}_{obj}$, and a set of arrows $\mathcal{G}(A_0, \dots, A_n; B_0, \dots B_n)$ for sequences $A_0, \dots, A_n$ and $B_0, \dots, B_n$.

EXAMPLE.

$$\mathcal{G} = \left\{ A, B, C \;\middle|\; \right.$$



$$\left. \right\}$$

Polygraphs give signatures for monoidal categories: string diagrams over a polygraph form the free strict monoidal category on it.

THEOREM. String diagrams construct an adjunction.

$$\begin{tikzcd} \text{POLYGRAPH} \arrow[r, bend left, "\text{Strings}"] \arrow[r, phantom, "\perp"] & \text{STRICTMONCAT} \arrow[l, bend left, "\text{Forget}"] \end{tikzcd}.$$