Constructing Cryptocurrency Ledgers with Monoidal Categories

Chad Nester
cnester@ed.ac.uk

Blockchains

In blockchain systems, a distributed consensus protocol is employed to maintain a record of activity called a *ledger*.

This talk is about the formal structure of such ledgers.

Motivation: "Smart Contracts". Complex ledger structures hard to reason about due to ad hoc design. Use category theory to guide system design.

UTXO Ledgers

A ledger is a list of *transactions*, each consuming specific *coins* as input and generating new coins as output.

Each coin can only be spent once, and money must not appear in or disappear from the system for no reason.

The coins that have not yet been spent are called "Unspent Transaction Outputs" (UTXOs). The UTXOs are the coins currently available in the system.

UTXO Ledgers

Each coin is associated with a *validator* (a computer program), which controls access to the coin.

To spend a coin with validator a, a transaction must supply a *redeemer* b such that $a(b) \downarrow$ in some fixed amount of time (varies according to protocol).

Example: Validator a such that $a(b) \downarrow$ if and only if b is the result of Alice signing the current block number with their private key. Alice effectively owns the associated coin.

Monoidal Categories as Resource Theories

Strict symmetric monoidal categories can be understood as theories of resource convertibility (Coecke, Fritz and Spekkens 2016).

Objects correspond to collections of resources: $A \otimes B$ is the collection composed of both A and B, The monoidal unit I is the empty collection.

Morphisms $f : A \to B$ are ways to convert the resources of A into those of B.

Monoidal Categories as Resource Theories

For example, consider the free symmetric strict monoidal category on atomic objects:

$$\{\texttt{bread}, \texttt{dough}, \texttt{water}, \texttt{flour}, \texttt{oven}\}$$

subject to additional axioms:

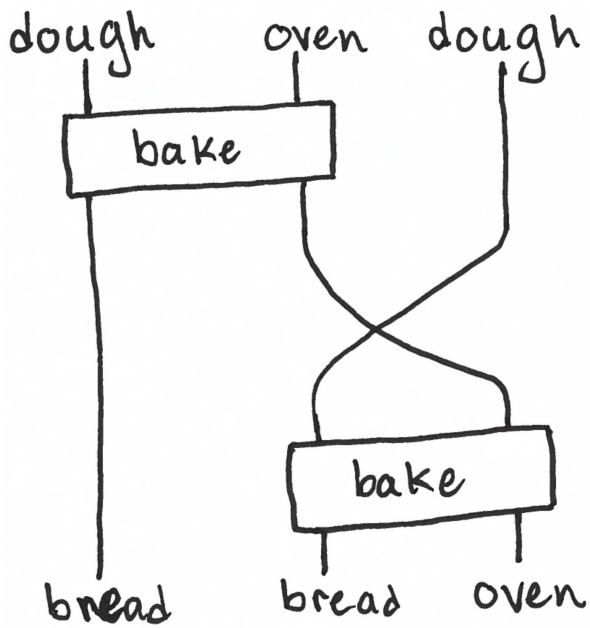
$$\texttt{mix} : \texttt{water} \otimes \texttt{flour} \to \texttt{dough} \qquad \texttt{knead} : \texttt{dough} \to \texttt{dough}$$

$$\texttt{bake} : \texttt{dough} \otimes \texttt{oven} \to \texttt{bread} \otimes \texttt{oven}$$

This category can be understood as a (rather naïve) theory of resource convertibility for baking bread.

Monoidal Categories as Resource Theories

Consider the morphism:



We can think of this as describing a procedure (bake the batches of dough one after the other), or as describing part of a *history*.

A Resource Theory of Coins

Define $\mathbb{M}$ to be the free symmetric strict monoidal category on atomic objects $\mathbb{N}^+$ subject to axioms:

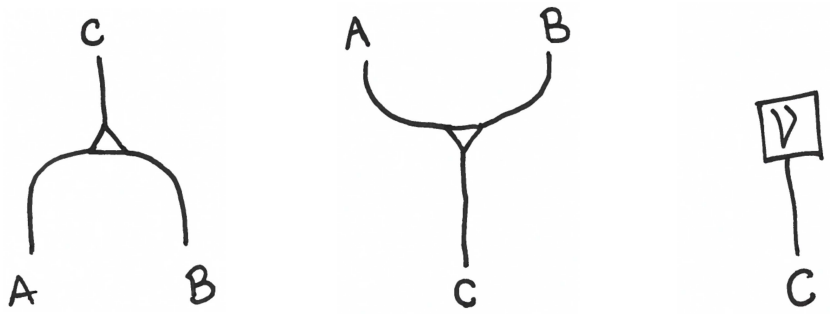
$$\frac{A, B, C \in \mathbb{N}^+ \qquad A + B = C}{\Delta_{A,B,C} : C \to A \otimes B} \qquad \frac{A, B, C \in \mathbb{N}^+ \qquad A + B = C}{\nabla_{A,B,C} : A \otimes B \to C}$$

$$\frac{C \in \mathbb{N}^+}{\nu_C : I \to C} \qquad \Delta_{A,B,C}\nabla_{A,B,C} = id_C \qquad \nabla_{A,B,C}\Delta_{A,B,C} = id_{A\otimes B}$$

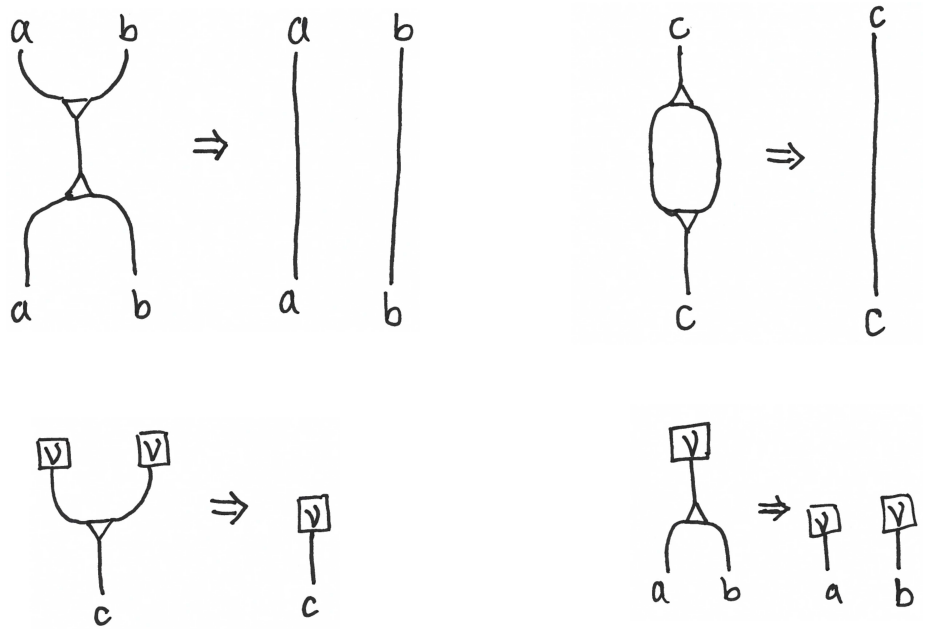
$$\nu_C\Delta_{A,B,C} = \nu_A \otimes \nu_B \qquad (\nu_A \otimes \nu_B)\nabla_{A,B,C} = \nu_C$$

A Resource Theory of Coins

In our string diagrams, we will represent the axioms as:



For cut elimination, we orient the equations as:



A Resource Theory of Coins

$\mathbb{M}$ captures the way systems like Bitcoin model currency.

View objects as *coins*. A ledger state is a morphism

$$l : I \to U_0 \otimes \cdots \otimes U_n$$

and the UTXOs (available coins) are the U_i of the codomain. Morphisms are then *transactions*, rearranging the available money.

Each output wire in the associated string diagram corresponds to an *address* associated with that UTXO.

A Resource Theory of Coins

$\mathbb{M}$ does not model the access control part of systems like Bitcoin.

Access control information is associated with the address of a coin. Addresses $\rightsquigarrow$ wires in string diagrams, so we *colour* our diagrams to encode this information.

In particular, we make heavy use of string diagrams for monoidal functors (McCurdy 2011).

(Co)Monoidal Functors

Let $(\mathbb{X}, \otimes, I)$ and $(\mathbb{Y}, \otimes, I)$ be symmetric strict monoidal categories.

A functor $F : \mathbb{X} \to \mathbb{Y}$ is *monoidal* in case there is a natural transformation $\phi_{x,y} : F(x) \otimes F(y) \to F(x \otimes y)$ together with a morphism $\phi_0 : I \to F(I)$ such that

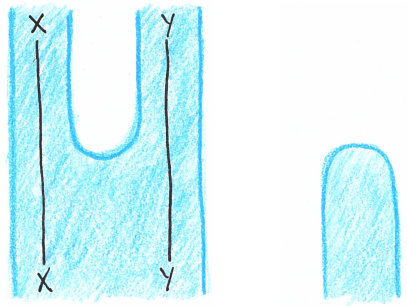
$$(\phi_{x,y} \otimes id_z)\phi_{x \otimes y,z} = (id_x \otimes \phi_{y,z})\phi_{x,y \otimes z}$$

and

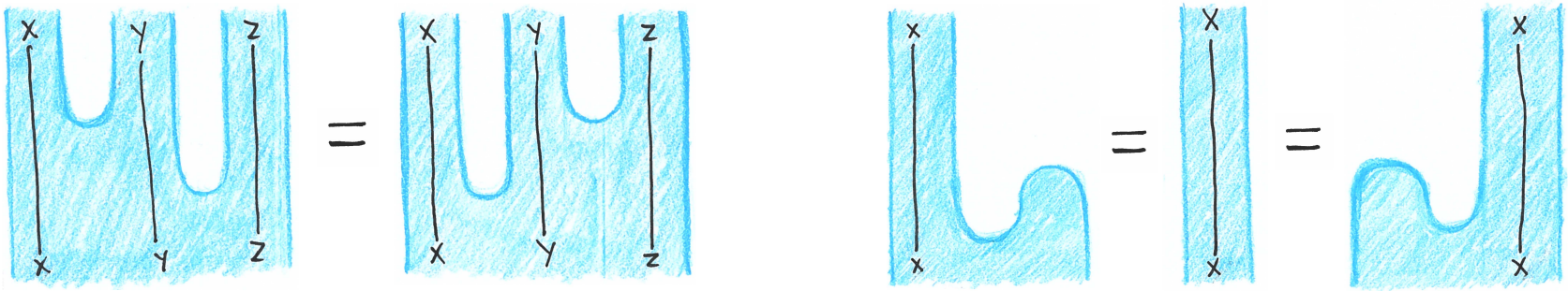
$$(id_x \otimes \phi_0)\phi_{x,I} = id_x = (\phi_0 \otimes id_x)\phi_{I,x}$$

(Co)Monoidal Functors

If we depict the data for a monoidal functor as:



then the required identities are



(Co)Monoidal Functors

Similarly, a functor $F : \mathbb{X} \to \mathbb{Y}$ is *comonoidal* in case there is a natural transformation $\psi_{x,y} : F(x \otimes y) \to F(x) \otimes F(y)$ together with a morphism $\psi_0 : F(I) \to I$ such that

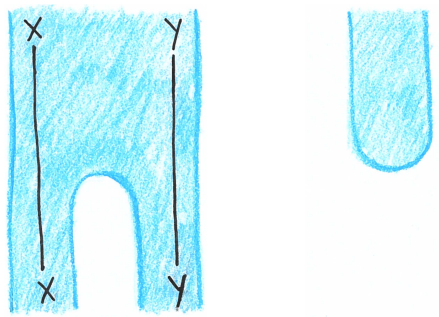
$$\psi_{x \otimes y, z}(\psi_{x,y} \otimes id_z) = \psi_{x, y \otimes z}(id_x \otimes \psi_{y,z})$$

and

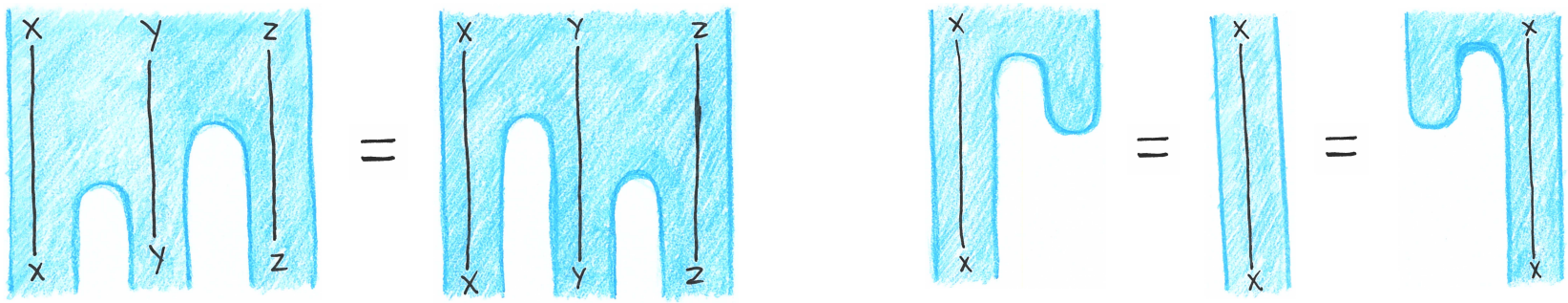
$$(id_x \otimes \psi_0)\psi_{x,I} = id_x = (\psi_0 \otimes id_x)\psi_{I,x}$$

(Co)Monoidal Functors

If we depict the data for a comonoidal functor as:

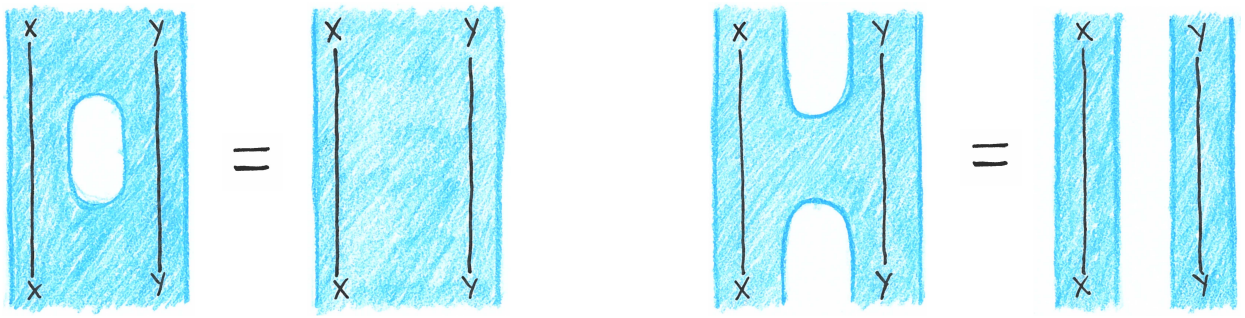


then the required identities are

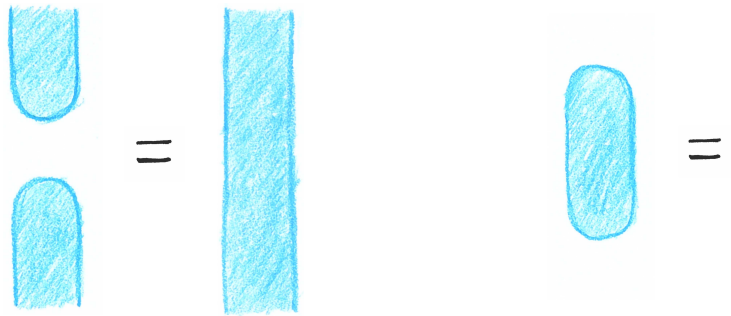


(Co)Monoidal Functors

A functor $F : \mathbb{X} \to \mathbb{Y}$ that is both monoidal and comonoidal is said to be *strong monoidal* in case $\phi_{x,y} = \psi_{x,y}^{-1}$ and $\phi_0 = \psi_0^{-1}$. That is

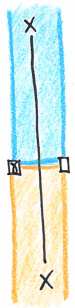


and

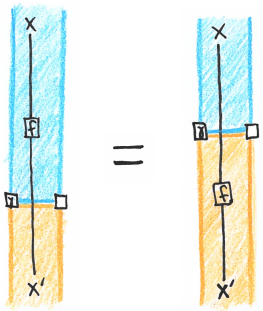


(Co)Monoidal Functors

For $F, G : \mathbb{X} \to \mathbb{Y}$, we depict natural transformations $\gamma : F \to G$ as

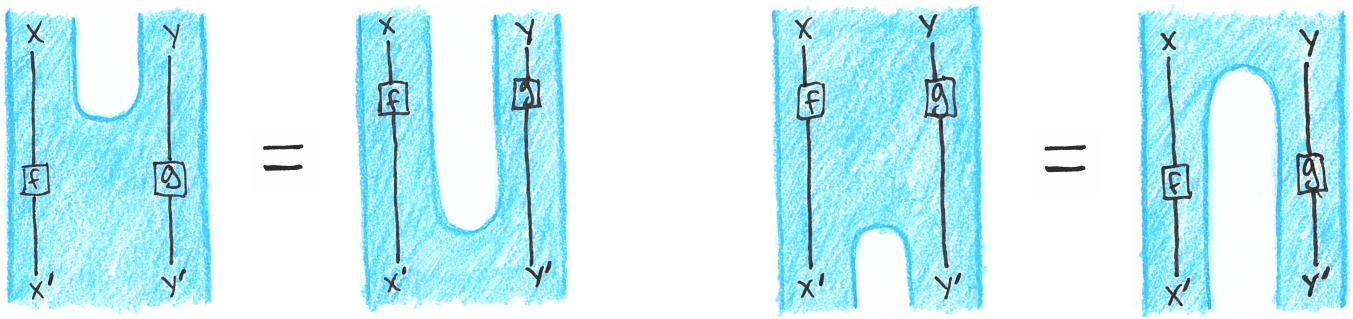


then naturality is the condition that



(Co)Monoidal Functors

Aside: Naturality of $\phi_{x,y}$ and $\psi_{x,y}$ (in both x and y) means



(Co)Monoidal Functors

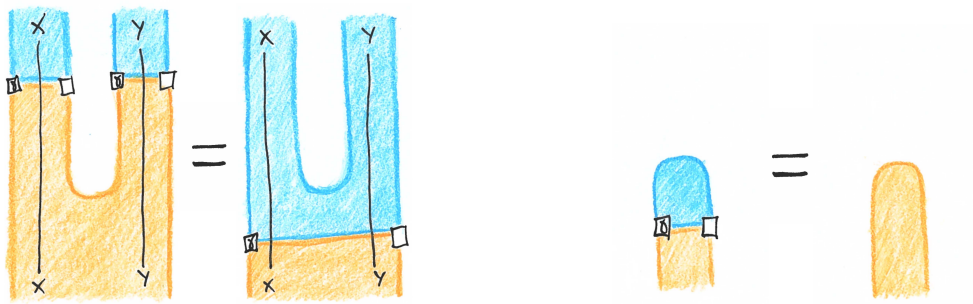
If $F, G : \mathbb{X} \to \mathbb{Y}$ monoidal functors, $\gamma : F \to G$ a natural transformation, we say that γ is *monoidal* in case

$$(\gamma_x \otimes \gamma_y)\phi_{x,y} = \phi_{x,y}\gamma_{x \otimes y}$$

and

$$\phi_0 \gamma_I = \phi_0$$

graphically, this is



(Co)Monoidal Functors

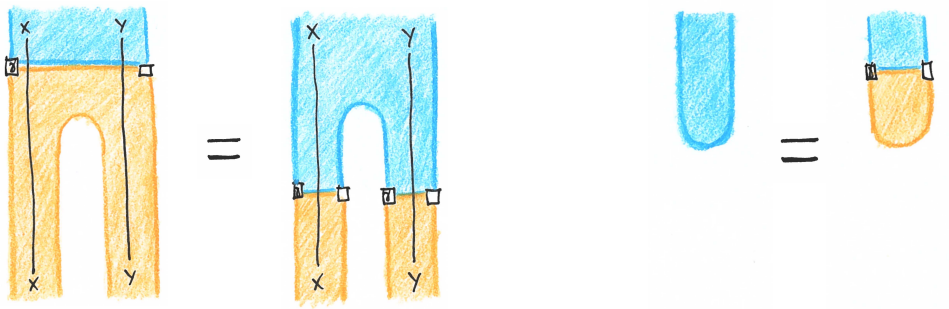
Similarly, if $F, G : \mathbb{X} \to \mathbb{Y}$ are comonoidal functors, $\gamma : F \to G$ a natural transformation, we say that γ is *comonoidal* in case

$$\gamma_{x \otimes y} \psi_{x,y} = \psi_{x,y} (\gamma_x \otimes \gamma_y)$$

and

$$\psi_0 = \gamma_I \psi_0$$

graphically, this is



A Construction

Let $\mathbb{X}$ be a theory of resources, and let $\mathcal{L}$ be a set of colours. Define $\mathcal{L}(\mathbb{X})$ to be the free symmetric strict monoidal category on $\mathbb{X} \times \mathcal{L}$ with the following additional structure:

For each $A \in \mathcal{L}$ and each $x, y \in \mathbb{X}_0$ morphisms:

$$\phi^A_{x,y} : (x, A) \otimes (y, A) \to (x \otimes y, A) \qquad \phi^A_0 : I \to (I, A)$$

$$\psi^A_{x,y} : (x \otimes y, A) \to (x, A) \otimes (y, A) \qquad \psi^A_0 : (I, A) \to I$$

in $\mathcal{L}(\mathbb{X})$ ...

A Construction

. . . such that for $f : x \to x'$, $g : y \to y'$ in $\mathbb{X}$:

[C.1] $\phi^A_{x,y}(f \otimes g, A) = ((f, A) \otimes (g, A))\phi^A_{x',y'}$

[C.2] $\psi^A_{x,y}((f, A) \otimes (g, A)) = (f \otimes g, A)\psi^A_{x',y'}$

[C.3] $(\phi^A_{x,y} \otimes id_{(z,A)})\phi^A_{x\otimes y,z} = (id_{(x,A)} \otimes \phi^A_{y,z})\phi^A_{x,y\otimes z}$

[C.4] $(id_{(x,A)} \otimes \phi^A_0)\phi^A_{x,I} = id_{(x,A)} = (\phi^A_0 \otimes id_{(x,A)})\phi^A_{I,x}$

[C.5] $\psi^A_{x,y\otimes z}(id_{(x,A)} \otimes \psi^A_{y,z}) = \psi^A_{x\otimes y,z}(\psi^A_{x,y} \otimes id_{(z,A)})$

[C.6] $\psi^A_{I,x}(\psi^A_0 \otimes id_{(x,A)}) = id_{(x,A)} = \psi^A_{x,I}(id_{(x,A)} \otimes \psi^A_0)$

[C.7] $\phi^A_{x,y}\psi^A_{x,y} = id_{(x,A)\otimes(y,A)}$

[C.8] $\psi^A_{x,y}\phi^A_{x,y} = id_{(x\otimes y,A)}$

[C.9] $\phi^A_0\psi^A_0 = id_I$

[C.10] $\psi^A_0\phi^A_0 = id_{(I,A)}$

[C.11] $\phi^A_{x,y}(\sigma_{x,y}, A) = \sigma_{(x,A),(y,A)}\phi_{y,x}$

. . .

A Construction

Additionally, for each $A, B \in \mathcal{L}$ and each $x \in \mathbb{X}_0$, a morphism:

$$\gamma_x^{A,B} : (x, A) \to (x, B)$$

in $\mathcal{L}(\mathbb{X})$ such that for $f : x \to x'$ in $\mathbb{X}$:

[G.1] $\gamma_x^{A,B}(f, B) = (f, A)\gamma_{x'}^{A,B}$

[G.2] $(\gamma_x^{A,B} \otimes \gamma_y^{A,B})\phi_{x,y}^B = \phi_{x,y}^A \gamma_{x\otimes y}^{A,B}$

[G.3] $\phi_0^A \gamma_I^{A,B} = \phi_0^B$

[G.4] $\gamma_{x\otimes y}^{A,B}\psi_{x,y}^B = \psi_{x,y}^A(\gamma_x^{A,B} \otimes \gamma_y^{A,B})$

[G.5] $\gamma_I^{A,B}\psi_0^B = \psi_0^A$

[G.6] $\gamma_x^{A,B}\gamma_x^{B,C} = \gamma_x^{A,C}$

[G.7] $\gamma_x^{A,A} = id_{(x,A)}$

A Construction

Proposition

If $\mathbb{X}$ is a symmetric strict monoidal category, and $\mathcal{L}$ is a set. There is a strong symmetric monoidal functor

$$A : \mathbb{X} \to \mathcal{L}(\mathbb{X})$$

for each $A \in \mathcal{L}$. Further, there is a monoidal and comonoidal natural transformation

$$\gamma^{A,B} : A \to B$$

between the functors corresponding to any two $A, B \in \mathcal{L}$.

A Construction

While the previous proposition does not require **[G.6]** and **[G.7]**, they give:

Proposition

For each $A \in \mathcal{L}$, the functor $A : \mathbb{X} \to \mathcal{L}(\mathbb{X})$ is the left adjoint in an equivalence of categories.

Interpretation

Interpret $A(x)$ as an instance of x owned by A. Note that every object of $\mathcal{L}(\mathbb{X})$ is $A(x)$ for some $A \in \mathcal{L}$.

The (co)monoidal functor maps ϕ^A, ψ^A are natural axioms for managing the logical grouping of A's resources. (like Δ, ∇ in $\mathbb{M}$, but for non-homogeneous resource theories).

UTXOs of type $A(x) \otimes A(y)$ correspond to two different addresses, and may be spent separately, while UTXOs of type $A(x \otimes y)$ correspond to one address, and must be spent together.

Interpretation

To spend a UTXO of type $A(x)$, must supply evidence that you have access to permission level A (e.g., Alice signs the block number).

The $\gamma_x^{A,B}$ maps allow us to "give" an x from A to B. (Co)monoidality of $\gamma^{A,B}$ ensures that giving away a collection of resources is the same as giving away its constituents.

Maps should be equal if they denote transactions with the same effect. (Syntax $\rightsquigarrow$ literal sequence of events, semantics $\rightsquigarrow$ effect).

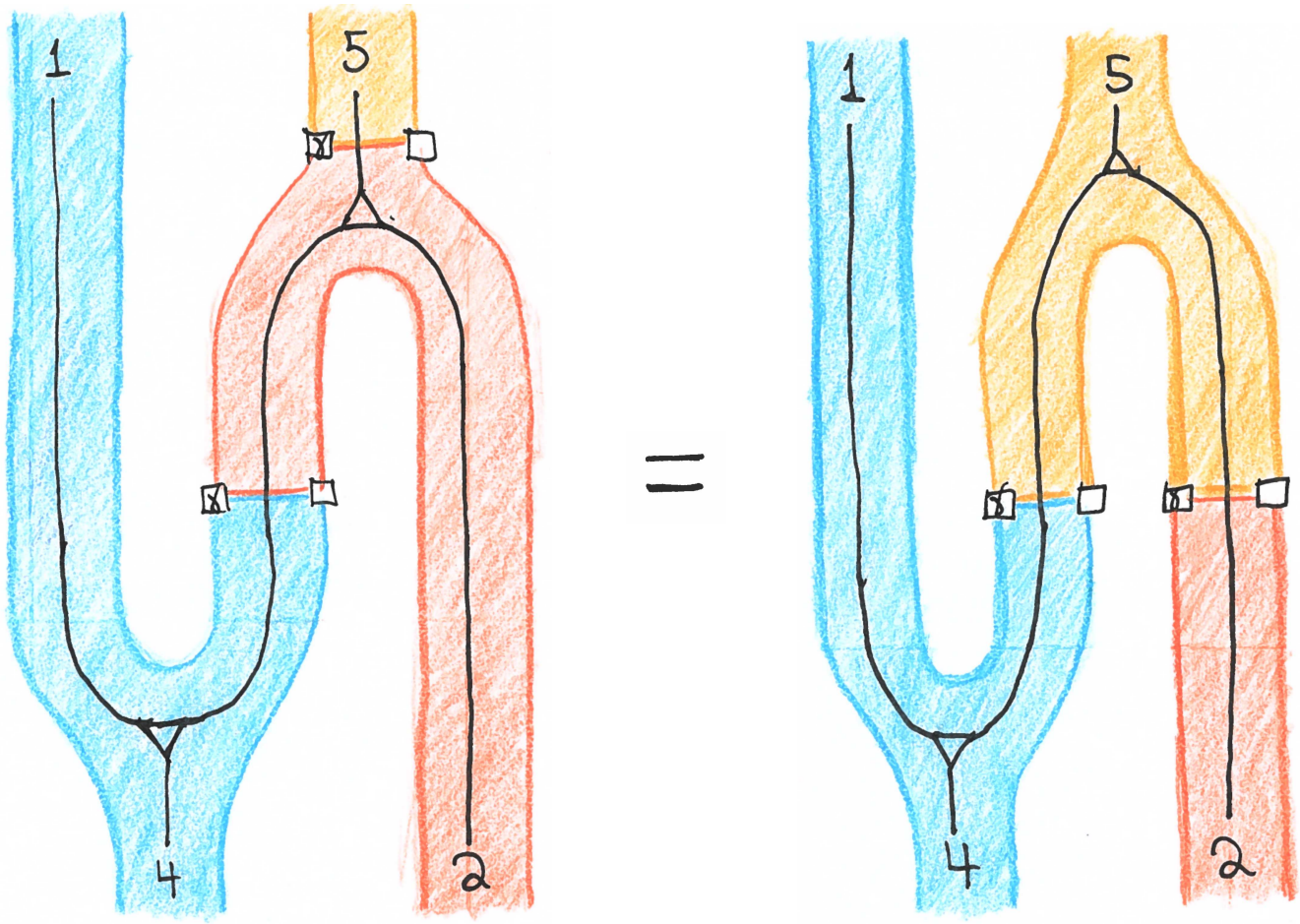
Let's Pretend Colours are People

We leave the precise nature of $\mathcal{L}$ unspecified, assuming for the sake of example that it contains four colours, each associated with a person:

Our next examples are with respect to $\mathcal{L}(\mathbb{M})$.

Small Example

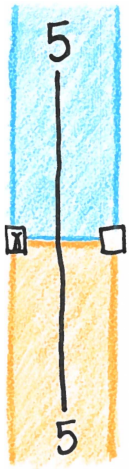
Transactions in $\mathcal{L}(\mathbb{M})$ are things like:



Big Example

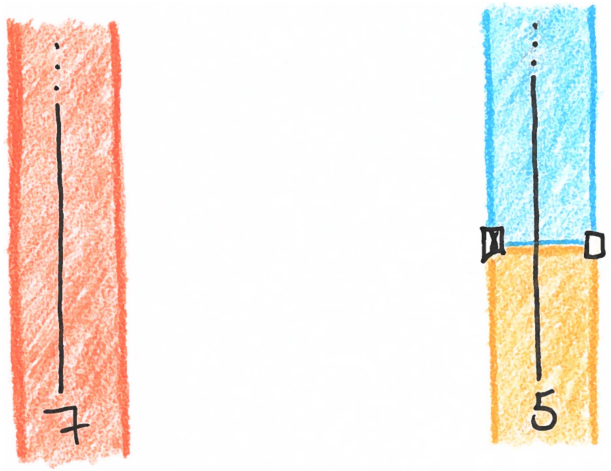
Supose the state of the ledger is:

Alice wishes to use their 5 to buy a bicycle from Bob, so submits the following transaction (along with proof of identity):

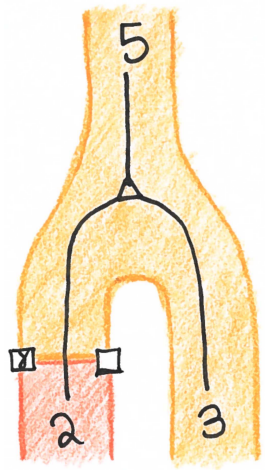


Big Example

The resulting state of the ledger is:

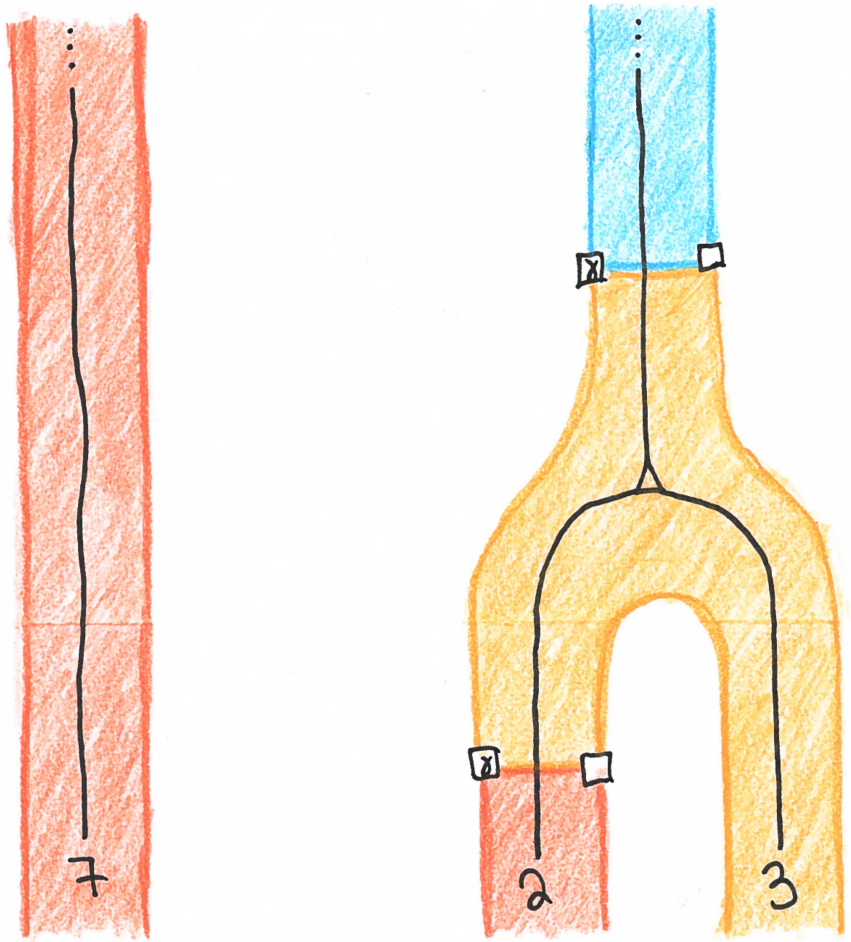


Now, Bob uses 2 of this to buy a coffee from Carol, keeping the other 3:



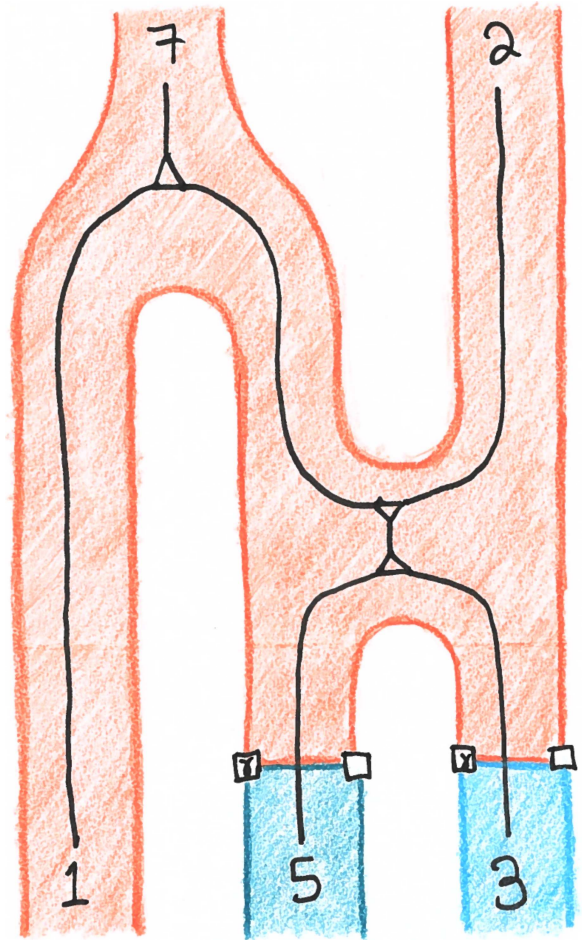
Big Example

The resulting state of the ledger is:



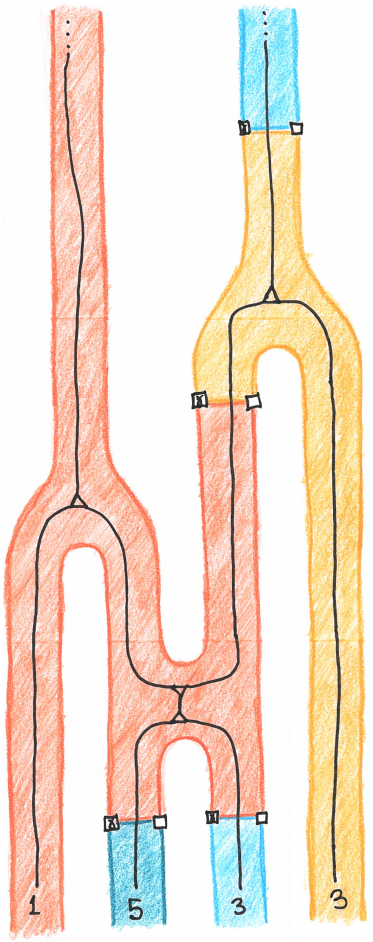
Big Example

Some times later, Carol buys a drum from Dave for 5, and some apples from Alice for 3. Carol must reorganize their coins to do this:



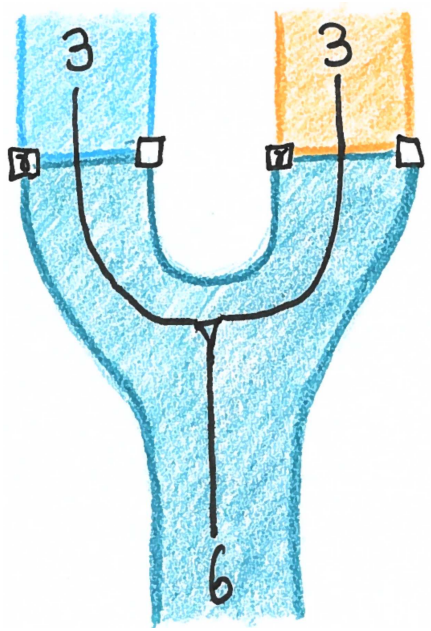
Big Example

The resulting state of the ledger is:



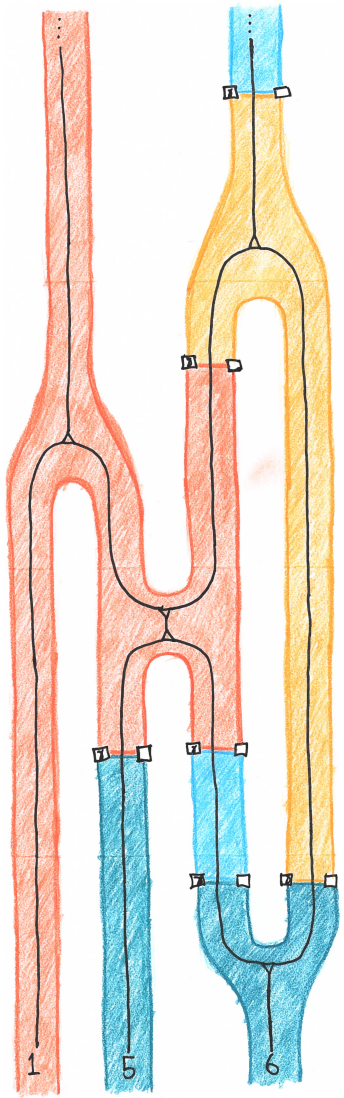
Big Example

Finally, Alice and Bob pool their resources to have Dave dig them a ditch. Alice and Bob must each supply proof of identity:



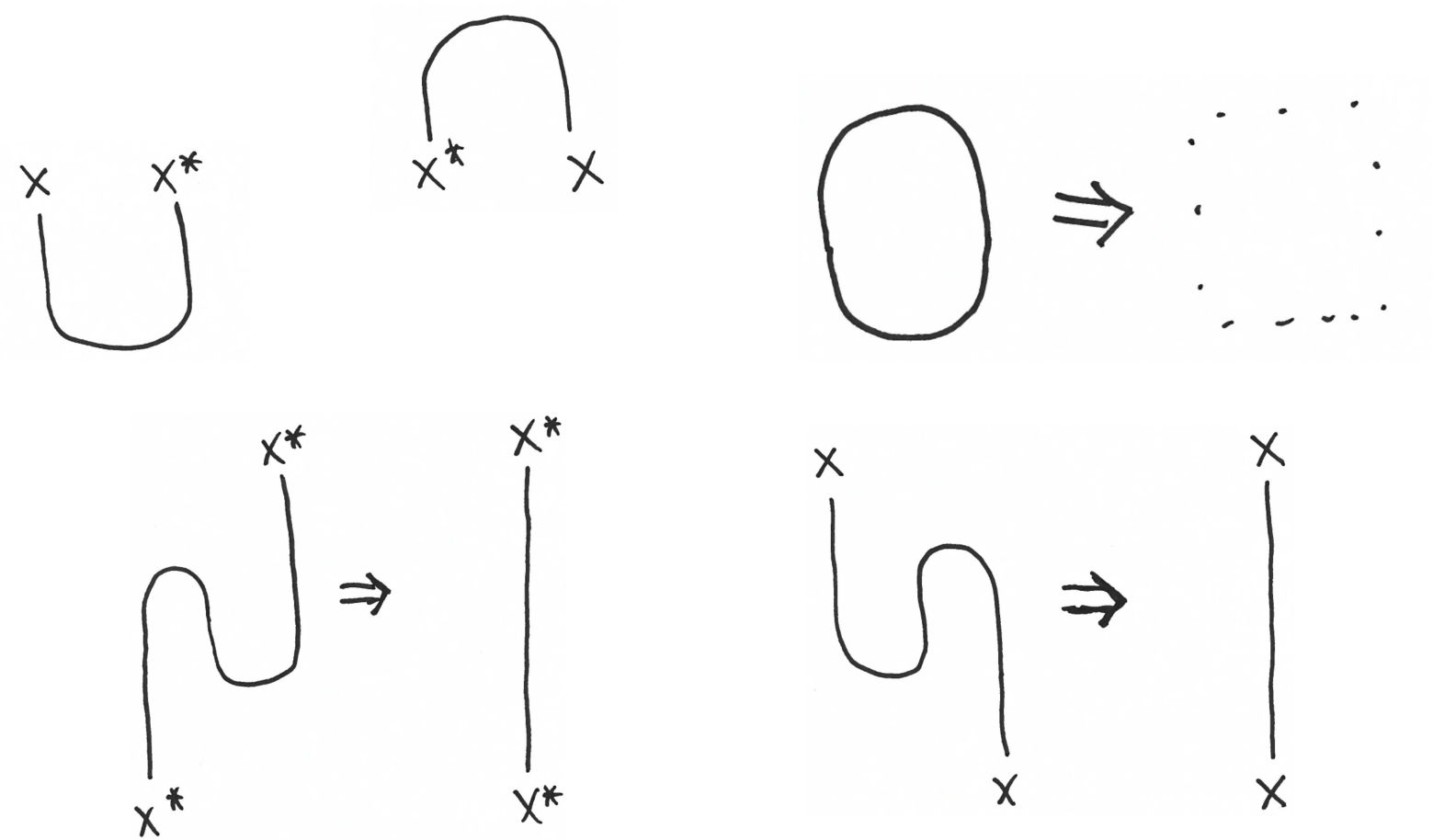
Big Example

The resulting state of the ledger is:



Compact Closed Resources

Suppose we freely add compact closed duals to our theory of resources:



Compact Closed Resources

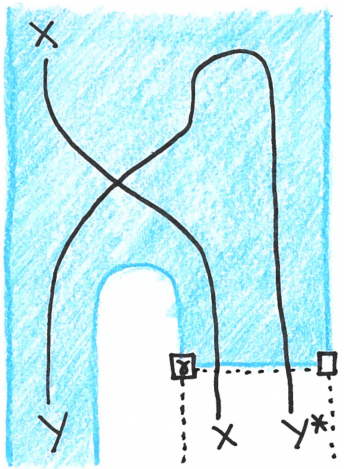
$I \to x \otimes x^*$ is like a promise to supply an x later. Spending the x^* is *supplying* the corresponding x.

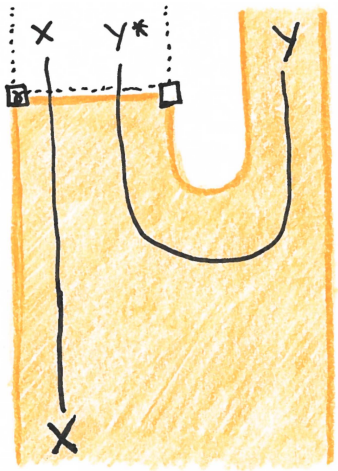
Most applications require us to disallow spending the x until the x^* is supplied. (Exception: fractional reserve banking).

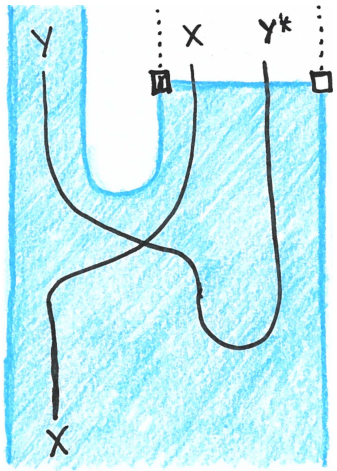
Erase colours and perform cut elimination to simplify this analysis.

Exchange Example

If our theory of resources $\mathbb{X}$ is compact closed, then in $\mathcal{L}(\mathbb{X})$ we can construct morphisms:



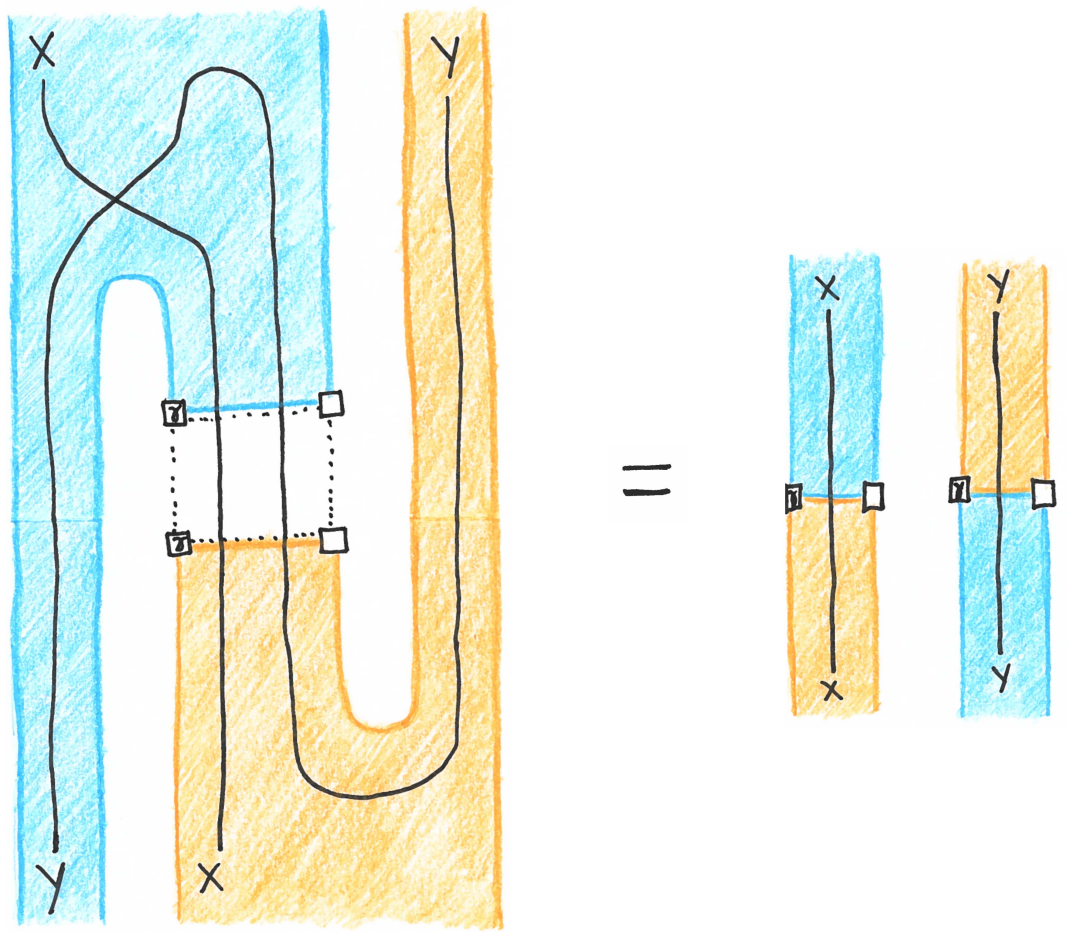




As transactions, the first has the effect of offering an exchange, the seconds accepts the exchange, and the third revokes the offer.

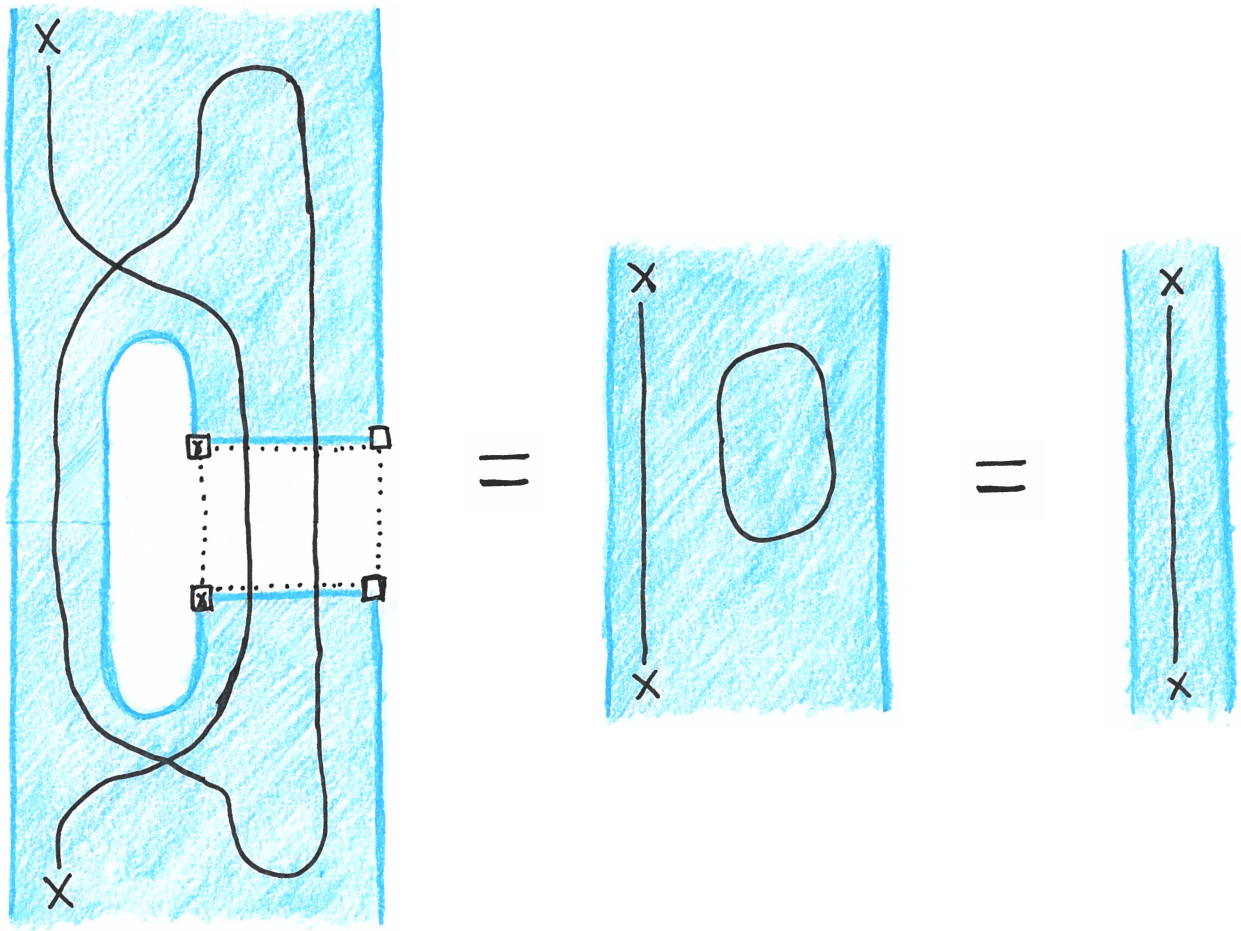
Exchange Example

Composing the first with the second results in a swap:



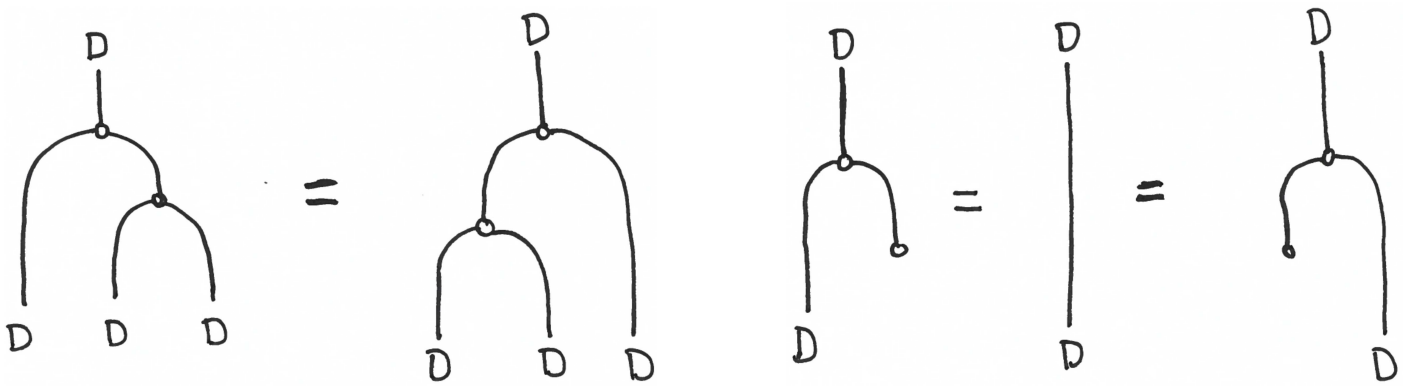
Exchange Example

While composing the first with the third results in the identity:



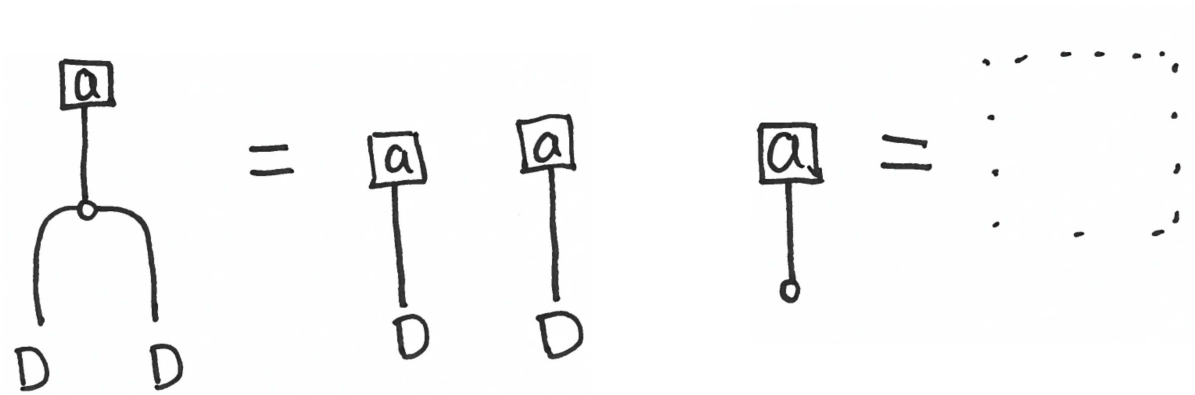
Comonoidal Data and PCAs

Commutative comonoids act like *data*:



Comonoidal Data and PCAs

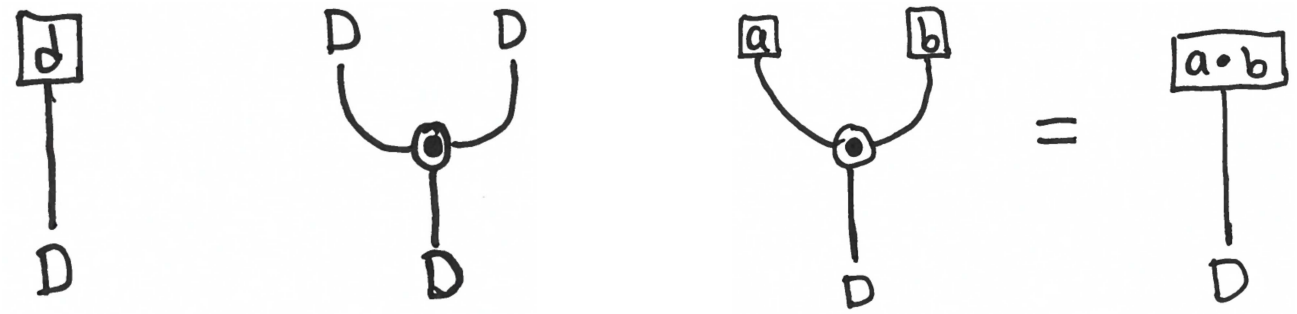
Each $a \in D$ manifests as a morphism of comonoids $a : I \to D$



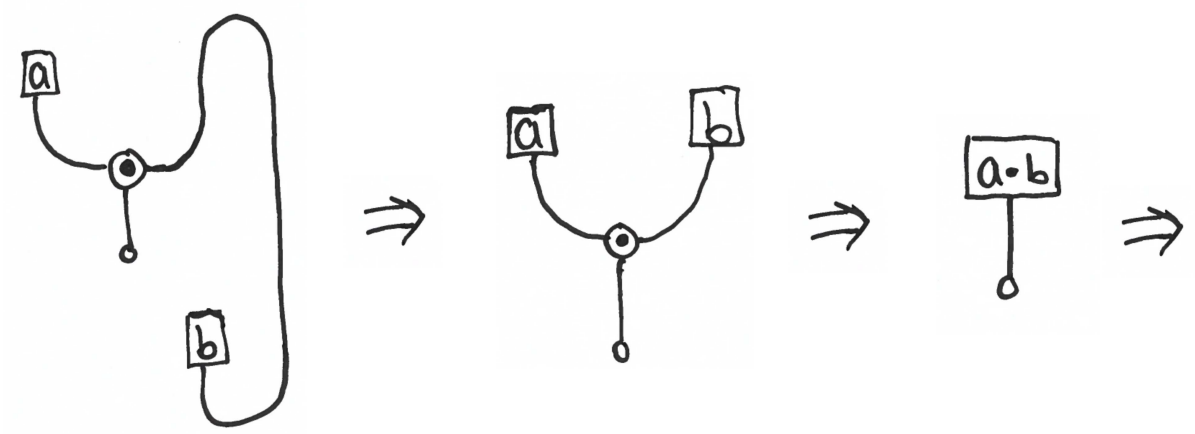
In any symmetric monoidal category, The subcategory of commutative comonoids and comonoid homomorphisms is cartesian.

Comonoidal Data and PCAs

Suppose D is a commutative comonoid with the structure of a partial combinatory algebra:

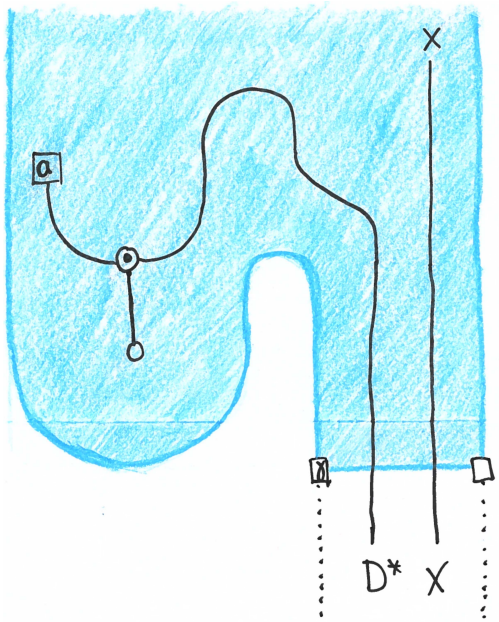


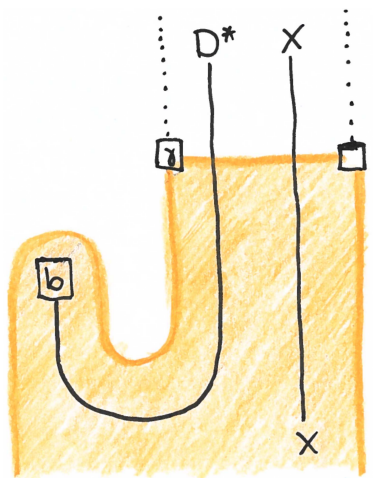
If we add such a D to our compact closed resource theory of money, we have:



Validator/Redeemer Example

If our theory of resources $\mathbb{X}$ is compact closed and contains such a PCA, then in $\mathcal{L}(\mathbb{X})$ we can construct morphisms:

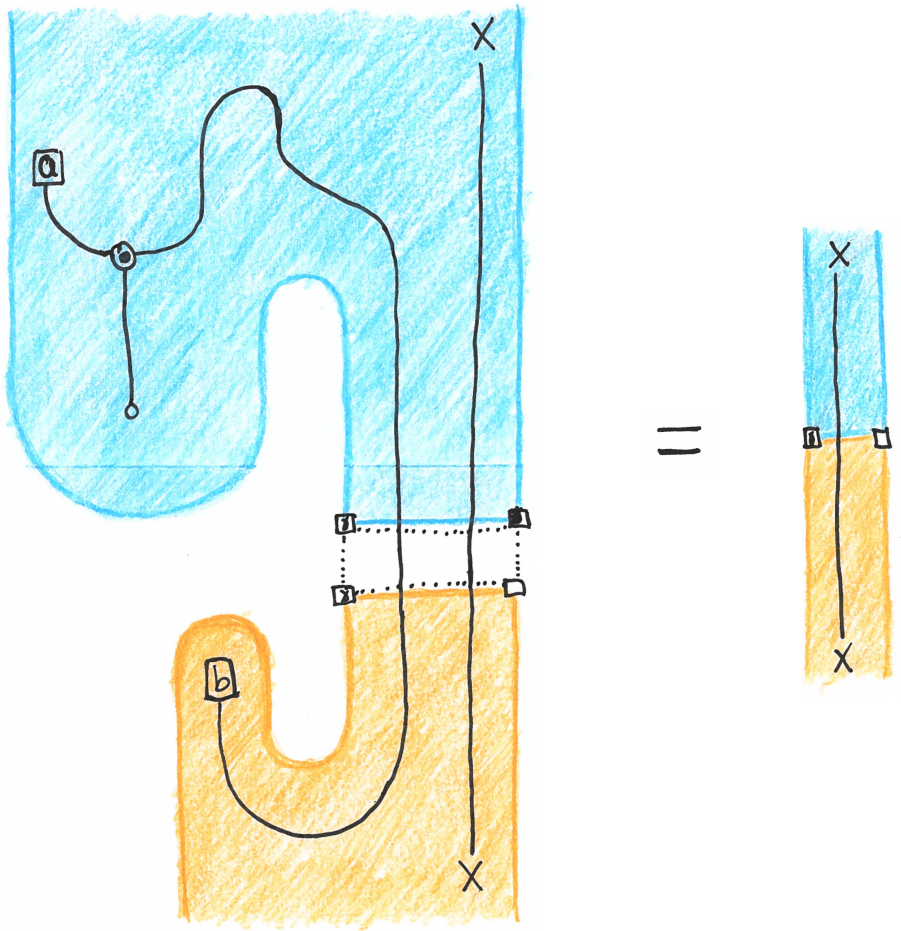




As transactions, the first locks the x with validator a, and the second attempts to spend the x with redeemer b.

Validator/Redeemer Example

If $a \bullet b \downarrow$, then the composite is $\gamma_x^{A,B}$:



If $a \bullet b \uparrow$, the transaction attempting to spend the x is rejected.

Future Work

Linear combinatory algebras for Ethereum-like smart contracts.

Which features do we want our ledger to have? Appropriate notion of simulation for ledgers.

Other applications of the $\mathcal{L}(\mathbb{X})$ construction.

Summary (Last Slide)

Blockchain $\rightsquigarrow$ Series of Tubes

Thanks for listening!