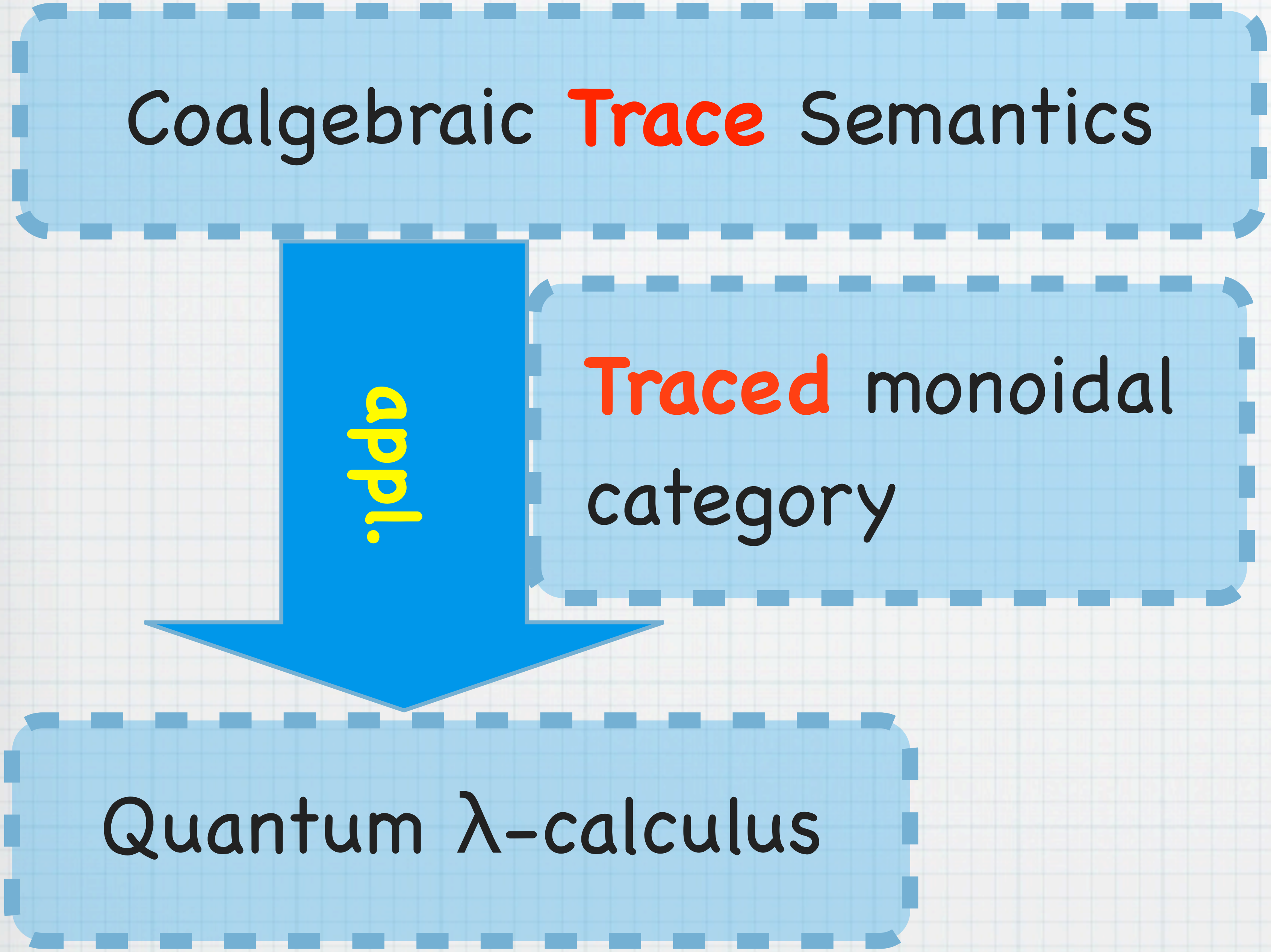
Trace Everywhere

Based on: IH & N. Hoshino, **Semantics of Higher-Order Quantum Computation via Geometry of Interaction,** Proc. LICS 2011

Ichiro Hasuo
University of Tokyo (JP)

東京大学
THE UNIVERSITY OF TOKYO

Three "Traces"



Coalgebraic **Trace** Semantics

appl.

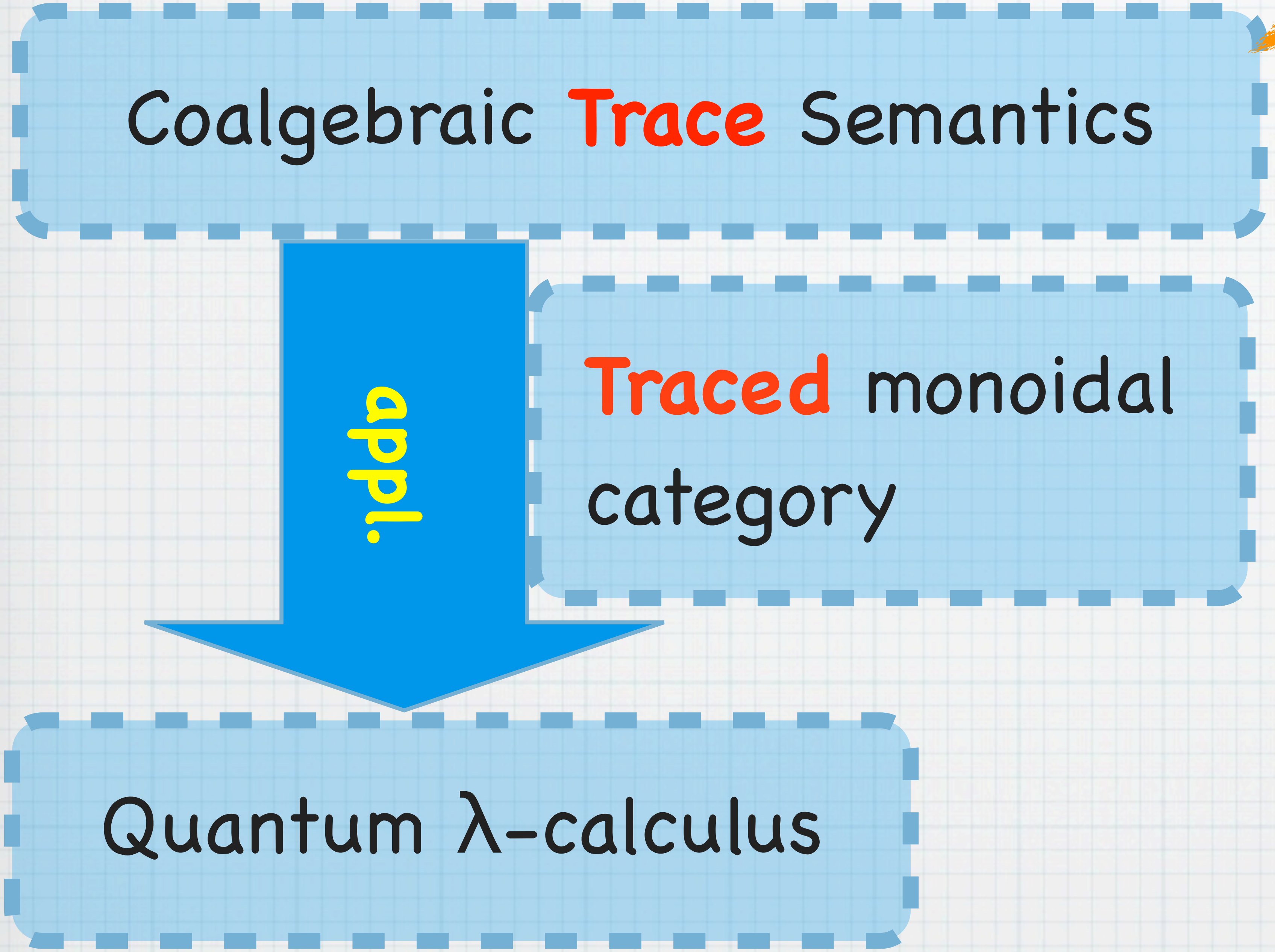
Traced monoidal category

Quantum λ-calculus

Three "Traces"

$$\begin{tikzcd}
\boldsymbol{FX} \arrow[r, dashed, "\boldsymbol{F}\mathbf{beh}(\boldsymbol{c})"] & \boldsymbol{FZ} \\
\boldsymbol{X} \arrow[u, "\boldsymbol{c}"] \arrow[r, dashed, "\mathbf{beh}(\boldsymbol{c})"'] & \boldsymbol{Y} \arrow[u, "\mathrm{final}"']
\end{tikzcd}$$

Coinduction in $\boldsymbol{Kl}(\mathbf{B})$



Coalgebraic **Trace** Semantics

Traced monoidal category

Quantum λ-calculus

Three "Traces"

$$\begin{tikzcd} FX \arrow[r, dashed, "F\mathsf{beh}(c)"] & FZ \\ X \arrow[u, "c"] \arrow[r, dashed, "\mathsf{beh}(c)"'] & Y \arrow[u, "\mathrm{final}"'] \end{tikzcd}$$

Coinduction in $\mathcal{K}\ell(\mathbf{B})$

Coalgebraic **Trace** Semantics

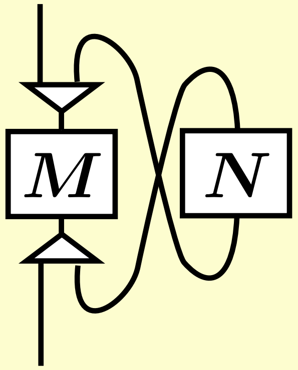
appl.

Traced monoidal category

Quantum λ-calculus

Categorical GoI

[Abramsky, Haghverdi, Scott]



Three "Traces"

$$\begin{tikzcd}
FX \arrow[r, dashed, "F\mathsf{beh}(c)"] & FZ \\
X \arrow[u, "c"] \arrow[r, dashed, "\mathsf{beh}(c)"'] & Y \arrow[u, "\mathrm{final}"']
\end{tikzcd}$$

Coinduction in $\mathcal{K}\ell(\mathrm{B})$

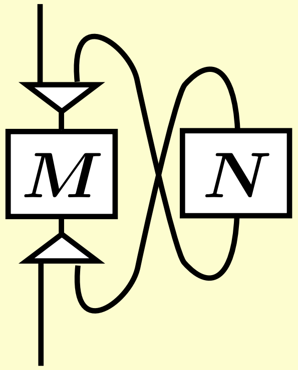
Coalgebraic **Trace** Semantics

appl.

Traced monoidal category

Categorical GoI

[Abramsky, Haghverdi, Scott]



Quantum λ-calculus

Measurements by **tracing out** matrices

Three “Traces”

Coalgebraic **Trace** Semantics

$$\begin{tikzcd} FX \arrow[r, dashed, "F\mathsf{beh}(c)"] & FZ \\ X \arrow[u, "c"] \arrow[r, dashed, "\mathsf{beh}(c)"'] & Y \arrow[u, "\mathrm{final}"'] \end{tikzcd}$$

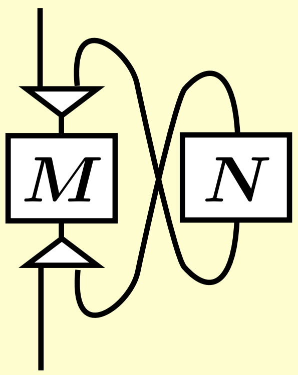
Coinduction in $\mathcal{K}\ell(\mathrm{B})$

appl.

Traced monoidal category

Categorical GoI

[Abramsky, Haghverdi, Scott]



Quantum λ-calculus

Measurements by **tracing out** matrices

- Goal: Denotational model of a quantum λ-calculus

Three "Traces"

$$\begin{tikzcd}
FX \arrow[r, dashed, "F\mathsf{beh}(c)"] & FZ \\
X \arrow[u, "c"] \arrow[r, dashed, "\mathsf{beh}(c)"'] & Y \arrow[u, "\mathrm{final}"']
\end{tikzcd}$$

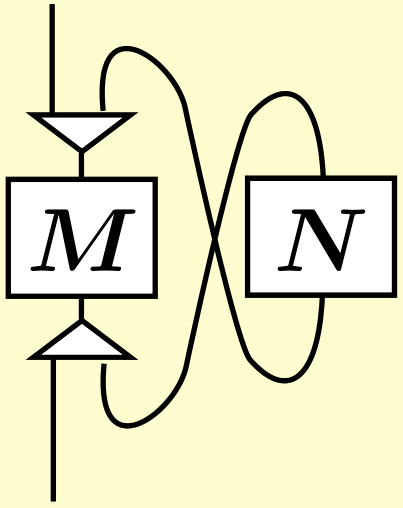
Coinduction in $\mathcal{K}l(\mathbf{B})$

Coalgebraic **Trace** Semantics

appl.

Traced monoidal category

Categorical GoI



[Abramsky, Haghverdi, Scott]

Quantum λ-calculus

Mea

trac

- Goal: Denotational model of a quantum λ-calculus

GoI:
Geometry of Interaction

- J.-Y. Girard, at Logic Colloquium '88

GoI:
Geometry of Interaction

- J.-Y. Girard, at Logic Colloquium '88
- Provides denotational semantics $[\![M]\!]$ for linear λ-term M

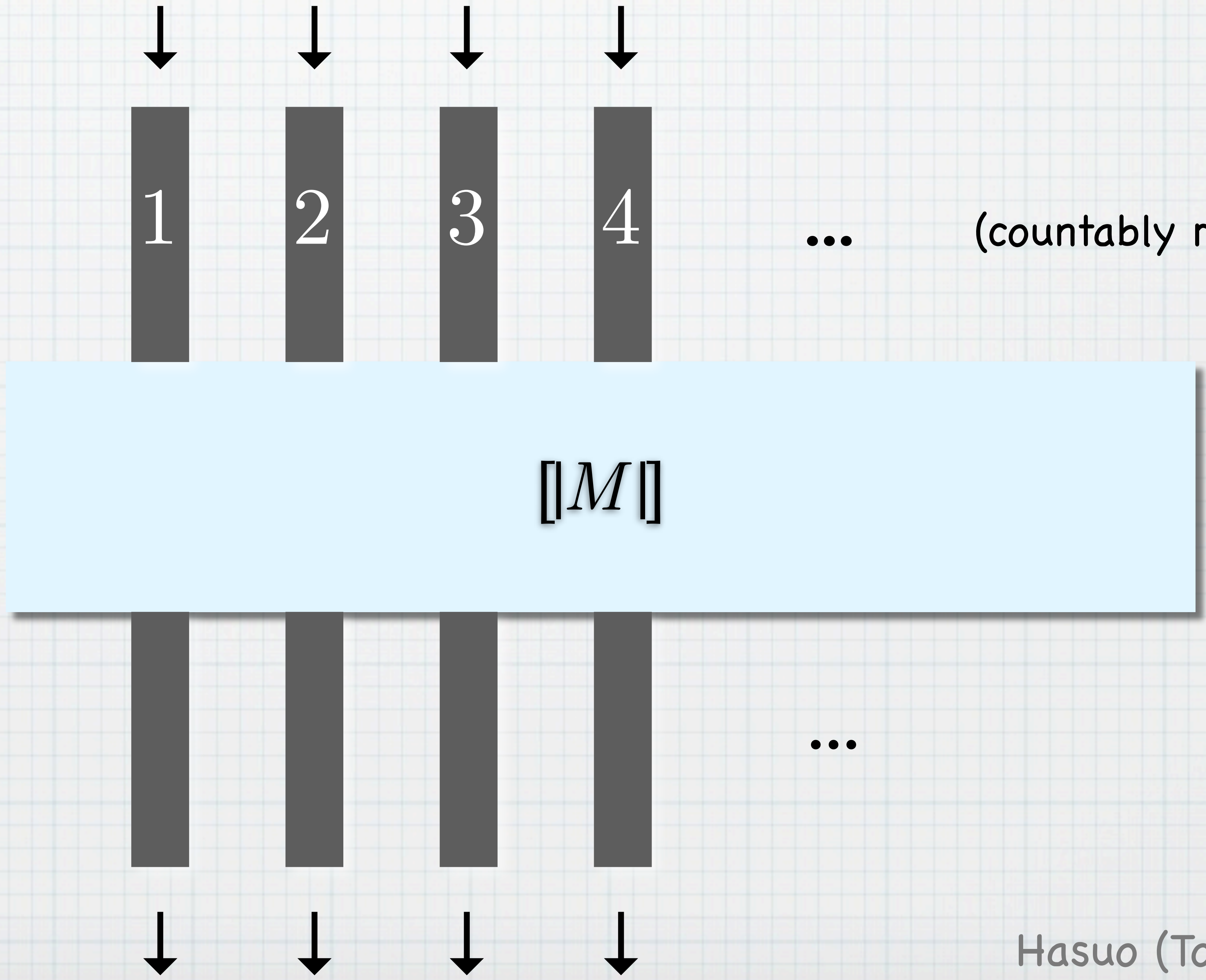
GoI:
Geometry of Interaction

- J.-Y. Girard, at Logic Colloquium '88
- Provides denotational semantics $[\![M]\!]$ for linear λ-term M
- In this talk:
  - Its categorical formulation [Abramsky, Haghverdi, Scott '02]
  - "The GoI Animation"

The GoI Animation

$$[\![M]\!] \;=\; (\,\mathbb{N} \rightharpoonup \mathbb{N},\;\text{a partial function}\,)$$

$$=\;\text{“piping”}$$

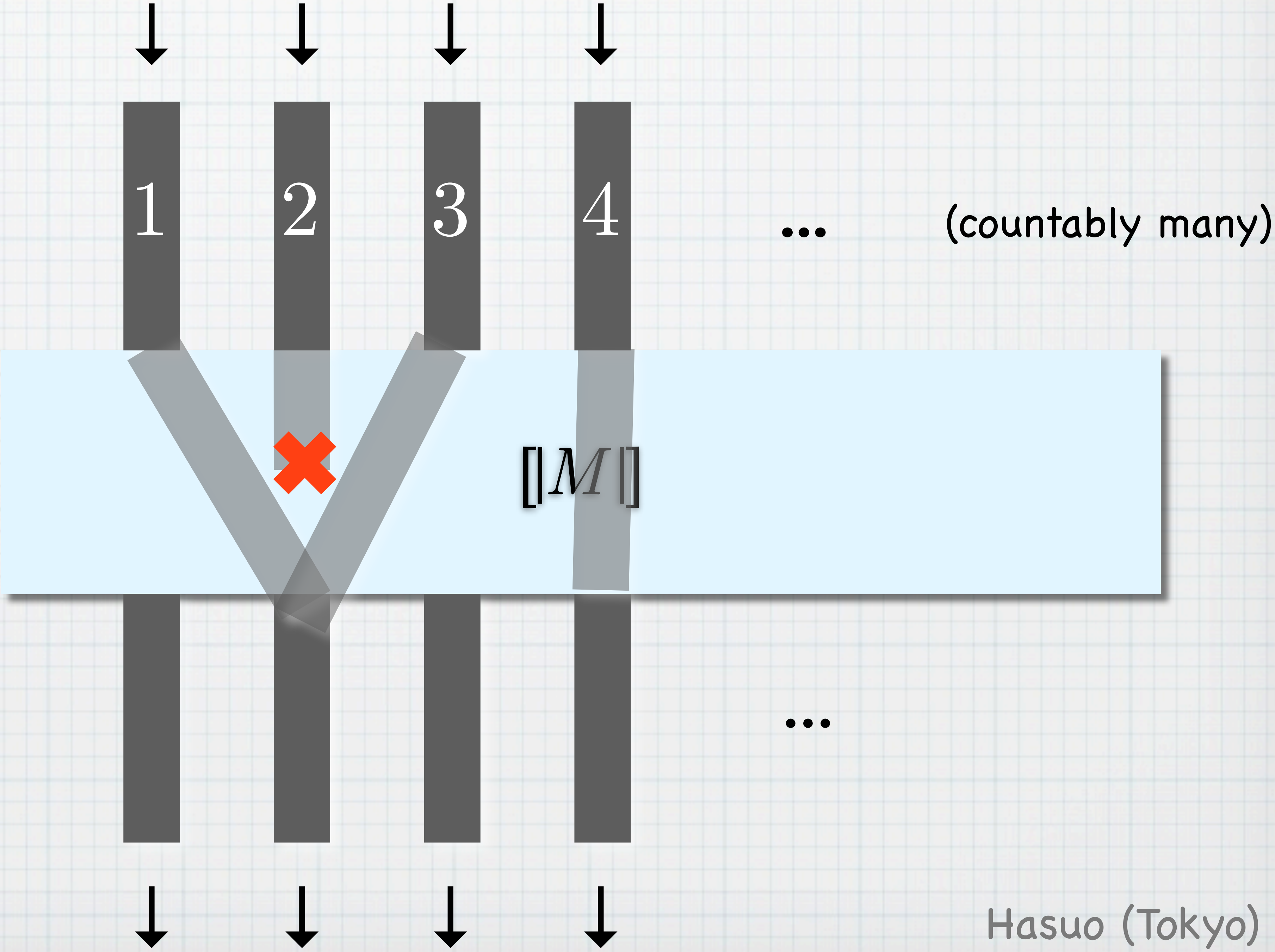


(countably many)

The GoI Animation

$$[\![M]\!] = (\ \mathbb{N} \rightharpoonup \mathbb{N},\ \text{a partial function}\ )$$

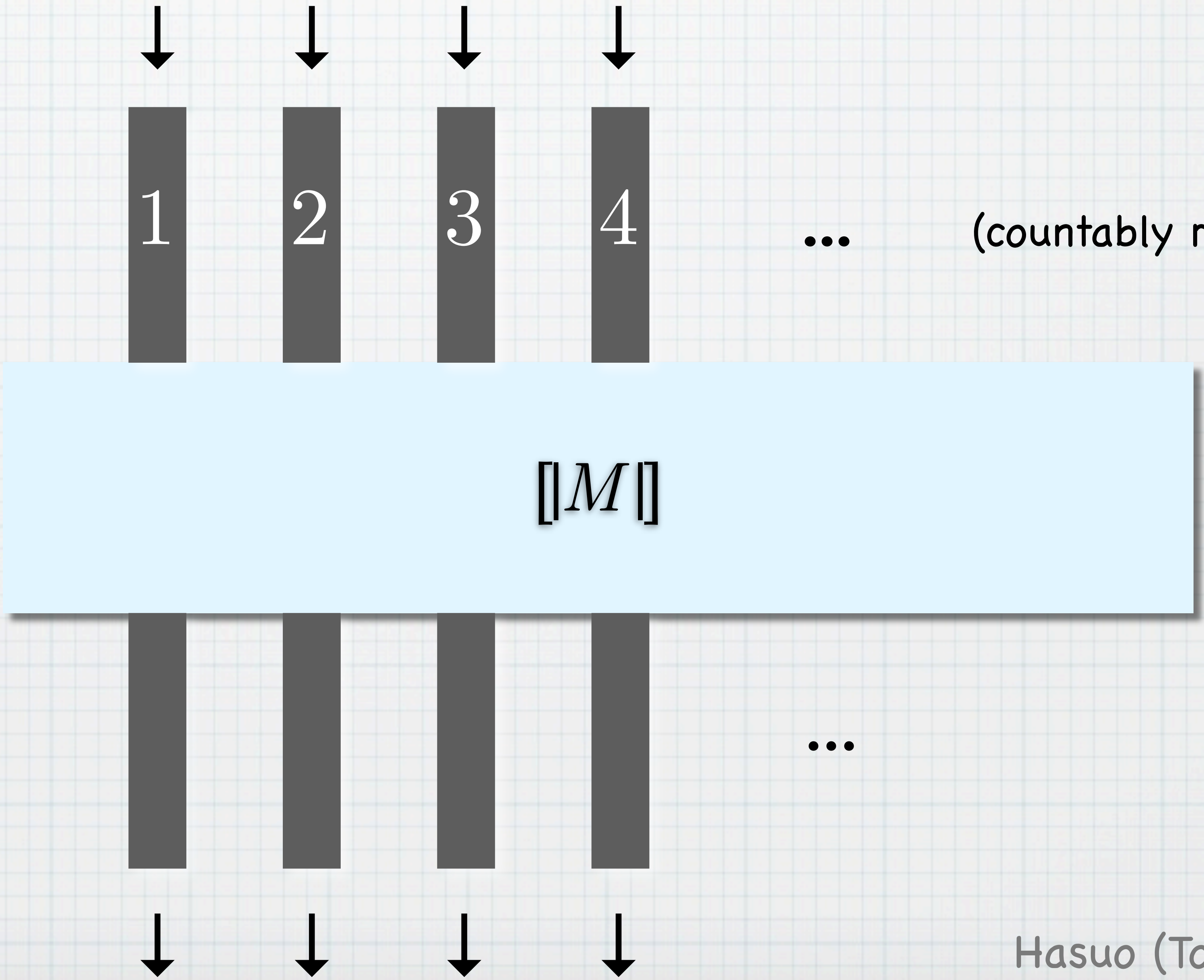
$$= \text{“piping”}$$



The GoI Animation

$$\llbracket M \rrbracket \;=\; (\,\mathbb{N} \rightharpoonup \mathbb{N},\ \text{a partial function}\,)$$

$=$ “piping”

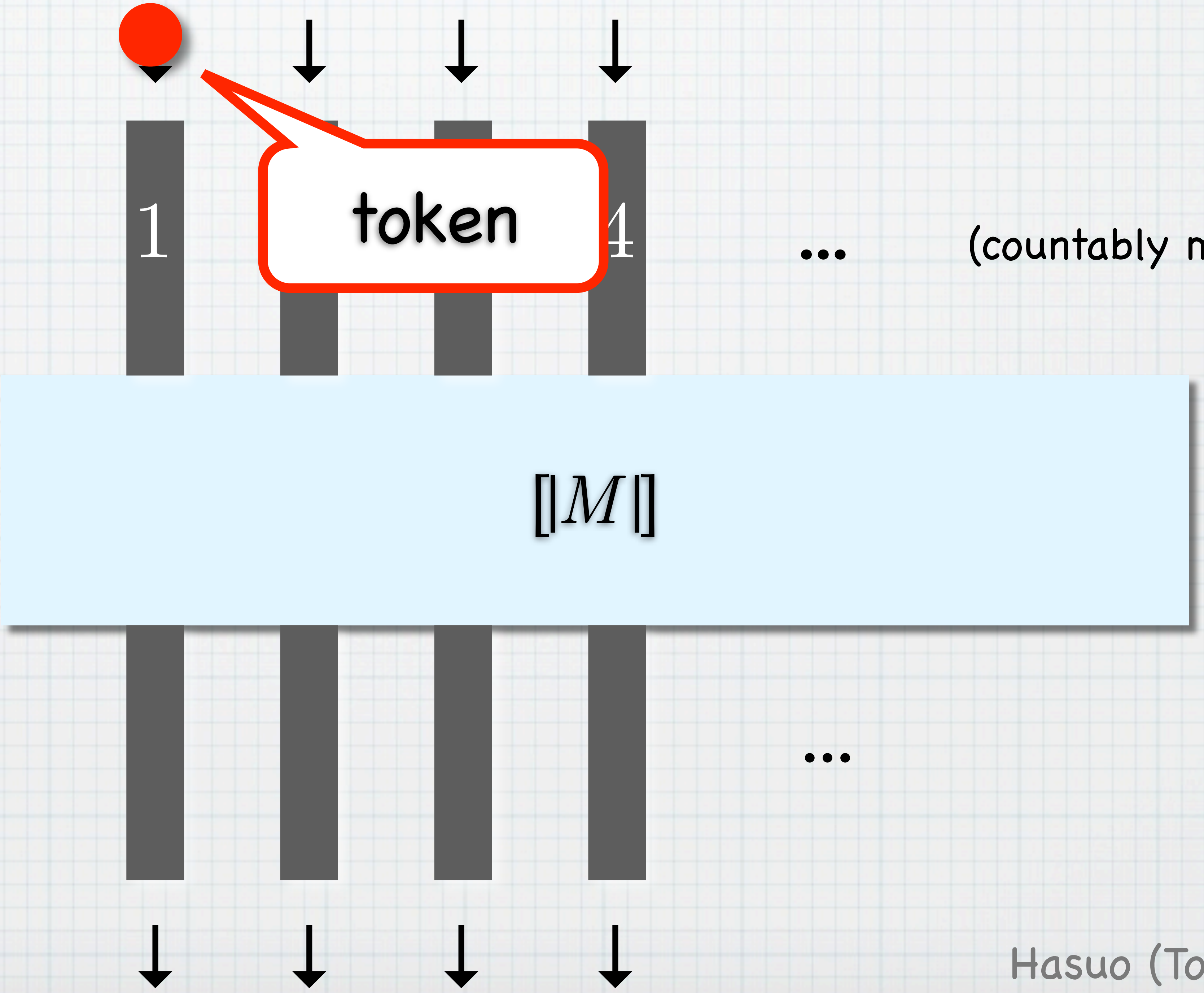


(countably many)

The GoI Animation

$$[\![M]\!] \;=\; (\,\mathbb{N} \rightharpoonup \mathbb{N},\ \text{a partial function}\,)$$

$$=\;\text{“piping”}$$

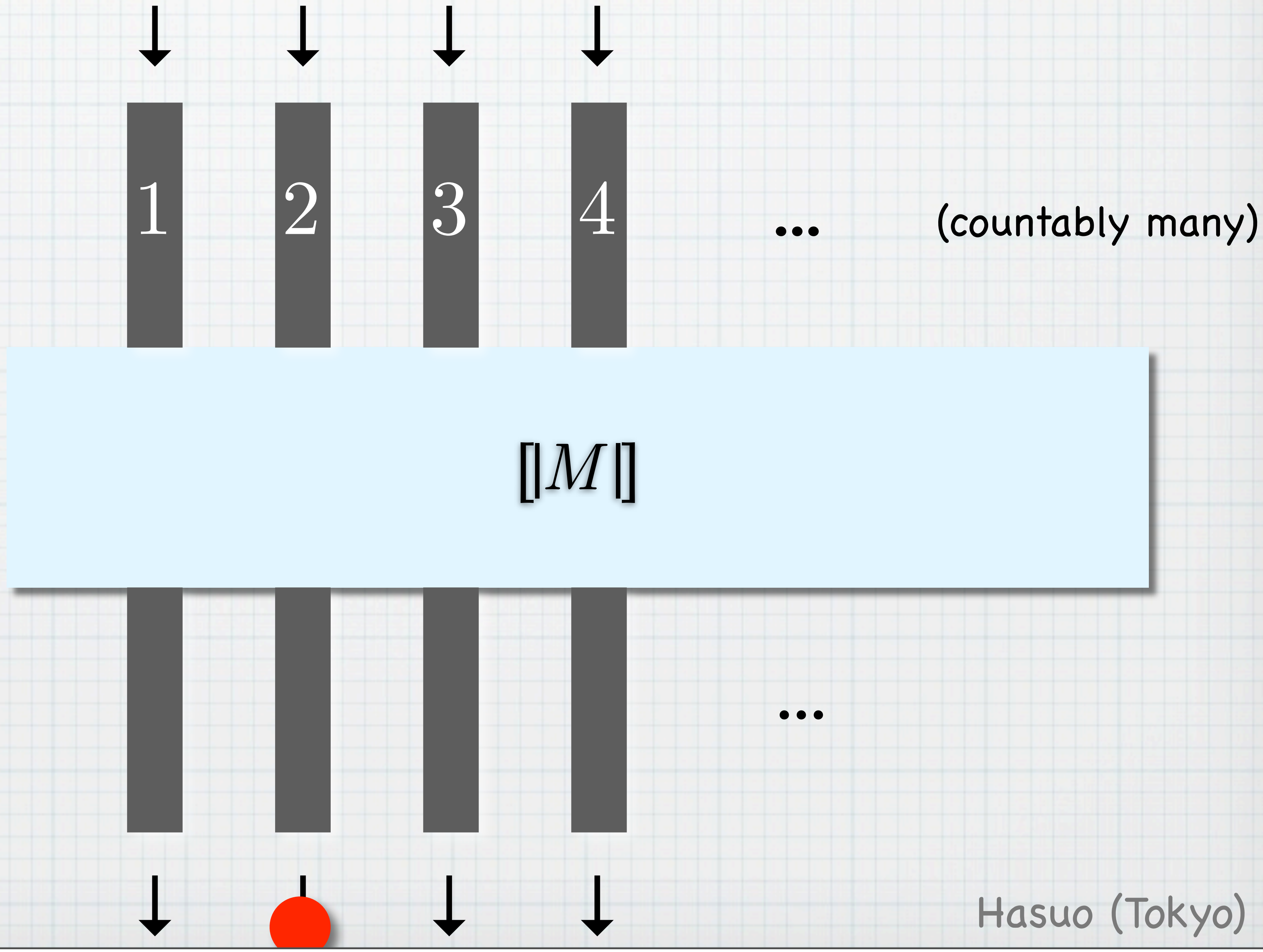


(countably many)

The GoI Animation

$$[\![M]\!] \;=\; (\,\mathbb{N} \rightharpoonup \mathbb{N},\ \text{a partial function}\,)$$

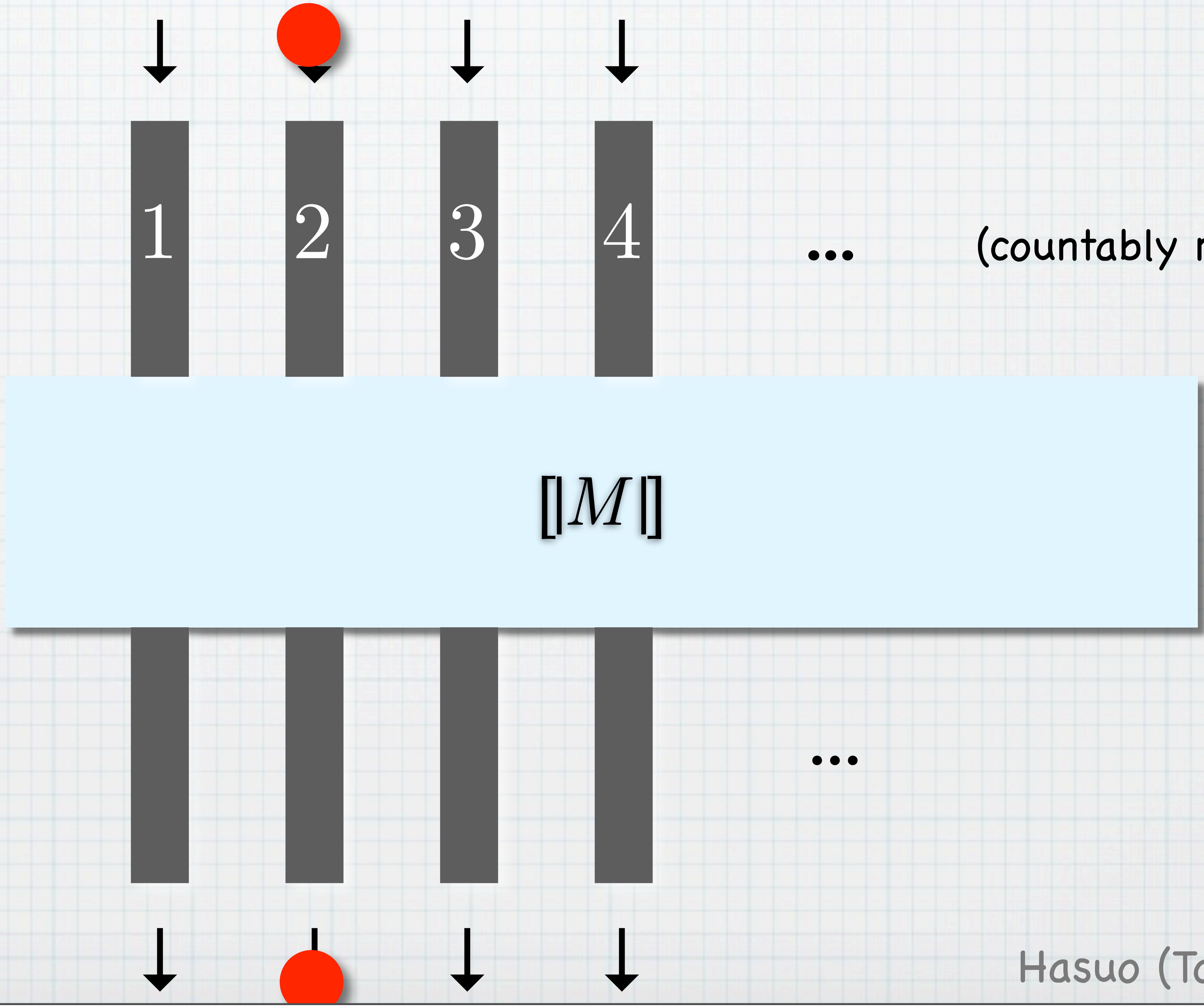
$$=\ \text{“piping”}$$



The GoI Animation

$$[\![M]\!] \;=\; (\,\mathbb{N} \rightharpoonup \mathbb{N},\ \text{a partial function}\,)$$

$$=\;\text{“piping”}$$

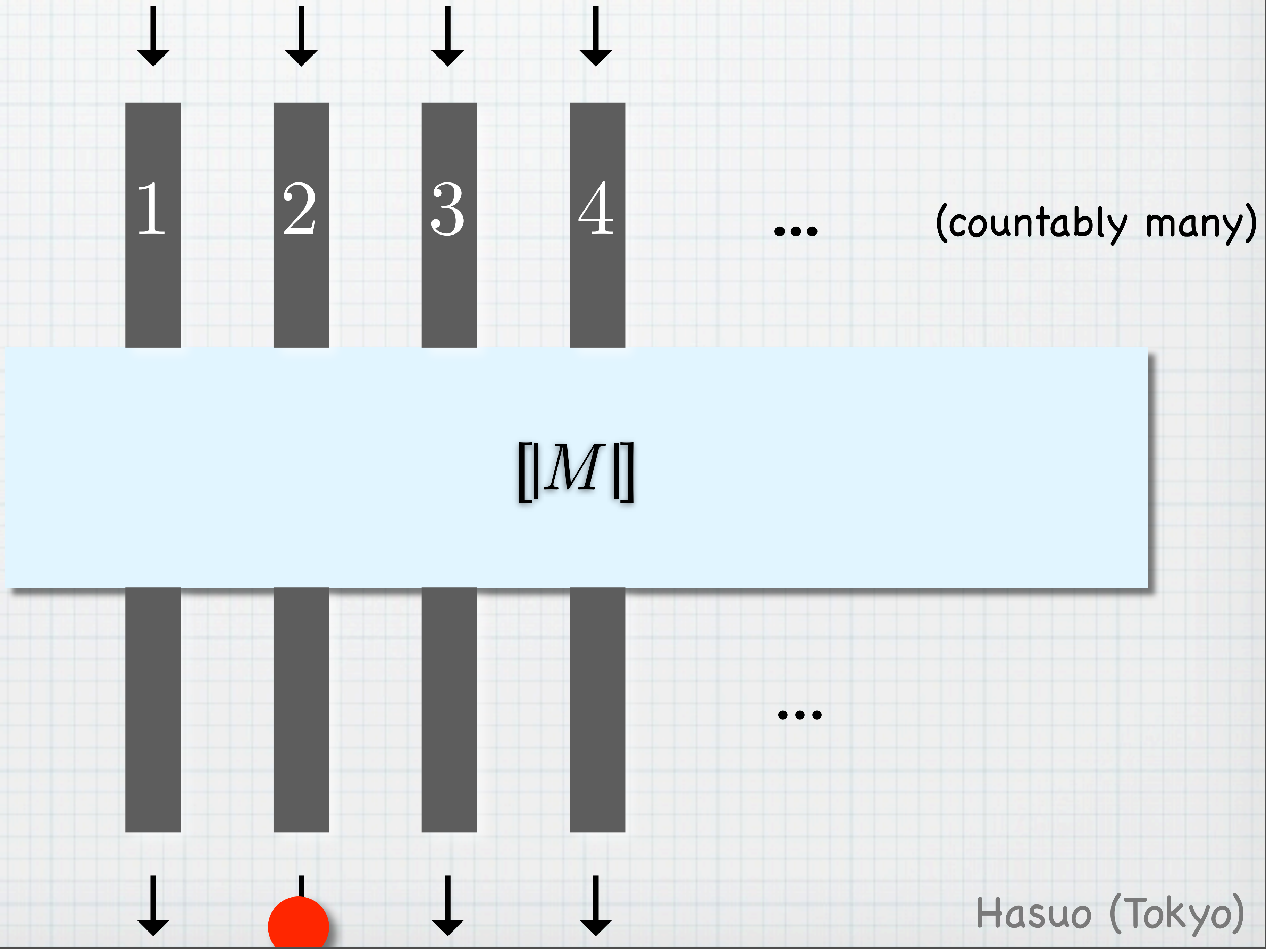


(countably many)

The GoI Animation

$$[\![M]\!] \;=\; (\,\mathbb{N} \rightharpoonup \mathbb{N},\ \text{a partial function}\,)$$

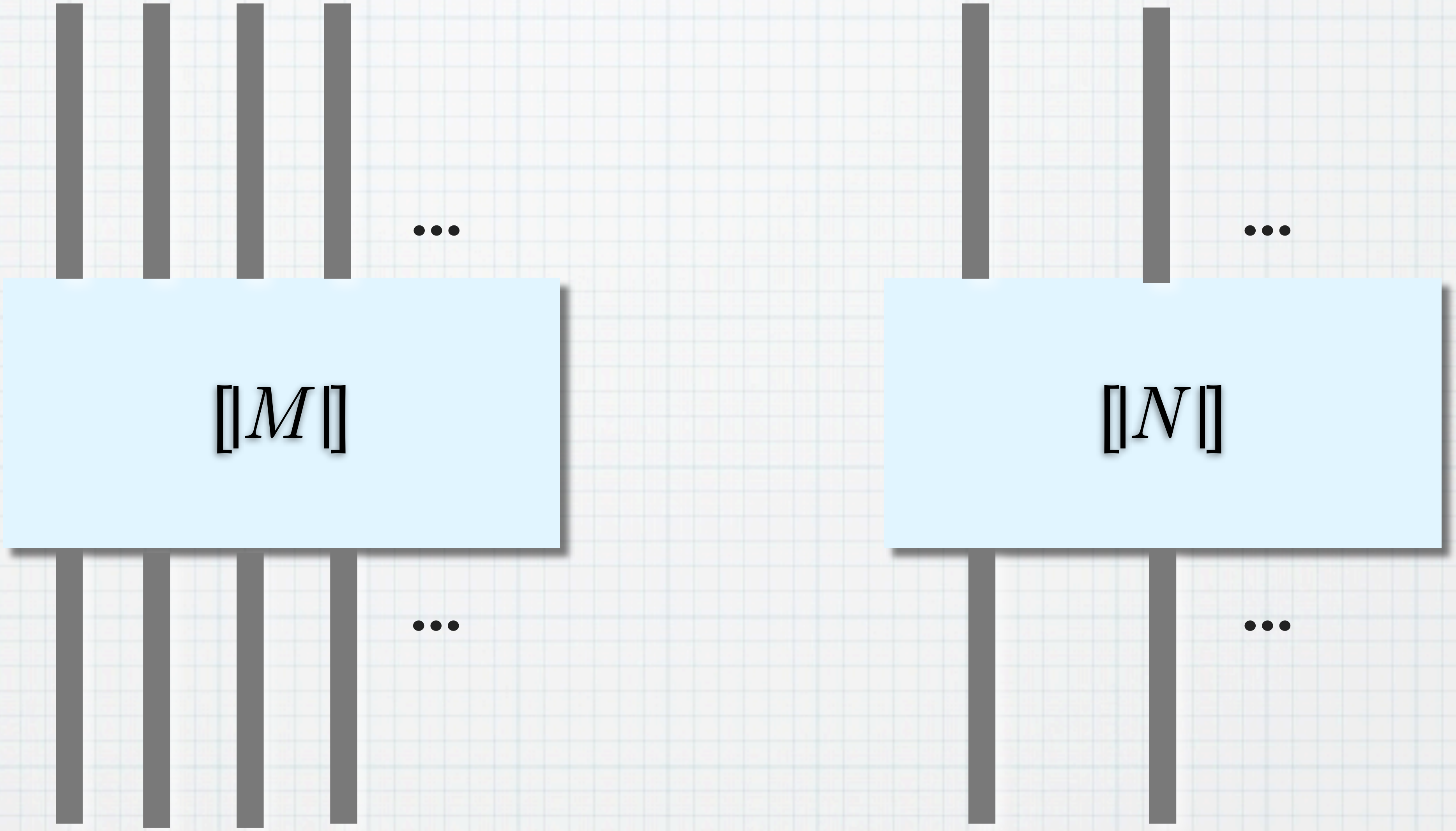
$= \;$ “piping”



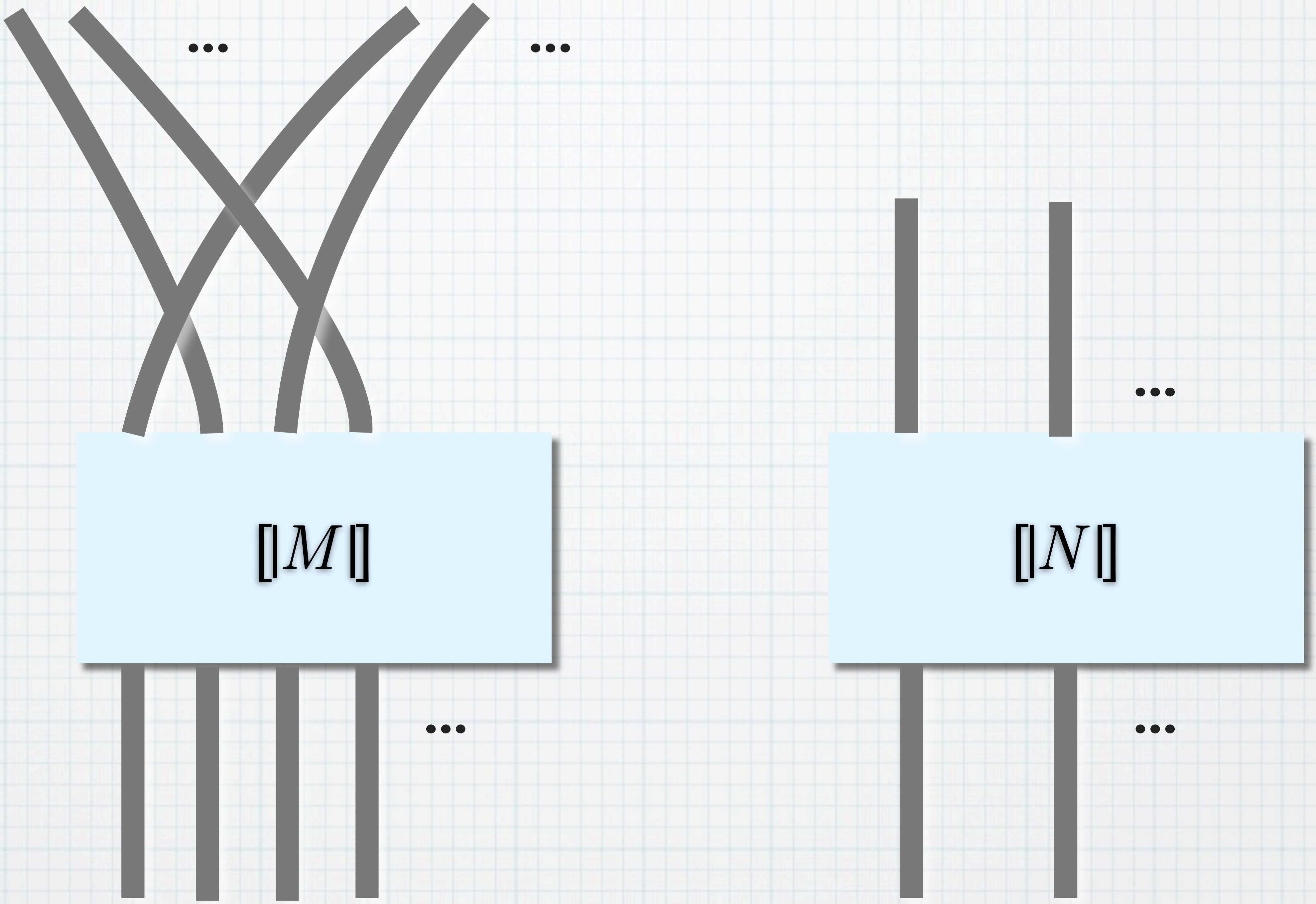
The GoI Animation

- Function application $[\![MN]\!]$
  - by “parallel composition + hiding”

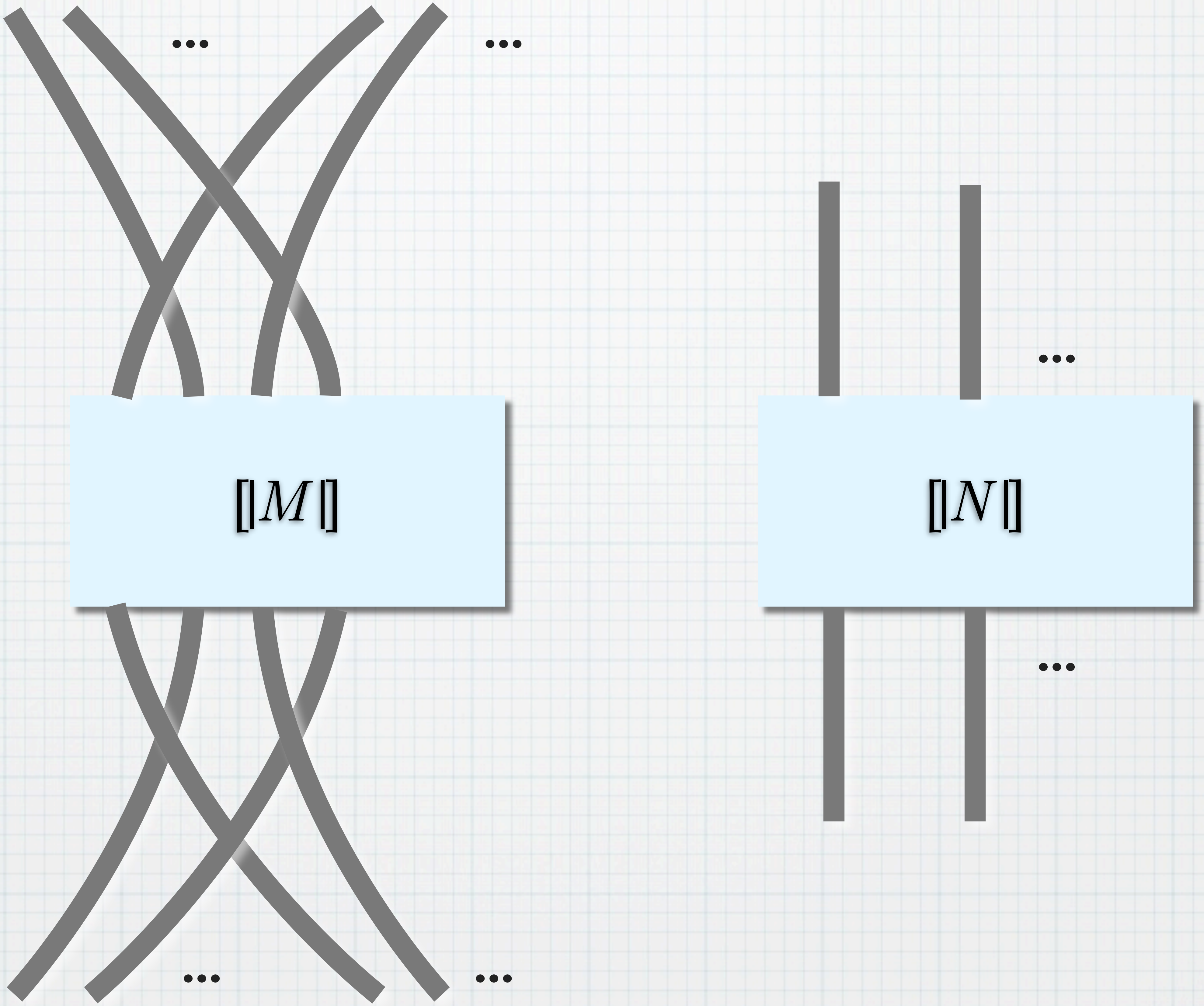
$$[\![\boldsymbol{M}\boldsymbol{N}]\!] =$$



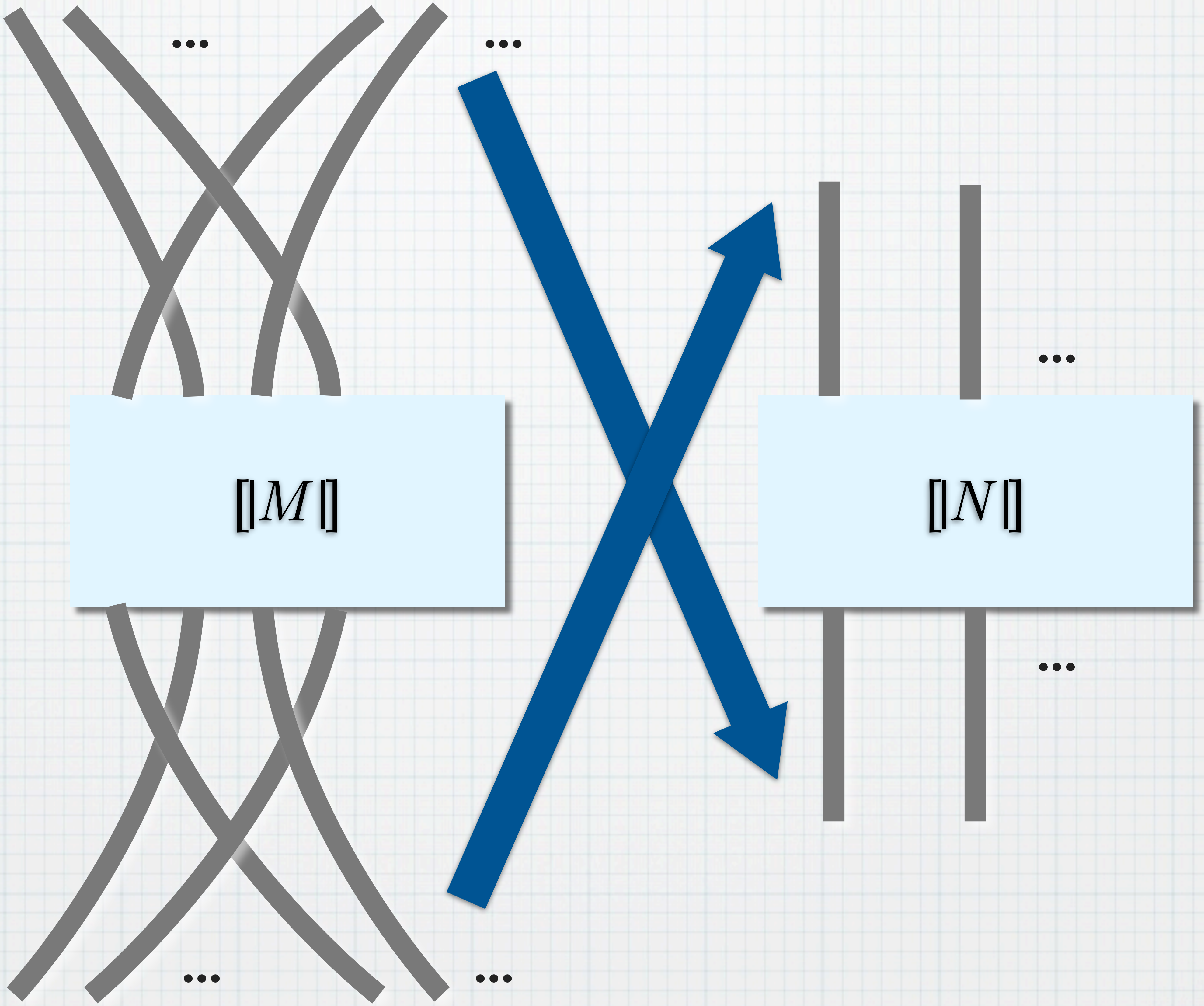
$$[\![\boldsymbol{M}\,\boldsymbol{N}]\!] =$$



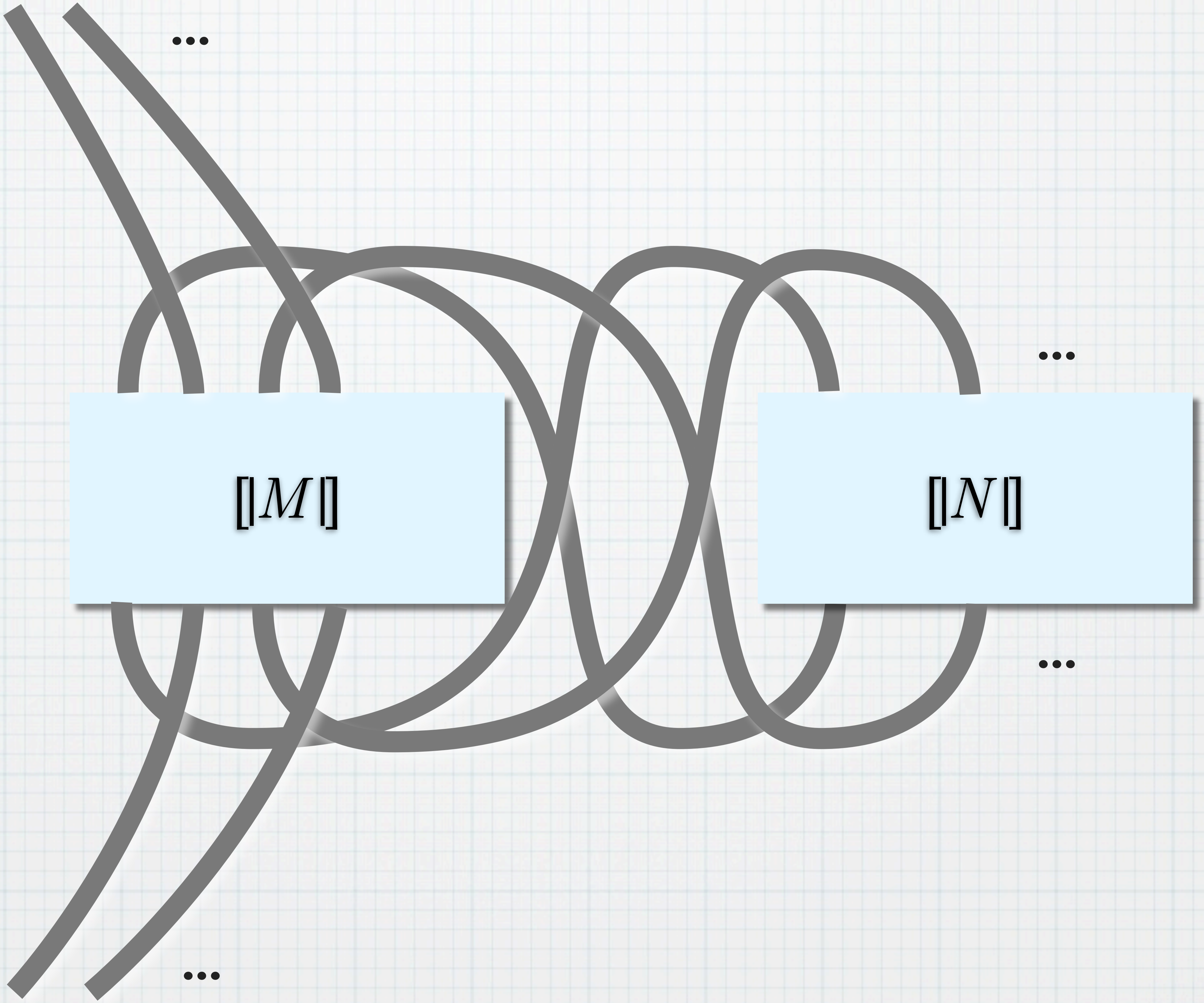
$$[\![\boldsymbol{M}\boldsymbol{N}]\!] =$$



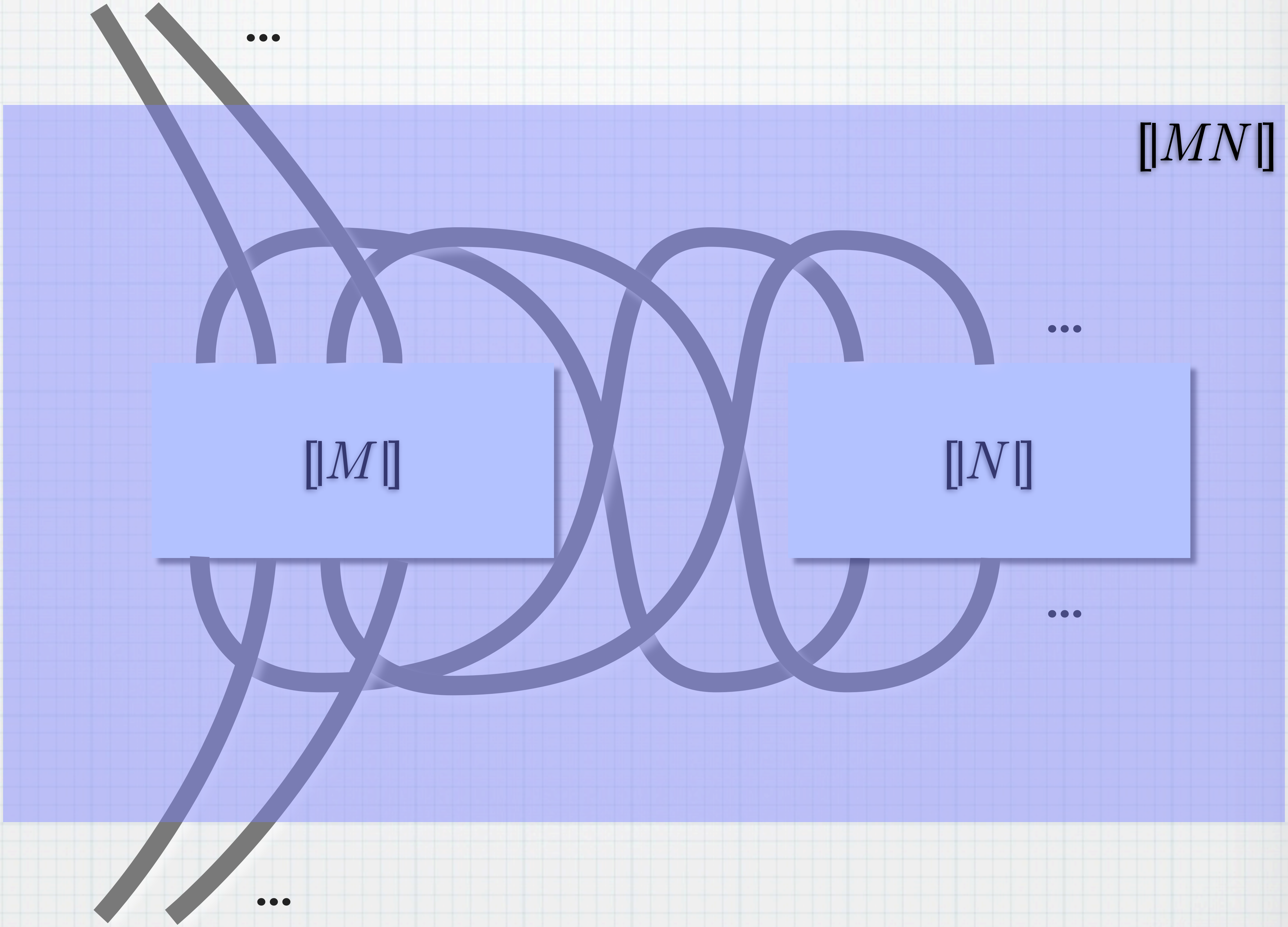
$$[\![MN]\!] =$$



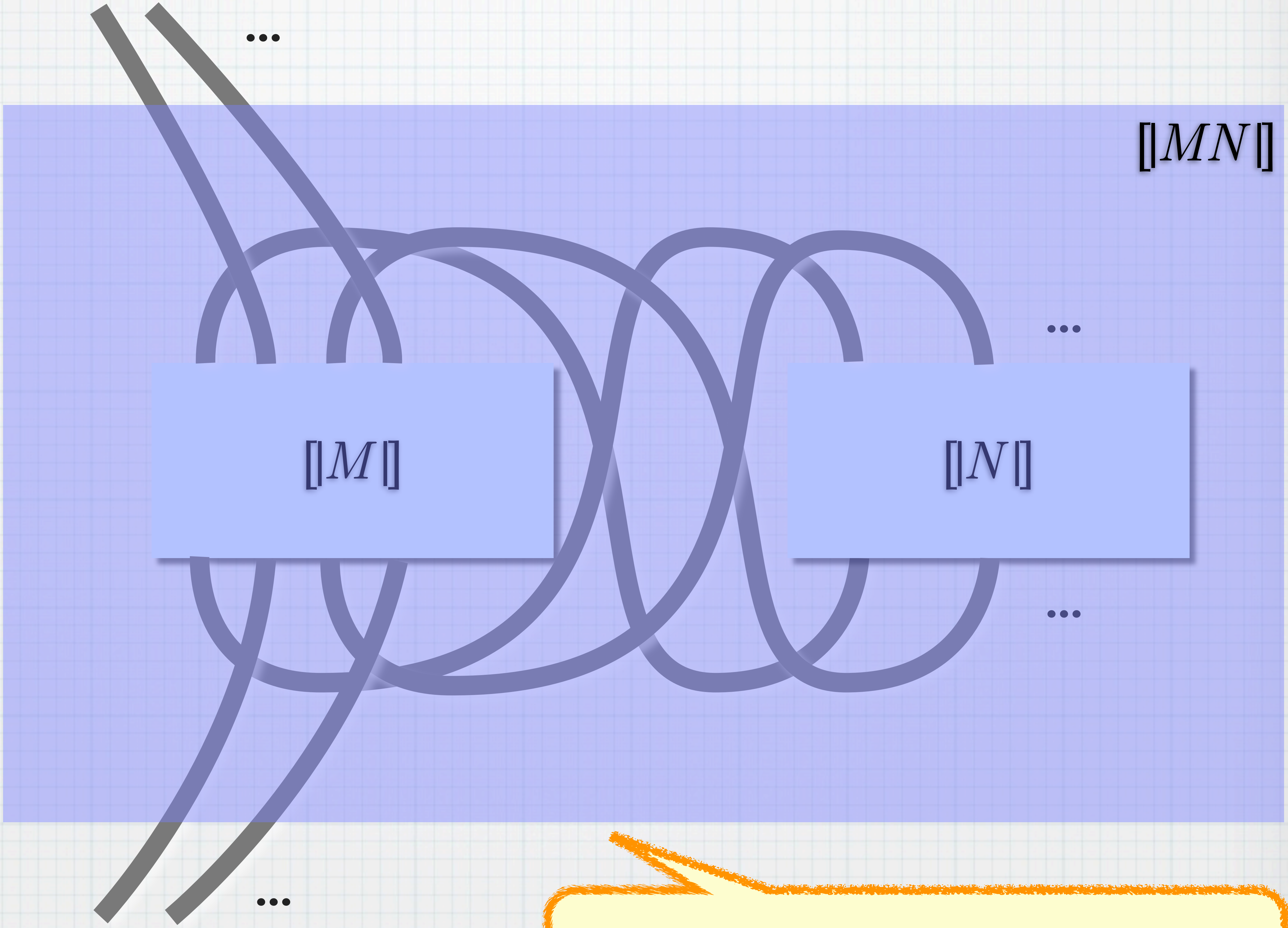
$$[\![\boldsymbol{M} \boldsymbol{N}]\!] =$$



$$[\![\boldsymbol{M}\boldsymbol{N}]\!] =$$

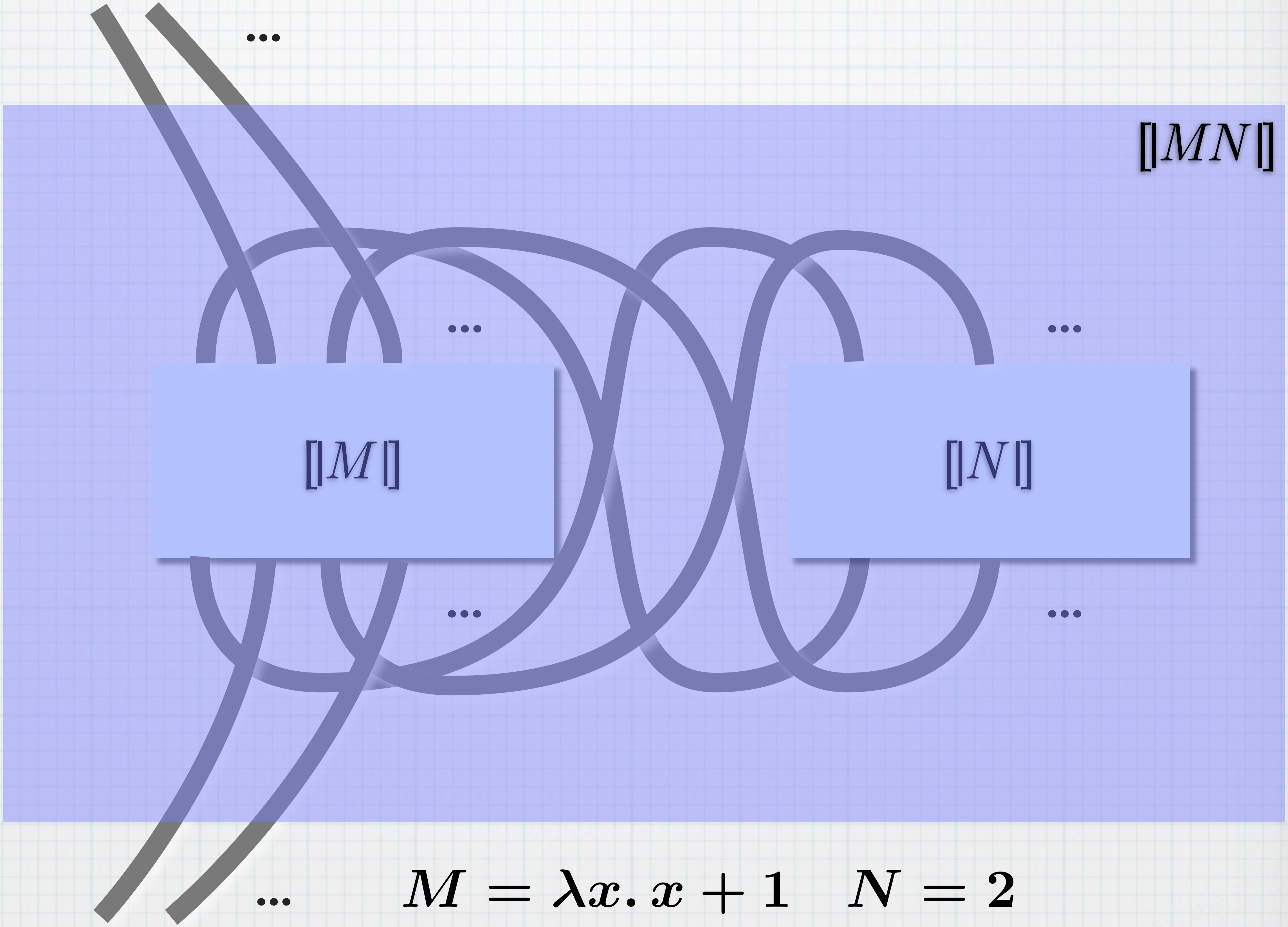


$$[\![MN]\!] =$$



"parallel composition + hiding"
(cf. AJM games)

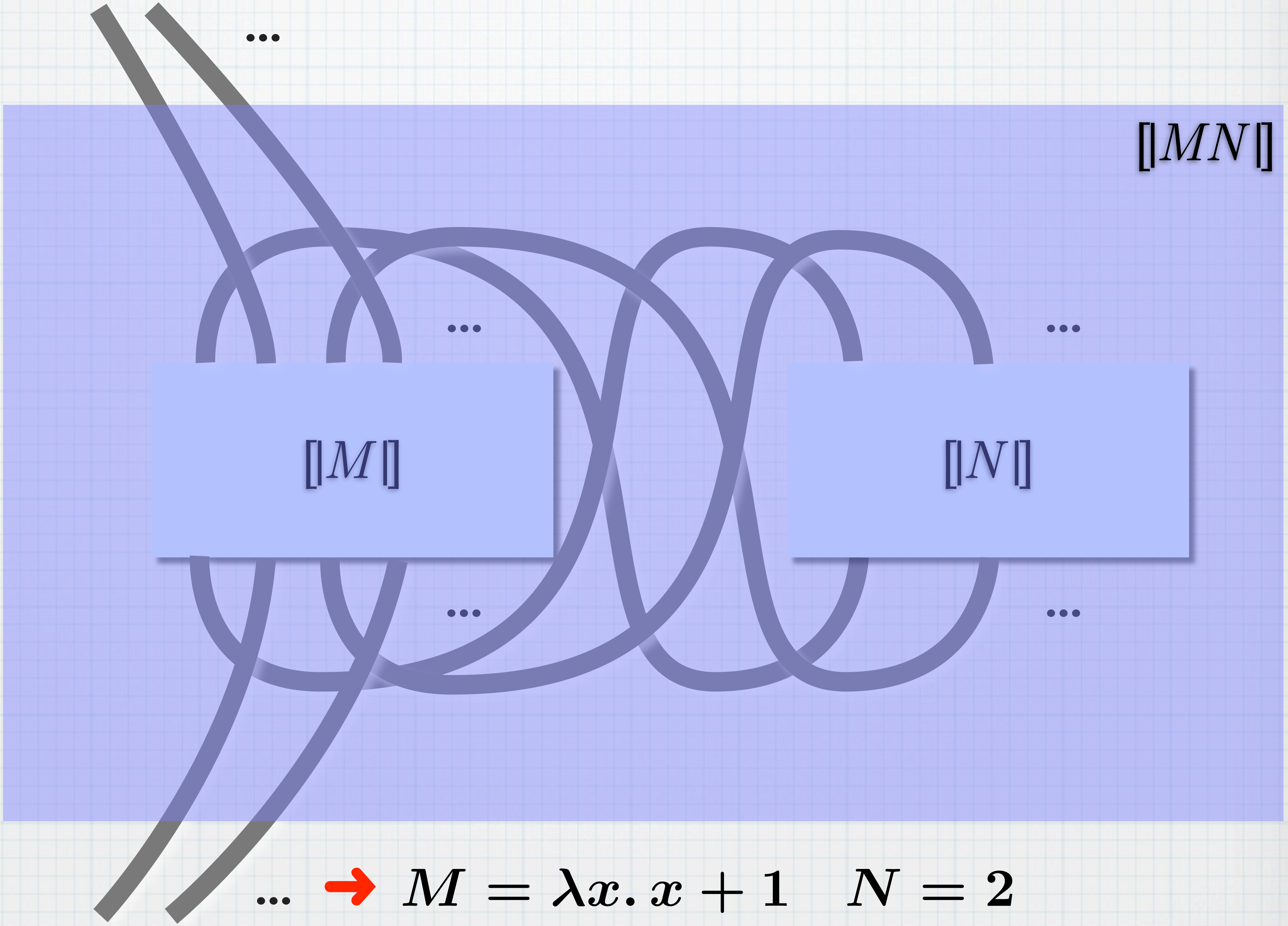
$$\llbracket \boldsymbol{MN} \rrbracket =$$



$$\boldsymbol{M = \lambda x.\, x + 1 \quad N = 2}$$
$$\boldsymbol{M = \lambda x.\, 1 \quad N = 2}$$
$$\boldsymbol{M = \lambda f.\, f 1 \quad N = \lambda x.\, (x + 1)}$$

$[\![MN]\!]$

$=$

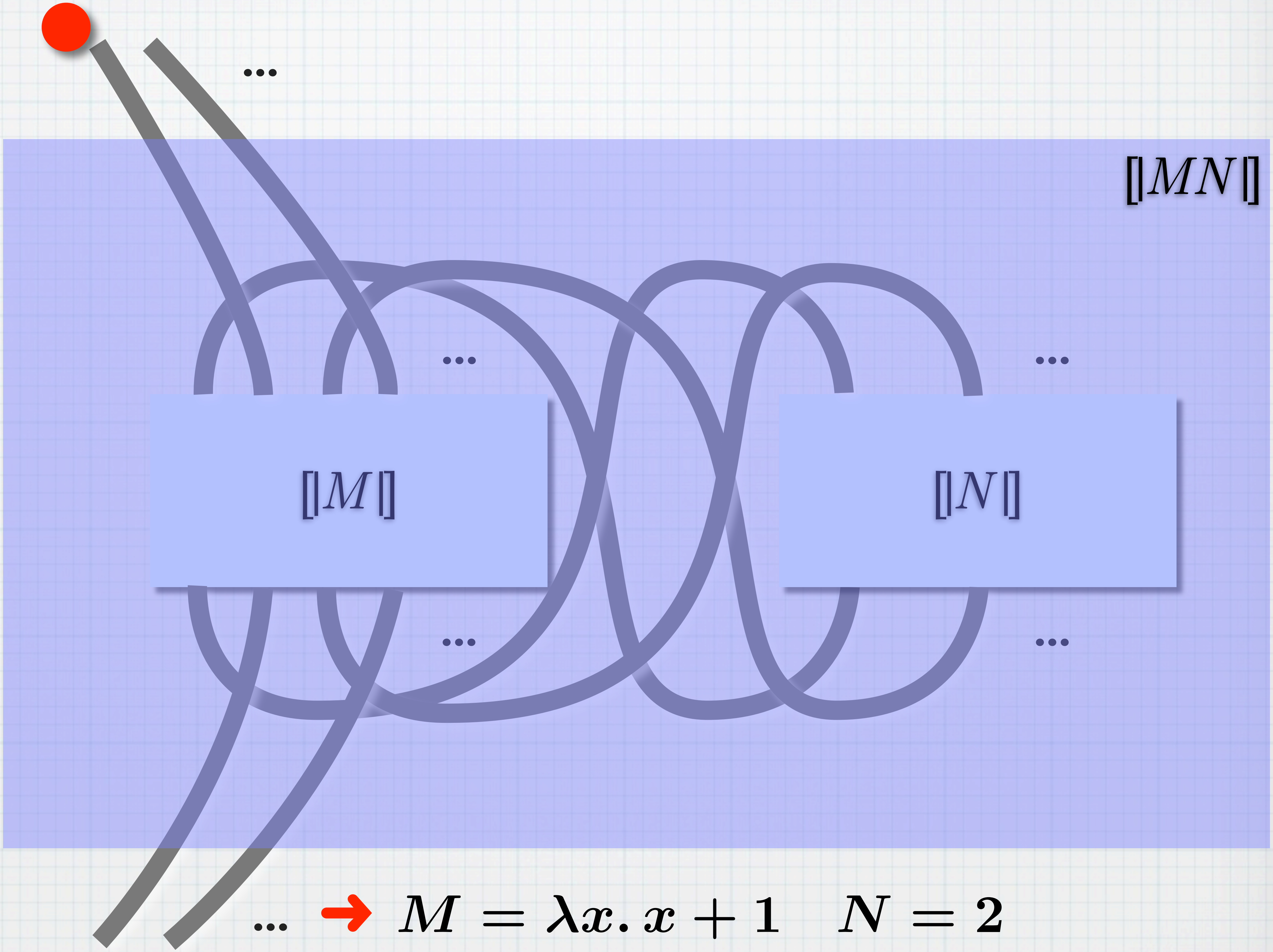


➜ $M = \lambda x.\, x + 1 \quad N = 2$

$M = \lambda x.\, 1 \quad N = 2$

$M = \lambda f.\, f 1 \quad N = \lambda x.\, (x + 1)$

$[\![MN]\!] =$

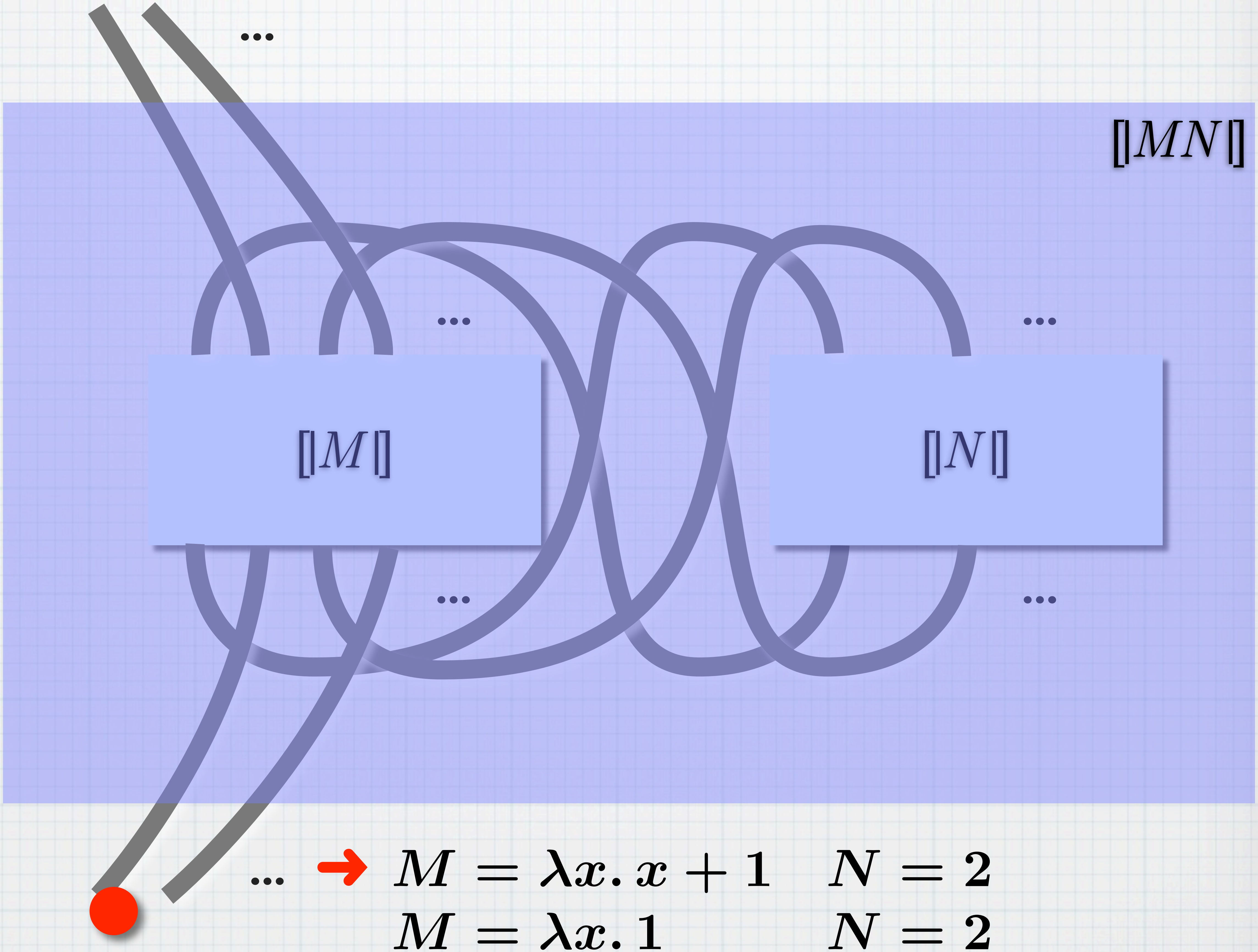


… ➜ $M = \lambda x.\, x + 1 \qquad N = 2$

$M = \lambda x.\, 1 \qquad N = 2$

$M = \lambda f.\, f 1 \qquad N = \lambda x.\, (x + 1)$

$$[\![MN]\!] =$$

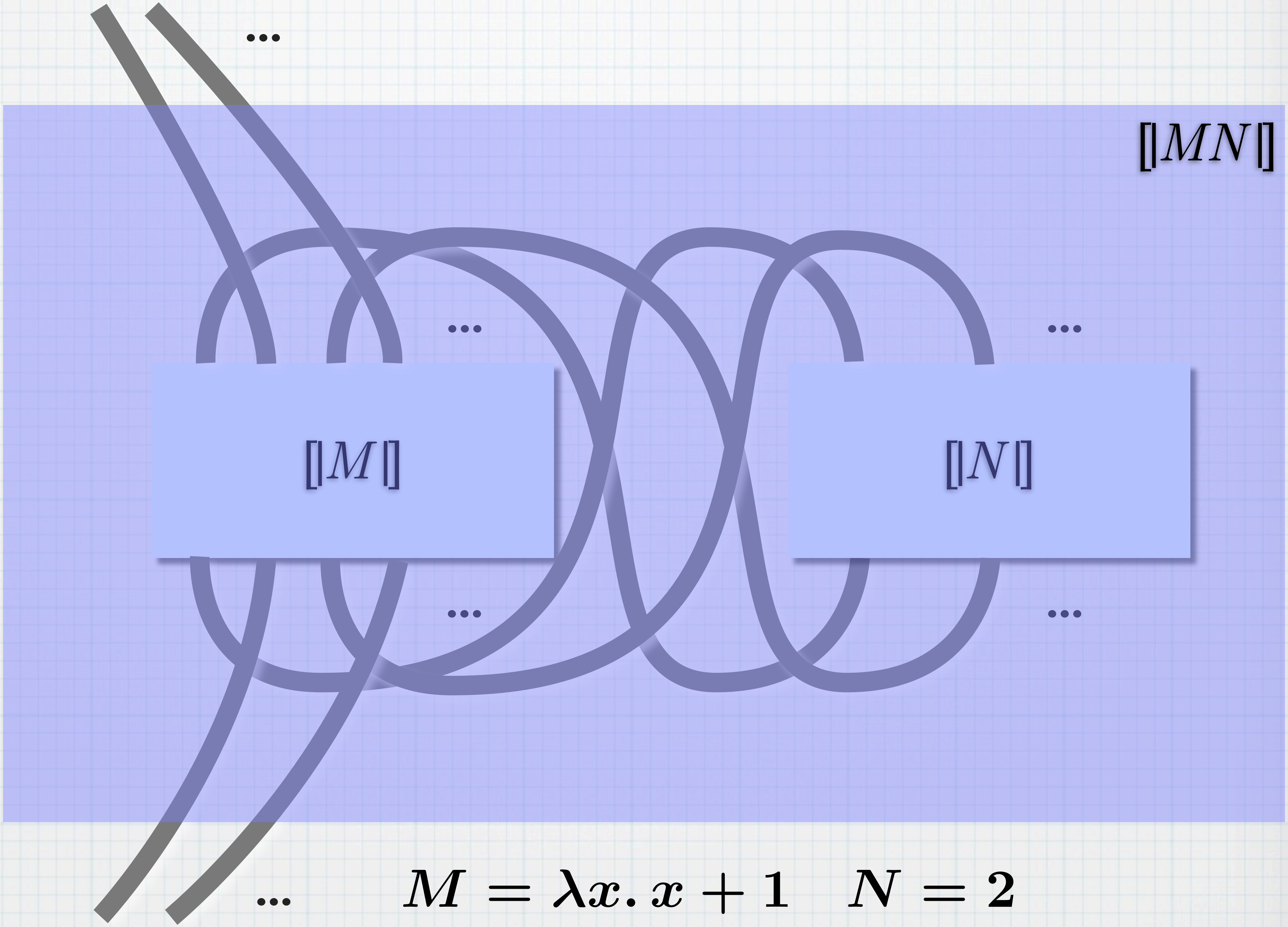


➜ $M = \lambda x.\, x + 1 \qquad N = 2$

$M = \lambda x.\, 1 \qquad N = 2$

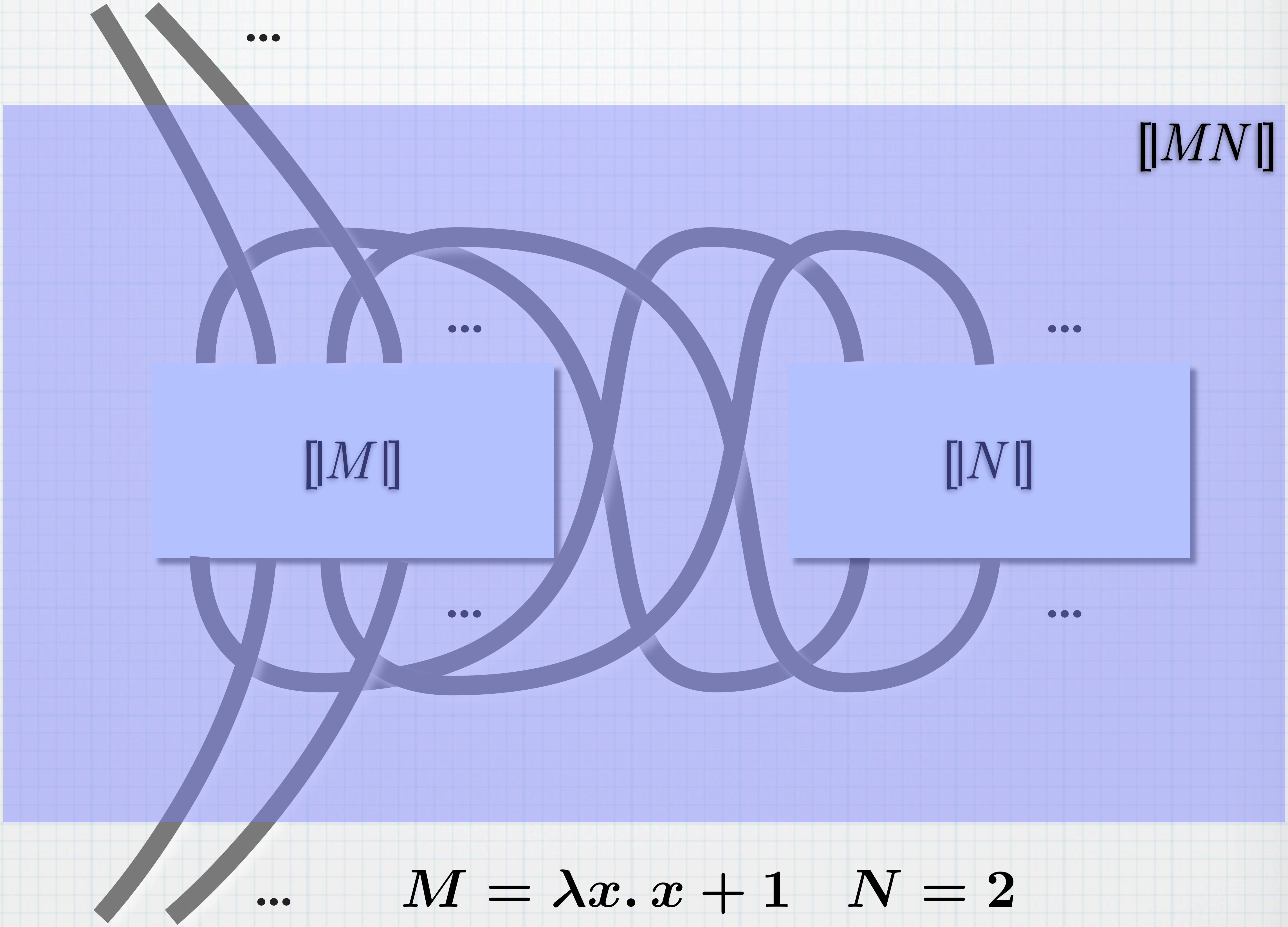
$M = \lambda f.\, f 1 \qquad N = \lambda x.\, (x + 1)$

$$[\![\boldsymbol{M N}]\!] =$$



$$\boldsymbol{M = \lambda x.\, x + 1 \quad N = 2}$$
$$\boldsymbol{M = \lambda x.\, 1 \quad N = 2}$$
$$\boldsymbol{M = \lambda f.\, f 1 \quad N = \lambda x.\, (x + 1)}$$

$$[\![MN]\!] =$$

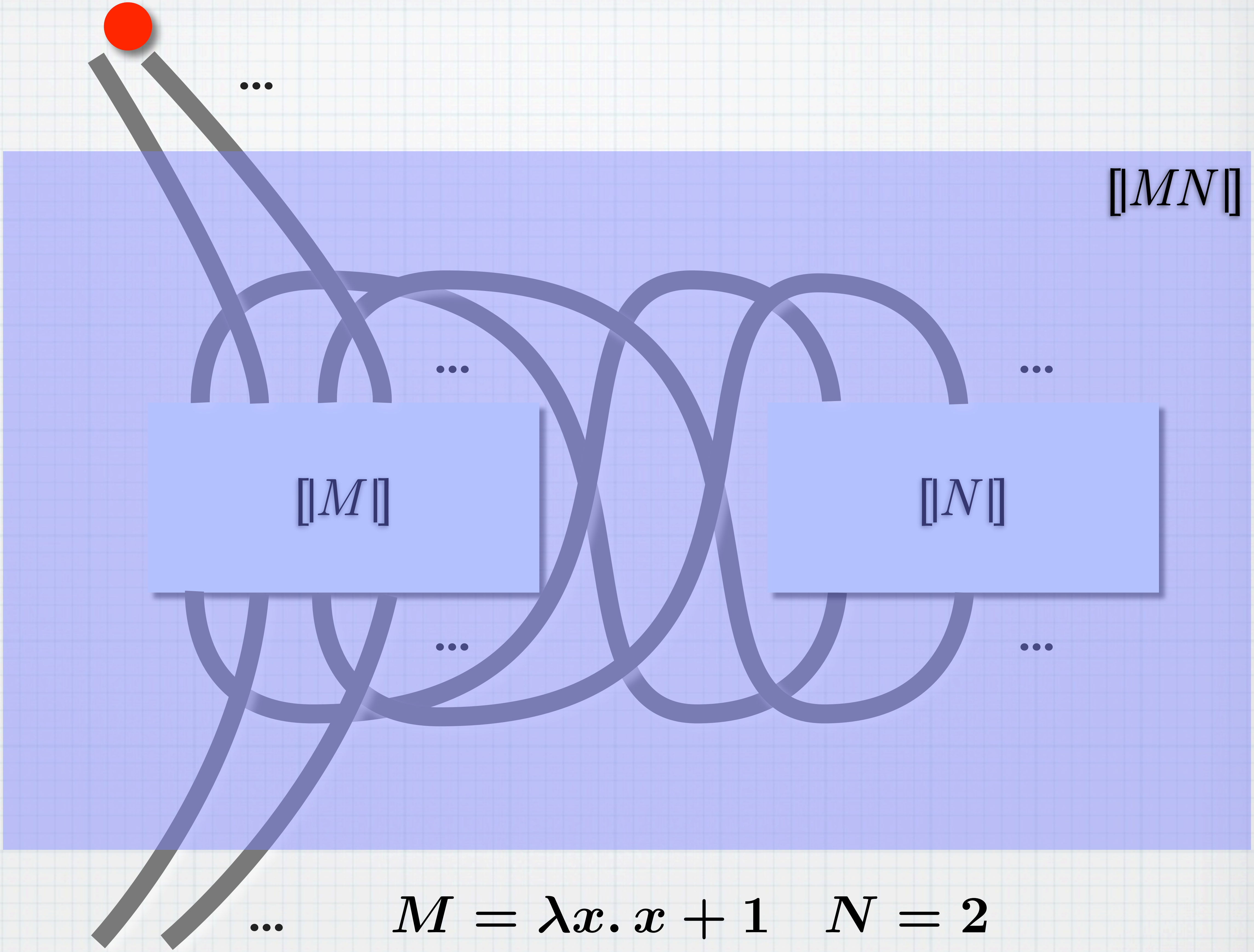


$$M = \lambda x.\, x + 1 \qquad N = 2$$

$$\rightarrow \; M = \lambda x.\, 1 \qquad N = 2$$

$$M = \lambda f.\, f 1 \qquad N = \lambda x.\, (x + 1)$$

$$[\![MN]\!] =$$



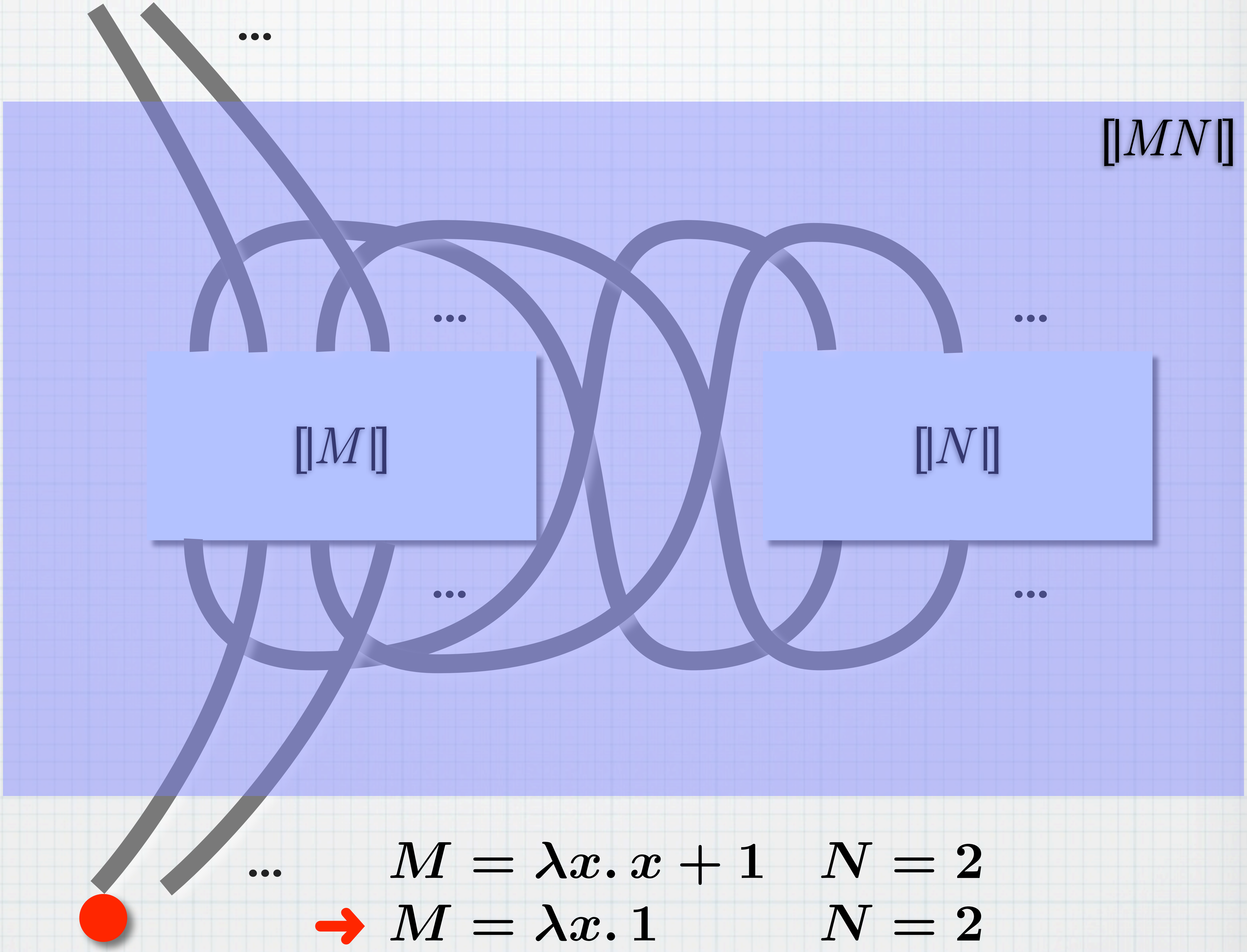
$\boldsymbol{M = \lambda x.\, x + 1} \qquad \boldsymbol{N = 2}$

➜ $\boldsymbol{M = \lambda x.\, 1} \qquad \boldsymbol{N = 2}$

$\boldsymbol{M = \lambda f.\, f 1} \qquad \boldsymbol{N = \lambda x.\, (x + 1)}$

$[\![MN]\!]$

$=$

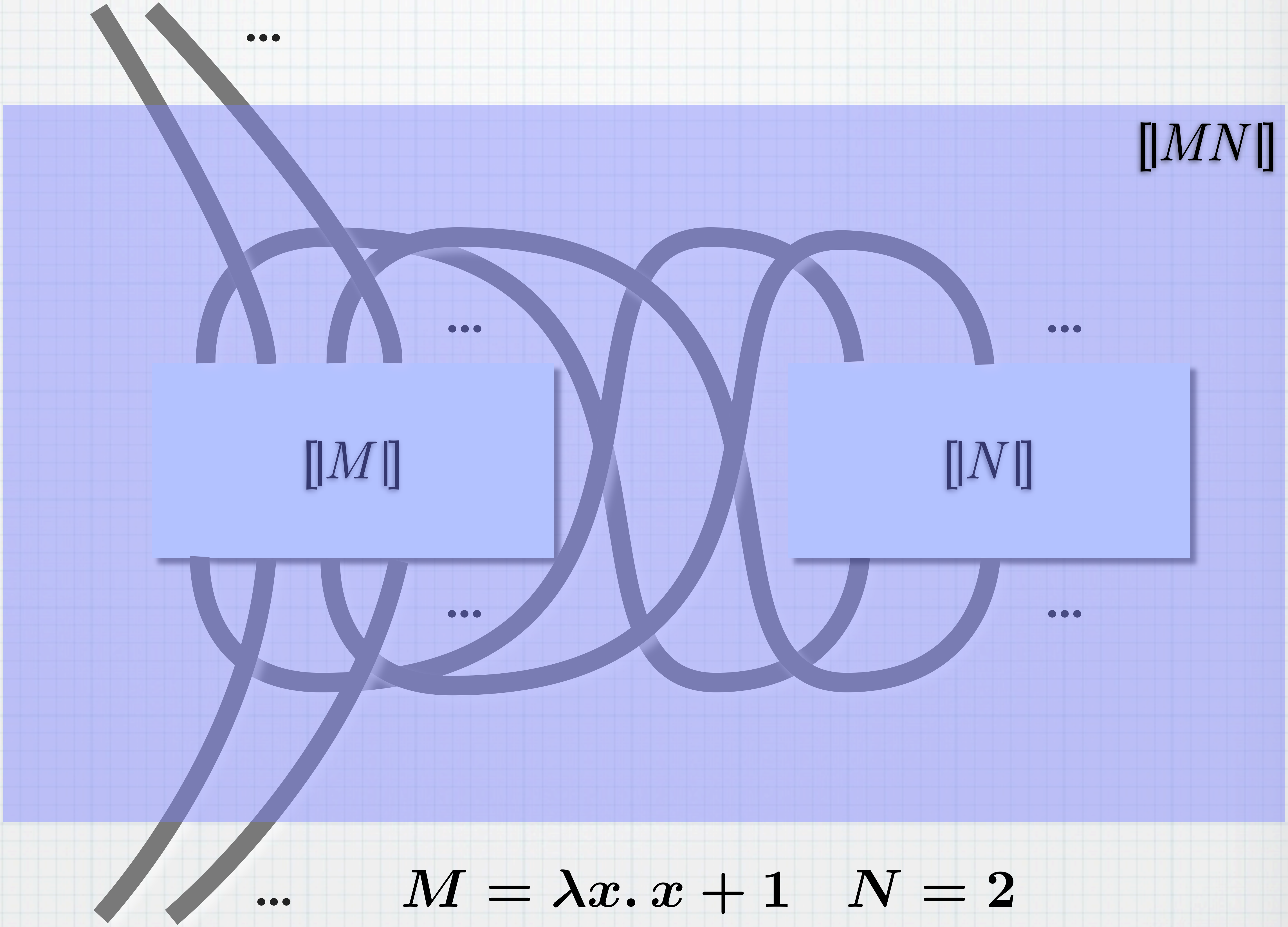


$M = \lambda x.\, x+1 \quad N = 2$

→ $M = \lambda x.\, 1 \quad N = 2$

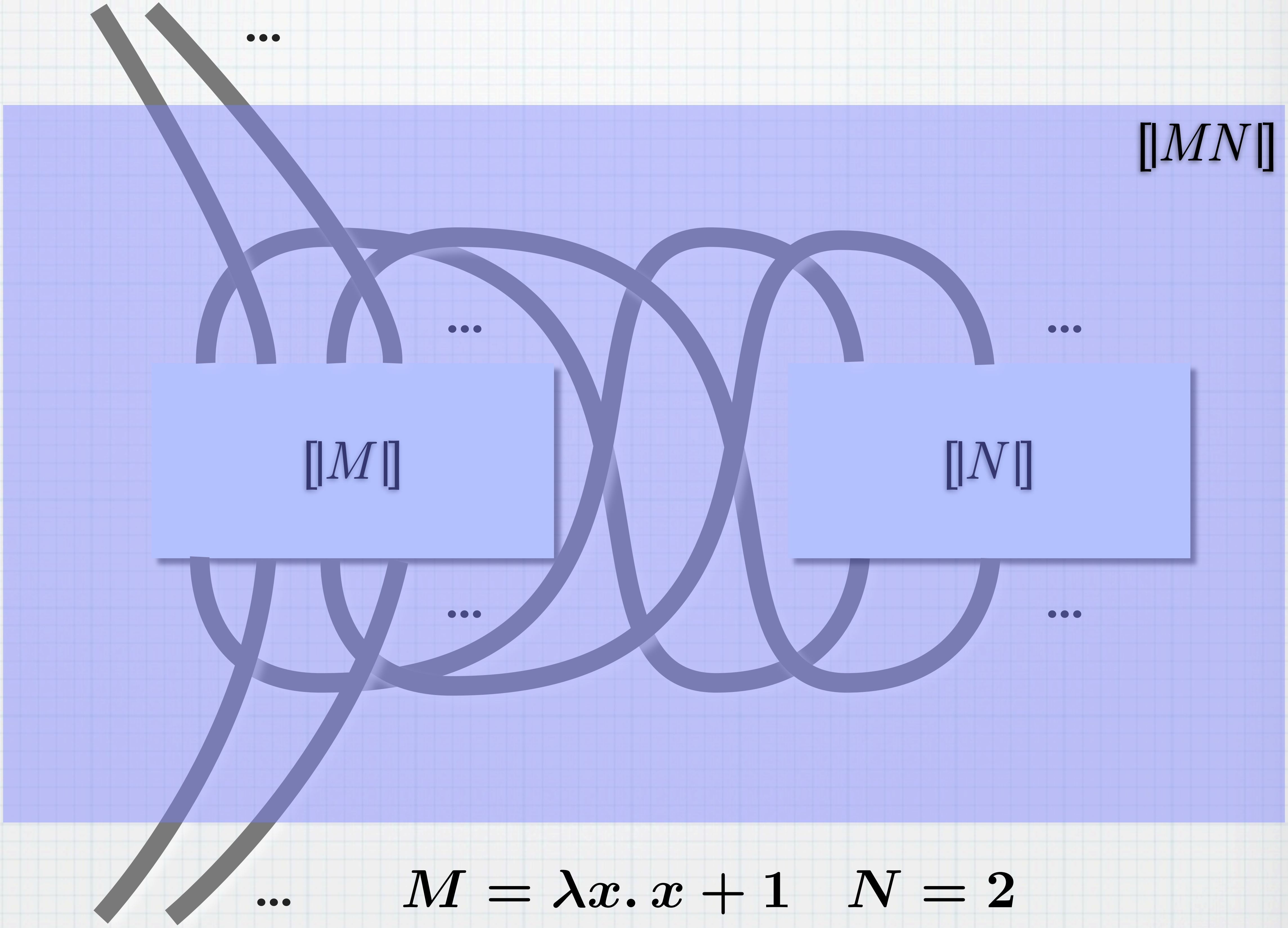
$M = \lambda f.\, f 1 \quad N = \lambda x.\, (x+1)$

$$\llbracket \boldsymbol{M N} \rrbracket =$$



$$\begin{array}{ll} \boldsymbol{M = \lambda x.\, x + 1} & \boldsymbol{N = 2} \\ \boldsymbol{M = \lambda x.\, 1} & \boldsymbol{N = 2} \\ \boldsymbol{M = \lambda f.\, f 1} & \boldsymbol{N = \lambda x.\, (x + 1)} \end{array}$$

$[\![MN]\!] =$

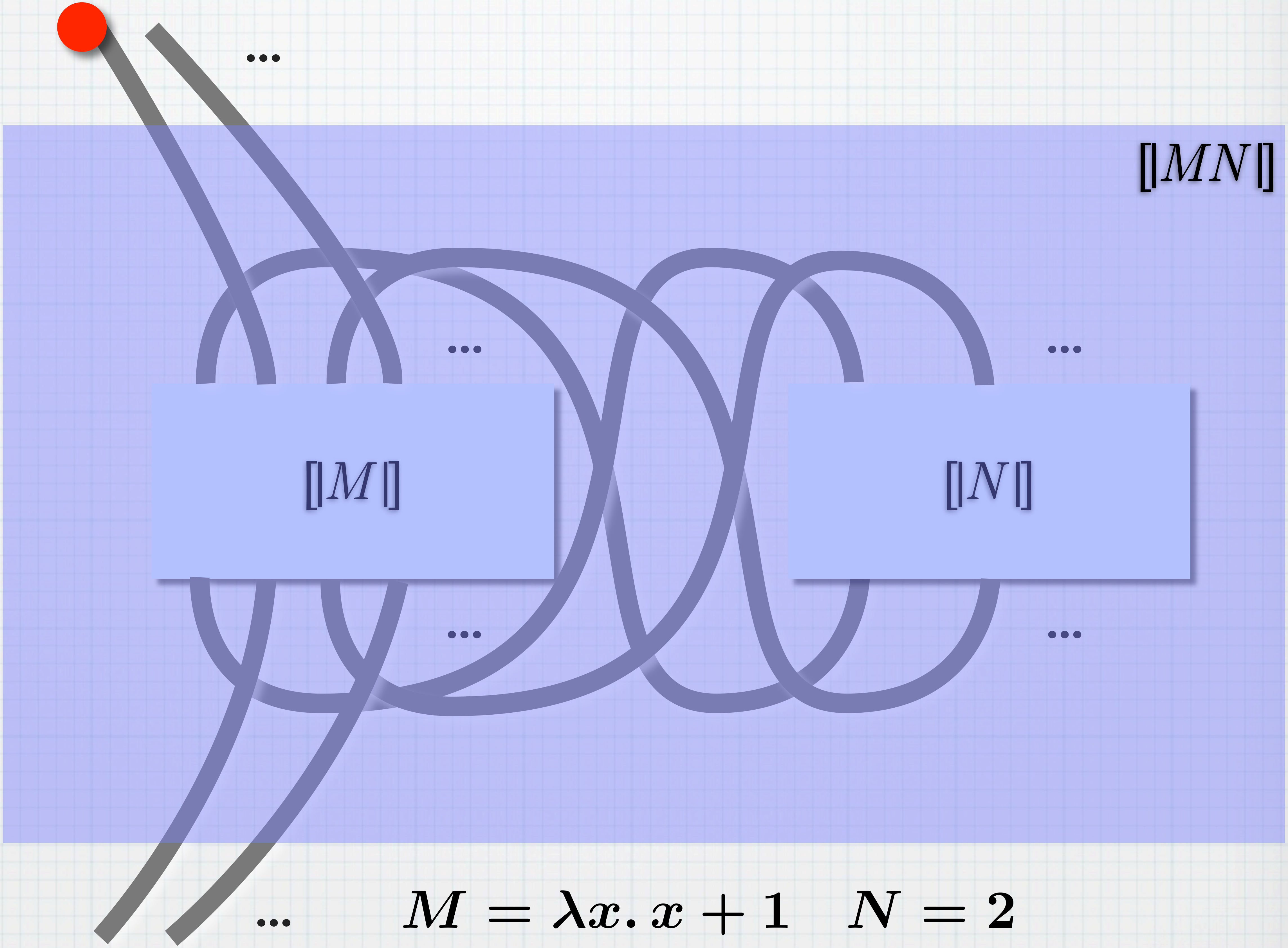


$M = \lambda x.\, x + 1 \qquad N = 2$

$M = \lambda x.\, 1 \qquad N = 2$

➜ $M = \lambda f.\, f 1 \qquad N = \lambda x.\, (x + 1)$

$$[\![MN]\!] =$$

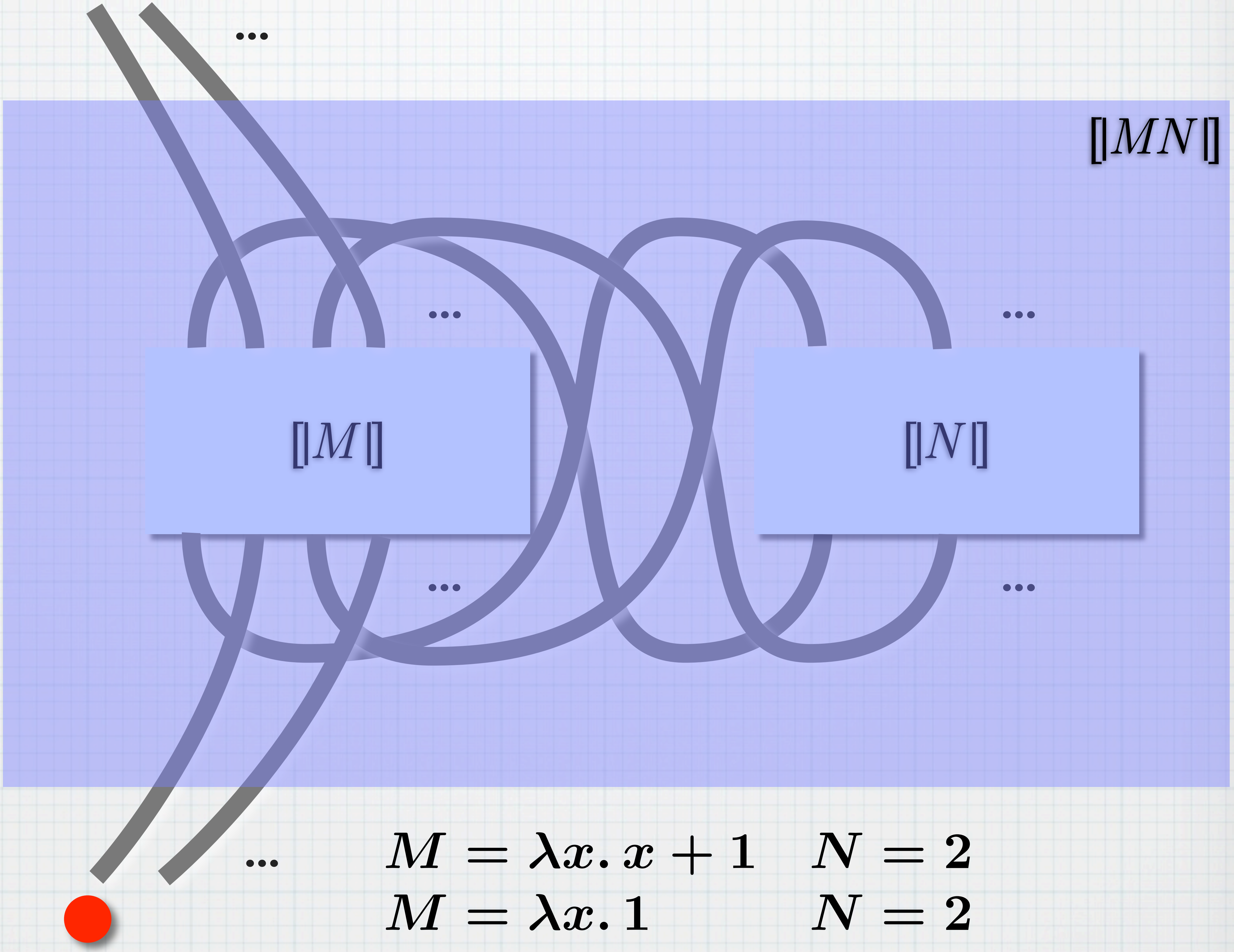


$$M = \lambda x.\, x + 1 \qquad N = 2$$

$$M = \lambda x.\, 1 \qquad N = 2$$

➜ $$M = \lambda f.\, f 1 \qquad N = \lambda x.\, (x + 1)$$

$[\![MN]\!] =$

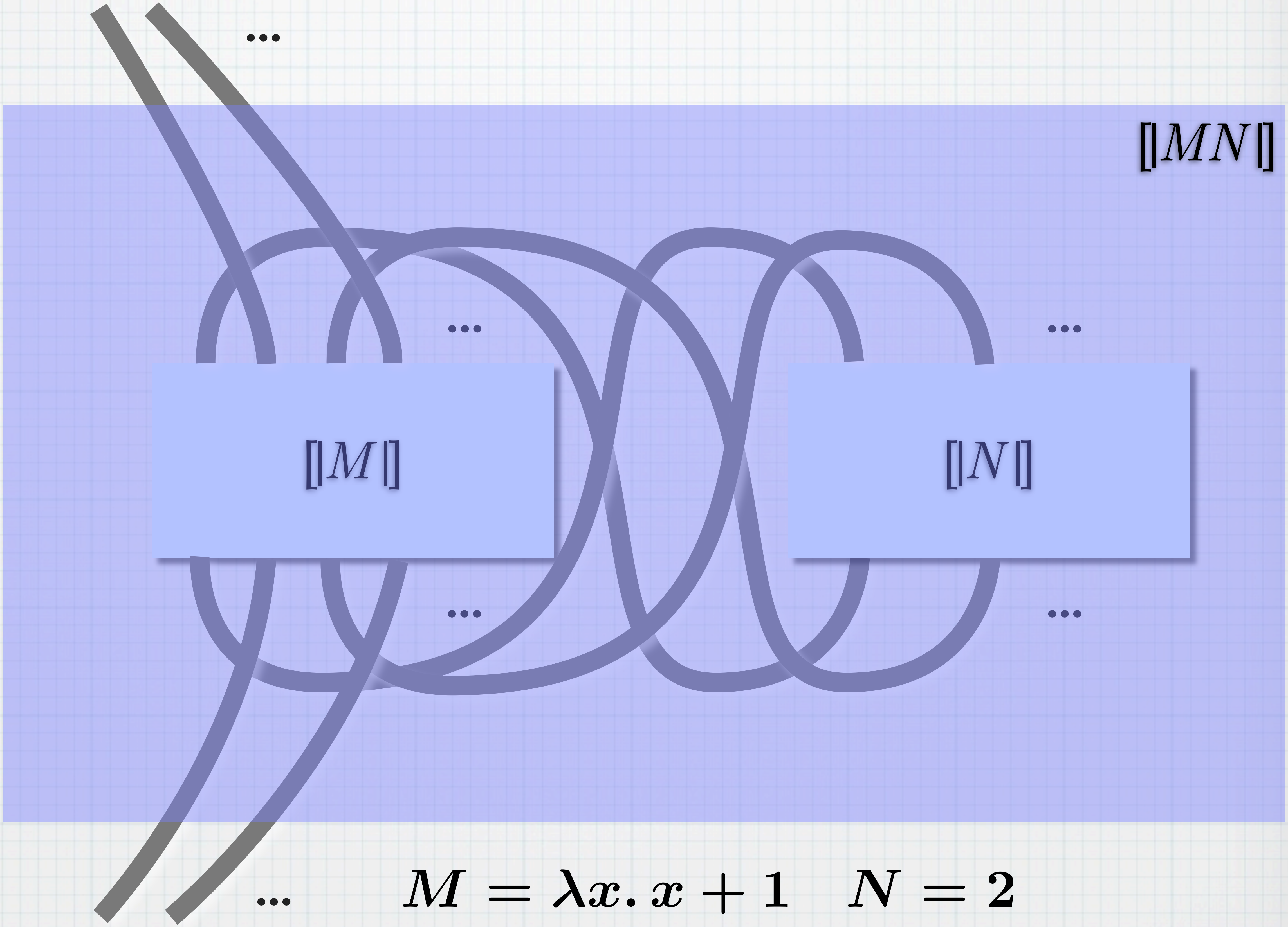


$M = \lambda x.\, x + 1 \quad N = 2$

$M = \lambda x.\, 1 \quad N = 2$

→ $M = \lambda f.\, f 1 \quad N = \lambda x.\, (x + 1)$

$$[\![M N]\!] =$$

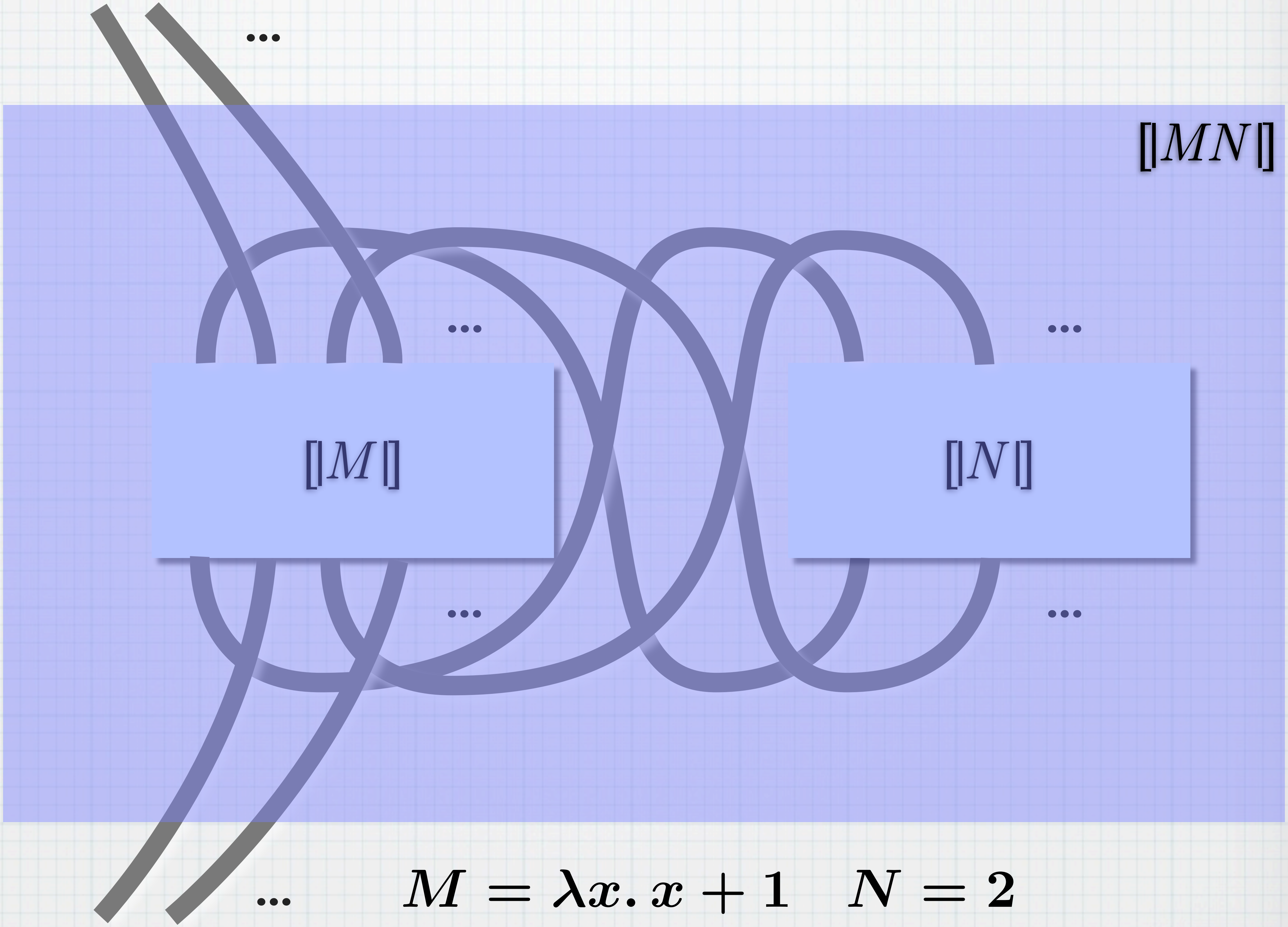


$M = \lambda x.\, x + 1 \quad N = 2$

$M = \lambda x.\, 1 \quad N = 2$

→ $M = \lambda f.\, f 1 \quad N = \lambda x.\, (x + 1)$

$$[\![\boldsymbol{M N}]\!] \\ =$$



$$\begin{array}{ll} \boldsymbol{M = \lambda x.\, x + 1} & \boldsymbol{N = 2} \\ \boldsymbol{M = \lambda x.\, 1} & \boldsymbol{N = 2} \\ \boldsymbol{M = \lambda f.\, f 1} & \boldsymbol{N = \lambda x.\, (x + 1)} \end{array}$$

GoI:
Geometry of Interaction

- J.-Y. Girard, at Logic Colloquium '88

GoI:
Geometry of Interaction

- J.-Y. Girard, at Logic Colloquium '88
- Provides denotational semantics $[\![M]\!]$ for linear λ-term M

GoI:
Geometry of Interaction

- J.-Y. Girard, at Logic Colloquium '88
- Provides denotational semantics $[\![M]\!]$ for linear λ-term M
  - Similar to **game semantics** [AJM/HO]

GoI:
Geometry of Interaction

- J.-Y. Girard, at Logic Colloquium '88
- Provides denotational semantics $[\![M]\!]$ for linear λ-term M
  - Similar to **game semantics** [AJM/HO]
  - Linearity: simplicity; no-cloning

GoI:
Geometry of Interaction

- J.-Y. Girard, at Logic Colloquium '88
- Provides denotational semantics $[\![M]\!]$ for linear λ-term M
  - Similar to **game semantics** [AJM/HO]
  - Linearity: simplicity; no-cloning
  - Girard translation

$A \to B$

as $!A \multimap B$

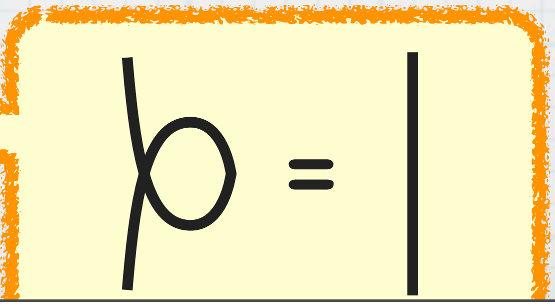
GoI: Geometry of Interaction

- J.-Y. Girard, at Logic Colloquium '88
- Provides denotational semantics $[\![M]\!]$ for linear λ-term M
  - Similar to **game semantics** [AJM/HO]
  - Linearity: simplicity; no-cloning
  - Girard translation

$A \to B$

as $!A \multimap B$

- "Geometry": invariant under β-reductions



(Tokyo)

Categorical GoI

- Axiomatics of GoI in the categorical language
- Our main reference:
  - [AHS02] S. Abramsky, E. Haghverdi, and P. Scott, "Geometry of interaction and linear combinatory algebras," MSCS 2002
  - Especially its technical report version (Oxford CL), since it's a bit more detailed

The Categorical GoI Workflow

Traced monoidal category $\mathbb{C}$

+ other constructs ➜ "GoI situation" [AHS02]

Categorical GoI [AHS02]

Linear combinatory algebra

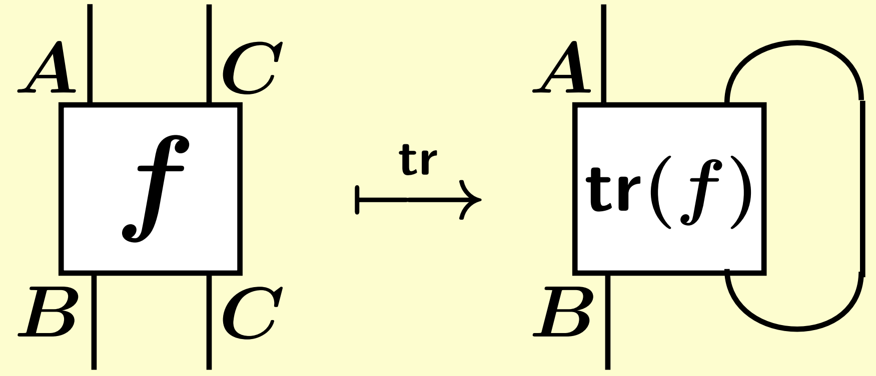
Realizability

Linear category

The Categorical GoI Workflow

Traced monoidal category $\mathbb{C}$

\+ other constructs ➜ "GoI situation" [AHS02]



Categorical GoI [AHS02]

Linear combinatory algebra

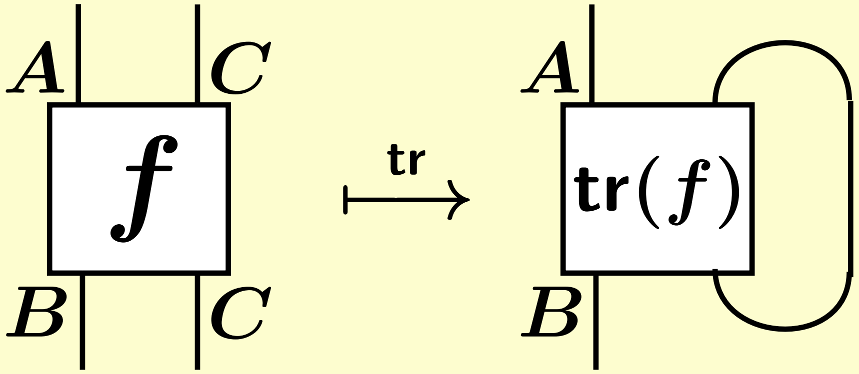
Realizability

Linear category

The Categorical GoI Workflow

Traced monoidal category $\mathbb{C}$

+ other constructs ➜ "GoI situation" [AHS02]



Categorical GoI [AHS02]

Linear combinatory algebra

- ✱ Applicative str. + combinators
- ✱ Model of untyped calculus

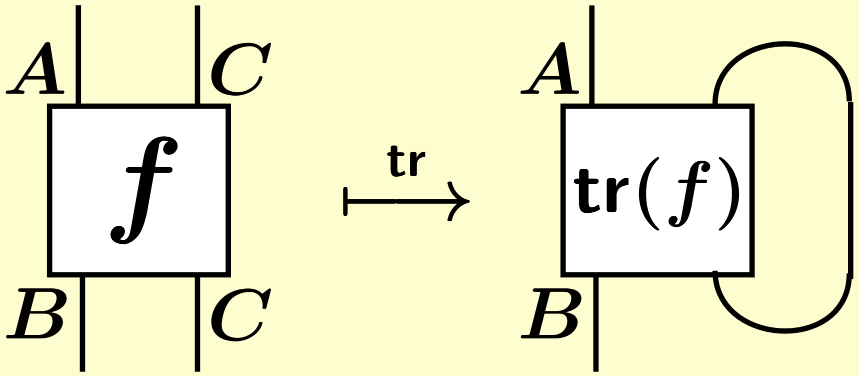
Realizability

Linear category

The Categorical GoI Workflow

Traced monoidal category $\mathbb{C}$

+ other constructs ➜ "GoI situation" [AHS02]



Categorical GoI [AHS02]

Linear combinatory algebra

- Applicative str. + combinators
- Model of untyped calculus

Realizability

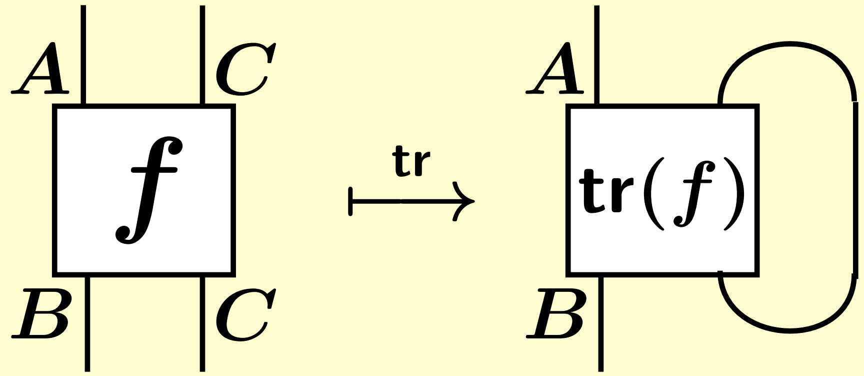
Linear category

Model of typed calculus

The Categorical GoI Workflow

Traced monoidal category $\mathbb{C}$

+ other constructs ➜ "GoI situation" [AHS02]



Categorical GoI [AHS02]

- Applicative str. + combinators
- Model of untyped calculus

Linear combinatory algebra

Realizability

- PER, ω-set, assembly, ...
- "Programming in untyped λ"

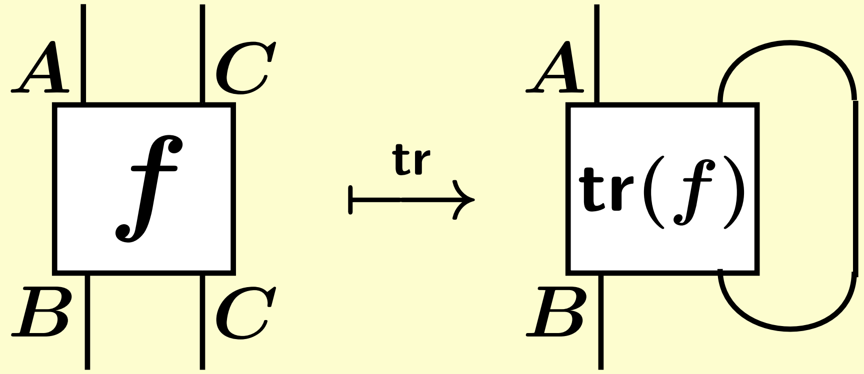
Linear category

Model of typed calculus

The Categorical GoI Workflow

Traced monoidal category $\mathbb{C}$

+ other constructs ➜ "GoI situation" [AHS02]



Categorical GoI [AHS02]

- ✱ Applicative str. + combinators
- ✱ Model of untyped calculus

Linear combinatory algebra

Realizability

- ✱ PER, ω-set, assembly, ...
- ✱ "Programming in untyped λ"

Linear category

Model of typed calculus

Linear Combinatory Algebra (LCA)

Defn. (LCA)
A *linear combinatory algebra (LCA)* is a set $\boldsymbol{A}$ equipped with

- a binary operator (called an *applicative structure*)

$$\cdot \;:\; \boldsymbol{A}^2 \longrightarrow \boldsymbol{A}$$

- a unary operator

$$! \;:\; \boldsymbol{A} \longrightarrow \boldsymbol{A}$$

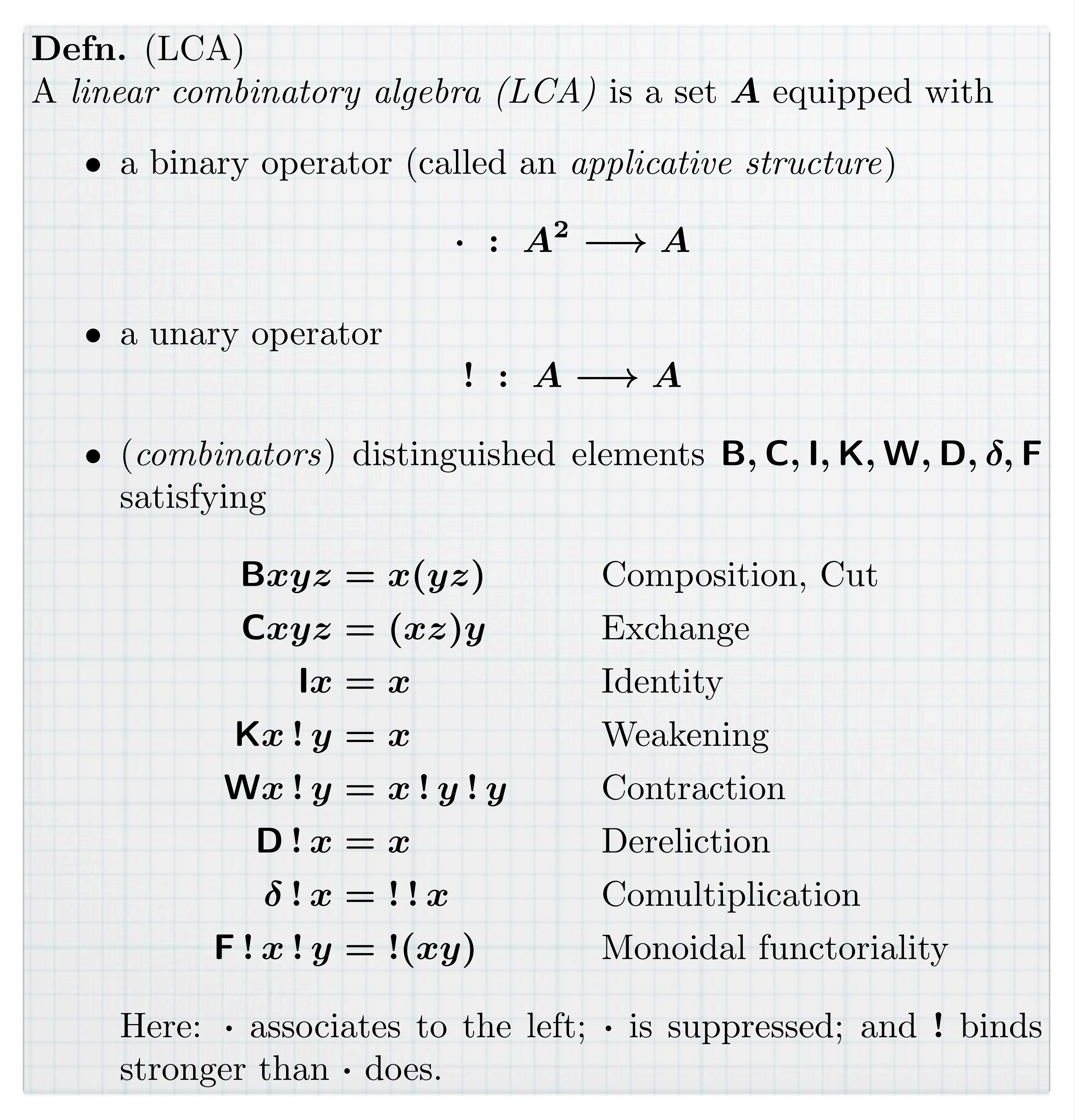
- (*combinators*) distinguished elements $\mathsf{B}, \mathsf{C}, \mathsf{I}, \mathsf{K}, \mathsf{W}, \mathsf{D}, \delta, \mathsf{F}$ satisfying

$\mathsf{B}xyz = x(yz)$	Composition, Cut
$\mathsf{C}xyz = (xz)y$	Exchange
$\mathsf{I}x = x$	Identity
$\mathsf{K}x\,!\,y = x$	Weakening
$\mathsf{W}x\,!\,y = x\,!\,y\,!\,y$	Contraction
$\mathsf{D}\,!\,x = x$	Dereliction
$\delta\,!\,x = !\,!\,x$	Comultiplication
$\mathsf{F}\,!\,x\,!\,y = !(xy)$	Monoidal functoriality

Here: $\cdot$ associates to the left; $\cdot$ is suppressed; and $!$ binds stronger than $\cdot$ does.

Linear Combinatory Algebra (LCA)

What
we want (outcome)



Defn. (LCA)
A *linear combinatory algebra (LCA)* is a set $\boldsymbol{A}$ equipped with

- a binary operator (called an *applicative structure*)

$$\cdot \; : \; \boldsymbol{A}^2 \longrightarrow \boldsymbol{A}$$

- a unary operator

$$! \; : \; \boldsymbol{A} \longrightarrow \boldsymbol{A}$$

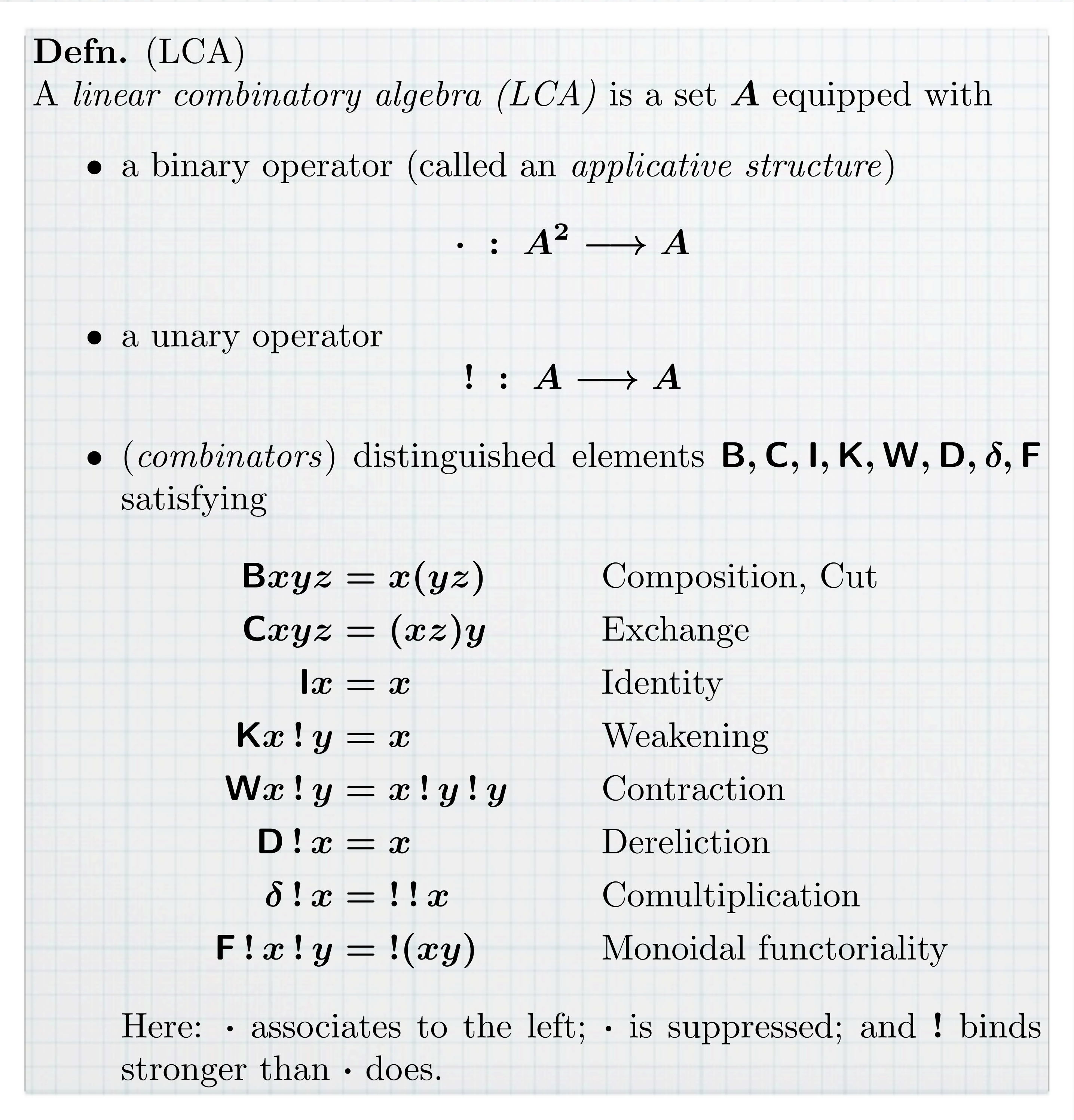
- (*combinators*) distinguished elements $\mathsf{B}, \mathsf{C}, \mathsf{I}, \mathsf{K}, \mathsf{W}, \mathsf{D}, \delta, \mathsf{F}$ satisfying

$\mathsf{B}xyz = x(yz)$	Composition, Cut
$\mathsf{C}xyz = (xz)y$	Exchange
$\mathsf{I}x = x$	Identity
$\mathsf{K}x\,!\,y = x$	Weakening
$\mathsf{W}x\,!\,y = x\,!\,y\,!\,y$	Contraction
$\mathsf{D}\,!\,x = x$	Dereliction
$\delta\,!\,x = !\,!\,x$	Comultiplication
$\mathsf{F}\,!\,x\,!\,y = !(xy)$	Monoidal functoriality

Here: $\cdot$ associates to the left; $\cdot$ is suppressed; and ! binds stronger than $\cdot$ does.

Linear Combinatory Algebra (LCA)

What
we want (outcome)



Defn. (LCA)
A *linear combinatory algebra (LCA)* is a set $\boldsymbol{A}$ equipped with

- a binary operator (called an *applicative structure*)

$$\cdot \; : \; \boldsymbol{A}^2 \longrightarrow \boldsymbol{A}$$

- a unary operator

$$! \; : \; \boldsymbol{A} \longrightarrow \boldsymbol{A}$$

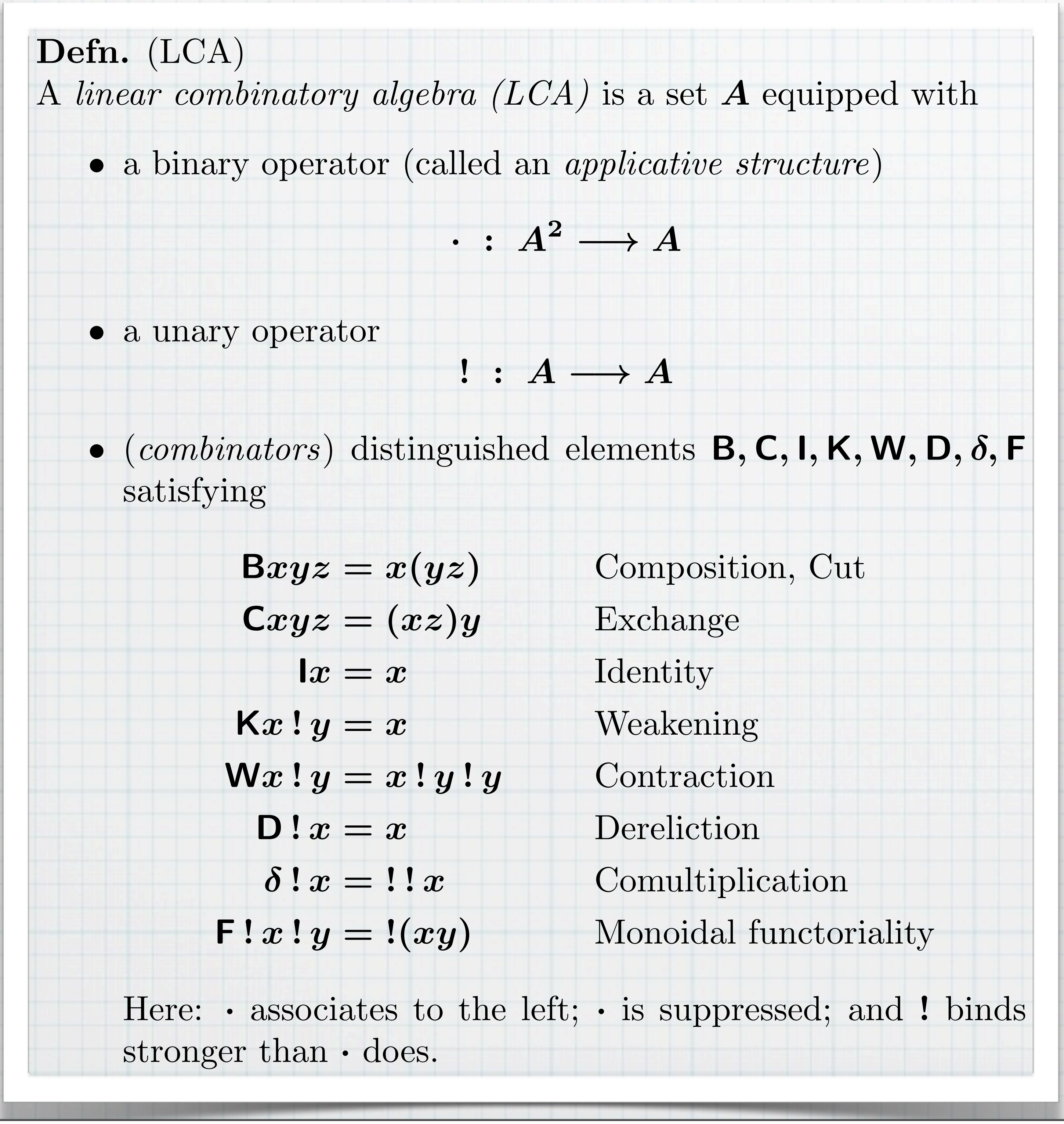
- (*combinators*) distinguished elements $\mathsf{B}, \mathsf{C}, \mathsf{I}, \mathsf{K}, \mathsf{W}, \mathsf{D}, \delta, \mathsf{F}$ satisfying

$\mathsf{B}xyz = x(yz)$	Composition, Cut
$\mathsf{C}xyz = (xz)y$	Exchange
$\mathsf{I}x = x$	Identity
$\mathsf{K}x\,!y = x$	Weakening
$\mathsf{W}x\,!y = x\,!y\,!y$	Contraction
$\mathsf{D}\,!x = x$	Dereliction
$\delta\,!x = !\,!x$	Comultiplication
$\mathsf{F}\,!x\,!y = !(xy)$	Monoidal functoriality

Here: $\cdot$ associates to the left; $\cdot$ is suppressed; and ! binds stronger than $\cdot$ does.

- Model of untyped linear λ

Linear Combinatory Algebra (LCA)



Defn. (LCA)
A *linear combinatory algebra (LCA)* is a set $\boldsymbol{A}$ equipped with

- a binary operator (called an *applicative structure*)

$$\cdot \; : \; \boldsymbol{A}^2 \longrightarrow \boldsymbol{A}$$

- a unary operator

$$! \; : \; \boldsymbol{A} \longrightarrow \boldsymbol{A}$$

- (*combinators*) distinguished elements $\mathsf{B}, \mathsf{C}, \mathsf{I}, \mathsf{K}, \mathsf{W}, \mathsf{D}, \delta, \mathsf{F}$ satisfying

$\mathsf{B}xyz = x(yz)$	Composition, Cut
$\mathsf{C}xyz = (xz)y$	Exchange
$\mathsf{I}x = x$	Identity
$\mathsf{K}x\,!\,y = x$	Weakening
$\mathsf{W}x\,!\,y = x\,!\,y\,!\,y$	Contraction
$\mathsf{D}\,!\,x = x$	Dereliction
$\delta\,!\,x = !\,!\,x$	Comultiplication
$\mathsf{F}\,!\,x\,!\,y = !(xy)$	Monoidal functoriality

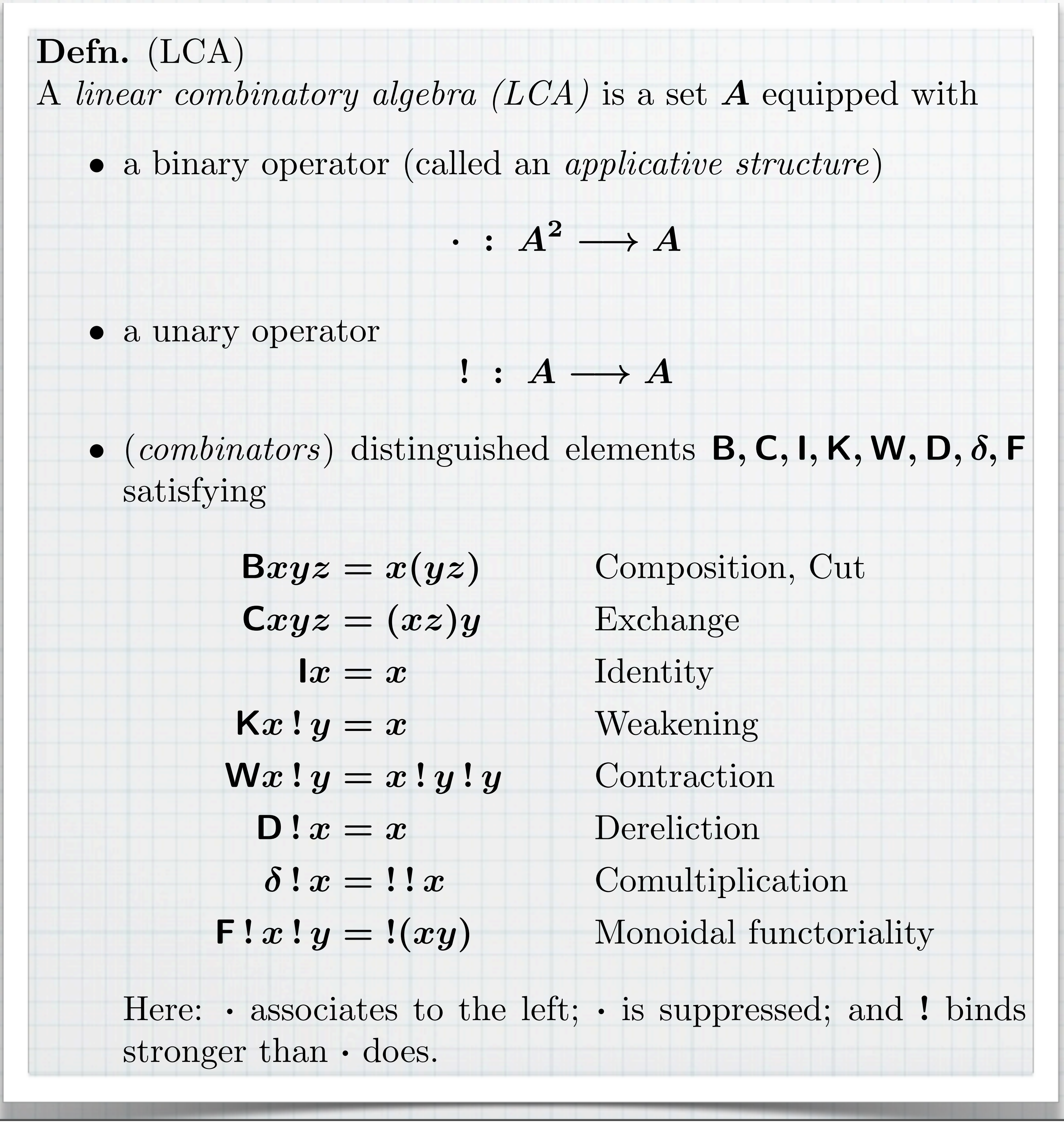
Here: $\cdot$ associates to the left; $\cdot$ is suppressed; and $!$ binds stronger than $\cdot$ does.

What
we want (outcome)

- Model of
untyped linear λ
- a ∈ A ≈
closed linear λ-term

Linear Combinatory Algebra (LCA)

What
we want (outcome)



Defn. (LCA)
A *linear combinatory algebra (LCA)* is a set $\boldsymbol{A}$ equipped with

- a binary operator (called an *applicative structure*)

$$\cdot \; : \; \boldsymbol{A}^2 \longrightarrow \boldsymbol{A}$$

- a unary operator

$$! \; : \; \boldsymbol{A} \longrightarrow \boldsymbol{A}$$

- (*combinators*) distinguished elements $\mathbf{B}, \mathbf{C}, \mathbf{I}, \mathbf{K}, \mathbf{W}, \mathbf{D}, \delta, \mathbf{F}$ satisfying

$\mathbf{B}xyz = x(yz)$	Composition, Cut
$\mathbf{C}xyz = (xz)y$	Exchange
$\mathbf{I}x = x$	Identity
$\mathbf{K}x\,!\,y = x$	Weakening
$\mathbf{W}x\,!\,y = x\,!\,y\,!\,y$	Contraction
$\mathbf{D}\,!\,x = x$	Dereliction
$\delta\,!\,x = !\,!\,x$	Comultiplication
$\mathbf{F}\,!\,x\,!\,y = !(xy)$	Monoidal functoriality

Here: $\cdot$ associates to the left; $\cdot$ is suppressed; and $!$ binds stronger than $\cdot$ does.

- Model of untyped linear λ
- a ∈ A ≈ closed linear λ-term
- No **S** or **K** (linear!)
- Combinatory completeness: e.g.

$$\lambda xyz.\, zxy$$

designates an elem. of A

What we use (ingredient)

GoI situation

Defn. (GoI situation [AHS02])
A *GoI situation* is a triple $(\mathbb{C}, F, U)$ where

- $\mathbb{C} = (\mathbb{C}, \otimes, I)$ is a traced symmetric monoidal category (TSMC);
- $F : \mathbb{C} \to \mathbb{C}$ is a traced symmetric monoidal functor, equipped with the following retractions (which are monoidal natural transformations).

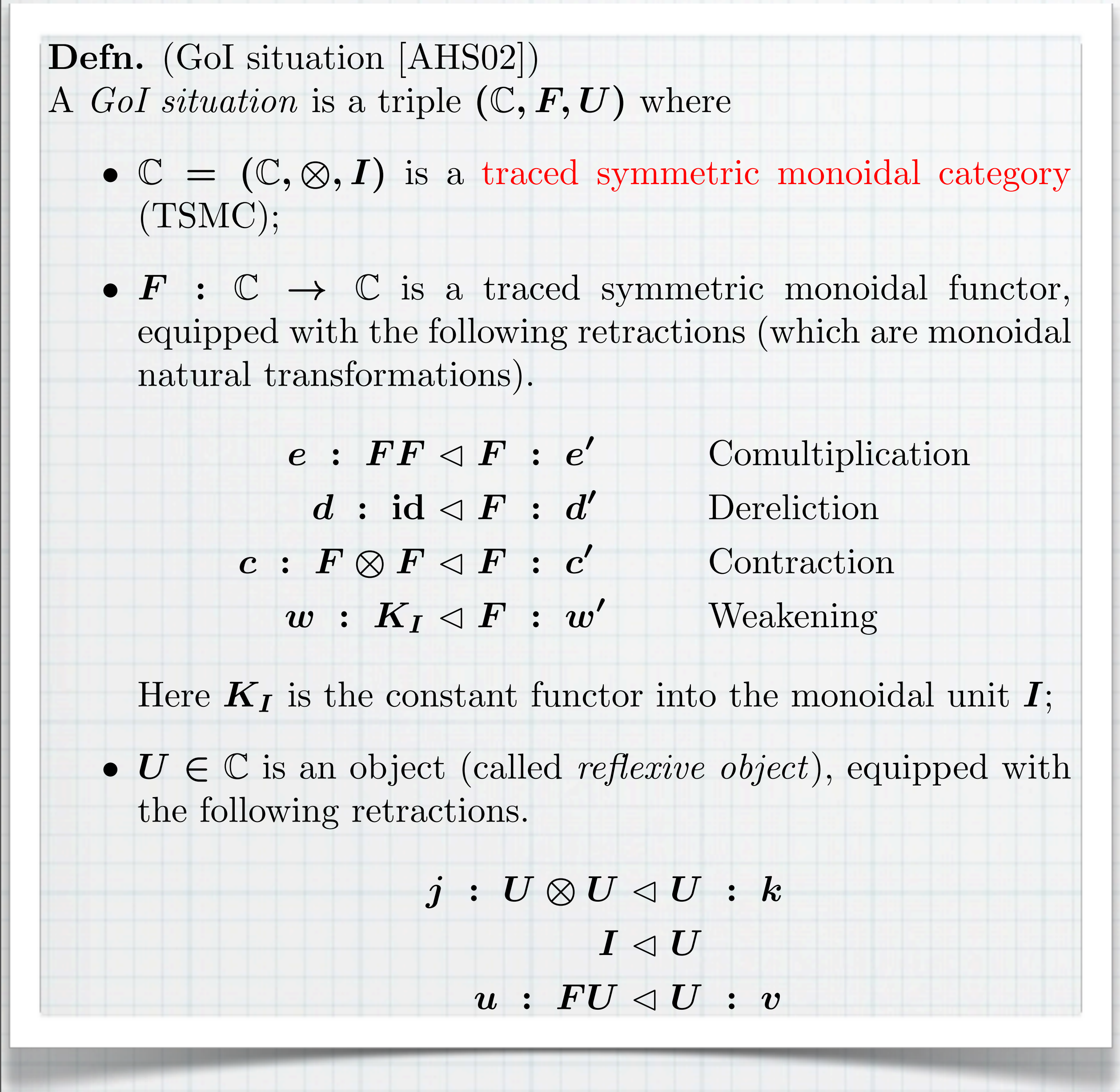
$$\begin{array}{rcl l}
e : & FF \lhd F & : e' & \text{Comultiplication} \\
d : & \mathrm{id} \lhd F & : d' & \text{Dereliction} \\
c : & F \otimes F \lhd F & : c' & \text{Contraction} \\
w : & K_I \lhd F & : w' & \text{Weakening}
\end{array}$$

Here K_I is the constant functor into the monoidal unit I;

- $U \in \mathbb{C}$ is an object (called *reflexive object*), equipped with the following retractions.

$$\begin{array}{rcl}
j : & U \otimes U \lhd U & : k \\
 & I \lhd U & \\
u : & FU \lhd U & : v
\end{array}$$

GoI situation



Defn. (GoI situation [AHS02])
A *GoI situation* is a triple $(\mathbb{C}, \boldsymbol{F}, \boldsymbol{U})$ where

- $\mathbb{C} = (\mathbb{C}, \otimes, \boldsymbol{I})$ is a traced symmetric monoidal category (TSMC);
- $\boldsymbol{F} : \mathbb{C} \to \mathbb{C}$ is a traced symmetric monoidal functor, equipped with the following retractions (which are monoidal natural transformations).

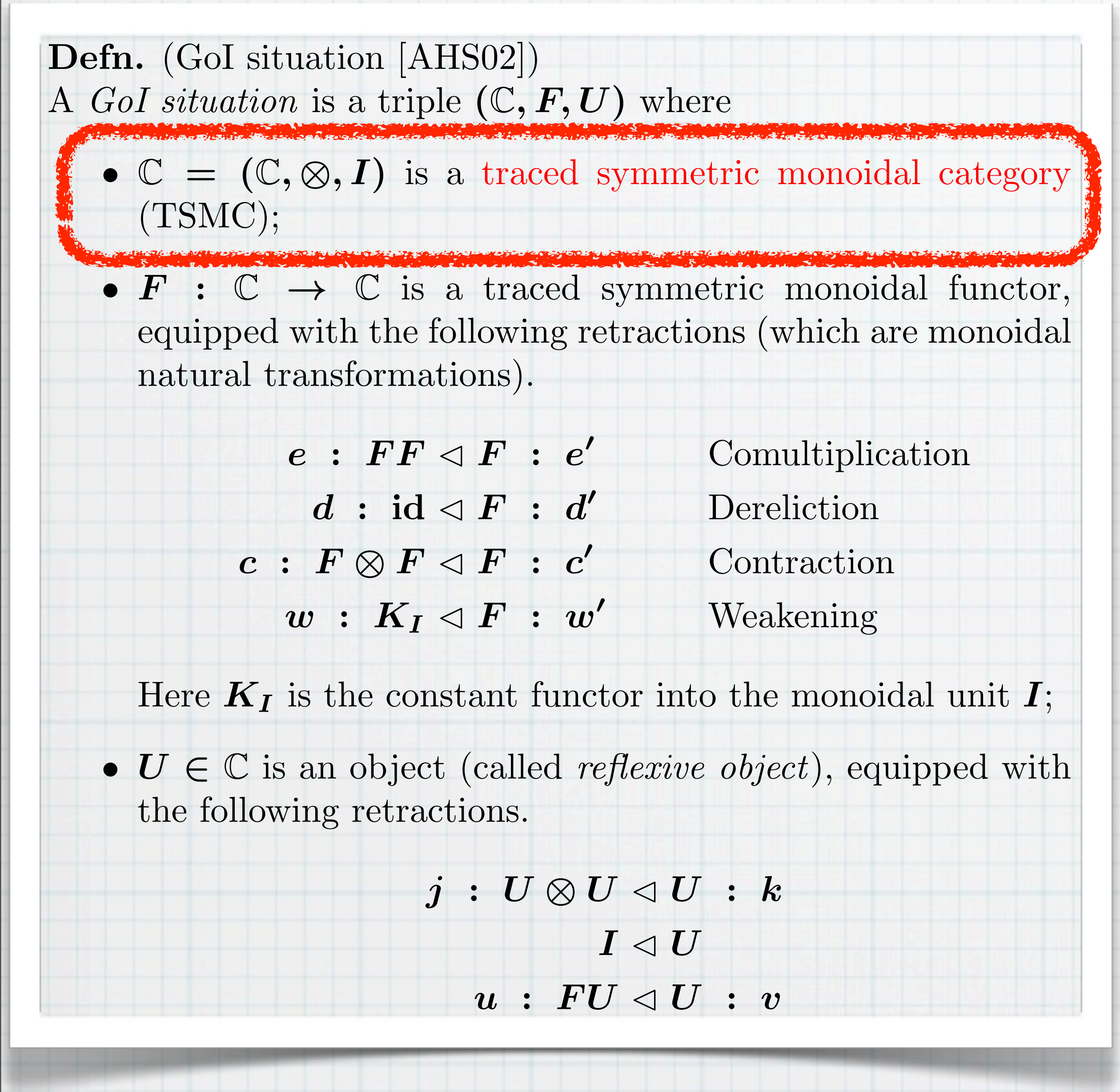
$$\begin{array}{rcl}
\boldsymbol{e} : \boldsymbol{F}\boldsymbol{F} \lhd \boldsymbol{F} : \boldsymbol{e}' & & \text{Comultiplication} \\
\boldsymbol{d} : \mathbf{id} \lhd \boldsymbol{F} : \boldsymbol{d}' & & \text{Dereliction} \\
\boldsymbol{c} : \boldsymbol{F} \otimes \boldsymbol{F} \lhd \boldsymbol{F} : \boldsymbol{c}' & & \text{Contraction} \\
\boldsymbol{w} : \boldsymbol{K_I} \lhd \boldsymbol{F} : \boldsymbol{w}' & & \text{Weakening}
\end{array}$$

  Here $\boldsymbol{K_I}$ is the constant functor into the monoidal unit $\boldsymbol{I}$;
- $\boldsymbol{U} \in \mathbb{C}$ is an object (called *reflexive object*), equipped with the following retractions.

$$\begin{array}{c}
\boldsymbol{j} : \boldsymbol{U} \otimes \boldsymbol{U} \lhd \boldsymbol{U} : \boldsymbol{k} \\
\boldsymbol{I} \lhd \boldsymbol{U} \\
\boldsymbol{u} : \boldsymbol{F}\boldsymbol{U} \lhd \boldsymbol{U} : \boldsymbol{v}
\end{array}$$

- Monoidal category $(\mathbb{C}, \otimes, I)$
  - String diagrams

GoI situation



Defn. (GoI situation [AHS02])
A *GoI situation* is a triple $(\mathbb{C}, \boldsymbol{F}, \boldsymbol{U})$ where

- $\mathbb{C} = (\mathbb{C}, \otimes, \boldsymbol{I})$ is a traced symmetric monoidal category (TSMC);
- $\boldsymbol{F} : \mathbb{C} \to \mathbb{C}$ is a traced symmetric monoidal functor, equipped with the following retractions (which are monoidal natural transformations).

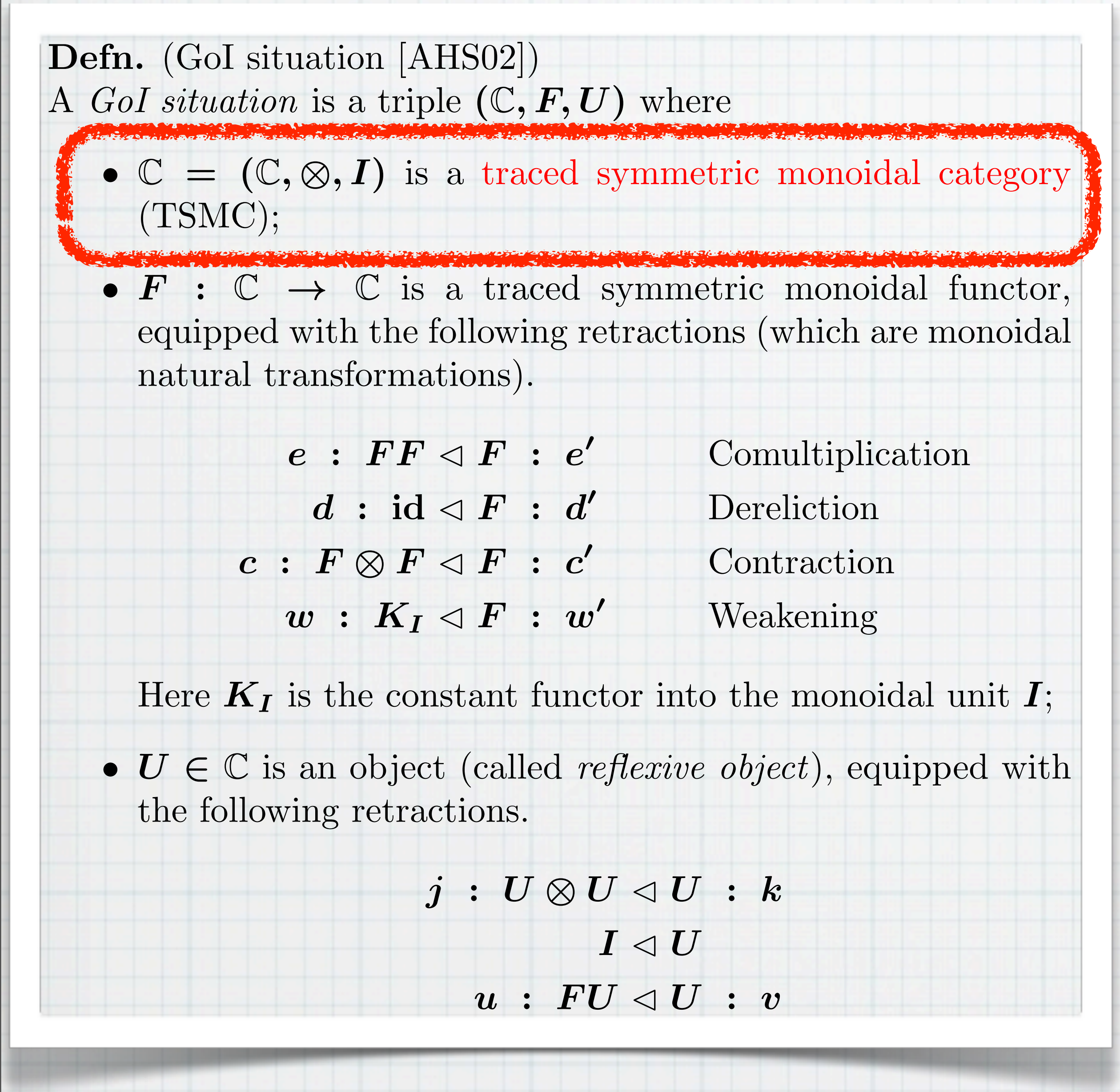
$$\begin{array}{rcccl l}
\boldsymbol{e} & : & \boldsymbol{FF} \lhd \boldsymbol{F} & : & \boldsymbol{e'} & \text{Comultiplication} \\
\boldsymbol{d} & : & \mathbf{id} \lhd \boldsymbol{F} & : & \boldsymbol{d'} & \text{Dereliction} \\
\boldsymbol{c} & : & \boldsymbol{F} \otimes \boldsymbol{F} \lhd \boldsymbol{F} & : & \boldsymbol{c'} & \text{Contraction} \\
\boldsymbol{w} & : & \boldsymbol{K_I} \lhd \boldsymbol{F} & : & \boldsymbol{w'} & \text{Weakening}
\end{array}$$

  Here $\boldsymbol{K_I}$ is the constant functor into the monoidal unit $\boldsymbol{I}$;
- $\boldsymbol{U} \in \mathbb{C}$ is an object (called *reflexive object*), equipped with the following retractions.

$$\begin{array}{rcccl}
\boldsymbol{j} & : & \boldsymbol{U} \otimes \boldsymbol{U} \lhd \boldsymbol{U} & : & \boldsymbol{k} \\
 & & \boldsymbol{I} \lhd \boldsymbol{U} & & \\
\boldsymbol{u} & : & \boldsymbol{FU} \lhd \boldsymbol{U} & : & \boldsymbol{v}
\end{array}$$

- **Monoidal category** $(\mathbb{C}, \otimes, \boldsymbol{I})$
  - **String diagrams**

GoI situation



Defn. (GoI situation [AHS02])
A *GoI situation* is a triple $(\mathbb{C}, F, U)$ where

- $\mathbb{C} = (\mathbb{C}, \otimes, I)$ is a traced symmetric monoidal category (TSMC);
- $F : \mathbb{C} \to \mathbb{C}$ is a traced symmetric monoidal functor, equipped with the following retractions (which are monoidal natural transformations).

  $e : FF \lhd F : e'$ Comultiplication

  $d : \mathrm{id} \lhd F : d'$ Dereliction

  $c : F \otimes F \lhd F : c'$ Contraction

  $w : K_I \lhd F : w'$ Weakening

  Here K_I is the constant functor into the monoidal unit I;
- $U \in \mathbb{C}$ is an object (called *reflexive object*), equipped with the following retractions.

  $$j : U \otimes U \lhd U : k$$

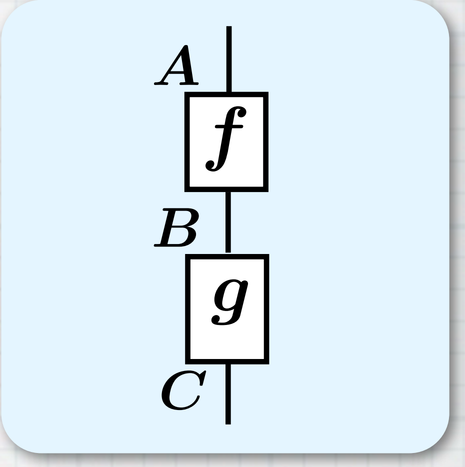
  $$I \lhd U$$

  $$u : FU \lhd U : v$$

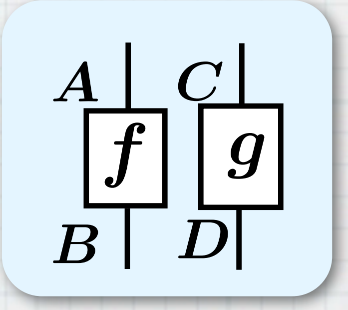
- Monoidal category $(\mathbb{C}, \otimes, I)$

- String diagrams

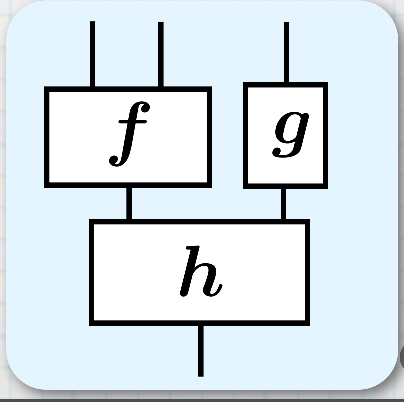
$$\frac{A \xrightarrow{f} B \quad B \xrightarrow{g} C}{A \xrightarrow{g \circ f} C}$$



$$\frac{A \xrightarrow{f} B \quad C \xrightarrow{g} D}{A \otimes C \xrightarrow{f \otimes g} B \otimes D}$$

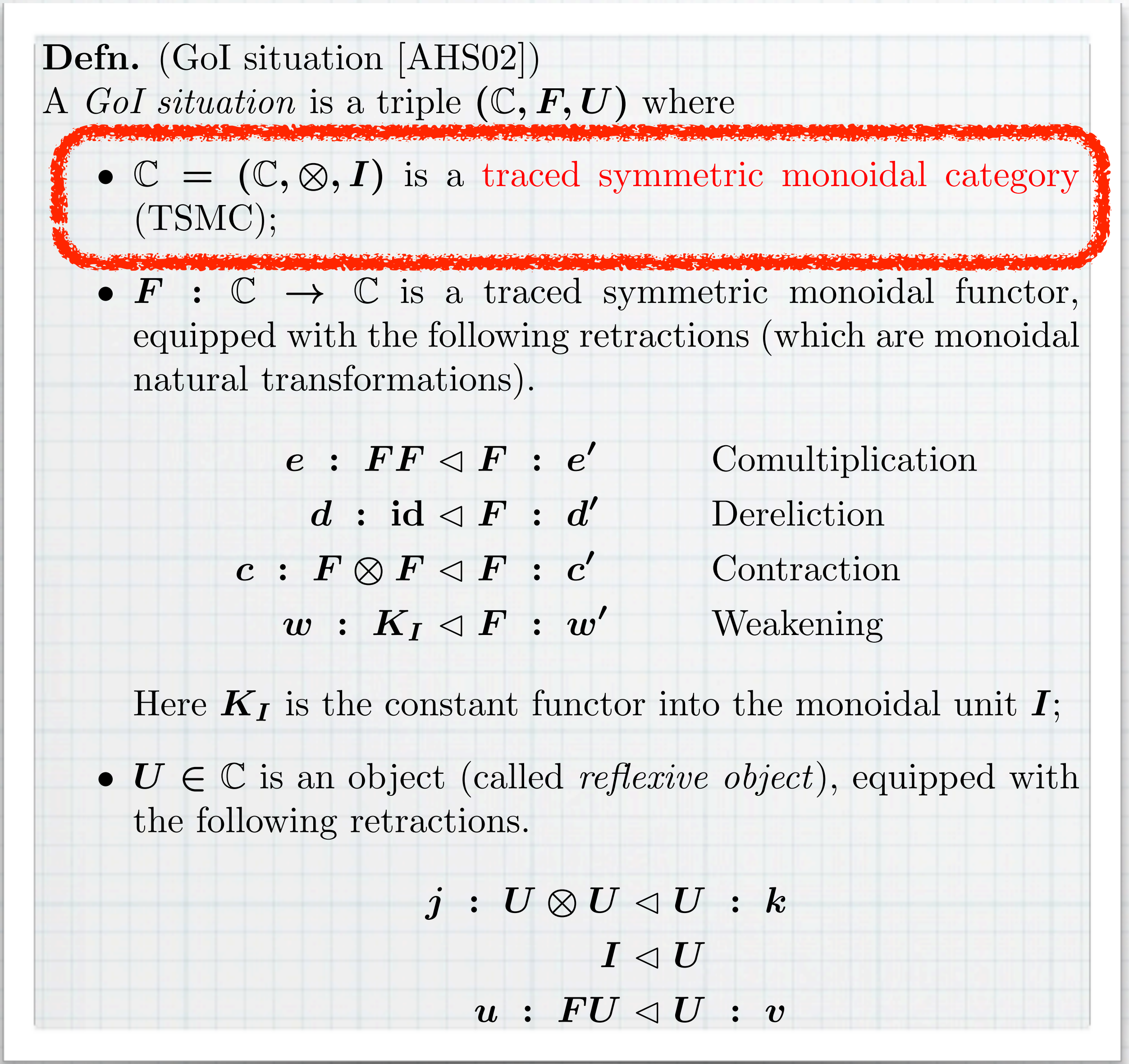


$$h \circ (f \otimes g)$$



o)

GoI situation



Defn. (GoI situation [AHS02])
A *GoI situation* is a triple $(\mathbb{C}, F, U)$ where

- $\mathbb{C} = (\mathbb{C}, \otimes, I)$ is a traced symmetric monoidal category (TSMC);
- $F : \mathbb{C} \to \mathbb{C}$ is a traced symmetric monoidal functor, equipped with the following retractions (which are monoidal natural transformations).

$e : FF \lhd F : e'$	Comultiplication
$d : \mathrm{id} \lhd F : d'$	Dereliction
$c : F \otimes F \lhd F : c'$	Contraction
$w : K_I \lhd F : w'$	Weakening

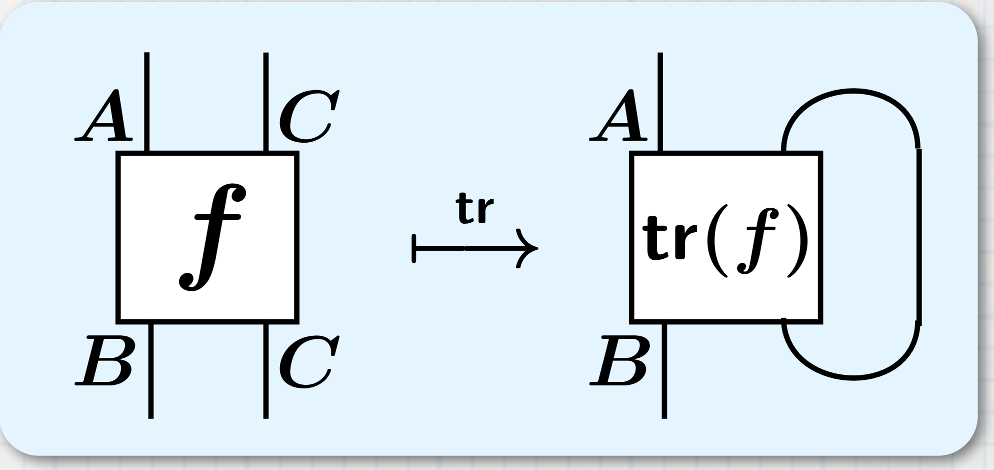
  Here K_I is the constant functor into the monoidal unit I;
- $U \in \mathbb{C}$ is an object (called *reflexive object*), equipped with the following retractions.

$$j : U \otimes U \lhd U : k$$
$$I \lhd U$$
$$u : FU \lhd U : v$$

- **Traced** monoidal category
  - “feedback”

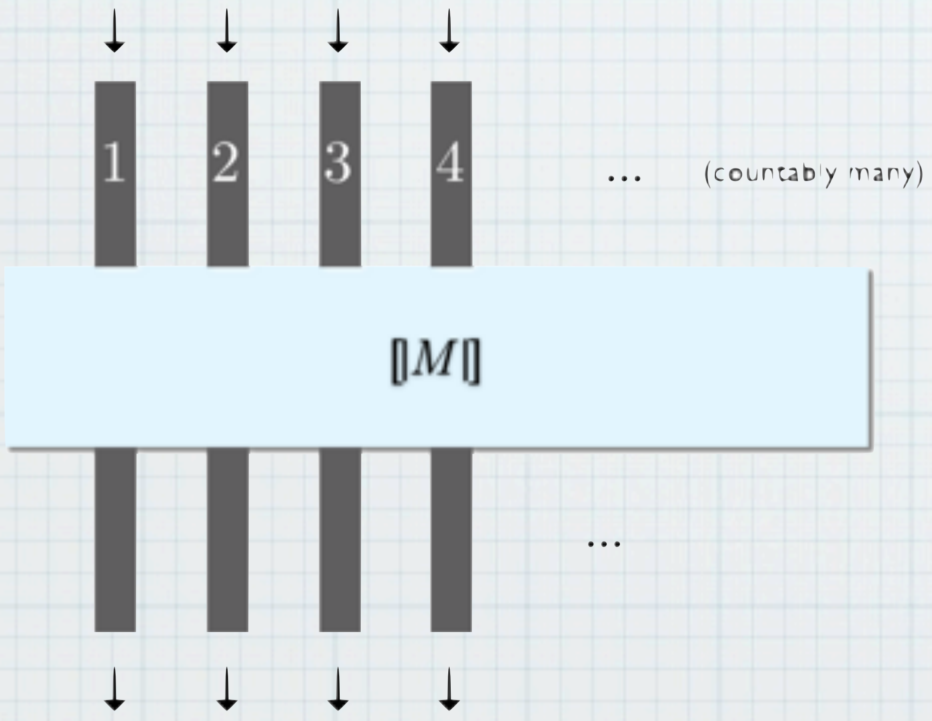
$$\frac{A \otimes C \xrightarrow{f} B \otimes C}{A \xrightarrow{\mathrm{tr}(f)} B}$$

that is

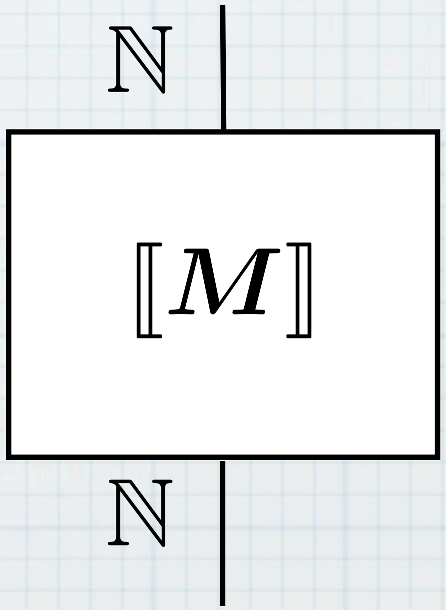


String Diagram vs. "Pipe Diagram"

- I use two ways of depicting partial functions $\mathbb{N} \rightharpoonup \mathbb{N}$



Pipe diagram

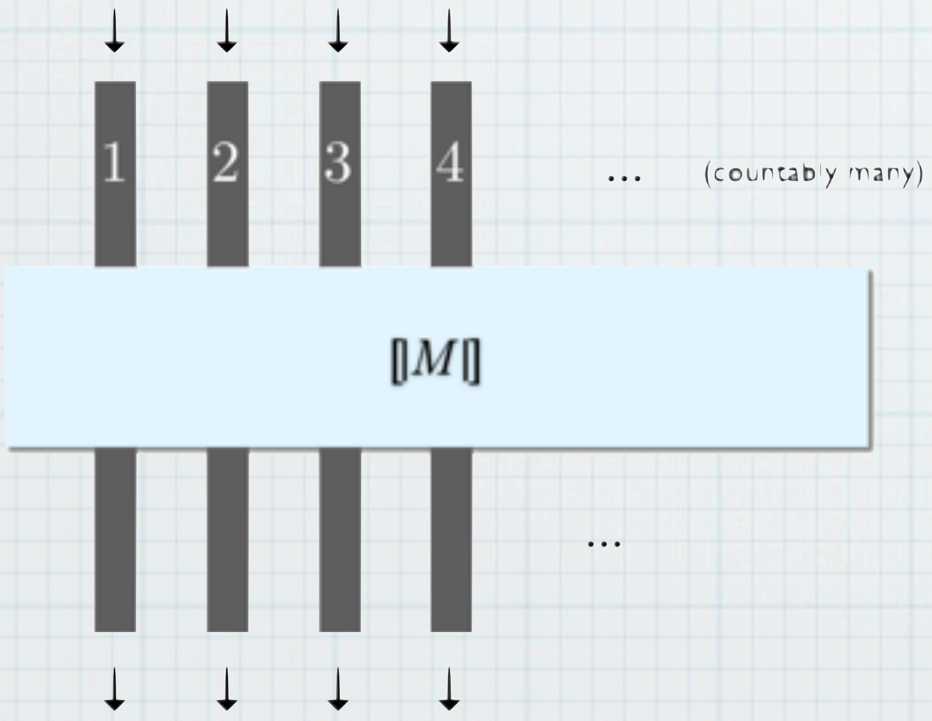


String diagram

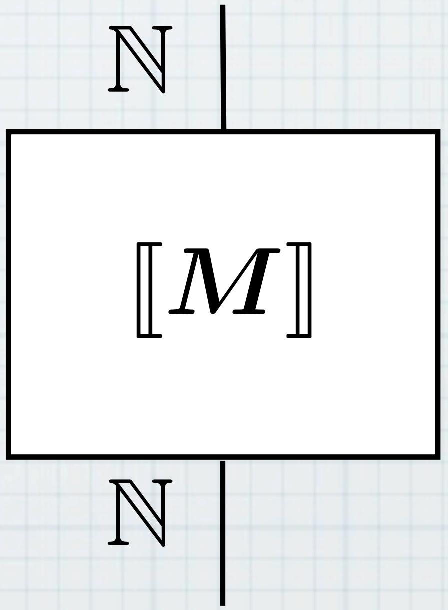
String Diagram vs. “Pipe Diagram”

- I use two ways of depicting partial functions $\mathbb{N} \rightharpoonup \mathbb{N}$

In the monoidal category $(\mathbf{Pfn}, +, 0)$



Pipe diagram



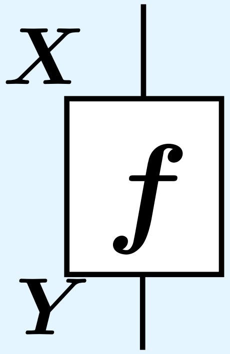
String diagram

Traced Sym. Monoidal Category

$(\mathbf{Pfn}, +, 0)$

- Category **Pfn** of partial functions
  - **Obj.** A set X
  - **Arr.** A partial function

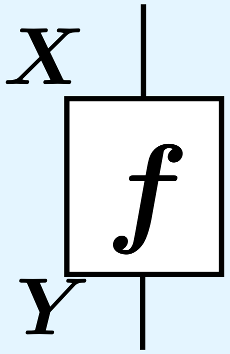
$$\begin{array}{c} X \to Y \ \text{ in } \mathbf{Pfn} \\ \hline\hline X \rightharpoonup Y, \text{ partial function} \end{array}$$



Traced Sym. Monoidal Category
$(\mathbf{Pfn}, +, 0)$

- Category **Pfn** of partial functions
  - Obj. A set X
  - Arr. A partial function

$$\begin{array}{c} \boldsymbol{X} \to \boldsymbol{Y} \ \text{ in } \mathbf{Pfn} \\ \hline\hline \boldsymbol{X} \rightharpoonup \boldsymbol{Y}, \text{ partial function} \end{array}$$



- is traced symmetric monoidal

Traced Sym. Monoidal Category

$(\mathbf{Pfn}, +, 0)$

$$\frac{X + Z \xrightarrow{f} Y + Z \qquad \text{in } \mathbf{Pfn}}{X \xrightarrow{\mathbf{tr}(f)} Y \qquad \text{in } \mathbf{Pfn}}$$

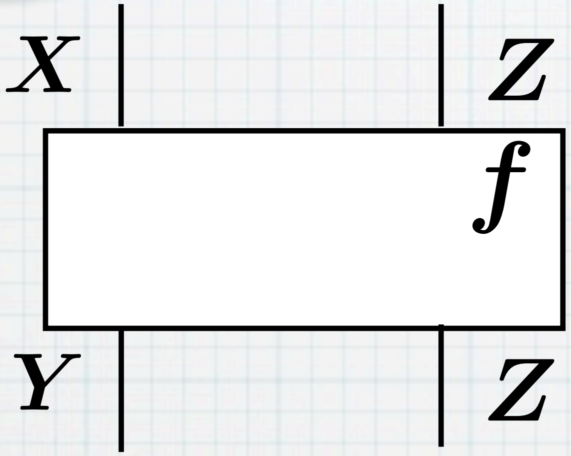
How?

Traced Sym. Monoidal Category

$(\mathbf{Pfn}, +, 0)$

- $$\frac{X + Z \xrightarrow{f} Y + Z \quad \text{in } \mathbf{Pfn}}{X \xrightarrow{\mathsf{tr}(f)} Y \quad \text{in } \mathbf{Pfn}}$$

How?

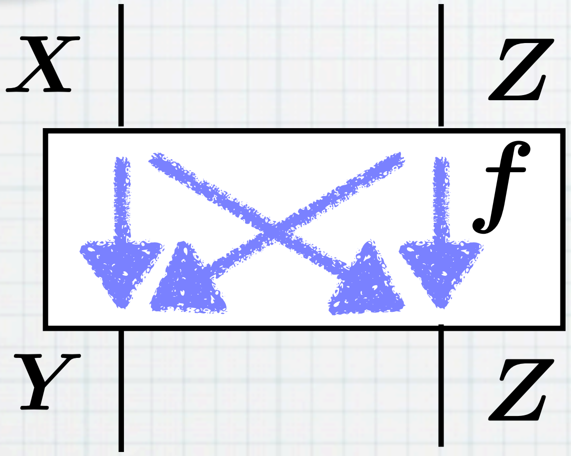
- 

Traced Sym. Monoidal Category

$(\mathbf{Pfn}, +, 0)$

- $$\frac{X + Z \xrightarrow{f} Y + Z \quad \text{in } \mathbf{Pfn}}{X \xrightarrow{\mathbf{tr}(f)} Y \quad \text{in } \mathbf{Pfn}}$$

How?

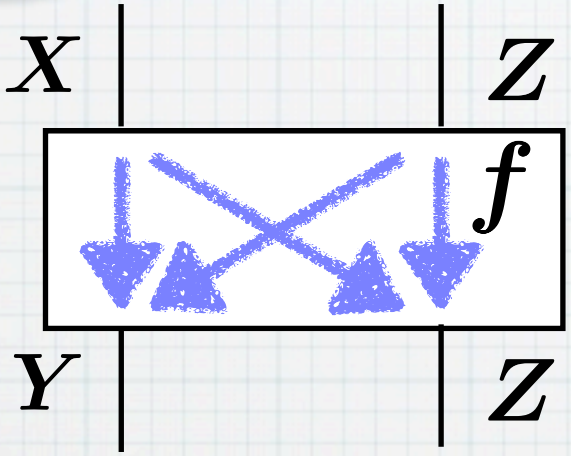
- 

Traced Sym. Monoidal Category

$(\mathbf{Pfn}, +, 0)$

- $$\frac{X + Z \xrightarrow{f} Y + Z \quad \text{in } \mathbf{Pfn}}{X \xrightarrow{\mathrm{tr}(f)} Y \quad \text{in } \mathbf{Pfn}}$$

How?

- 

$$f_{XY}(x) := \begin{cases} f(x) & \text{if } f(x) \in Y \\ \perp & \text{o.w.} \end{cases}$$

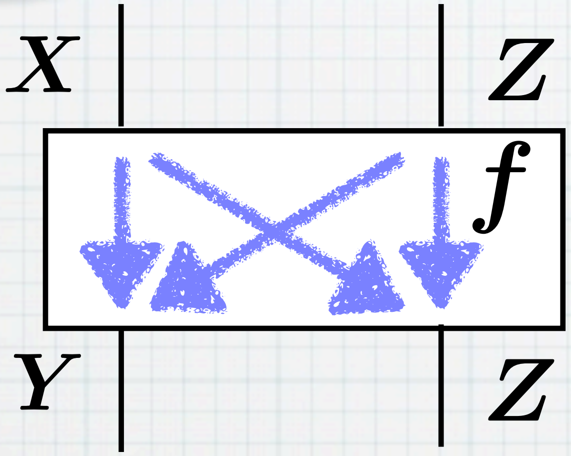
Similar for f_{XZ}, f_{ZY}, f_{ZZ}

Traced Sym. Monoidal Category

$(\mathbf{Pfn}, +, 0)$

- $$\frac{X+Z \xrightarrow{f} Y+Z \quad \text{in } \mathbf{Pfn}}{X \xrightarrow{\mathbf{tr}(f)} Y \quad \text{in } \mathbf{Pfn}}$$

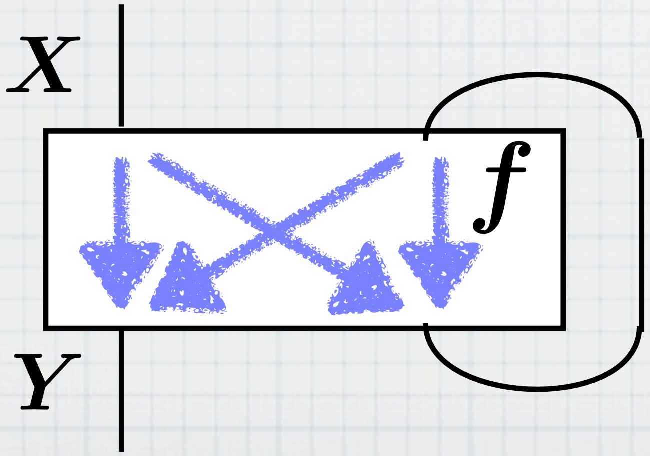
How?

- 

$$f_{XY}(x) := \begin{cases} f(x) & \text{if } f(x) \in Y \\ \perp & \text{o.w.} \end{cases}$$

Similar for f_{XZ}, f_{ZY}, f_{ZZ}

- Trace operator:

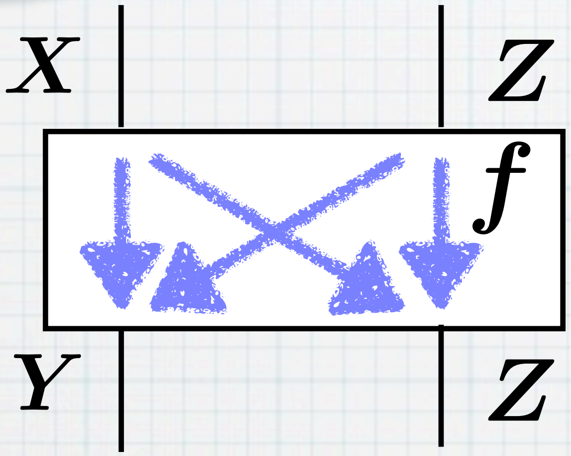


Traced Sym. Monoidal Category

$(\mathbf{Pfn}, +, 0)$

- $$\frac{X + Z \xrightarrow{f} Y + Z \qquad \text{in } \mathbf{Pfn}}{X \xrightarrow{\mathsf{tr}(f)} Y \qquad \text{in } \mathbf{Pfn}}$$

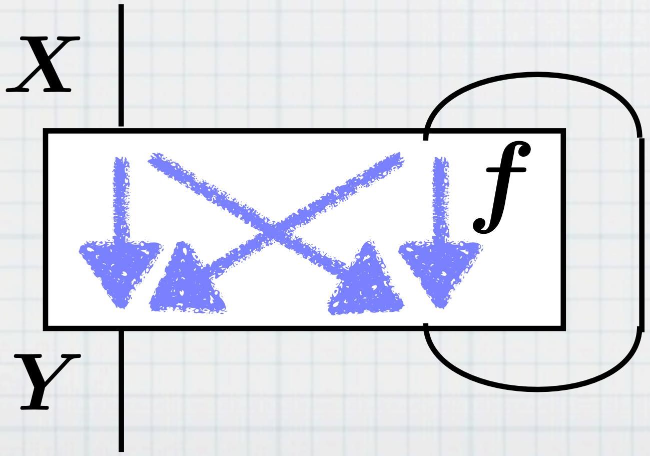
How?

- 

$$f_{XY}(x) := \begin{cases} f(x) & \text{if } f(x) \in Y \\ \bot & \text{o.w.} \end{cases}$$

Similar for f_{XZ}, f_{ZY}, f_{ZZ}

- Trace operator:



$$\mathsf{tr}(f) =$$

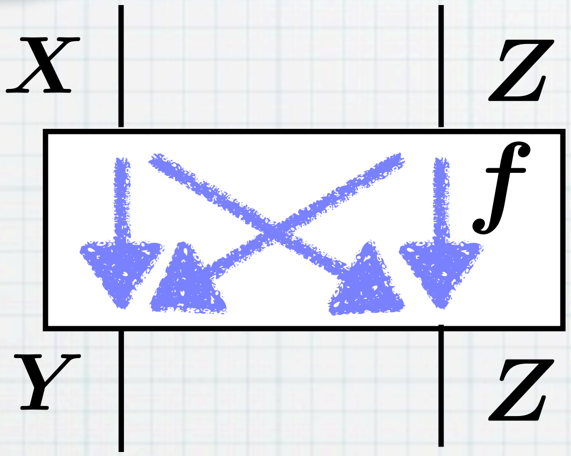
$$f_{XY} \sqcup \left(\coprod_{n \in \mathbb{N}} f_{ZY} \circ (f_{ZZ})^n \circ f_{XZ} \right)$$

Traced Sym. Monoidal Category

$(\mathbf{Pfn}, +, 0)$

- $$\frac{\boldsymbol{X} + \boldsymbol{Z} \xrightarrow{f} \boldsymbol{Y} + \boldsymbol{Z} \qquad \text{in } \mathbf{Pfn}}{\boldsymbol{X} \xrightarrow{\mathsf{tr}(f)} \boldsymbol{Y} \qquad \text{in } \mathbf{Pfn}}$$

How?

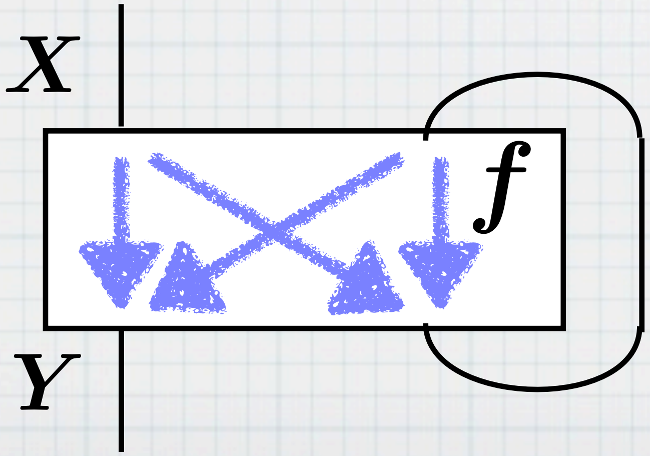
- 

$$f_{XY}(x) := \begin{cases} f(x) & \text{if } f(x) \in Y \\ \bot & \text{o.w.} \end{cases}$$

Similar for f_{XZ}, f_{ZY}, f_{ZZ}

- **Execution formula** (Girard)
- Partiality is essential (infinite loop)

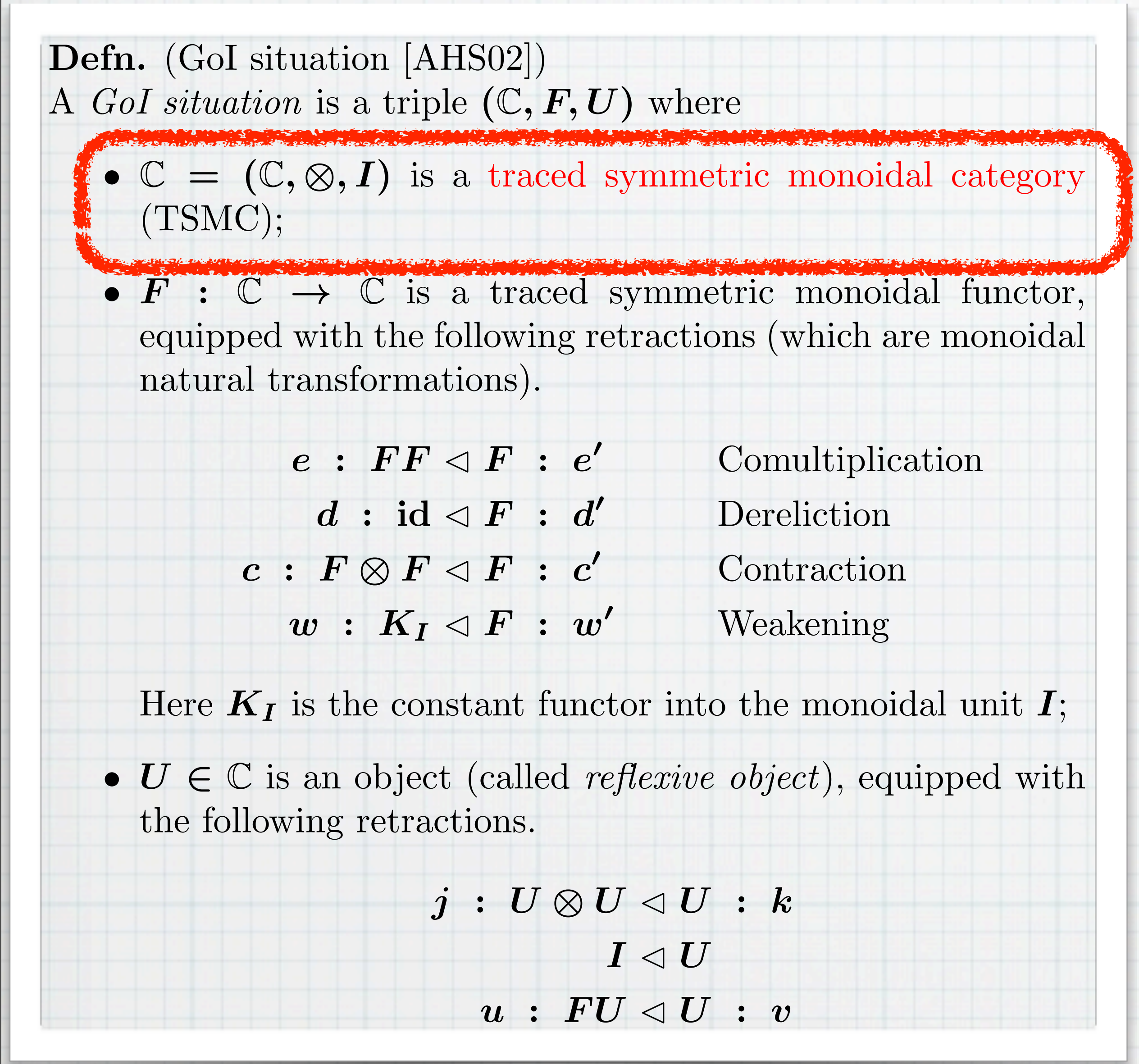
- Trace operator:



$$\mathsf{tr}(f) =$$

$$f_{XY} \sqcup \left(\coprod_{n \in \mathbb{N}} f_{ZY} \circ (f_{ZZ})^n \circ f_{XZ} \right)$$

GoI situation



Defn. (GoI situation [AHS02])
A *GoI situation* is a triple $(\mathbb{C}, \boldsymbol{F}, \boldsymbol{U})$ where

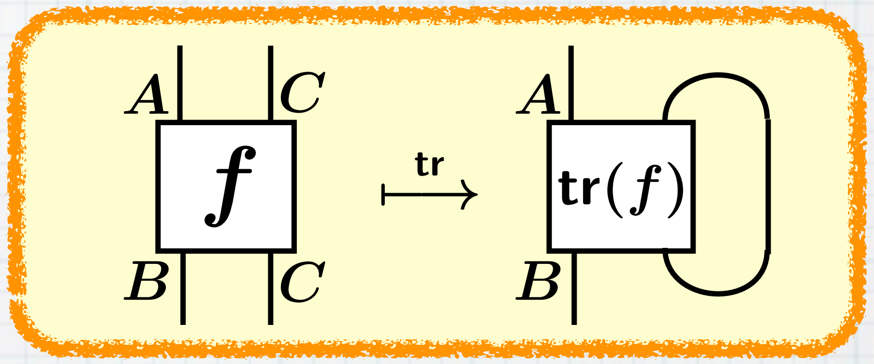
- $\mathbb{C} = (\mathbb{C}, \otimes, \boldsymbol{I})$ is a traced symmetric monoidal category (TSMC);
- $\boldsymbol{F} : \mathbb{C} \to \mathbb{C}$ is a traced symmetric monoidal functor, equipped with the following retractions (which are monoidal natural transformations).

$$\begin{array}{rcll} \boldsymbol{e} : \boldsymbol{FF} \lhd \boldsymbol{F} : \boldsymbol{e'} & & \text{Comultiplication} \\ \boldsymbol{d} : \mathbf{id} \lhd \boldsymbol{F} : \boldsymbol{d'} & & \text{Dereliction} \\ \boldsymbol{c} : \boldsymbol{F} \otimes \boldsymbol{F} \lhd \boldsymbol{F} : \boldsymbol{c'} & & \text{Contraction} \\ \boldsymbol{w} : \boldsymbol{K_I} \lhd \boldsymbol{F} : \boldsymbol{w'} & & \text{Weakening} \end{array}$$

  Here $\boldsymbol{K_I}$ is the constant functor into the monoidal unit $\boldsymbol{I}$;
- $\boldsymbol{U} \in \mathbb{C}$ is an object (called *reflexive object*), equipped with the following retractions.

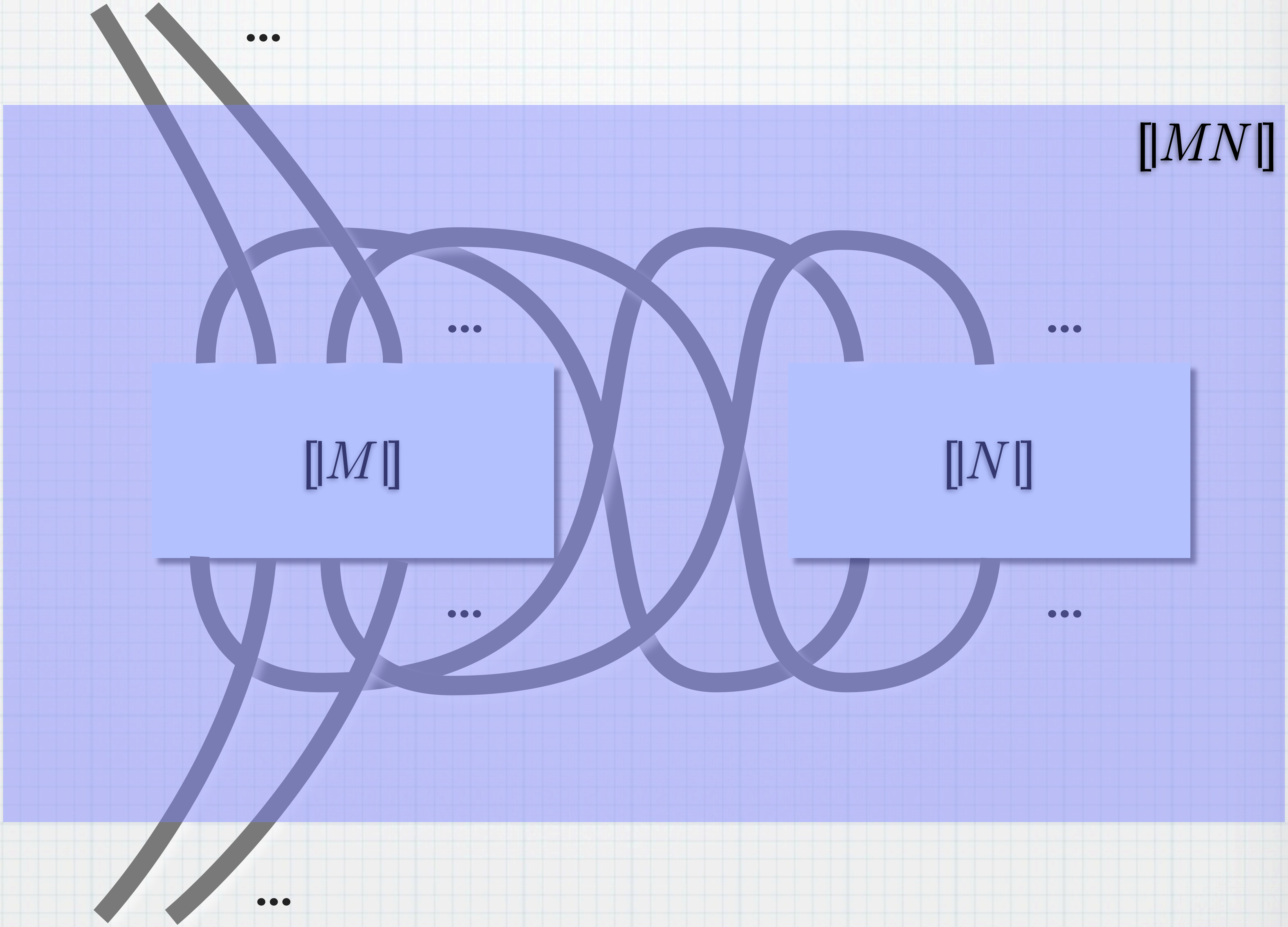
$$\begin{array}{c} \boldsymbol{j} : \boldsymbol{U} \otimes \boldsymbol{U} \lhd \boldsymbol{U} : \boldsymbol{k} \\ \boldsymbol{I} \lhd \boldsymbol{U} \\ \boldsymbol{u} : \boldsymbol{FU} \lhd \boldsymbol{U} : \boldsymbol{v} \end{array}$$

- Traced sym. monoidal cat.
  - Where one can "feedback"

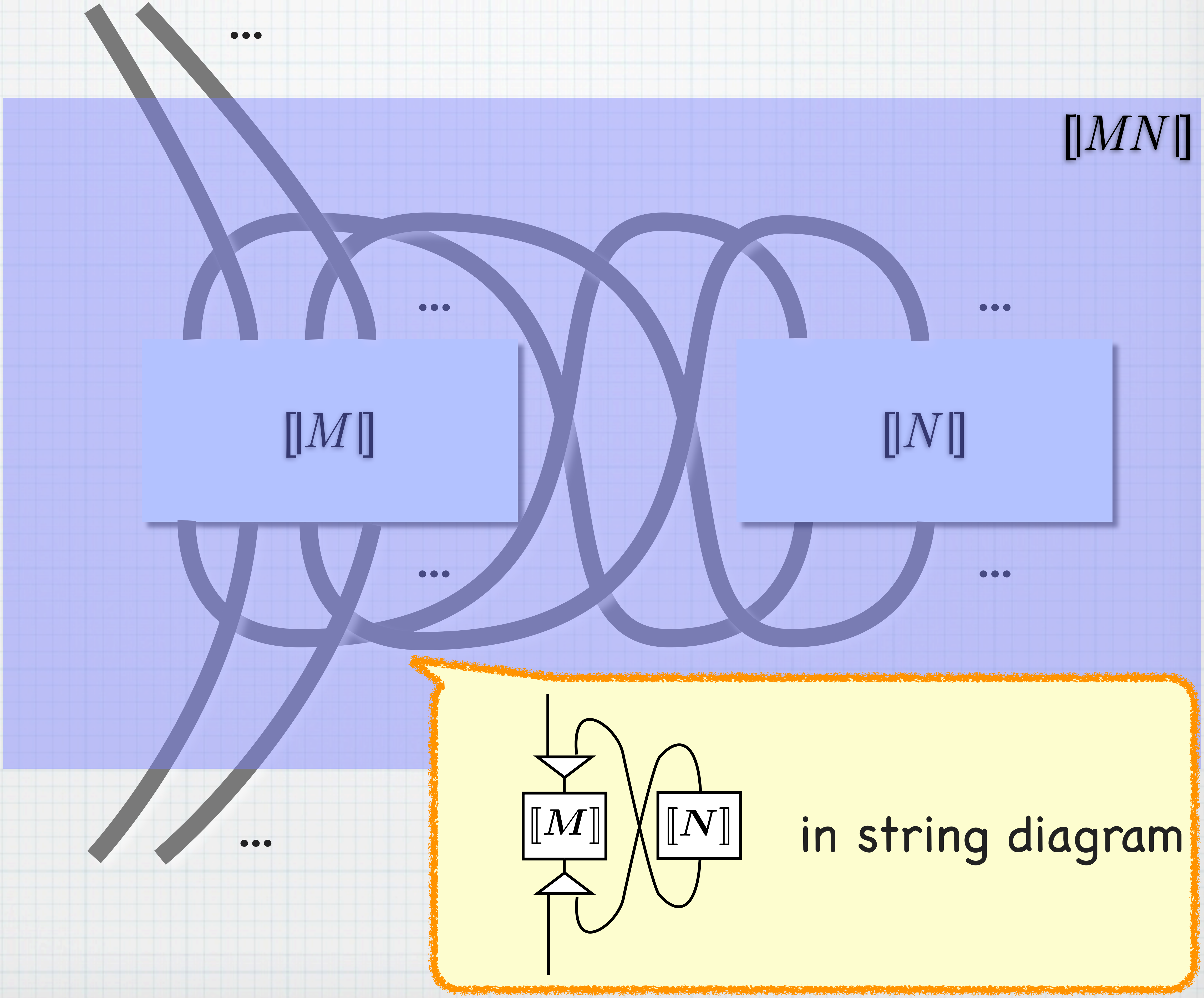


  - Why for GoI?

$$[\![\boldsymbol{M N}]\!] =$$

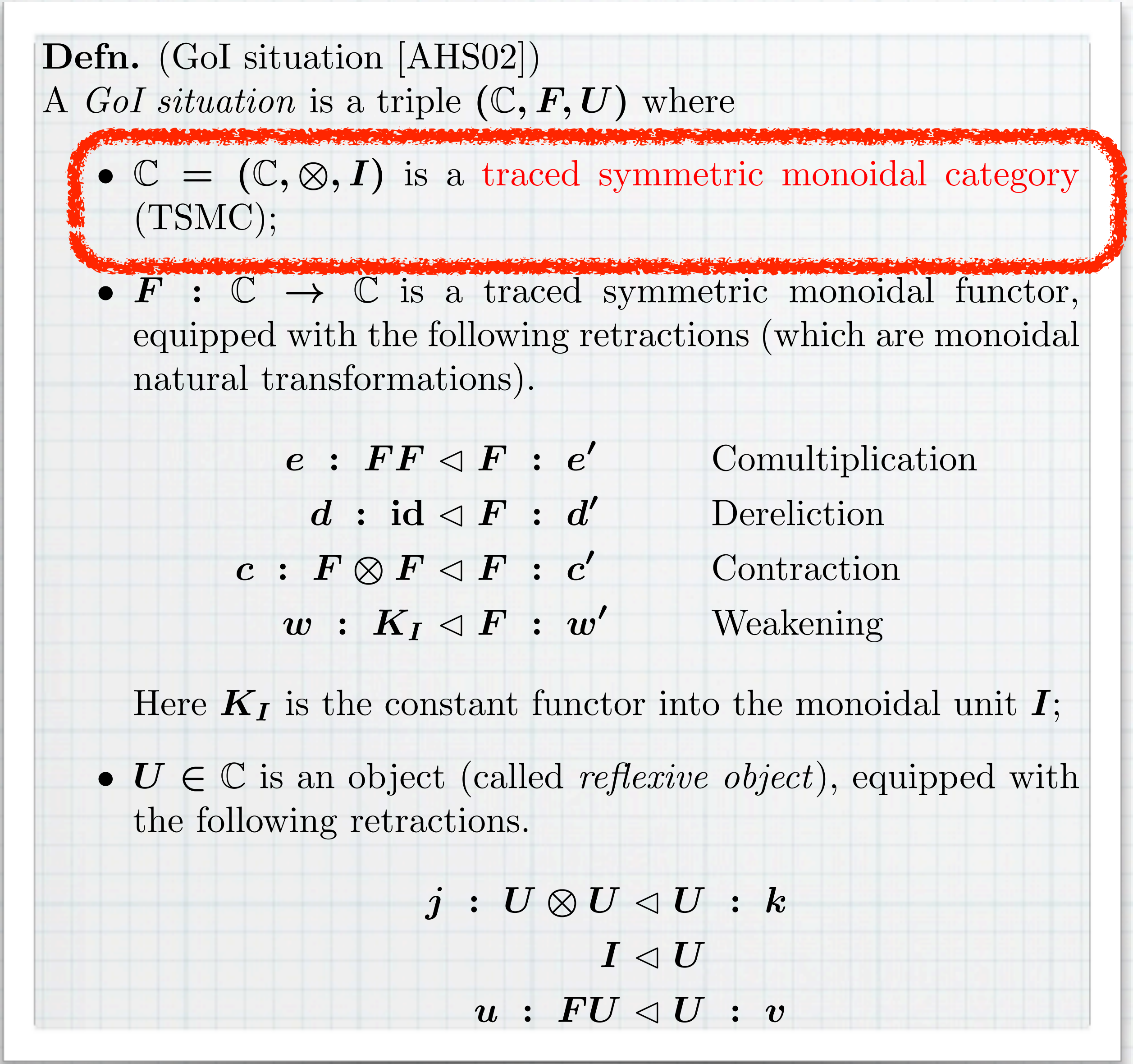


$$[\![MN]\!] =$$



in string diagram

GoI situation



Defn. (GoI situation [AHS02])
A *GoI situation* is a triple $(\mathbb{C}, \boldsymbol{F}, \boldsymbol{U})$ where

- $\mathbb{C} = (\mathbb{C}, \otimes, \boldsymbol{I})$ is a traced symmetric monoidal category (TSMC);
- $\boldsymbol{F} : \mathbb{C} \to \mathbb{C}$ is a traced symmetric monoidal functor, equipped with the following retractions (which are monoidal natural transformations).

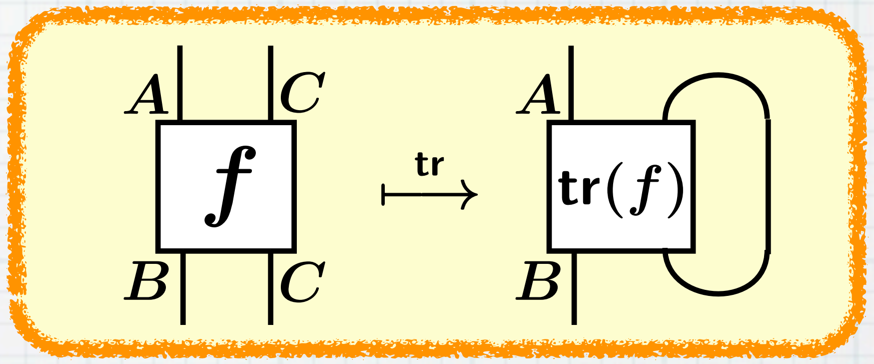
$\boldsymbol{e} : \boldsymbol{F}\boldsymbol{F} \lhd \boldsymbol{F} : \boldsymbol{e}'$	Comultiplication
$\boldsymbol{d} : \mathbf{id} \lhd \boldsymbol{F} : \boldsymbol{d}'$	Dereliction
$\boldsymbol{c} : \boldsymbol{F} \otimes \boldsymbol{F} \lhd \boldsymbol{F} : \boldsymbol{c}'$	Contraction
$\boldsymbol{w} : \boldsymbol{K_I} \lhd \boldsymbol{F} : \boldsymbol{w}'$	Weakening

  Here $\boldsymbol{K_I}$ is the constant functor into the monoidal unit $\boldsymbol{I}$;

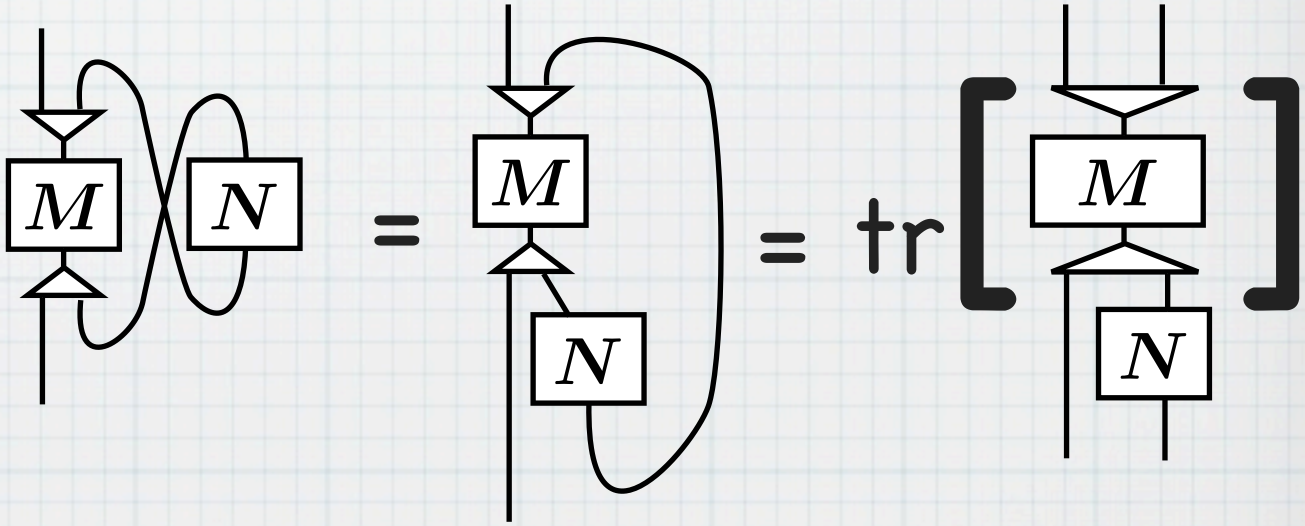
- $\boldsymbol{U} \in \mathbb{C}$ is an object (called *reflexive object*), equipped with the following retractions.

$$\begin{aligned} \boldsymbol{j} : \boldsymbol{U} \otimes \boldsymbol{U} &\lhd \boldsymbol{U} : \boldsymbol{k} \\ \boldsymbol{I} &\lhd \boldsymbol{U} \\ \boldsymbol{u} : \boldsymbol{F}\boldsymbol{U} &\lhd \boldsymbol{U} : \boldsymbol{v} \end{aligned}$$

- Traced sym. monoidal cat.
  - Where one can "feedback"



  - Why for GoI?



- Leading example: **Pfn**

GoI situation

Defn. (GoI situation [AHS02])
A *GoI situation* is a triple $(\mathbb{C}, F, U)$ where

- $\mathbb{C} = (\mathbb{C}, \otimes, I)$ is a traced symmetric monoidal category (TSMC);
- $F : \mathbb{C} \to \mathbb{C}$ is a traced symmetric monoidal functor, equipped with the following retractions (which are monoidal natural transformations).

$e : FF \lhd F : e'$	Comultiplication
$d : \mathrm{id} \lhd F : d'$	Dereliction
$c : F \otimes F \lhd F : c'$	Contraction
$w : K_I \lhd F : w'$	Weakening

  Here K_I is the constant functor into the monoidal unit I;
- $U \in \mathbb{C}$ is an object (called *reflexive object*), equipped with the following retractions.

$$\begin{aligned} j : U \otimes U \lhd U : k \\ I \lhd U \\ u : FU \lhd U : v \end{aligned}$$

Defn. (Retraction)
A *retraction* from X to Y,

$$f : X \lhd Y : g \ ,$$

is a pair of arrows

$$\begin{tikzcd} X \arrow[loop left, "\mathrm{id}"] \arrow[r, bend left, "f"] & Y \arrow[l, bend left, "g"] \end{tikzcd}$$

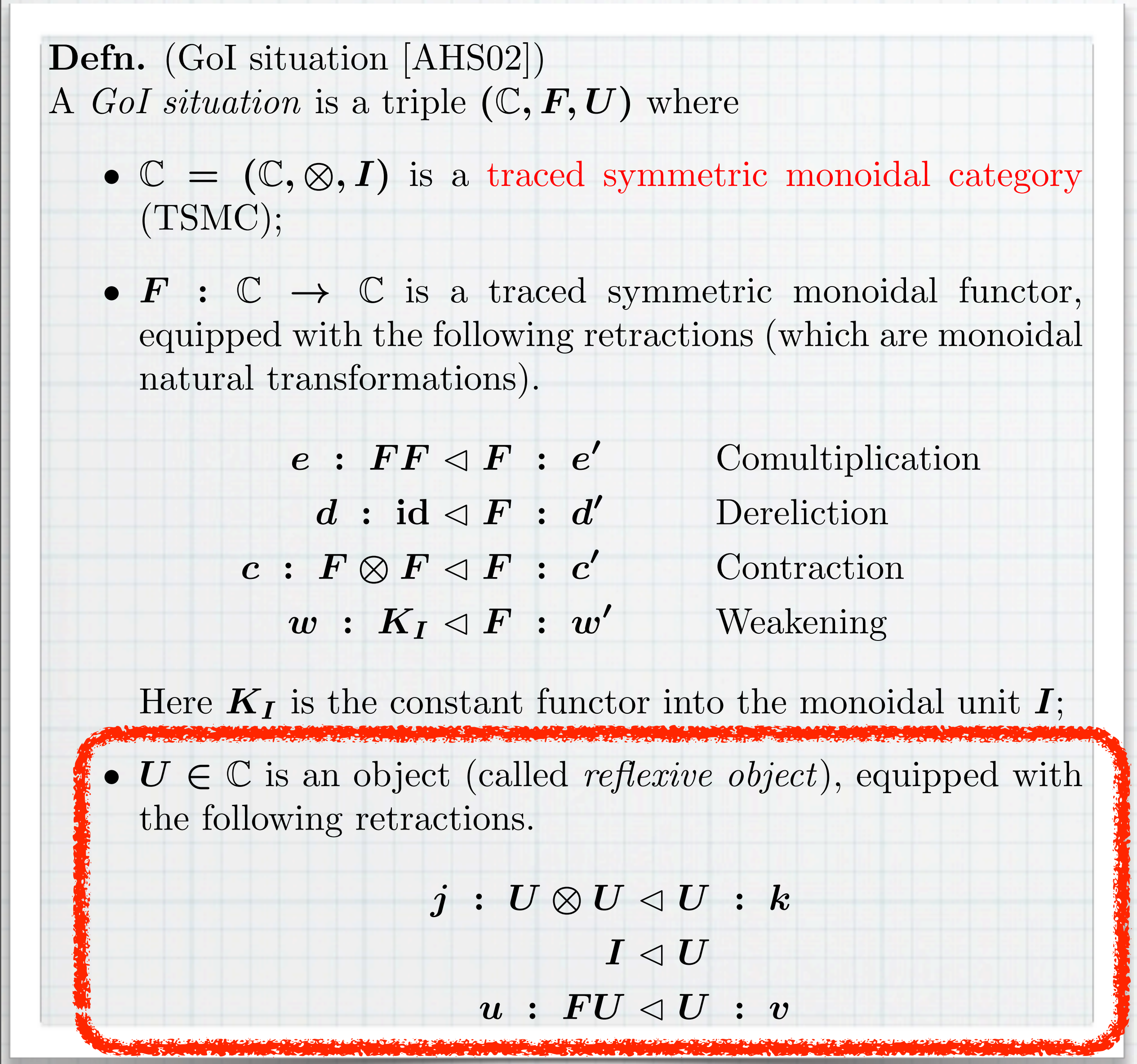
"embedding"

"projection"

such that $g \circ f = \mathrm{id}_X$.

- ✱ Functor F
  - ✱ For obtaining $! : A \to A$

GoI situation



Defn. (GoI situation [AHS02])
A *GoI situation* is a triple $(\mathbb{C}, \boldsymbol{F}, \boldsymbol{U})$ where

- $\mathbb{C} = (\mathbb{C}, \otimes, \boldsymbol{I})$ is a traced symmetric monoidal category (TSMC);
- $\boldsymbol{F} : \mathbb{C} \to \mathbb{C}$ is a traced symmetric monoidal functor, equipped with the following retractions (which are monoidal natural transformations).

$\boldsymbol{e} : \boldsymbol{FF} \lhd \boldsymbol{F} : \boldsymbol{e'}$	Comultiplication
$\boldsymbol{d} : \mathbf{id} \lhd \boldsymbol{F} : \boldsymbol{d'}$	Dereliction
$\boldsymbol{c} : \boldsymbol{F} \otimes \boldsymbol{F} \lhd \boldsymbol{F} : \boldsymbol{c'}$	Contraction
$\boldsymbol{w} : \boldsymbol{K_I} \lhd \boldsymbol{F} : \boldsymbol{w'}$	Weakening

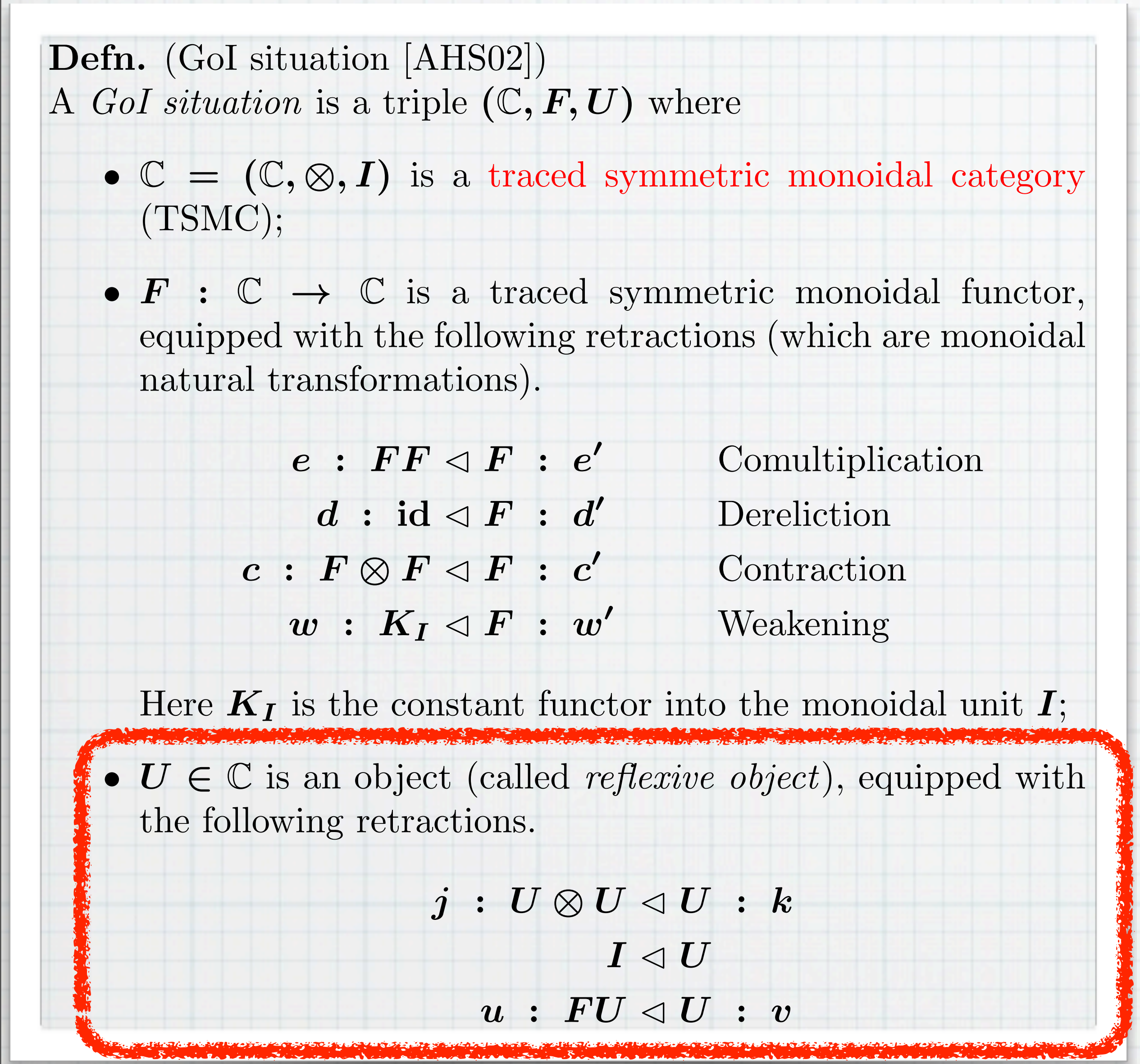
  Here $\boldsymbol{K_I}$ is the constant functor into the monoidal unit $\boldsymbol{I}$;

- $\boldsymbol{U} \in \mathbb{C}$ is an object (called *reflexive object*), equipped with the following retractions.

$$\begin{aligned} \boldsymbol{j} : \boldsymbol{U} \otimes \boldsymbol{U} \lhd \boldsymbol{U} &: \boldsymbol{k} \\ \boldsymbol{I} \lhd \boldsymbol{U} & \\ \boldsymbol{u} : \boldsymbol{FU} \lhd \boldsymbol{U} &: \boldsymbol{v} \end{aligned}$$

- The reflexive object U
- Retr. $$\begin{tikzcd} \boldsymbol{U} \otimes \boldsymbol{U} \arrow[r, bend left, "\boldsymbol{j}"] & \boldsymbol{U} \arrow[l, bend left, "\boldsymbol{k}"] \end{tikzcd}$$

GoI situation



Defn. (GoI situation [AHS02])
A *GoI situation* is a triple $(\mathbb{C}, \boldsymbol{F}, \boldsymbol{U})$ where

- $\mathbb{C} = (\mathbb{C}, \otimes, \boldsymbol{I})$ is a traced symmetric monoidal category (TSMC);
- $\boldsymbol{F} : \mathbb{C} \to \mathbb{C}$ is a traced symmetric monoidal functor, equipped with the following retractions (which are monoidal natural transformations).

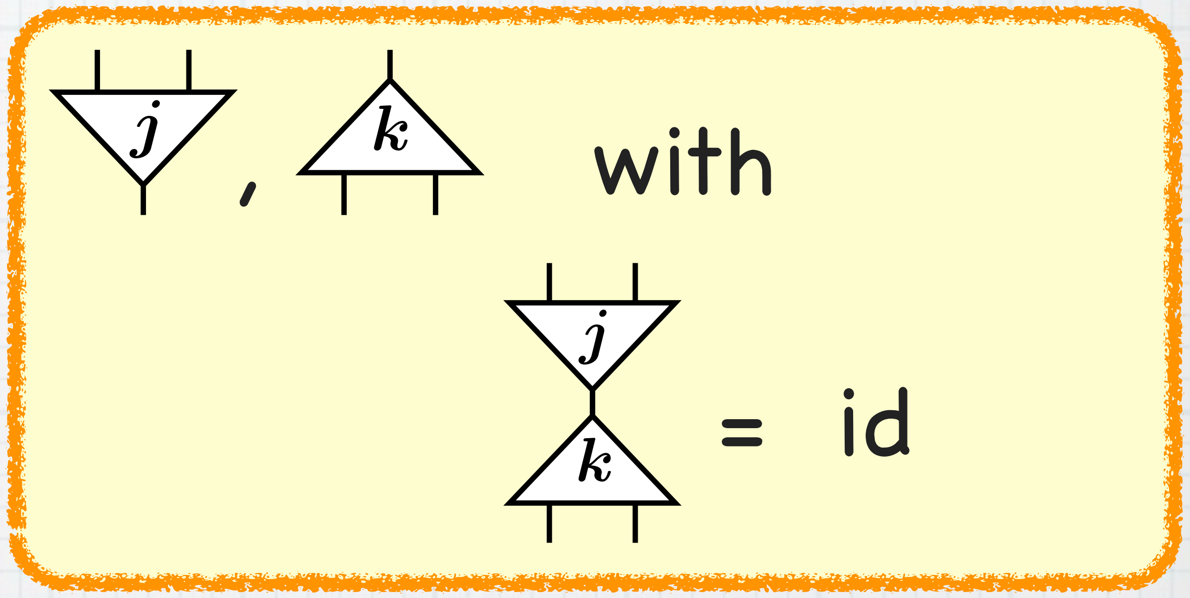
$e : FF \lhd F : e'$	Comultiplication
$d : \mathrm{id} \lhd F : d'$	Dereliction
$c : F \otimes F \lhd F : c'$	Contraction
$w : K_I \lhd F : w'$	Weakening

Here $\boldsymbol{K_I}$ is the constant functor into the monoidal unit $\boldsymbol{I}$;

- $\boldsymbol{U} \in \mathbb{C}$ is an object (called *reflexive object*), equipped with the following retractions.

$$j : U \otimes U \lhd U : k$$
$$I \lhd U$$
$$u : FU \lhd U : v$$

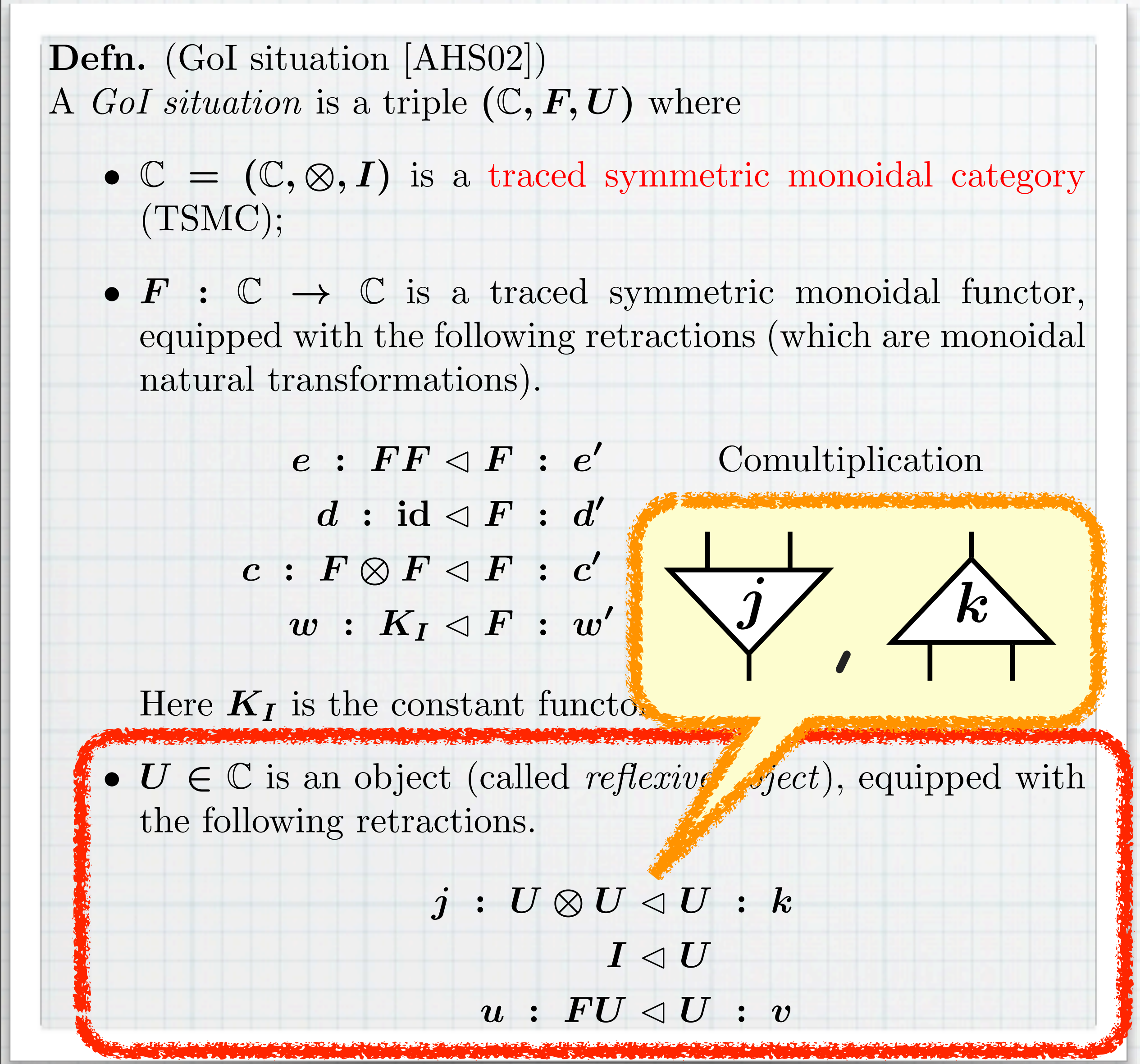
- The reflexive object U
- Retr. $$\begin{tikzcd} U \otimes U \arrow[r, bend left, "j"] & U \arrow[l, bend left, "k"] \end{tikzcd}$$



with

= id

GoI situation



Defn. (GoI situation [AHS02])
A *GoI situation* is a triple $(\mathbb{C}, F, U)$ where

- $\mathbb{C} = (\mathbb{C}, \otimes, I)$ is a traced symmetric monoidal category (TSMC);
- $F : \mathbb{C} \to \mathbb{C}$ is a traced symmetric monoidal functor, equipped with the following retractions (which are monoidal natural transformations).

$$\begin{array}{rl} e : FF \lhd F : e' & \text{Comultiplication} \\ d : \mathrm{id} \lhd F : d' & \\ c : F \otimes F \lhd F : c' & \\ w : K_I \lhd F : w' & \end{array}$$

Here K_I is the constant functo

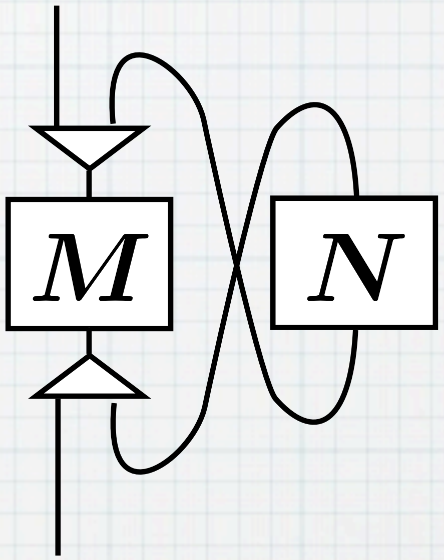
- $U \in \mathbb{C}$ is an object (called *reflexive* *ject*), equipped with the following retractions.

$$j : U \otimes U \lhd U : k$$

$$I \lhd U$$

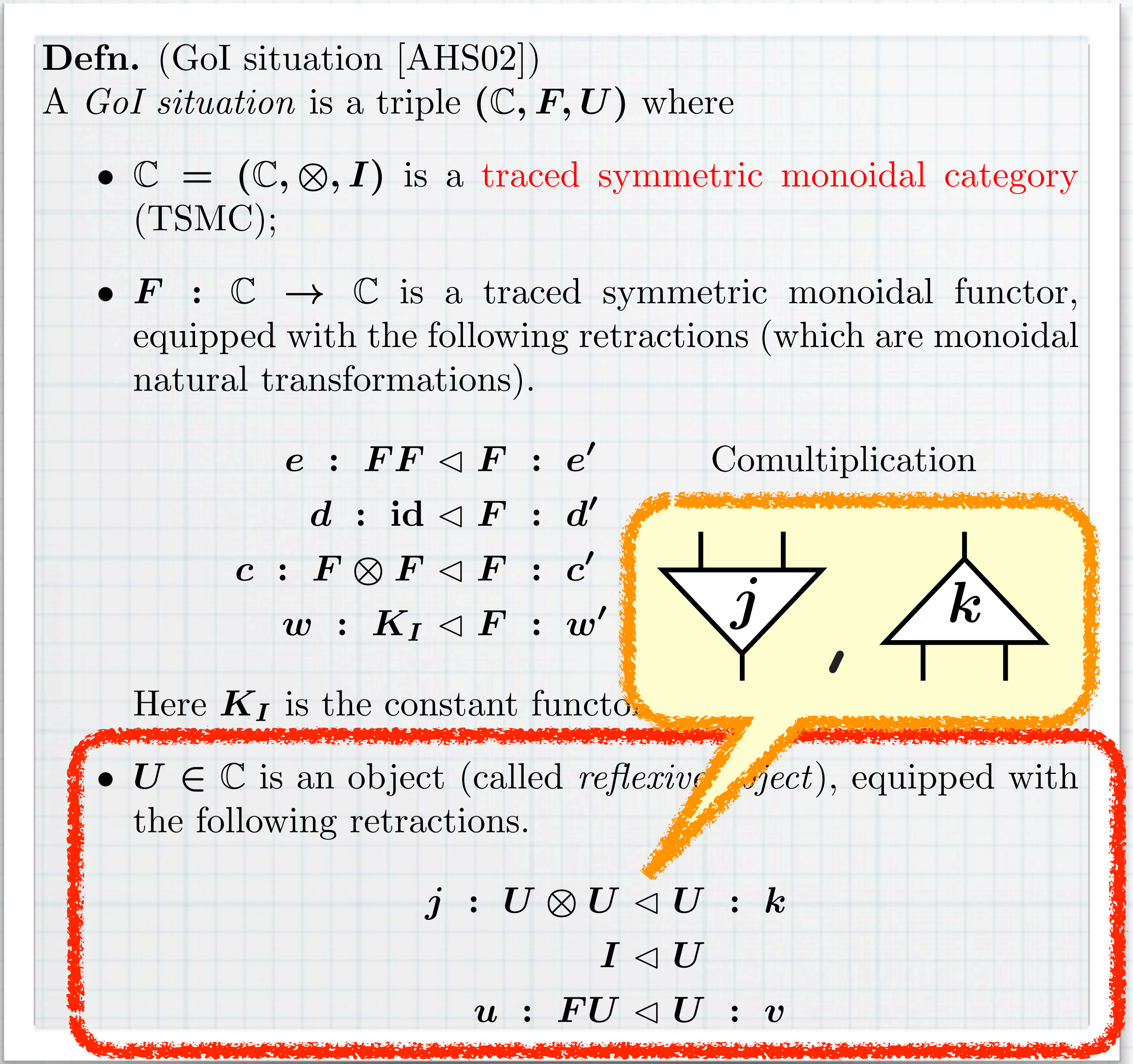
$$u : FU \lhd U : v$$

- The reflexive object U
  - Why for GoI?



  - Example in **Pfn**:

GoI situation



Defn. (GoI situation [AHS02])
A *GoI situation* is a triple $(\mathbb{C}, F, U)$ where

- $\mathbb{C} = (\mathbb{C}, \otimes, I)$ is a traced symmetric monoidal category (TSMC);
- $F : \mathbb{C} \to \mathbb{C}$ is a traced symmetric monoidal functor, equipped with the following retractions (which are monoidal natural transformations).

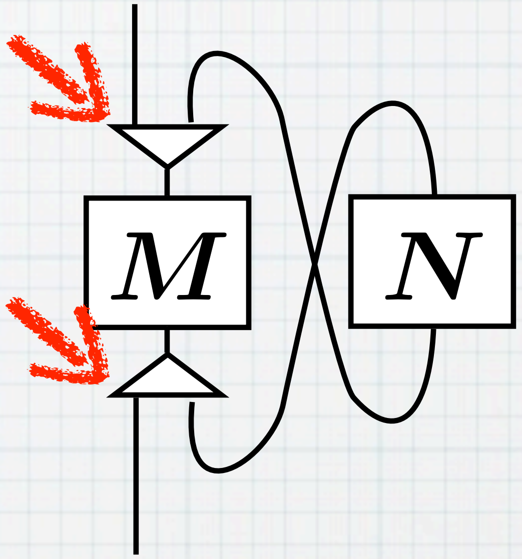
$$\begin{aligned} e &: FF \lhd F : e' \qquad \text{Comultiplication} \\ d &: \mathrm{id} \lhd F : d' \\ c &: F \otimes F \lhd F : c' \\ w &: K_I \lhd F : w' \end{aligned}$$

Here K_I is the constant functo

- $U \in \mathbb{C}$ is an object (called *reflexive ject*), equipped with the following retractions.

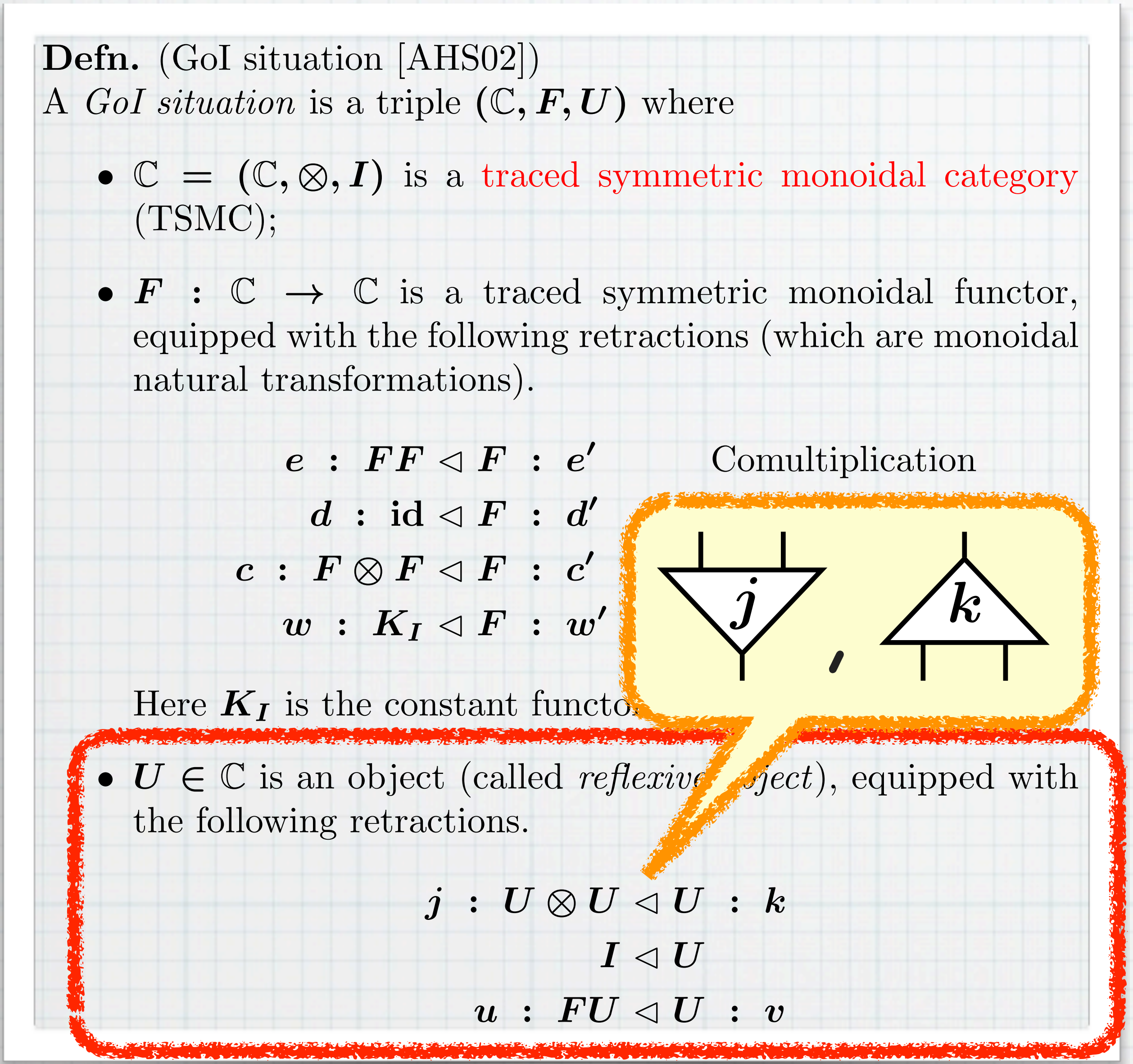
$$\begin{aligned} j : U \otimes U &\lhd U : k \\ I &\lhd U \\ u : FU &\lhd U : v \end{aligned}$$

- The reflexive object U
  - Why for GoI?



  - Example in **Pfn**:

GoI situation



Defn. (GoI situation [AHS02])
A *GoI situation* is a triple $(\mathbb{C}, F, U)$ where

- $\mathbb{C} = (\mathbb{C}, \otimes, I)$ is a traced symmetric monoidal category (TSMC);
- $F : \mathbb{C} \to \mathbb{C}$ is a traced symmetric monoidal functor, equipped with the following retractions (which are monoidal natural transformations).

$$e : FF \lhd F : e' \qquad \text{Comultiplication}$$

$$d : \mathrm{id} \lhd F : d'$$

$$c : F \otimes F \lhd F : c'$$

$$w : K_I \lhd F : w'$$

Here K_I is the constant functo

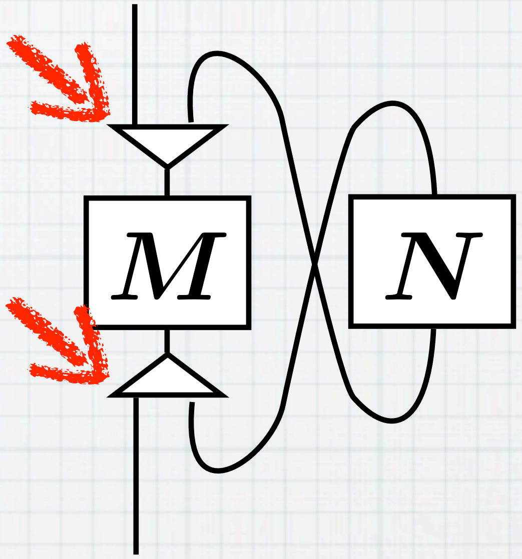
- $U \in \mathbb{C}$ is an object (called *reflexiv* *bject*), equipped with the following retractions.

$$j : U \otimes U \lhd U : k$$

$$I \lhd U$$

$$u : FU \lhd U : v$$

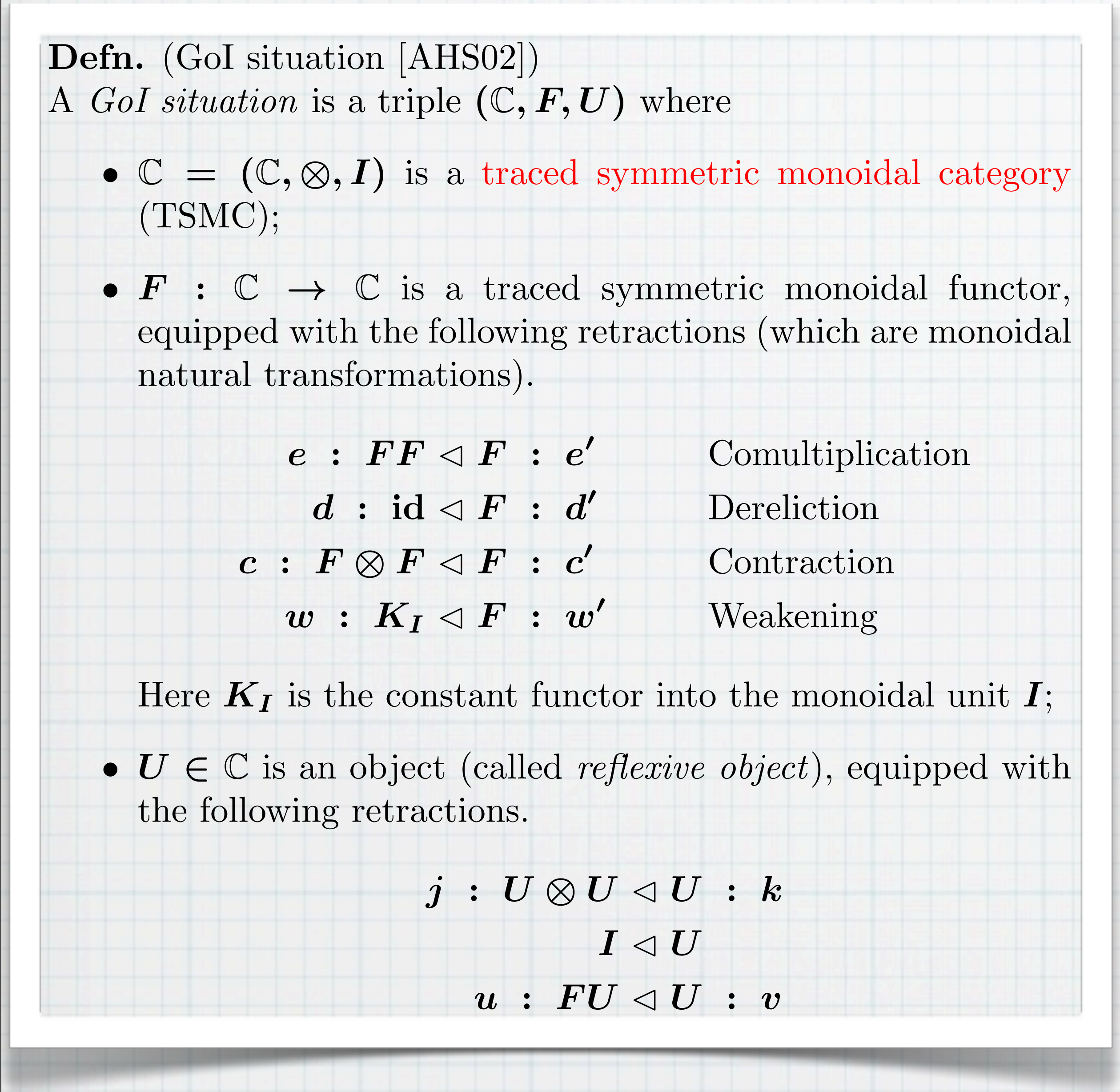
- The reflexive object U
  - Why for GoI?



  - Example in **Pfn**:

$$\mathbb{N} \in \mathbf{Pfn}, \text{ with}$$
$$\mathbb{N} + \mathbb{N} \cong \mathbb{N},$$
$$\mathbb{N} \cdot \mathbb{N} \cong \mathbb{N}$$

GoI Situation: Summary



Defn. (GoI situation [AHS02])
A *GoI situation* is a triple $(\mathbb{C}, \boldsymbol{F}, \boldsymbol{U})$ where

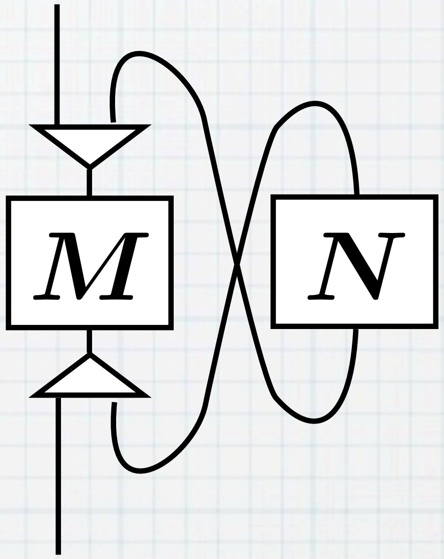
- $\mathbb{C} = (\mathbb{C}, \otimes, \boldsymbol{I})$ is a traced symmetric monoidal category (TSMC);
- $\boldsymbol{F} : \mathbb{C} \to \mathbb{C}$ is a traced symmetric monoidal functor, equipped with the following retractions (which are monoidal natural transformations).

$\boldsymbol{e} : \boldsymbol{F}\boldsymbol{F} \lhd \boldsymbol{F} : \boldsymbol{e}'$	Comultiplication
$\boldsymbol{d} : \mathbf{id} \lhd \boldsymbol{F} : \boldsymbol{d}'$	Dereliction
$\boldsymbol{c} : \boldsymbol{F} \otimes \boldsymbol{F} \lhd \boldsymbol{F} : \boldsymbol{c}'$	Contraction
$\boldsymbol{w} : \boldsymbol{K_I} \lhd \boldsymbol{F} : \boldsymbol{w}'$	Weakening

  Here $\boldsymbol{K_I}$ is the constant functor into the monoidal unit $\boldsymbol{I}$;
- $\boldsymbol{U} \in \mathbb{C}$ is an object (called *reflexive object*), equipped with the following retractions.

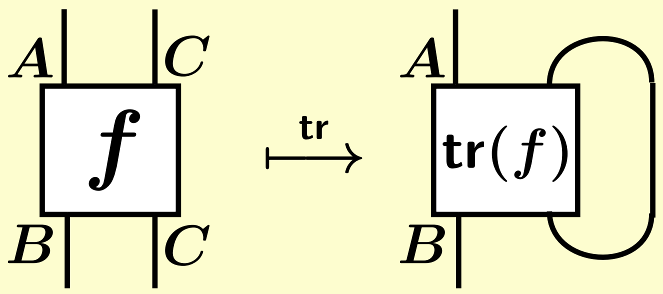
$$\boldsymbol{j} : \boldsymbol{U} \otimes \boldsymbol{U} \lhd \boldsymbol{U} : \boldsymbol{k}$$
$$\boldsymbol{I} \lhd \boldsymbol{U}$$
$$\boldsymbol{u} : \boldsymbol{F}\boldsymbol{U} \lhd \boldsymbol{U} : \boldsymbol{v}$$

- Categorical axiomatics of the "GoI animation"



- Example:

$$(\mathbf{Pfn}, \mathbb{N} \cdot \_\,, \mathbb{N})$$



tuation: Summary

Defn. (GoI situation [AHS02])
A *GoI situation* is a triple $(\mathbb{C}, F, U)$ w e

- $\mathbb{C} = (\mathbb{C}, \otimes, I)$ is a traced symmetric monoidal category (TSMC);
- $F : \mathbb{C} \to \mathbb{C}$ is a traced symmetric monoidal functor, equipped with the following retractions (which are monoidal natural transformations).

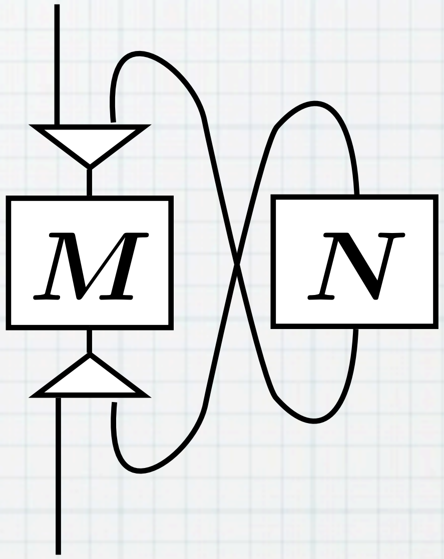
$$\begin{array}{rll} e : FF \lhd F : e' & & \text{Comultiplication} \\ d : \mathrm{id} \lhd F : d' & & \text{Dereliction} \\ c : F \otimes F \lhd F : c' & & \text{Contraction} \\ w : K_I \lhd F : w' & & \text{Weakening} \end{array}$$

  Here K_I is the constant functor into the monoidal unit I;

- $U \in \mathbb{C}$ is an object (called *reflexive object*), equipped with the following retractions.

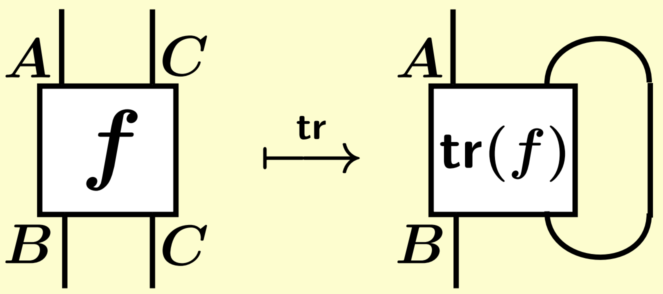
$$\begin{array}{c} j : U \otimes U \lhd U : k \\ I \lhd U \\ u : FU \lhd U : v \end{array}$$

- Categorical axiomatics of the "GoI animation"

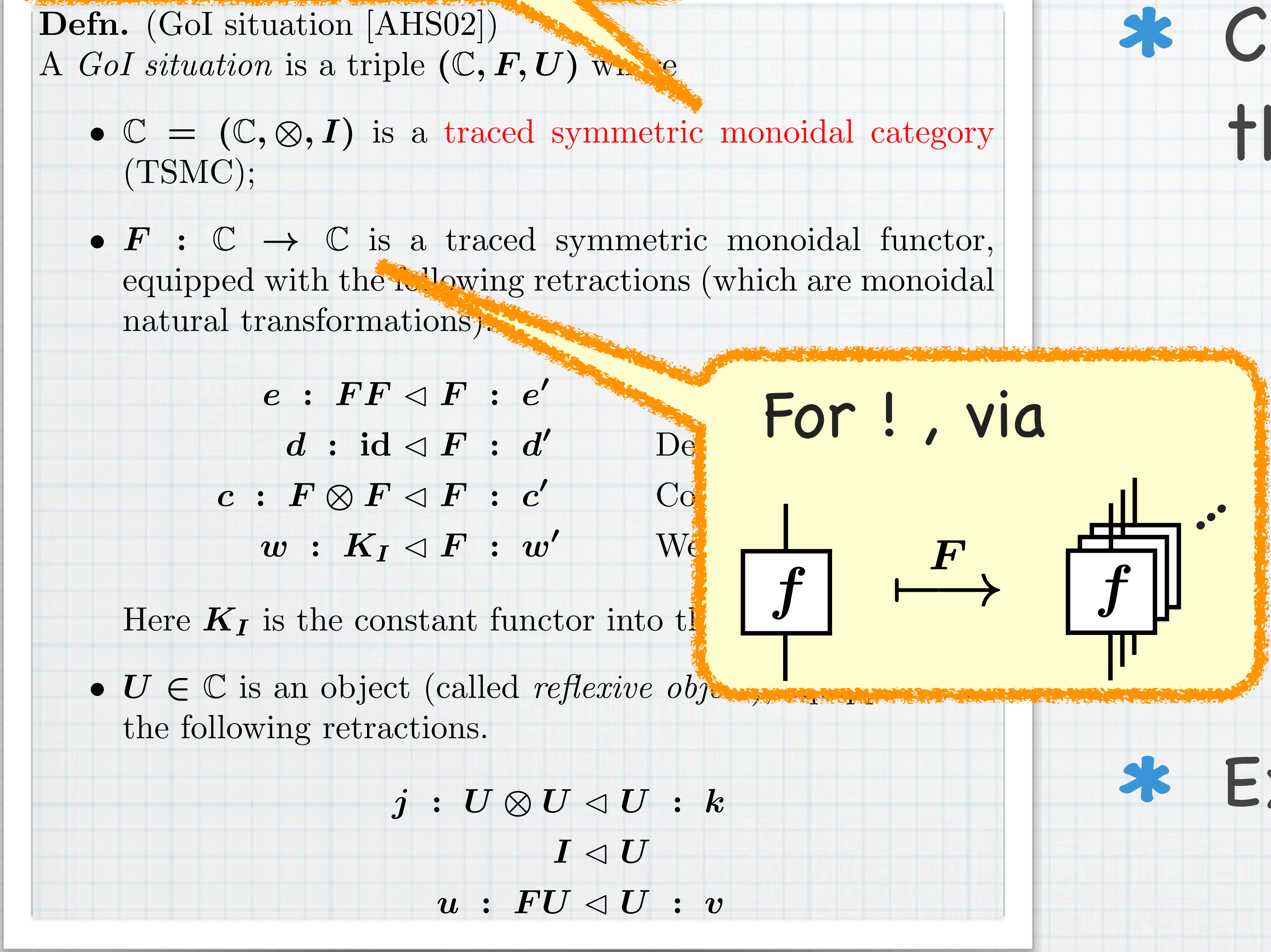


- Example:

$$(\mathbf{Pfn}, \mathbb{N} \cdot \_\,, \mathbb{N})$$



tuation: Summary



Defn. (GoI situation [AHS02])
A *GoI situation* is a triple $(\mathbb{C}, \boldsymbol{F}, \boldsymbol{U})$ where

- $\mathbb{C} = (\mathbb{C}, \otimes, \boldsymbol{I})$ is a traced symmetric monoidal category (TSMC);
- $\boldsymbol{F} : \mathbb{C} \to \mathbb{C}$ is a traced symmetric monoidal functor, equipped with the following retractions (which are monoidal natural transformations):

$$\begin{aligned} \boldsymbol{e} &: \boldsymbol{FF} \lhd \boldsymbol{F} : \boldsymbol{e}' \\ \boldsymbol{d} &: \mathbf{id} \lhd \boldsymbol{F} : \boldsymbol{d}' \\ \boldsymbol{c} &: \boldsymbol{F} \otimes \boldsymbol{F} \lhd \boldsymbol{F} : \boldsymbol{c}' \\ \boldsymbol{w} &: \boldsymbol{K_I} \lhd \boldsymbol{F} : \boldsymbol{w}' \end{aligned}$$

De

Co

We

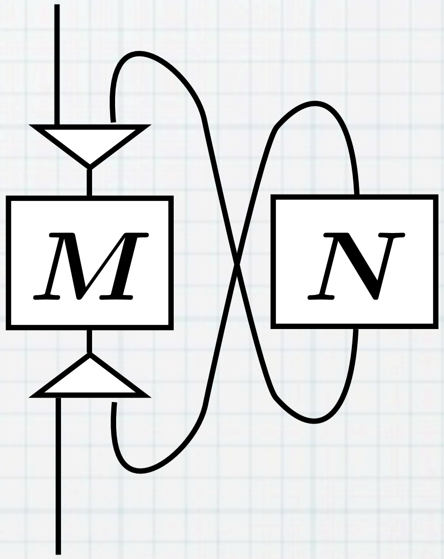
Here $\boldsymbol{K_I}$ is the constant functor into th

- $\boldsymbol{U} \in \mathbb{C}$ is an object (called *reflexive obj* the following retractions.

$$\begin{aligned} \boldsymbol{j} : \boldsymbol{U} \otimes \boldsymbol{U} &\lhd \boldsymbol{U} : \boldsymbol{k} \\ \boldsymbol{I} &\lhd \boldsymbol{U} \\ \boldsymbol{u} : \boldsymbol{FU} &\lhd \boldsymbol{U} : \boldsymbol{v} \end{aligned}$$

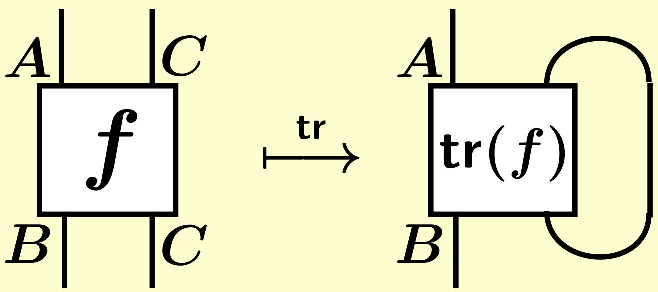
For ! , via

- Categorical axiomatics of the "GoI animation"



- Example:

$$(\mathbf{Pfn}, \mathbb{N} \cdot \_\_, \mathbb{N})$$



tuation: Summary

Defn. (GoI situation [AHS02])
A *GoI situation* is a triple $(\mathbb{C}, \boldsymbol{F}, \boldsymbol{U})$ w e

- $\mathbb{C} = (\mathbb{C}, \otimes, \boldsymbol{I})$ is a traced symmetric monoidal category (TSMC);
- $\boldsymbol{F} : \mathbb{C} \to \mathbb{C}$ is a traced symmetric monoidal functor, equipped with the following retractions (which are monoidal natural transformations).

$$\boldsymbol{e} : \boldsymbol{FF} \lhd \boldsymbol{F} : \boldsymbol{e}'$$

$$\boldsymbol{d} : \mathbf{id} \lhd \boldsymbol{F} : \boldsymbol{d}' \qquad \text{De}$$

$$\boldsymbol{c} : \boldsymbol{F} \otimes \boldsymbol{F} \lhd \boldsymbol{F} : \boldsymbol{c}' \qquad \text{Co}$$

$$\boldsymbol{w} : \boldsymbol{K_I} \lhd \boldsymbol{F} : \boldsymbol{w}' \qquad \text{We}$$

Here $\boldsymbol{K_I}$ is the constant functor into th

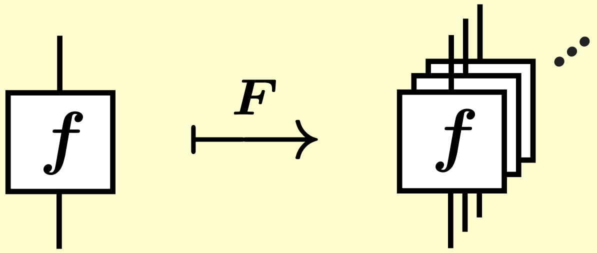
- $\boldsymbol{U} \in \mathbb{C}$ is an object (called *reflexive obj* the following retractions.

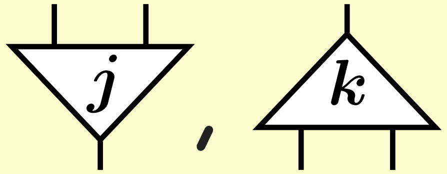
$$\boldsymbol{j} : \boldsymbol{U} \otimes \boldsymbol{U} \lhd \boldsymbol{U} : \boldsymbol{k}$$

$$\boldsymbol{I} \lhd \boldsymbol{U}$$

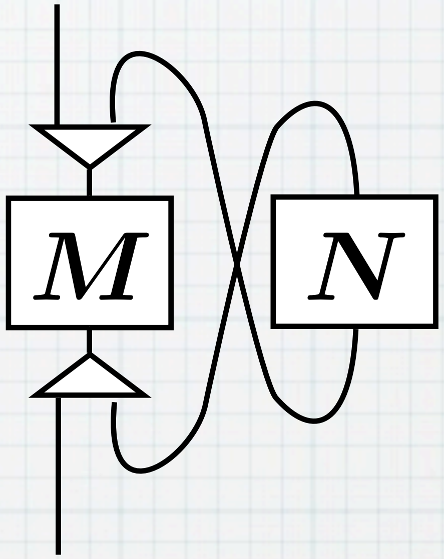
$$\boldsymbol{u} : \boldsymbol{FU} \lhd \boldsymbol{U} : \boldsymbol{v}$$

For ! , via





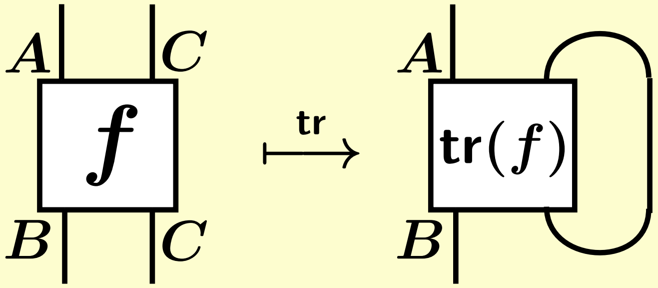
- Categorical axiomatics of the "GoI animation"

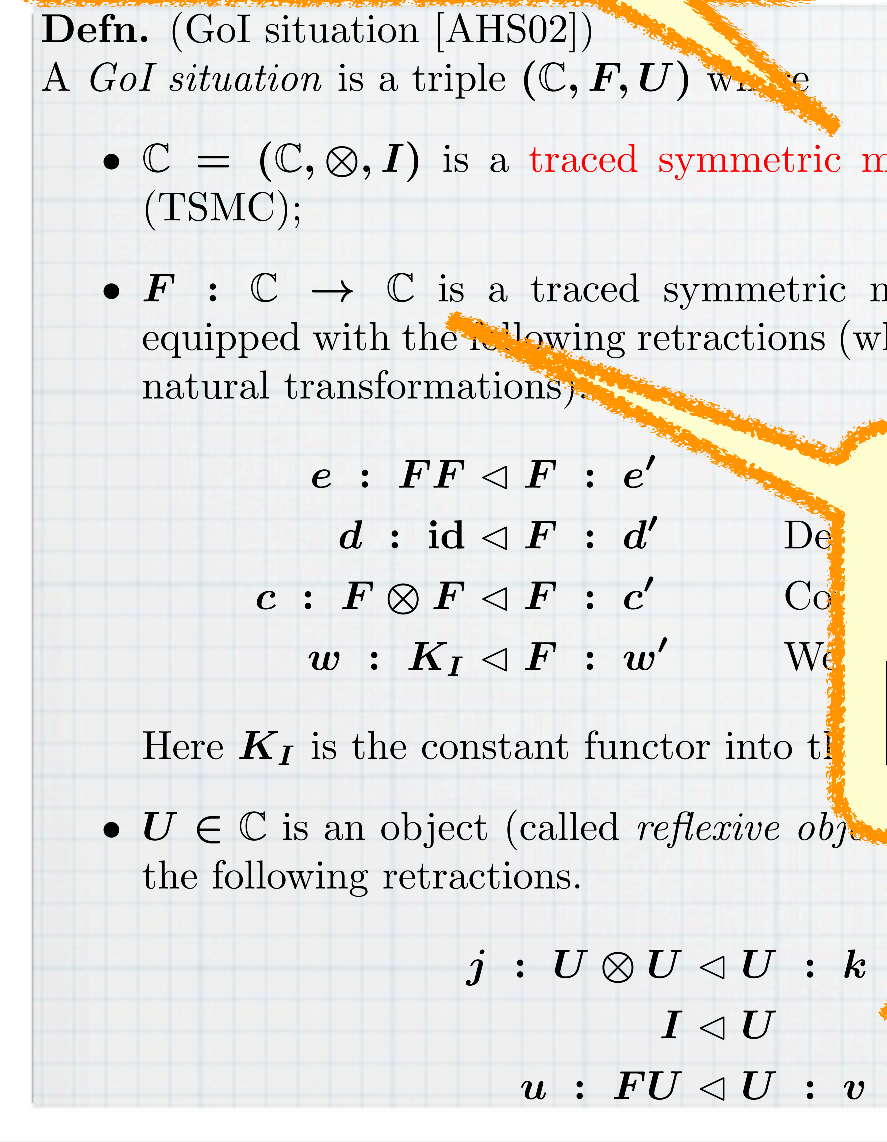


- Example:

$(\mathbf{Pfn}, \mathbb{N} \cdot \_\, , \mathbb{N})$

tuation: Summary





Defn. (GoI situation [AHS02])
A *GoI situation* is a triple $(\mathbb{C}, F, U)$ where

- $\mathbb{C} = (\mathbb{C}, \otimes, I)$ is a traced symmetric monoidal category (TSMC);
- $F : \mathbb{C} \to \mathbb{C}$ is a traced symmetric monoidal functor, equipped with the following retractions (which are monoidal natural transformations):

$e : FF \lhd F : e'$ De

$d : \mathrm{id} \lhd F : d'$ De

$c : F \otimes F \lhd F : c'$ Co

$w : K_I \lhd F : w'$ We

Here K_I is the constant functor into th

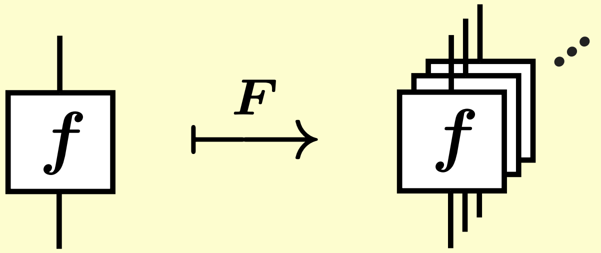
- $U \in \mathbb{C}$ is an object (called *reflexive obj* the following retractions.

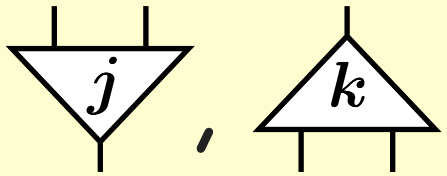
$j : U \otimes U \lhd U : k$

$I \lhd U$

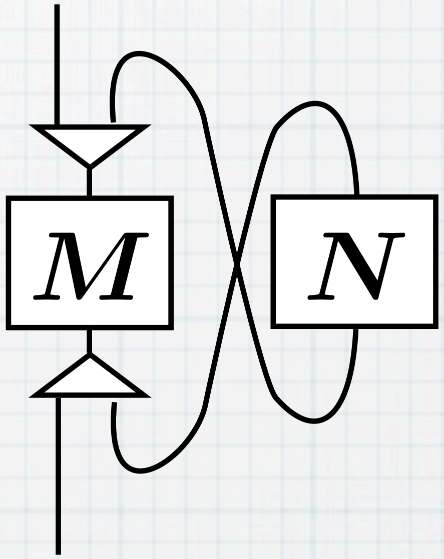
$u : FU \lhd U : v$

For ! , via



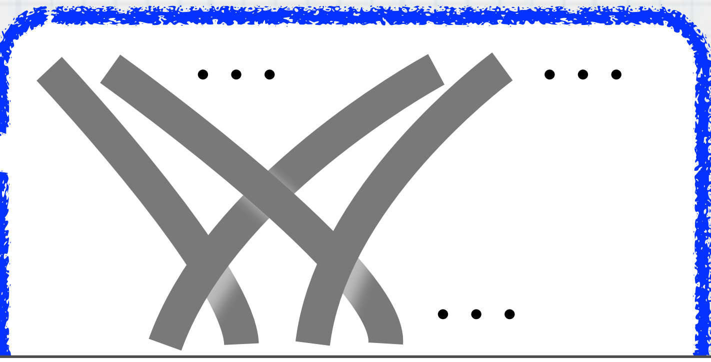


- Categorical axiomatics of the "GoI animation"



- Example:

$(\mathbf{Pfn}, \mathbb{N} \cdot \_\,, \mathbb{N})$



Categorical GoI: Constr. of an LCA

Thm. ([AHS02])
Given a GoI situation $(\mathbb{C}, \boldsymbol{F}, \boldsymbol{U})$, the homset

$$\mathbb{C}(\boldsymbol{U}, \boldsymbol{U})$$

carries a canonical LCA structure.

Categorical GoI: Constr. of an LCA

Thm. ([AHS02])
Given a GoI situation $(\mathbb{C}, \boldsymbol{F}, \boldsymbol{U})$, the homset

$$\mathbb{C}(\boldsymbol{U}, \boldsymbol{U})$$

carries a canonical LCA structure.

- Applicative str. $\cdot$
- ! operator
- Combinators B, C, I, ...

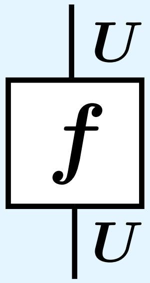
Categorical GoI: Constr. of an LCA

Thm. ([AHS02])
Given a GoI situation $(\mathbb{C}, F, U)$, the homset

$$\mathbb{C}(U, U)$$

carries a canonical LCA structure.

- Applicative str. $\cdot$
- ! operator
- Combinators B, C, I, ...



$\in \mathbb{C}(U, U)$

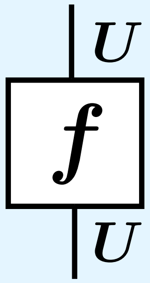
Categorical GoI: Constr. of an LCA

Thm. ([AHS02])
Given a GoI situation $(\mathbb{C}, F, U)$, the homset

$$\mathbb{C}(U, U)$$

carries a canonical LCA structure.

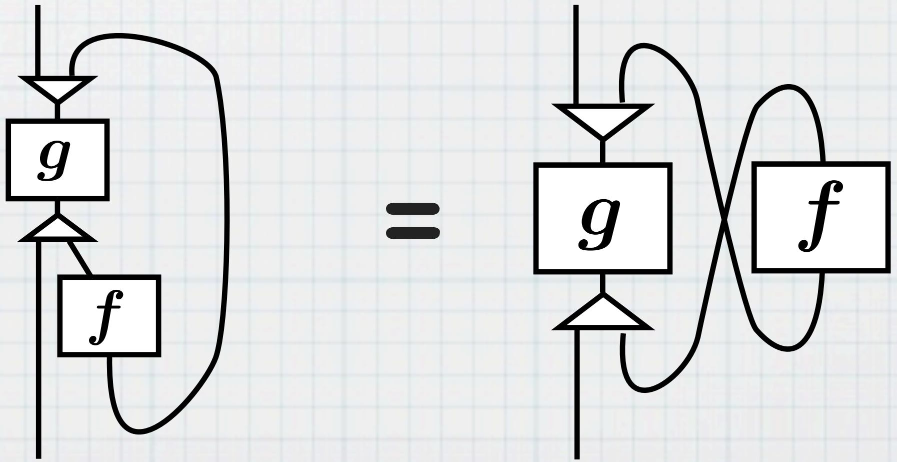
- Applicative str. $\cdot$
- ! operator
- Combinators B, C, I, ...



$\in \mathbb{C}(U, U)$

- $g \cdot f$

$$:= \mathbf{tr}\big((U \otimes f) \circ k \circ g \circ j \big)$$

$=$  $=$

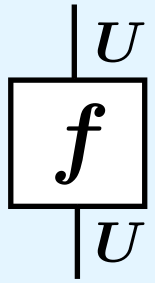
Categorical GoI: Constr. of an LCA

Thm. ([AHS02])
Given a GoI situation $(\mathbb{C}, \boldsymbol{F}, \boldsymbol{U})$, the homset

$$\mathbb{C}(\boldsymbol{U}, \boldsymbol{U})$$

carries a canonical LCA structure.

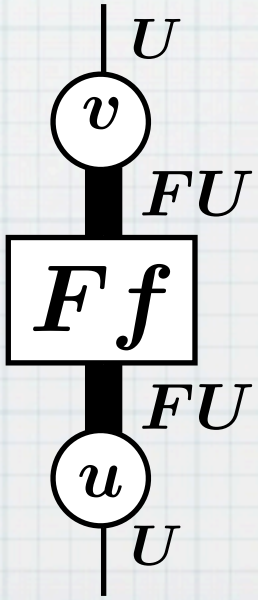
- Applicative str. $\cdot$
- ! operator
- Combinators B, C, I, ...



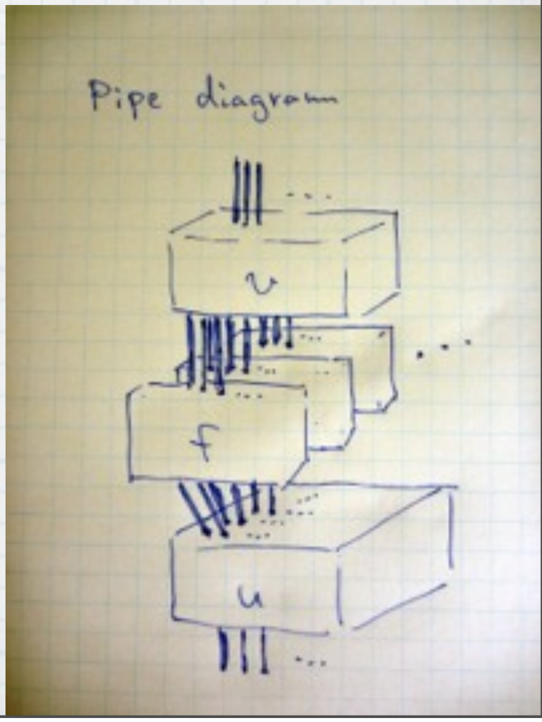
$\in \mathbb{C}(\boldsymbol{U}, \boldsymbol{U})$

- $!\boldsymbol{f} \;:=\; \boldsymbol{u} \circ \boldsymbol{F}\boldsymbol{f} \circ \boldsymbol{v}$

$=$



$=$



Categorical GoI: Constr. of an LCA

- Combinator $Bxyz = x(yz)$

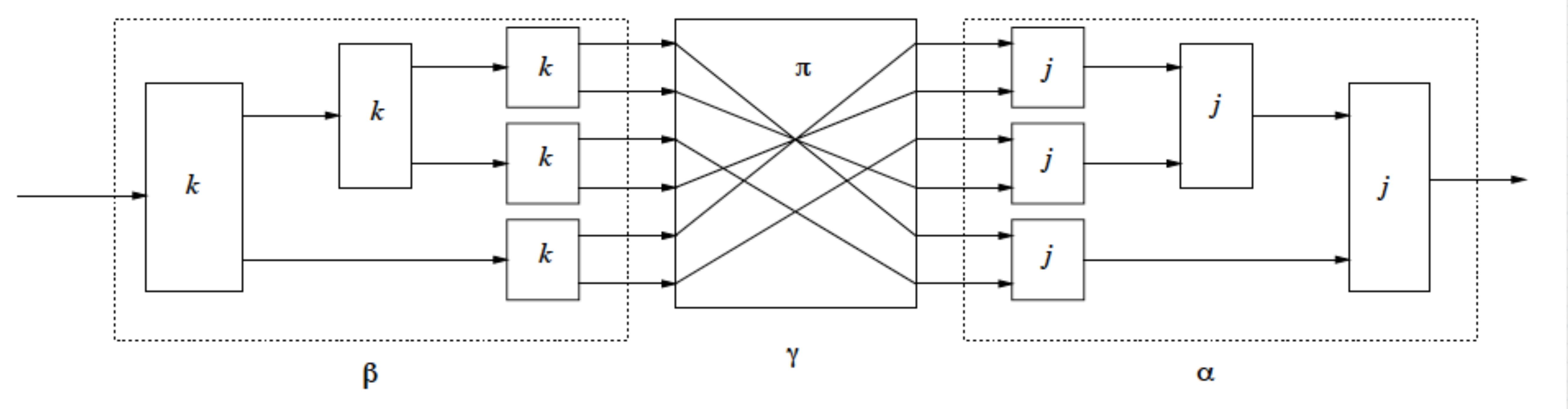
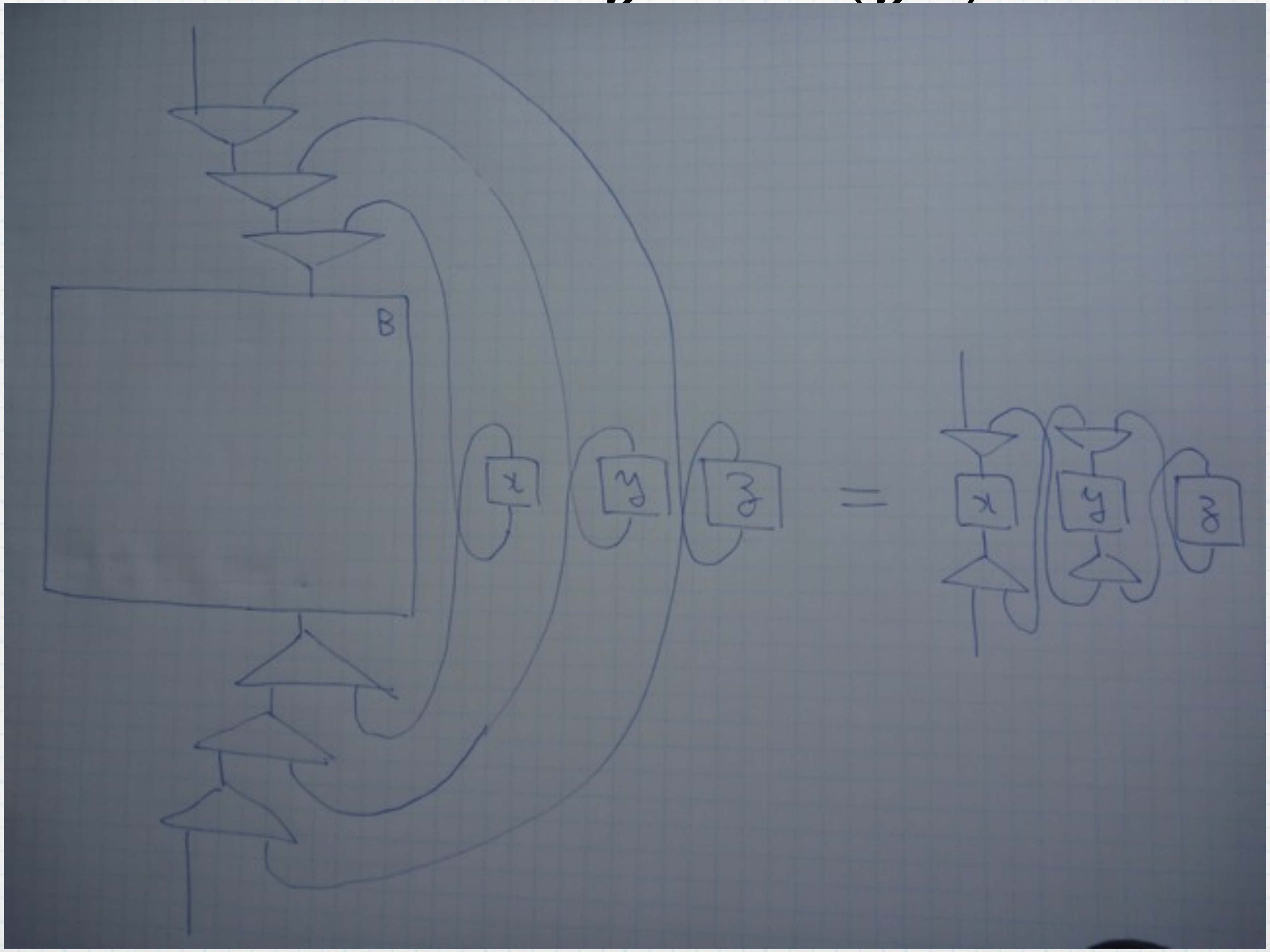


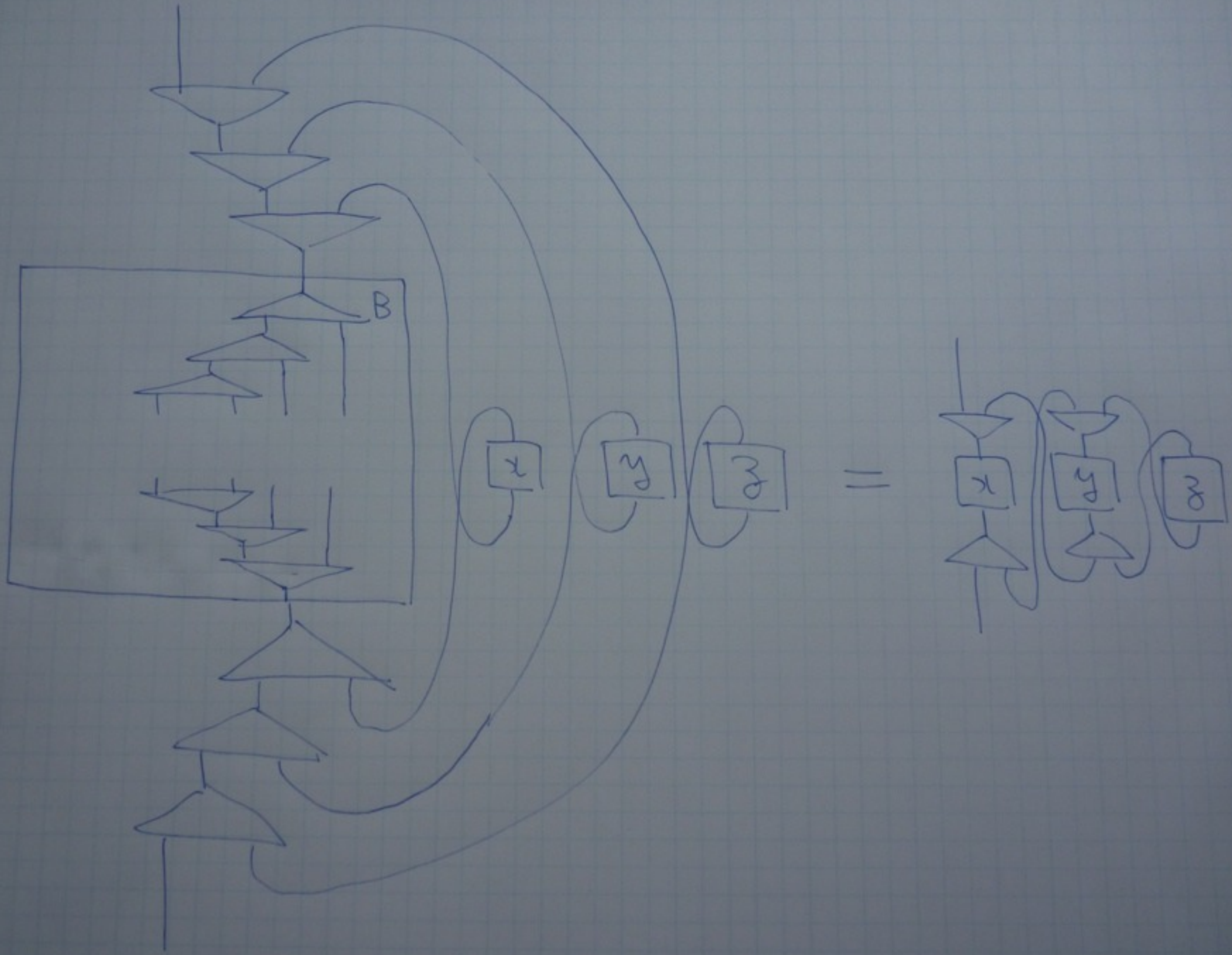
Figure 7: Composition Combinator B

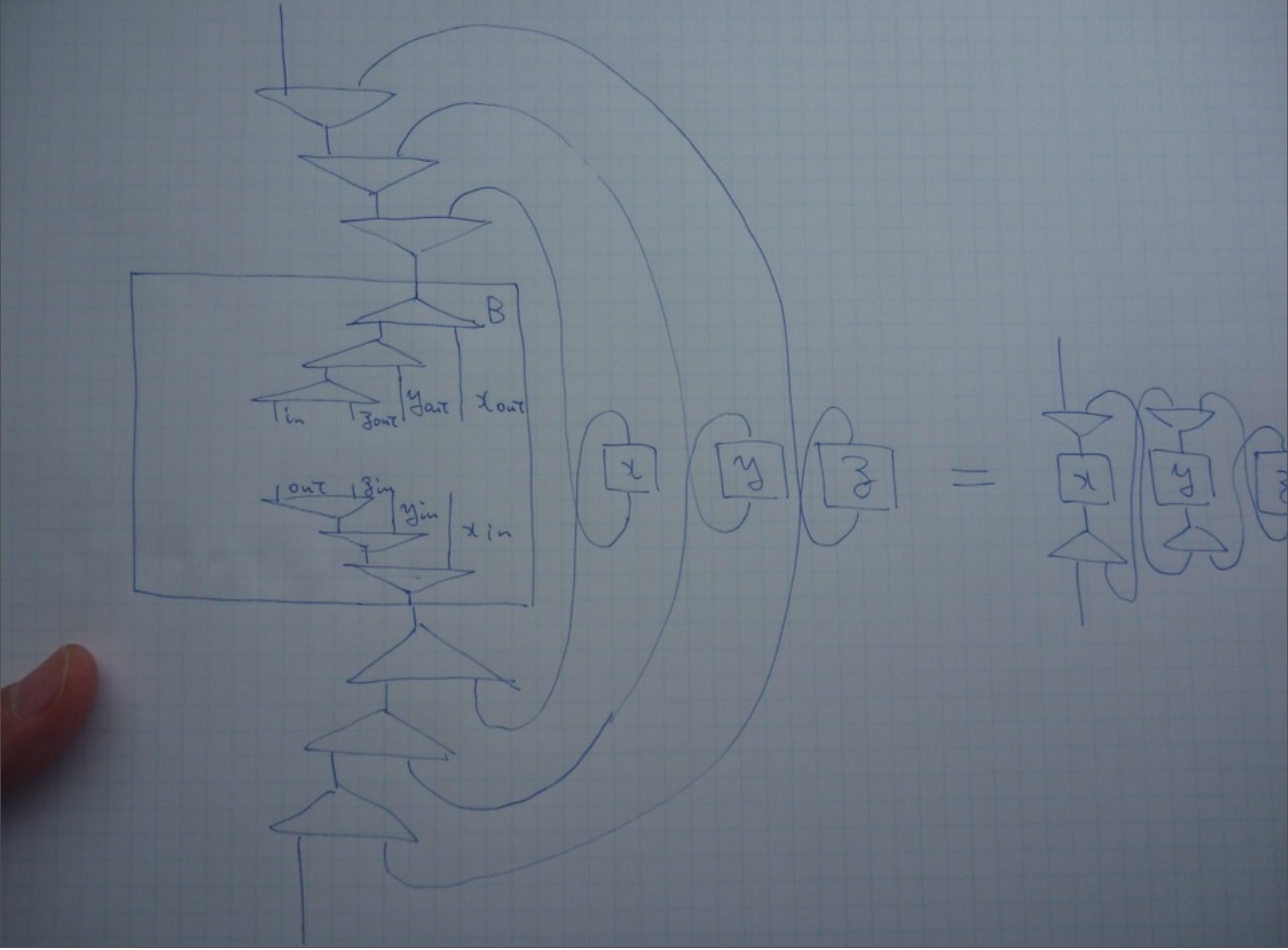
from [AHS02]

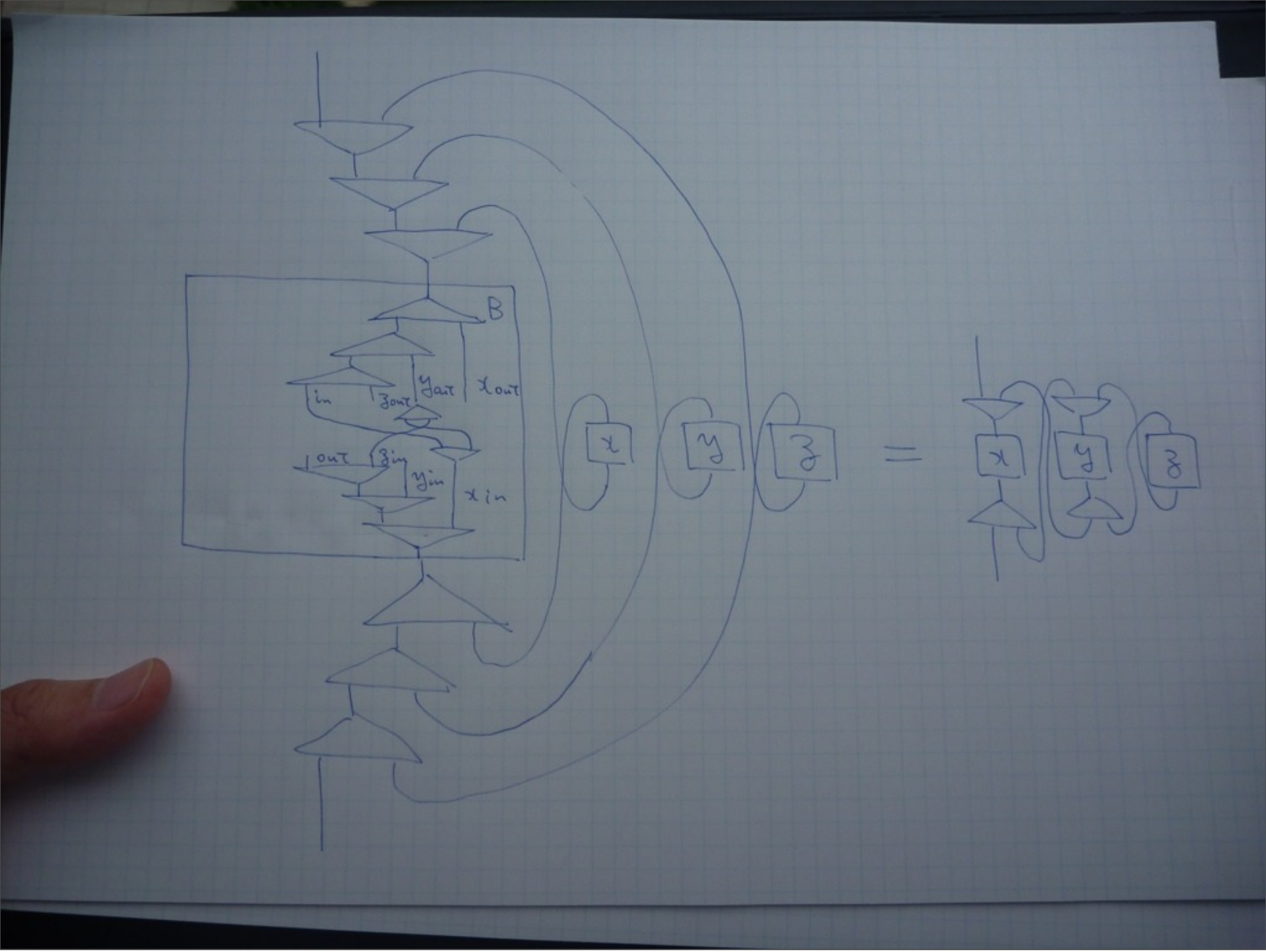
Categorical GoI: Constr. of an LCA

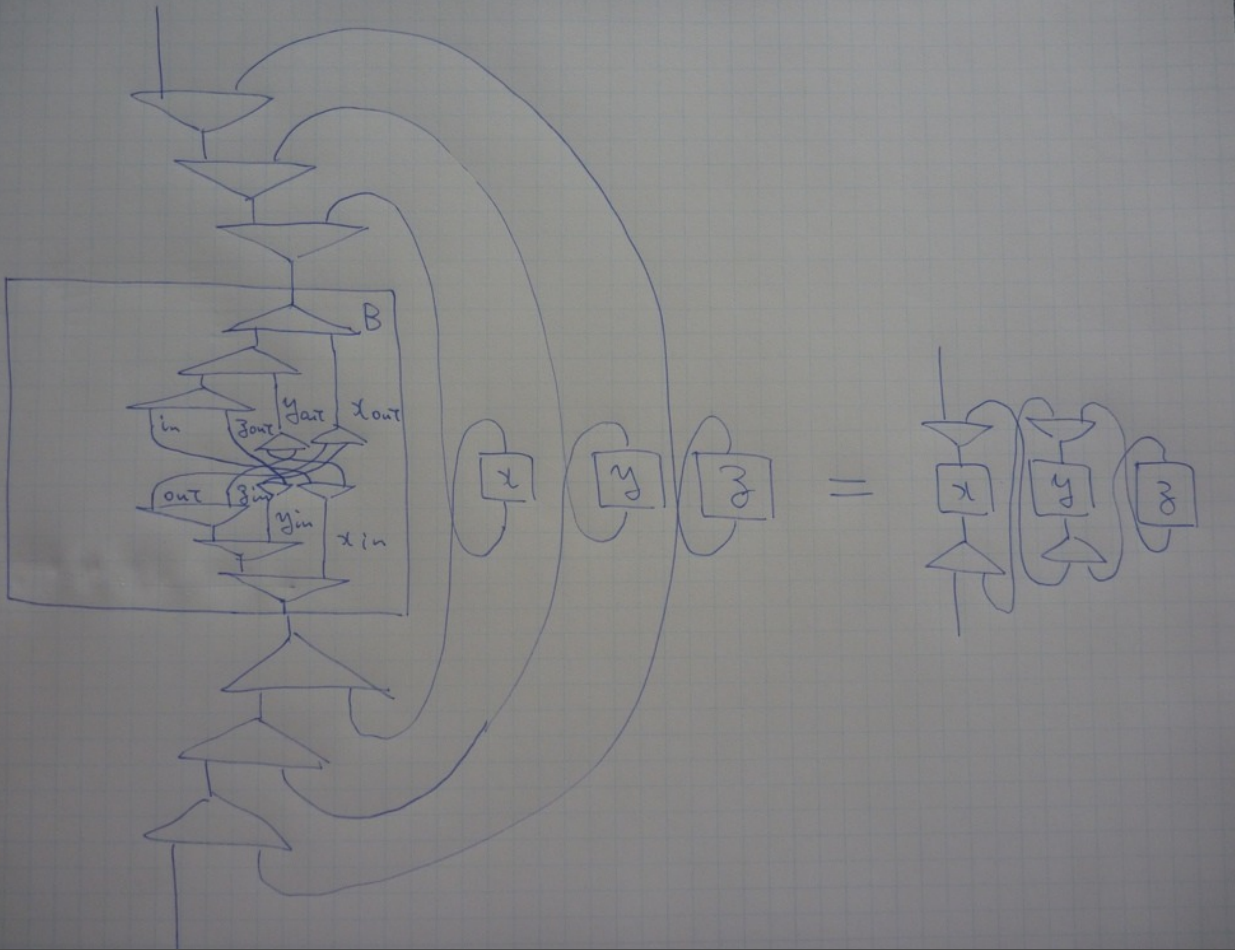
- Combinator $Bxyz = x(yz)$




B
x
y
z
=
x
y
z


B
in
z out
y out
x out
out
z in
y in
x in
x
y
z
=
x
y
z


B
in
z out
y out
x out
out
z in
y in
x in
x
y
z
=
x
y
z


B
in
zout
yout
xout
out
zin
yin
xin
x
y
z
=
x
y
z

Categorical GoI: Constr. of an LCA

- Combinator $Bxyz = x(yz)$

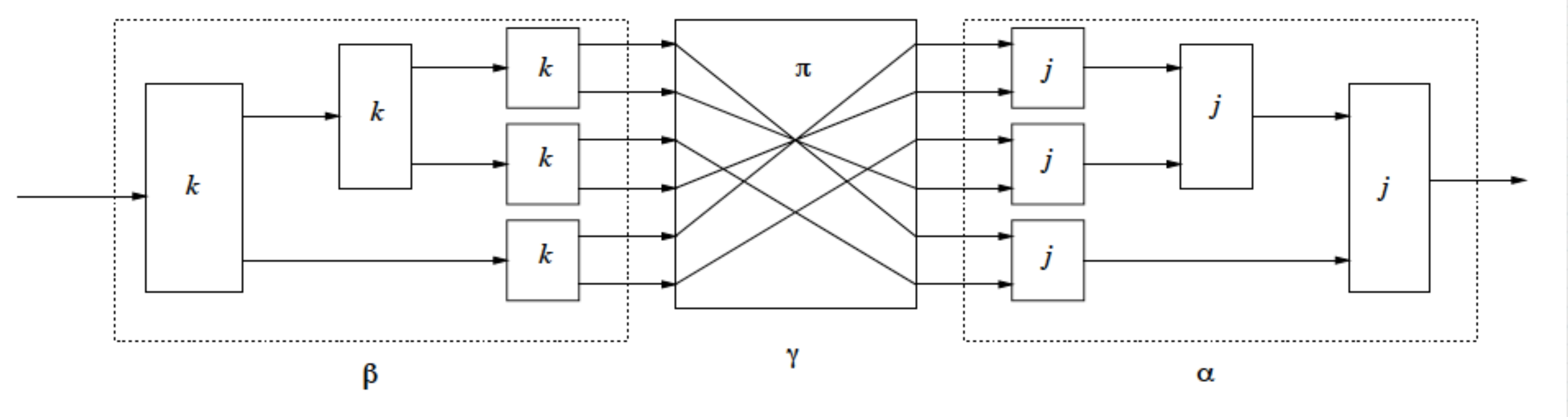


Figure 7: Composition Combinator B

from [AHS02]

Categorical GoI: Constr. of an LCA

- Combinator $Bxyz = x(yz)$

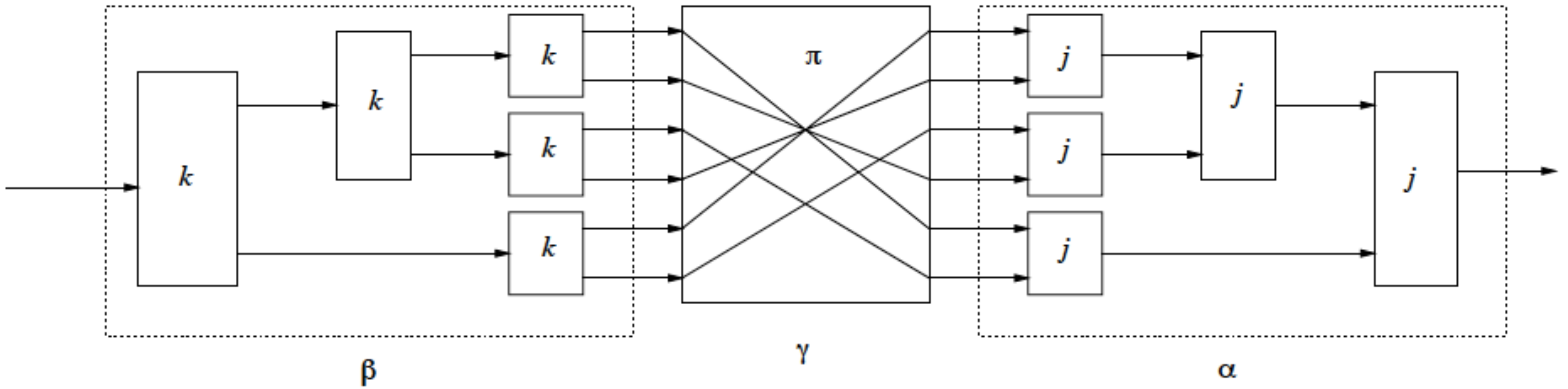


Figure 7: Composition Combinator B

from [AHS02]

Nice dynamic interpretation of (linear) computation!!

Summary: Categorical GoI

Defn. (GoI situation [AHS02])
A *GoI situation* is a triple $(\mathbb{C}, \boldsymbol{F}, \boldsymbol{U})$ where

- $\mathbb{C} = (\mathbb{C}, \otimes, \boldsymbol{I})$ is a traced symmetric monoidal category (TSMC);
- $\boldsymbol{F} : \mathbb{C} \to \mathbb{C}$ is a traced symmetric monoidal functor, equipped with the following retractions (which are monoidal natural transformations).

$$\begin{array}{rcll} \boldsymbol{e} : \boldsymbol{FF} & \lhd & \boldsymbol{F} : \boldsymbol{e'} & \text{Comultiplication} \\ \boldsymbol{d} : \mathbf{id} & \lhd & \boldsymbol{F} : \boldsymbol{d'} & \text{Dereliction} \\ \boldsymbol{c} : \boldsymbol{F} \otimes \boldsymbol{F} & \lhd & \boldsymbol{F} : \boldsymbol{c'} & \text{Contraction} \\ \boldsymbol{w} : \boldsymbol{K_I} & \lhd & \boldsymbol{F} : \boldsymbol{w'} & \text{Weakening} \end{array}$$

  Here $\boldsymbol{K_I}$ is the constant functor into the monoidal unit $\boldsymbol{I}$;
- $\boldsymbol{U} \in \mathbb{C}$ is an object (called *reflexive object*), equipped with the following retractions.

$$\begin{array}{c} \boldsymbol{j} : \boldsymbol{U} \otimes \boldsymbol{U} \lhd \boldsymbol{U} : \boldsymbol{k} \\ \boldsymbol{I} \lhd \boldsymbol{U} \\ \boldsymbol{u} : \boldsymbol{FU} \lhd \boldsymbol{U} : \boldsymbol{v} \end{array}$$

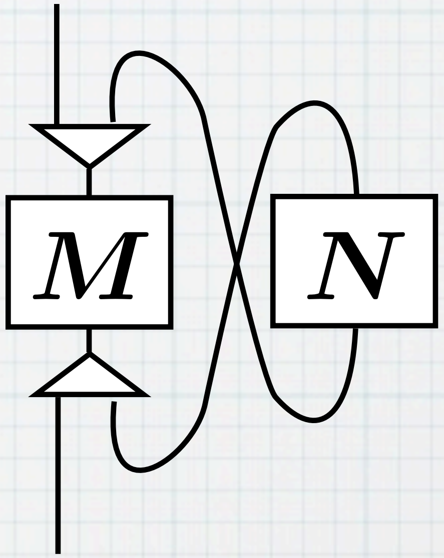
Thm. ([AHS02])
Given a GoI situation $(\mathbb{C}, \boldsymbol{F}, \boldsymbol{U})$, the homset

$$\mathbb{C}(\boldsymbol{U}, \boldsymbol{U})$$

carries a canonical LCA structure.

Why Categorical Generalization?: Examples Other Than Pfn [AHS02]

- Strategy: find a TSMC!

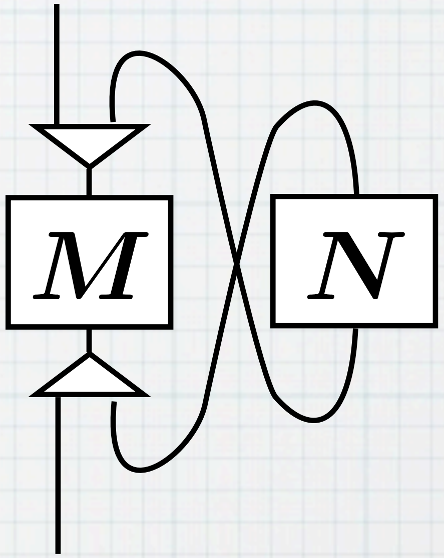


- "Wave-style" examples
  - ⊗ is Cartesian product(-like)
    - in which case,

      trace ≈ fixed point operator [Hasegawa/Hyland]
  - An example: $\big(\,(\omega\text{-}\mathbf{Cpo}, \times, \mathbf{1}),\ (\_)^{\mathbb{N}},\ A^{\mathbb{N}}\,\big)$
  - (... less of a dynamic flavor)

Why Categorical Generalization?: Examples Other Than $\mathbf{Pfn}$ [AHS02]

- "Particle-style" examples
  - Obj. X∈C is set-like; ⊗ is coproduct-like
    - The GoI animation is valid



  - Examples:
    - Partial functions $\big(\,(\mathbf{Pfn},+,\mathbf{0}),\ \mathbb{N}\cdot\_\,,\ \mathbb{N}\,\big)$
    - Binary relations $\big(\,(\mathbf{Rel},+,\mathbf{0}),\ \mathbb{N}\cdot\_\,,\ \mathbb{N}\,\big)$
    - "Discrete stochastic relations" $\big(\,(\mathbf{DSRel},+,\mathbf{0}),\ \mathbb{N}\cdot\_\,,\ \mathbb{N}\,\big)$

Why Categorical Generalization?: Examples Other Than Pfn [AHS02]

- **Pfn** (partial functions)

$$\begin{array}{c} X \to Y \text{ in } \mathbf{Pfn} \\ \hline\hline X \rightharpoonup Y, \text{ partial function} \\ \hline\hline X \to \mathcal{L}Y \text{ in } \mathbf{Sets} \end{array} \quad \text{where } \mathcal{L}Y = \{\bot\} + Y$$

- **Rel** (relations)

$$\begin{array}{c} X \to Y \text{ in } \mathbf{Rel} \\ \hline\hline R \subseteq X \times Y, \text{ relation} \\ \hline\hline X \to \mathcal{P}Y \text{ in } \mathbf{Sets} \end{array} \quad \text{where } \mathcal{P} \text{ is the powerset monad}$$

- **DSRel**

$$\begin{array}{c} X \to Y \text{ in } \mathbf{DSRel} \\ \hline\hline X \to \mathcal{D}Y \text{ in } \mathbf{Sets} \end{array}$$

$$\text{where } \mathcal{D}Y = \{d : Y \to [0,1] \mid \sum_{y} d(y) \leq 1\}$$

Why Catego
Examples Other Than Pfn [AHS02]

Categories of sets and
(functions with different branching/partiality)

- **Pfn** (partial functions)

$$\begin{array}{c} X \to Y \text{ in } \mathbf{Pfn} \\ \hline\hline X \rightharpoonup Y, \text{ partial function} \\ \hline\hline X \to \mathcal{L}Y \text{ in } \mathbf{Sets} \end{array} \quad \text{where } \mathcal{L}Y = \{\bot\} + Y$$

- **Rel** (relations)

$$\begin{array}{c} X \to Y \text{ in } \mathbf{Rel} \\ \hline\hline R \subseteq X \times Y, \text{ relation} \\ \hline\hline X \to \mathcal{P}Y \text{ in } \mathbf{Sets} \end{array} \quad \text{where } \mathcal{P} \text{ is the powerset monad}$$

- **DSRel**

$$\begin{array}{c} X \to Y \text{ in } \mathbf{DSRel} \\ \hline\hline X \to \mathcal{D}Y \text{ in } \mathbf{Sets} \end{array}$$

$$\text{where } \mathcal{D}Y = \{d : Y \to [0,1] \mid \sum_{y} d(y) \leq 1\}$$

Why Catego
Examples Other Than Pfn [AHS02]

Categories of sets and
(functions with different branching/partiality)

- **Pfn** (partial functions)

$$\begin{array}{c} X \to Y \text{ in } \mathbf{Pfn} \\ \hline\hline X \rightharpoonup Y, \text{ partial function} \\ \hline\hline X \to \mathcal{L}Y \text{ in } \mathbf{Sets} \end{array} \quad \text{where } \mathcal{L}Y = \{\perp\} + Y$$

(Potential) non-termination

- **Rel** (relations)

$$\begin{array}{c} X \to Y \text{ in } \mathbf{Rel} \\ \hline\hline R \subseteq X \times Y, \text{ relation} \\ \hline\hline X \to \mathcal{P}Y \text{ in } \mathbf{Sets} \end{array} \quad \text{where } \mathcal{P} \text{ is the powerset monad}$$

Non-determinism

- **DSRel**

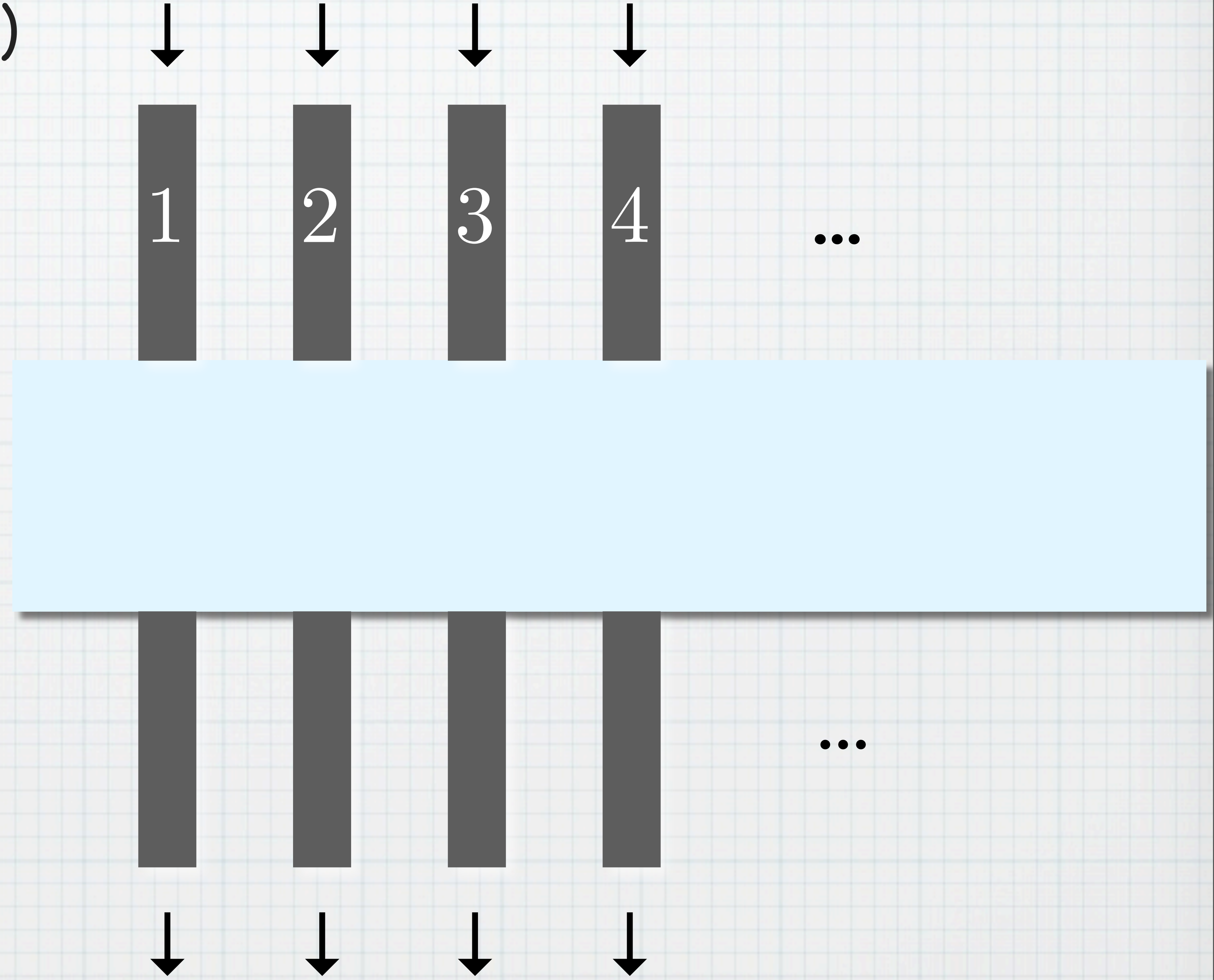
$$\begin{array}{c} X \to Y \text{ in } \mathbf{DSRel} \\ \hline\hline X \to \mathcal{D}Y \text{ in } \mathbf{Sets} \end{array}$$

$$\text{where } \mathcal{D}Y = \{d : Y \to [0,1] \mid \sum_y d(y) \leq 1\}$$

Probabilistic branching

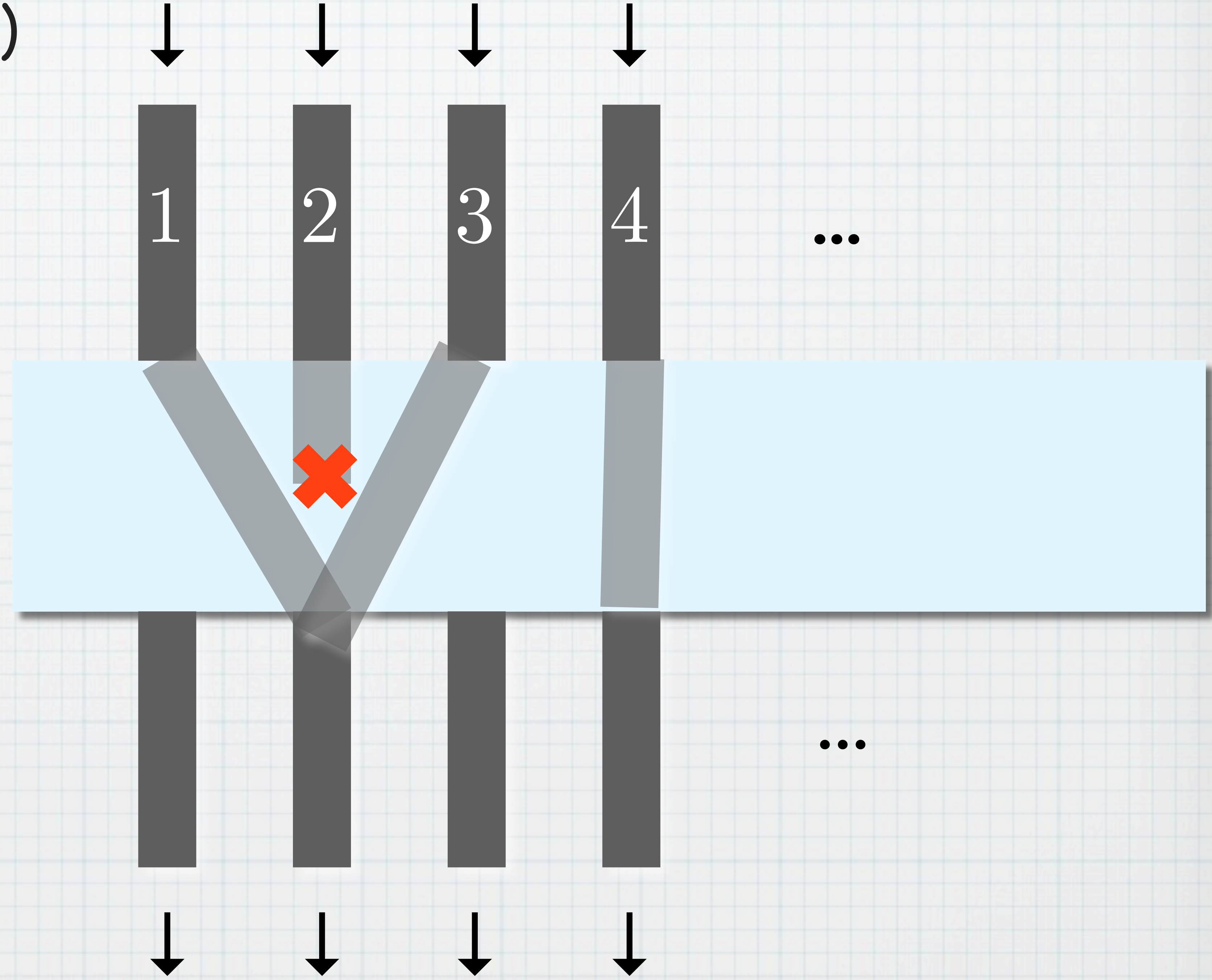
Different Branching in The GoI Animation

- **Pfn** (partial functions)
  - Pipes can be stuck
- **Rel** (relations)
  - Pipes can branch
- **DSRel**
  - Pipes can branch probabilistically



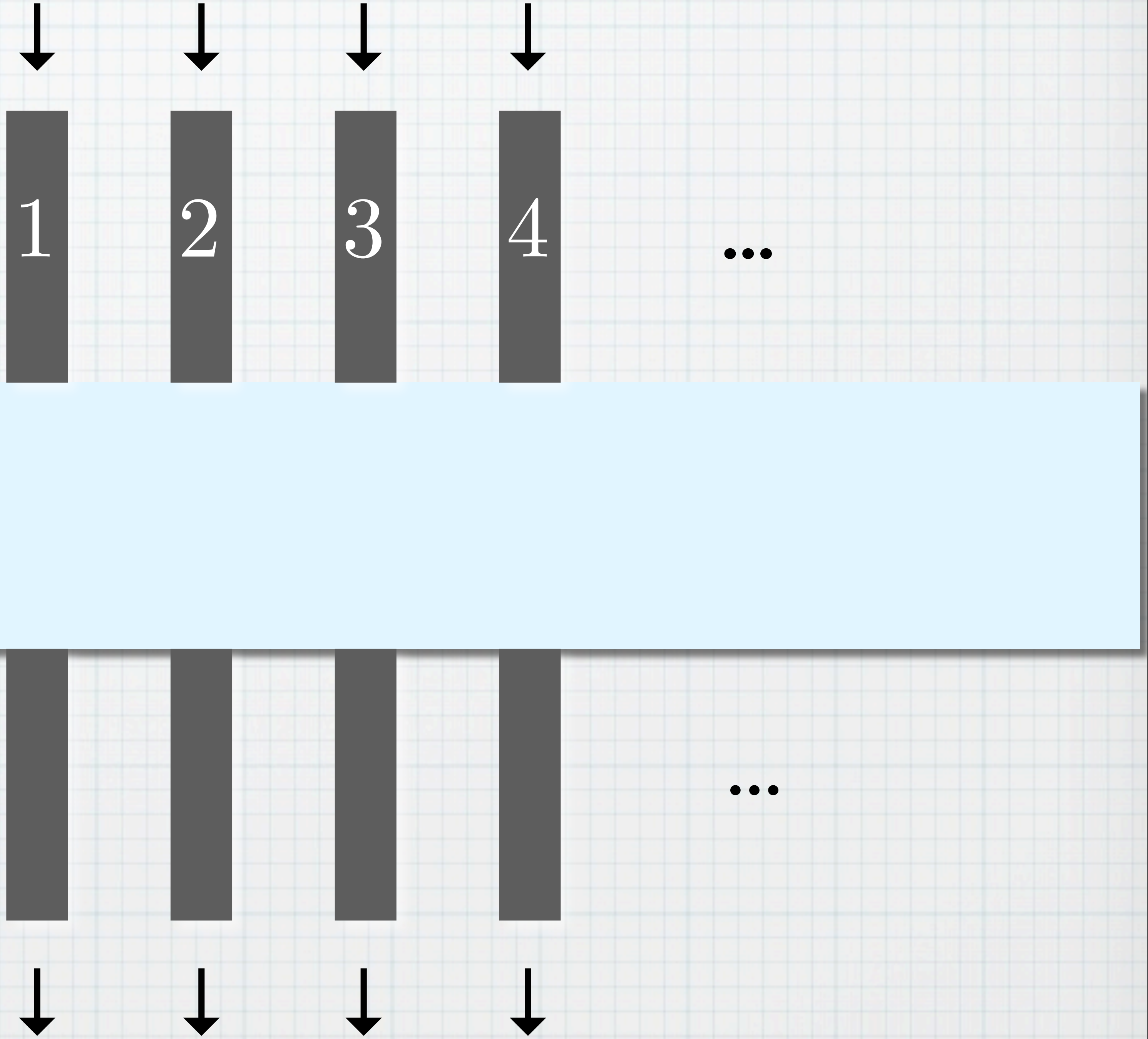
Different Branching in The GoI Animation

- **Pfn** (partial functions)
  - Pipes can be stuck
- **Rel** (relations)
  - Pipes can branch
- **DSRel**
  - Pipes can branch probabilistically



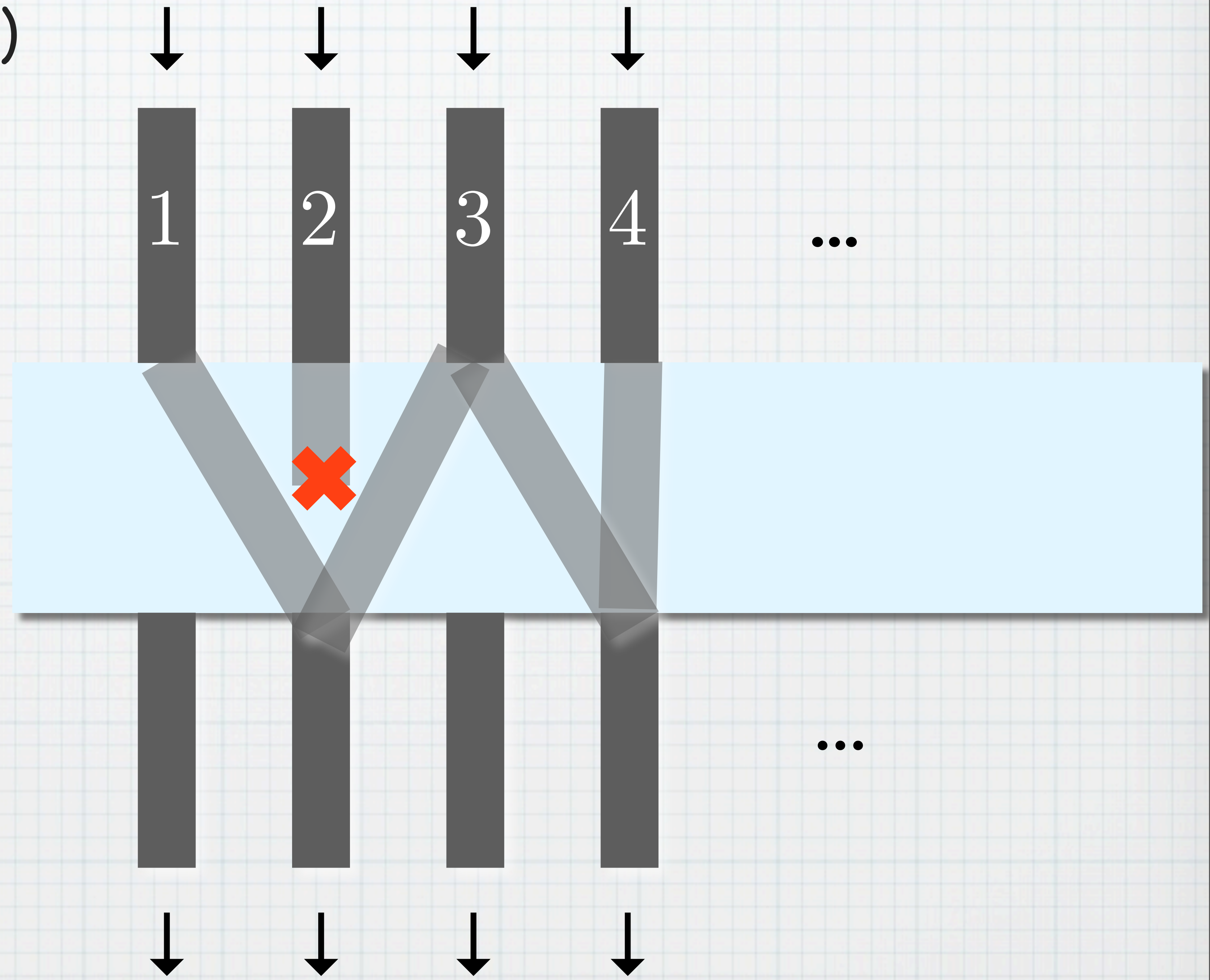
Different Branching in The GoI Animation

- **Pfn** (partial functions)
  - Pipes can be stuck
- **Rel** (relations)
  - Pipes can branch
- **DSRel**
  - Pipes can branch probabilistically



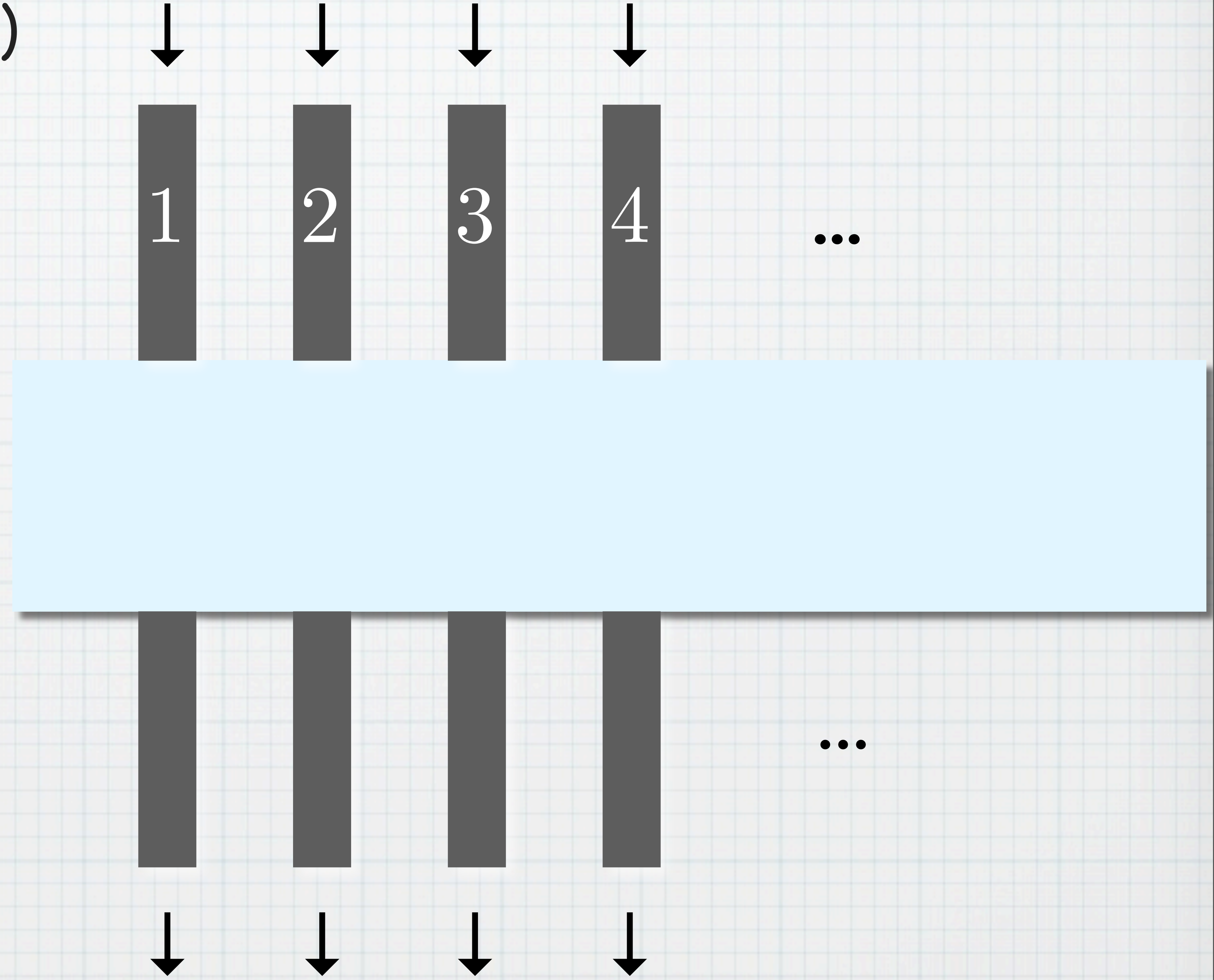
Different Branching in The GoI Animation

- **Pfn** (partial functions)
  - Pipes can be stuck
- **Rel** (relations)
  - Pipes can branch
- **DSRel**
  - Pipes can branch probabilistically



Different Branching in The GoI Animation

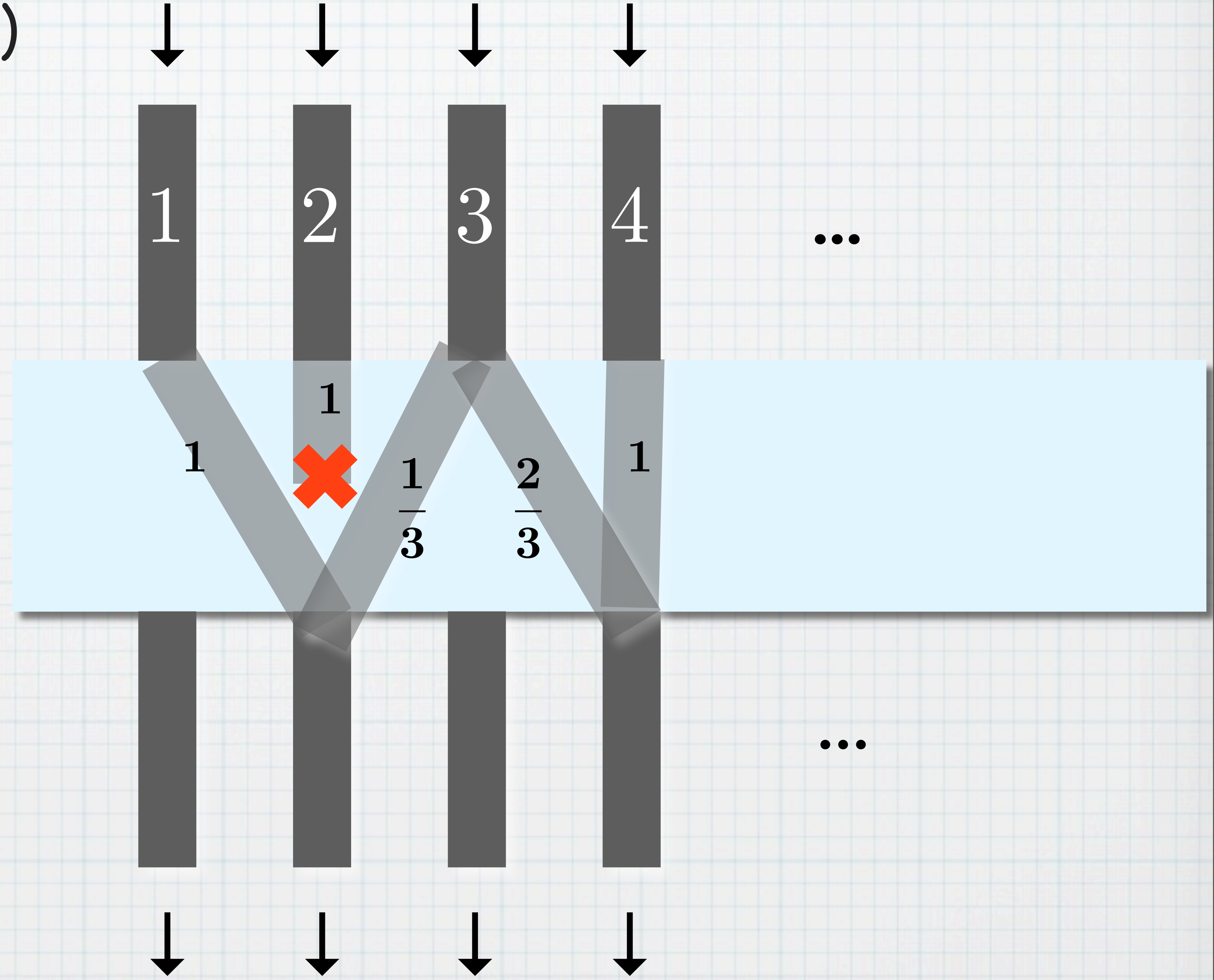
- **Pfn** (partial functions)
  - Pipes can be stuck
- **Rel** (relations)
  - Pipes can branch
- **DSRel**
  - Pipes can branch probabilistically



Hasuo (Tokyo)

Different Branching in The GoI Animation

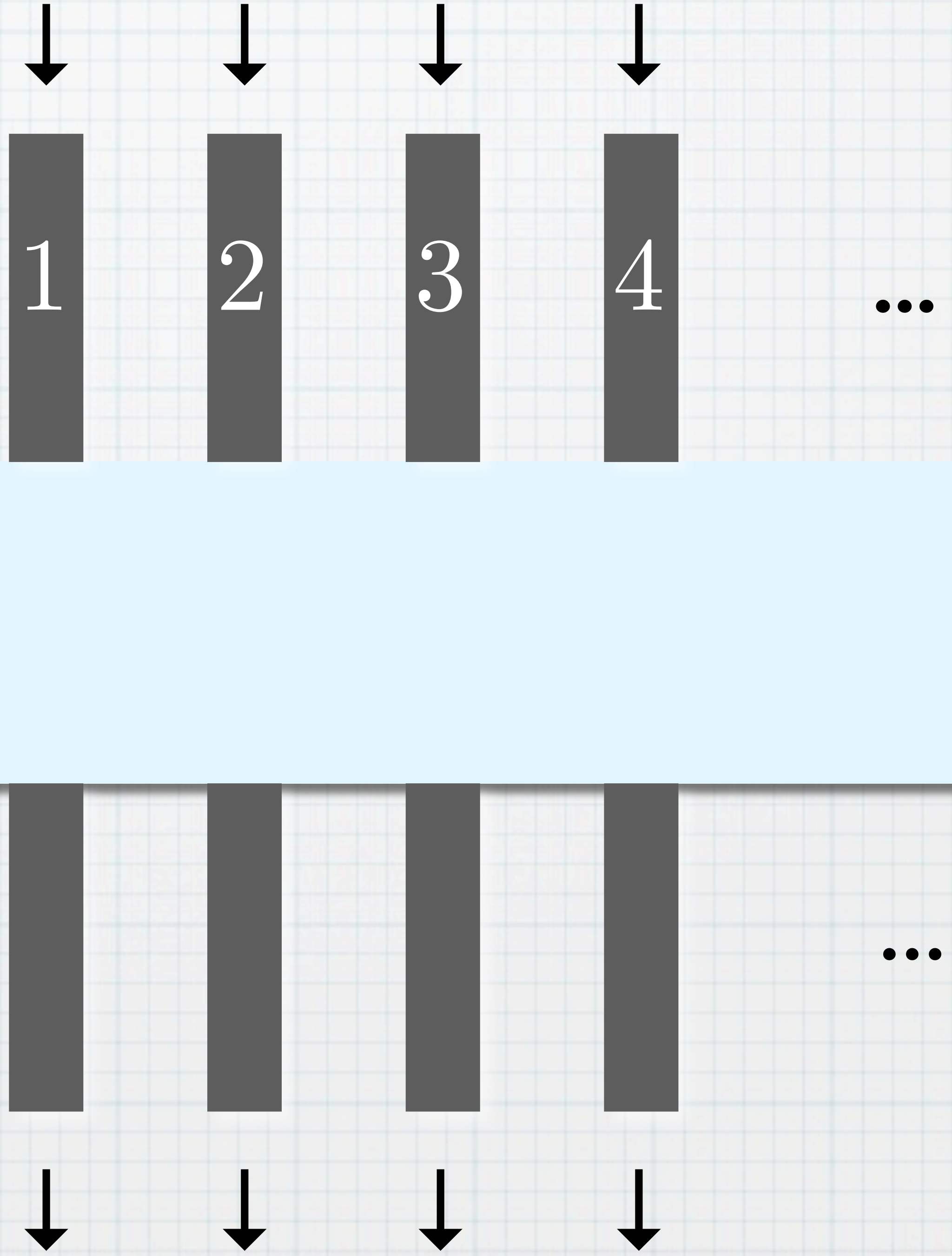
- **Pfn** (partial functions)
  - Pipes can be stuck
- **Rel** (relations)
  - Pipes can branch
- ➜ **DSRel**
  - Pipes can branch probabilistically



Hasuo (Tokyo)

Different Branching in The GoI Animation

- **Pfn** (partial functions)
  - Pipes can be stuck
- **Rel** (relations)
  - Pipes can branch
- **DSRel**
  - Pipes can branch probabilistically



Hasuo (Tokyo)

Why Catego
Examples

Categories of sets and
(functions with different branching/partiality)

(Potential) non-termination

- **Pfn** (partial functions)

$$\begin{array}{c} X \to Y \ \text{ in } \mathbf{Pfn} \\ \hline\hline X \rightharpoonup Y, \text{ partial function} \\ \hline\hline X \to \mathcal{L}Y \ \text{ in } \mathbf{Sets} \end{array} \quad \text{where } \mathcal{L}Y = \{\bot\} + Y$$

Non-determinism

- **Rel** (relations)

$$\begin{array}{c} X \to Y \ \text{ in } \mathbf{Rel} \\ \hline\hline R \subseteq X \times Y, \text{ relation} \\ \hline\hline X \to \mathcal{P}Y \ \text{ in } \mathbf{Sets} \end{array} \quad \text{where } \mathcal{P} \text{ is the powerset monad}$$

- **DSRel**

Probabilistic branching

$$\begin{array}{c} X \to Y \ \text{ in } \mathbf{DSRel} \\ \hline\hline X \to \mathcal{D}Y \ \text{ in } \mathbf{Sets} \end{array}$$

$$\text{where } \mathcal{D}Y = \{d : Y \to [0,1] \mid \sum_y d(y) \leq 1\}$$

Why Catego
Examples

$Kl(B)$ for different branching monads B

- **Pfn** (partial functions)

$$\begin{array}{c} X \to Y \text{ in } \mathbf{Pfn} \\ \hline\hline X \rightharpoonup Y, \text{ partial function} \\ \hline\hline X \to \mathcal{L}Y \text{ in } \mathbf{Sets} \end{array} \quad \text{where } \mathcal{L}Y = \{\bot\} + Y$$

(Potential) non-termination

- **Rel** (relations)

$$\begin{array}{c} X \to Y \text{ in } \mathbf{Rel} \\ \hline\hline R \subseteq X \times Y, \text{ relation} \\ \hline\hline X \to \mathcal{P}Y \text{ in } \mathbf{Sets} \end{array} \quad \text{where } \mathcal{P} \text{ is the powerset monad}$$

Non-determinism

- **DSRel**

$$\begin{array}{c} X \to Y \text{ in } \mathbf{DSRel} \\ \hline\hline X \to \mathcal{D}Y \text{ in } \mathbf{Sets} \end{array}$$

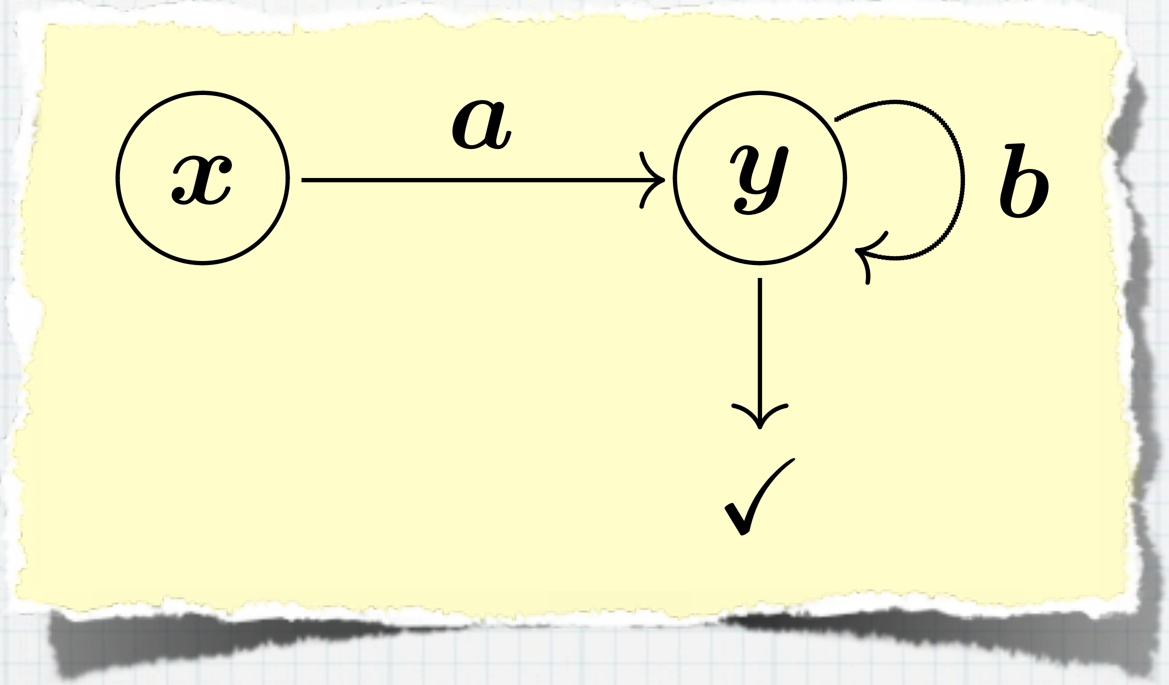
$$\text{where } \mathcal{D}Y = \{d : Y \to [0,1] \mid \sum_y d(y) \le 1\}$$

Probabilistic branching

Part 2

Coalgebraic
Trace Semantics

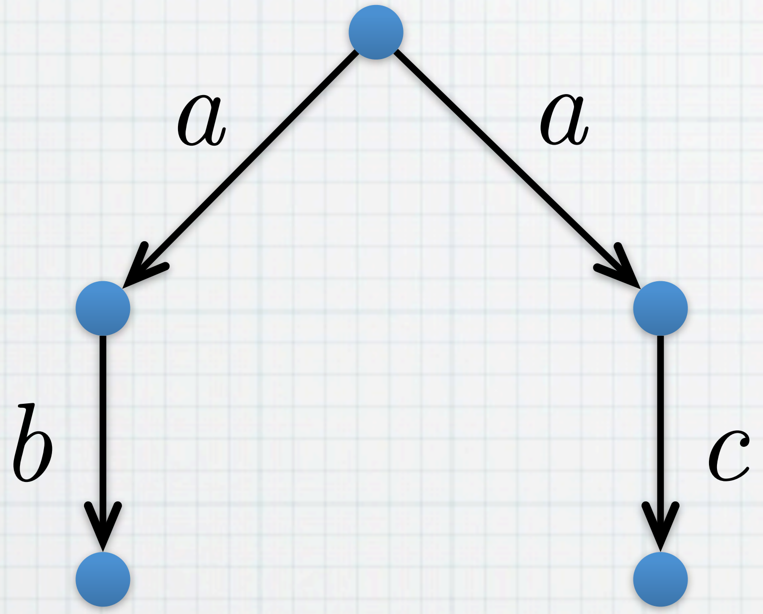
Trace Semantics of Systems

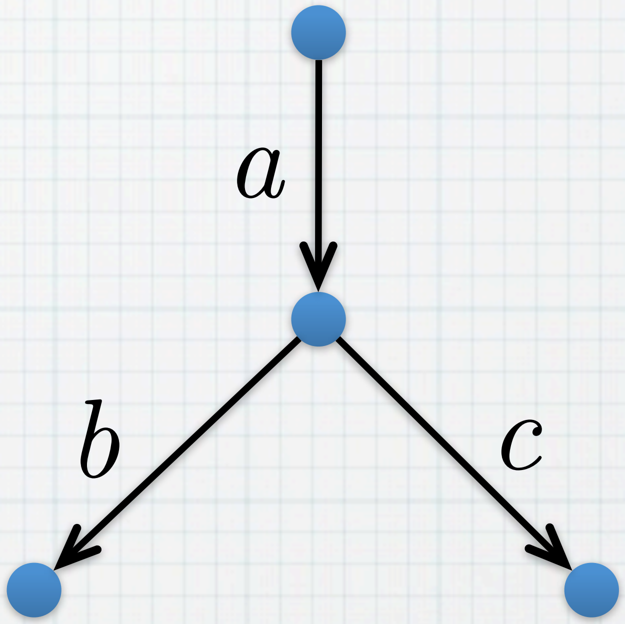


$$\mathbf{tr}(x) = \{a, ab, abb, \ldots\} = ab^*$$

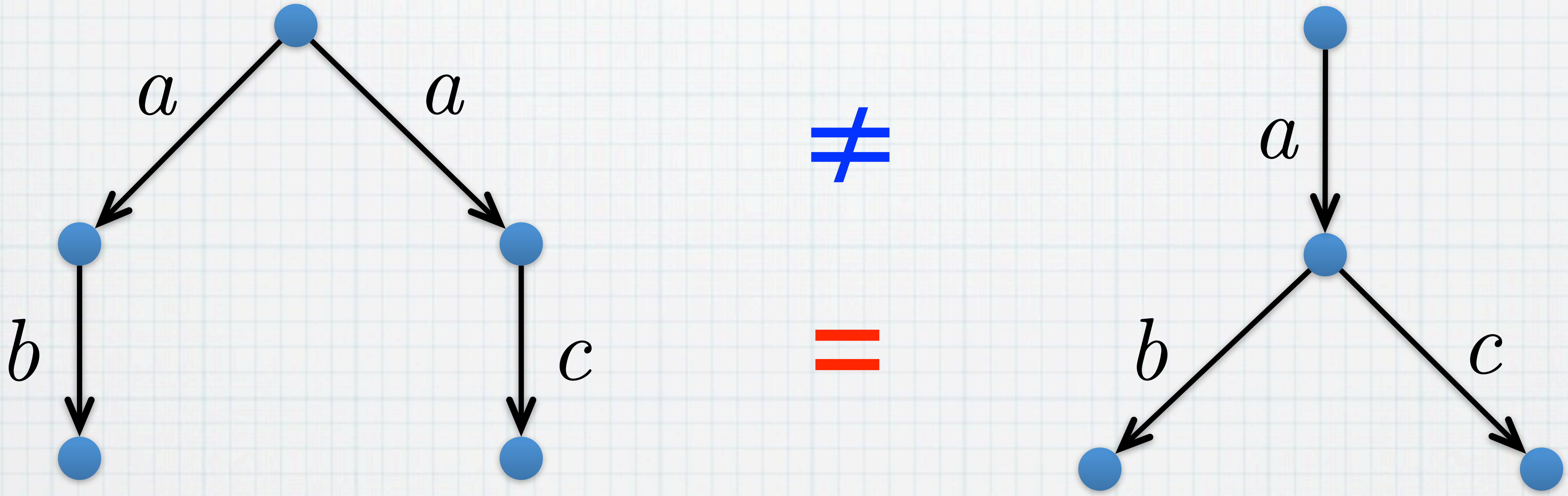
- **Non-deterministic** branching:
  sign. functor is $\mathcal{P}(1 + \Sigma \times \_)$

Bisimilarity vs. Trace Sem.

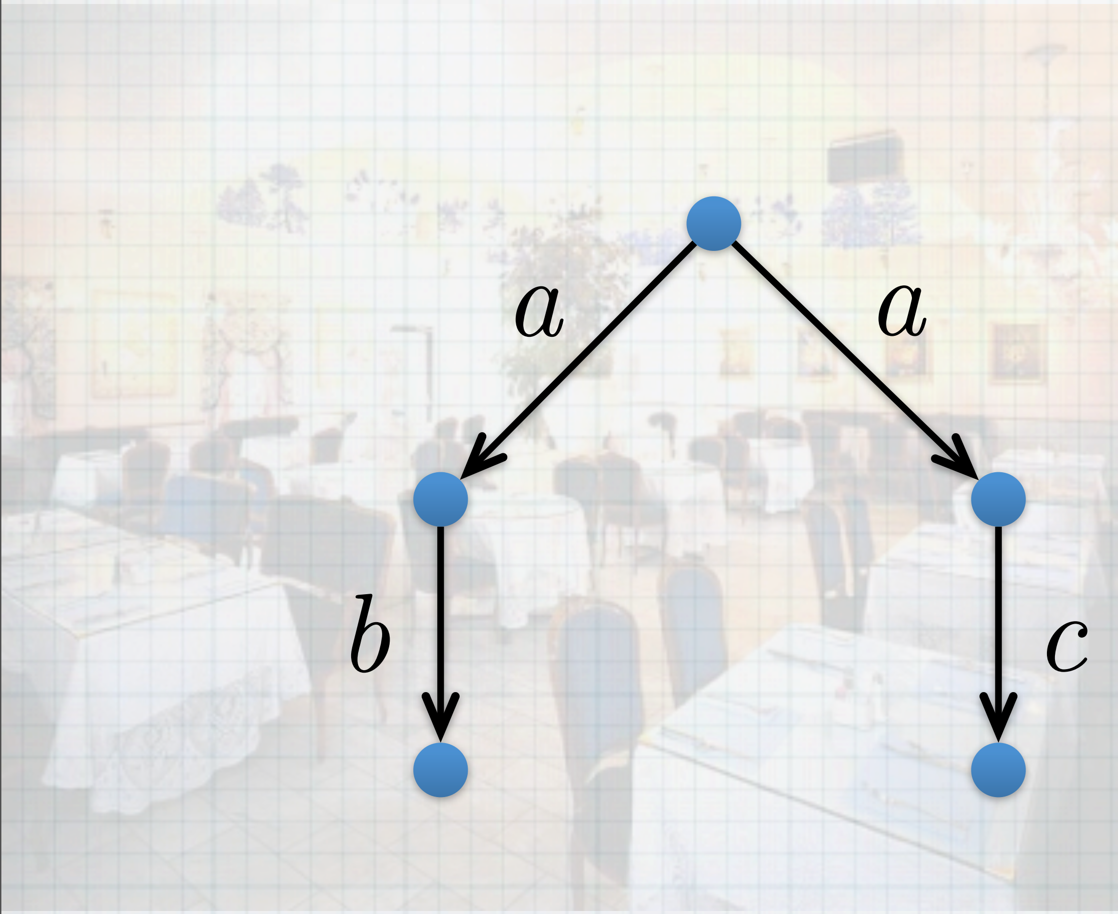




Bisimilarity vs. Trace Sem.

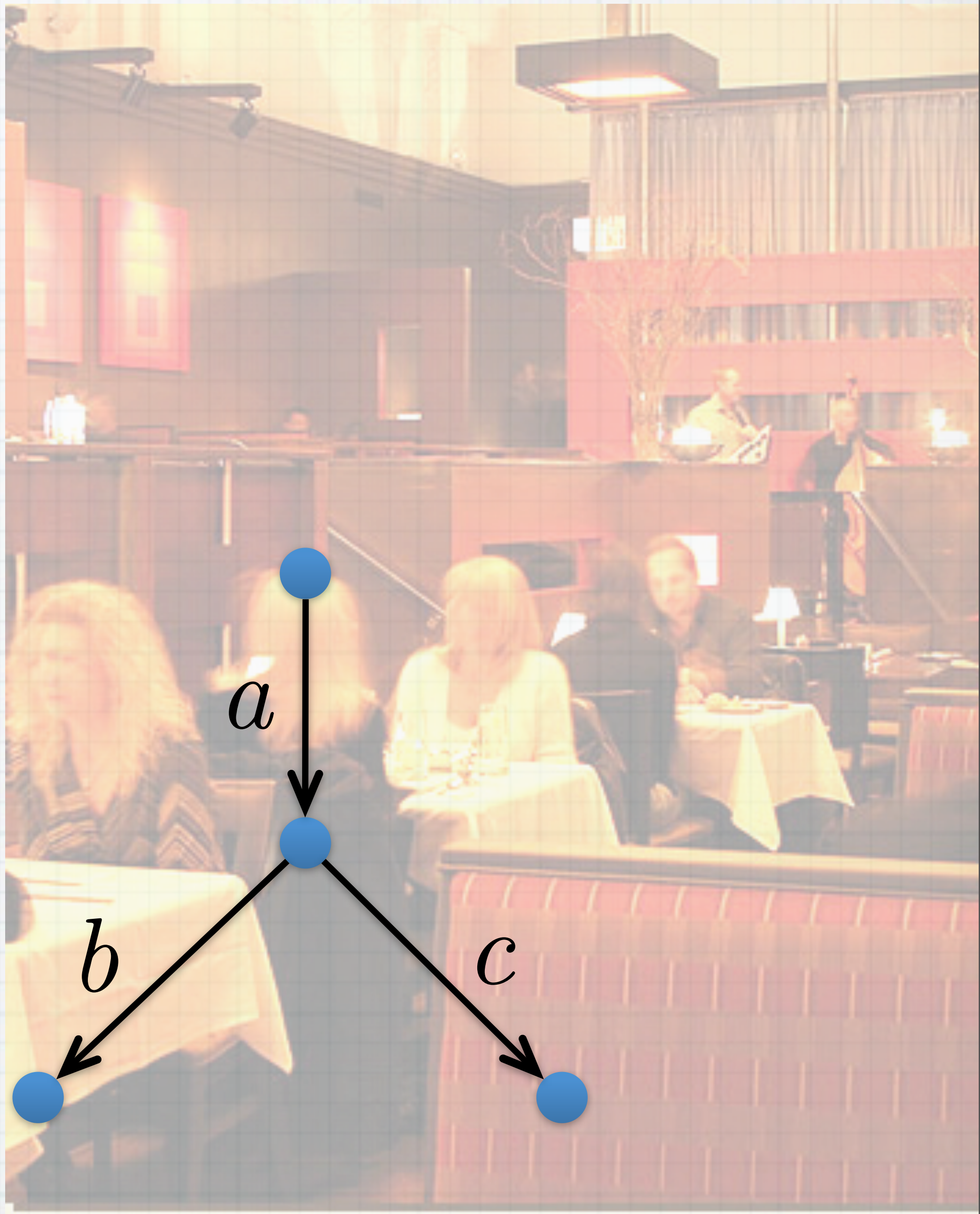


Bisimilarity vs. Trace Sem.

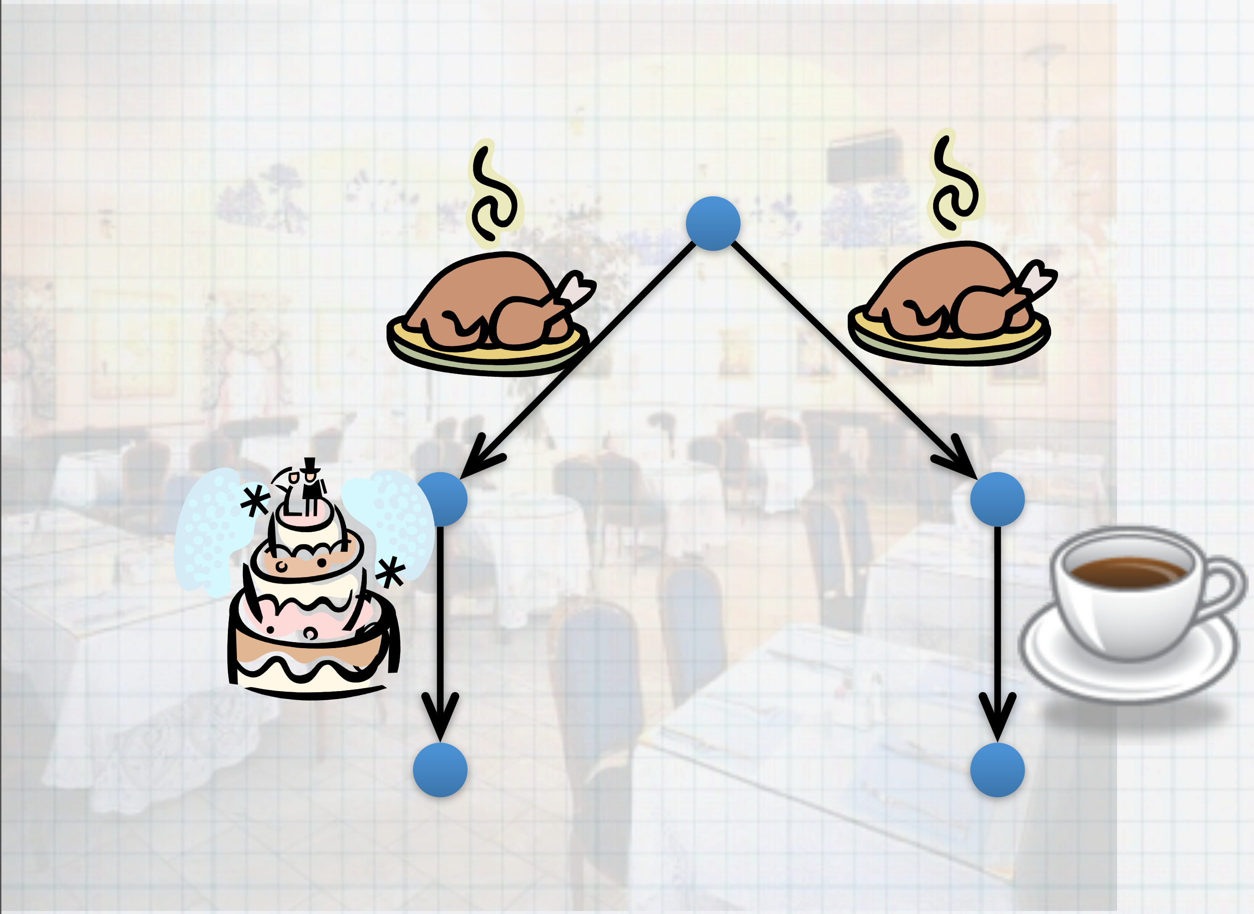


$\neq$

$=$

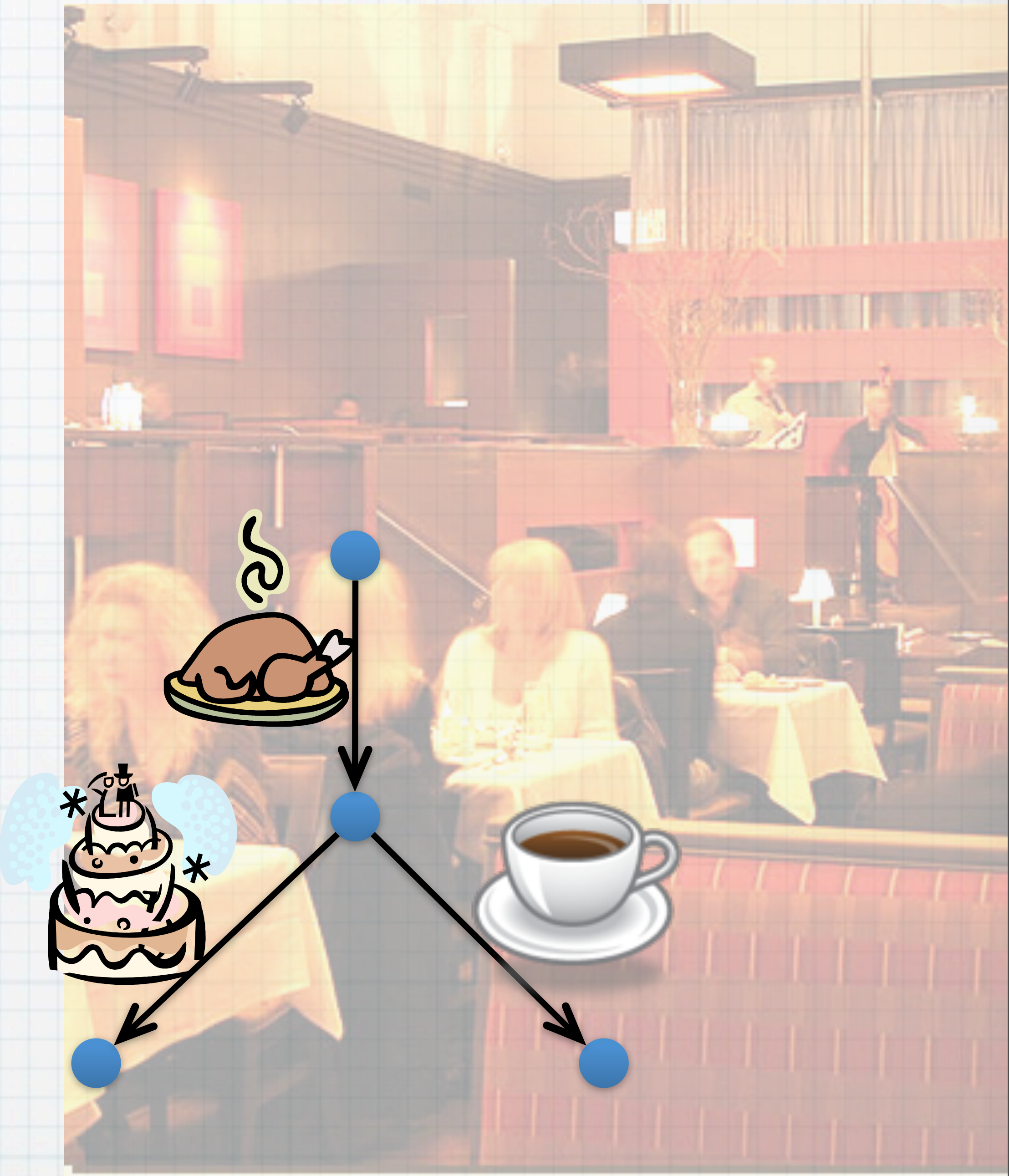


Bisimilarity vs. Trace Sem.



≠

=



Bisimilarity

Branching structure matters.

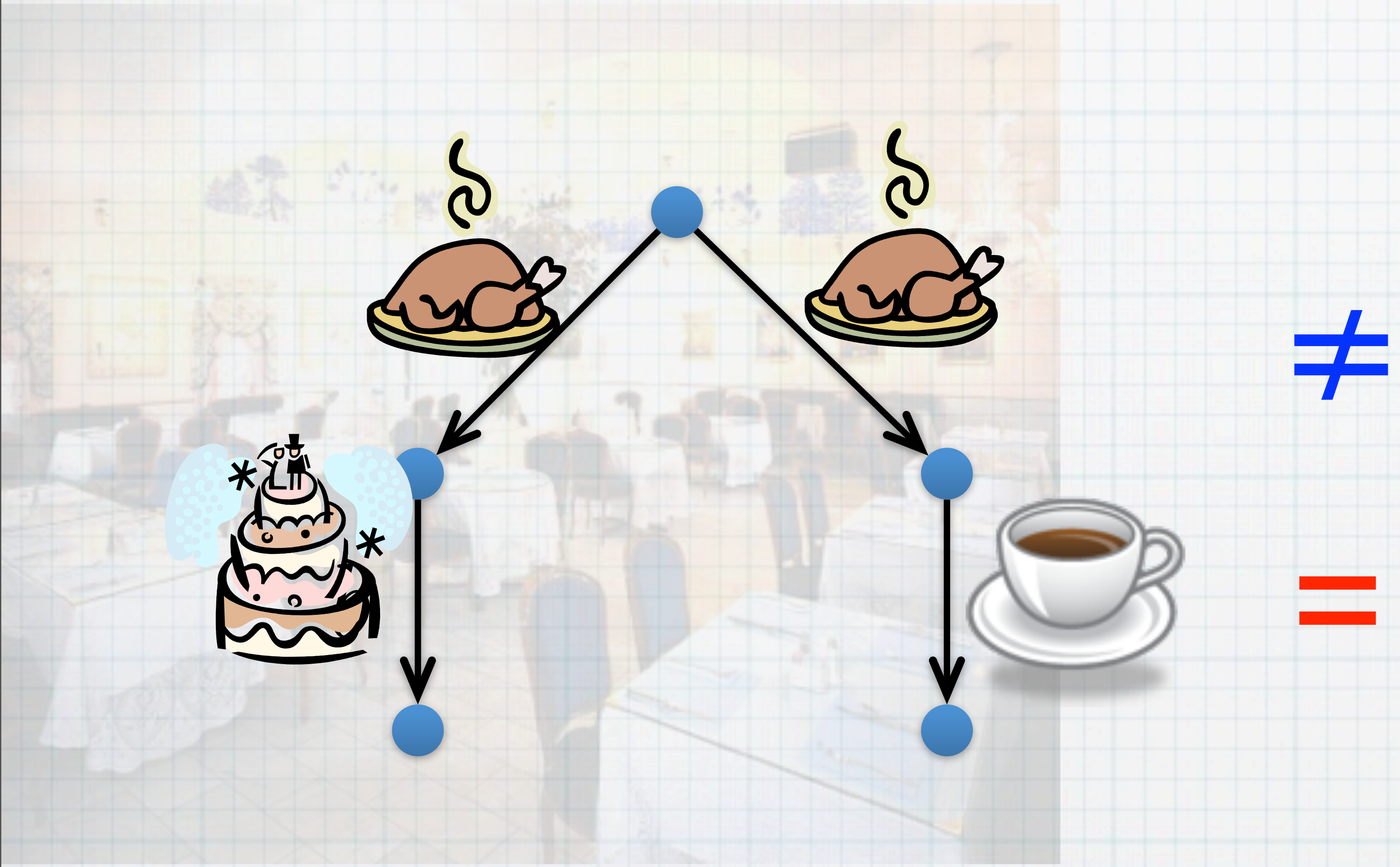
Can I choose later?

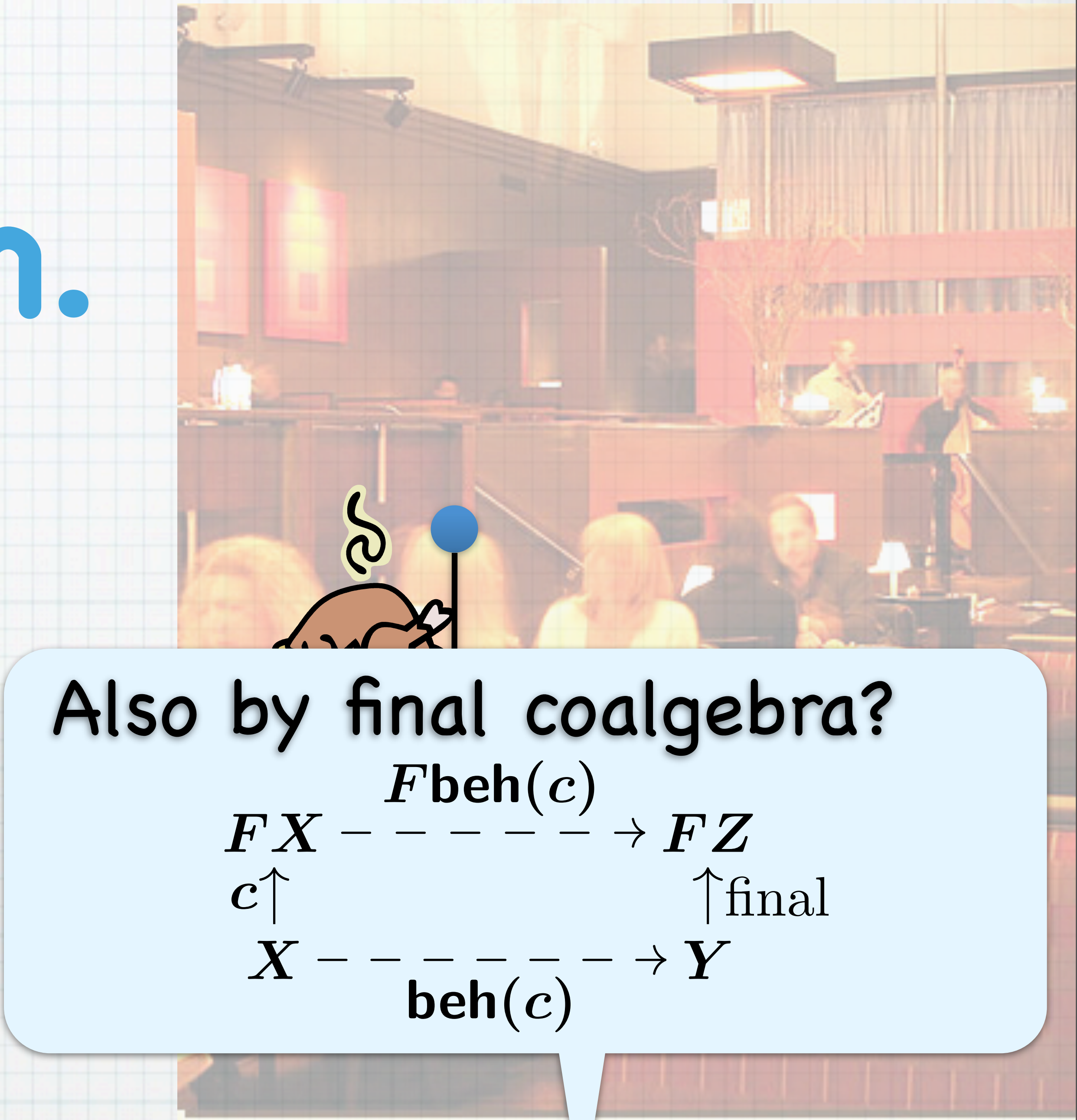
Trace semantics

Branching structure does not matter.

Anyway we'll get the same sets of food.

Bisimilarity vs. Trace Sem.





Also by final coalgebra?

$$\begin{tikzcd}
FX \arrow[r, dashed, "F\mathbf{beh}(c)"] & FZ \\
X \arrow[u, "c"] \arrow[r, dashed, "\mathbf{beh}(c)"'] & Y \arrow[u, "\mathrm{final}"']
\end{tikzcd}$$

Bisimilarity

Branching structure matters.

Can I choose later?

Trace semantics

Branching structure does not matter.

Anyway we'll get the same sets of food.

Coinduction in a Kleisli Category

[IH, Jacobs, Sokolova, '07]

$$\frac{X \longrightarrow\!\!\!\!\!\!\!+\;\; Y \quad \text{in } \mathcal{K}\ell(B)}{X \longrightarrow BY \quad \text{in } \mathbf{Sets}}$$

Thm. Let $\boldsymbol{F}$ be an endofunctor, and $\boldsymbol{B}$ be a monad, both on **Sets**. Assume:

1. We have a distributive law $\boldsymbol{\lambda : FB \Rightarrow BF}$.
2. The functor $\boldsymbol{F}$ preserves $\boldsymbol{\omega}$-colimits, yielding an initial algebra $\begin{tikzcd} \boldsymbol{FA} \arrow[d, "\boldsymbol{\alpha}", "\cong"'] \\ \boldsymbol{A} \end{tikzcd}$.
3. The Kleisli category $\mathcal{K}\ell(\boldsymbol{B})$ is $\mathbf{Cpo}_{\perp}$-enriched and composition in $\mathcal{K}\ell(\boldsymbol{B})$ is left-strict.

Then:

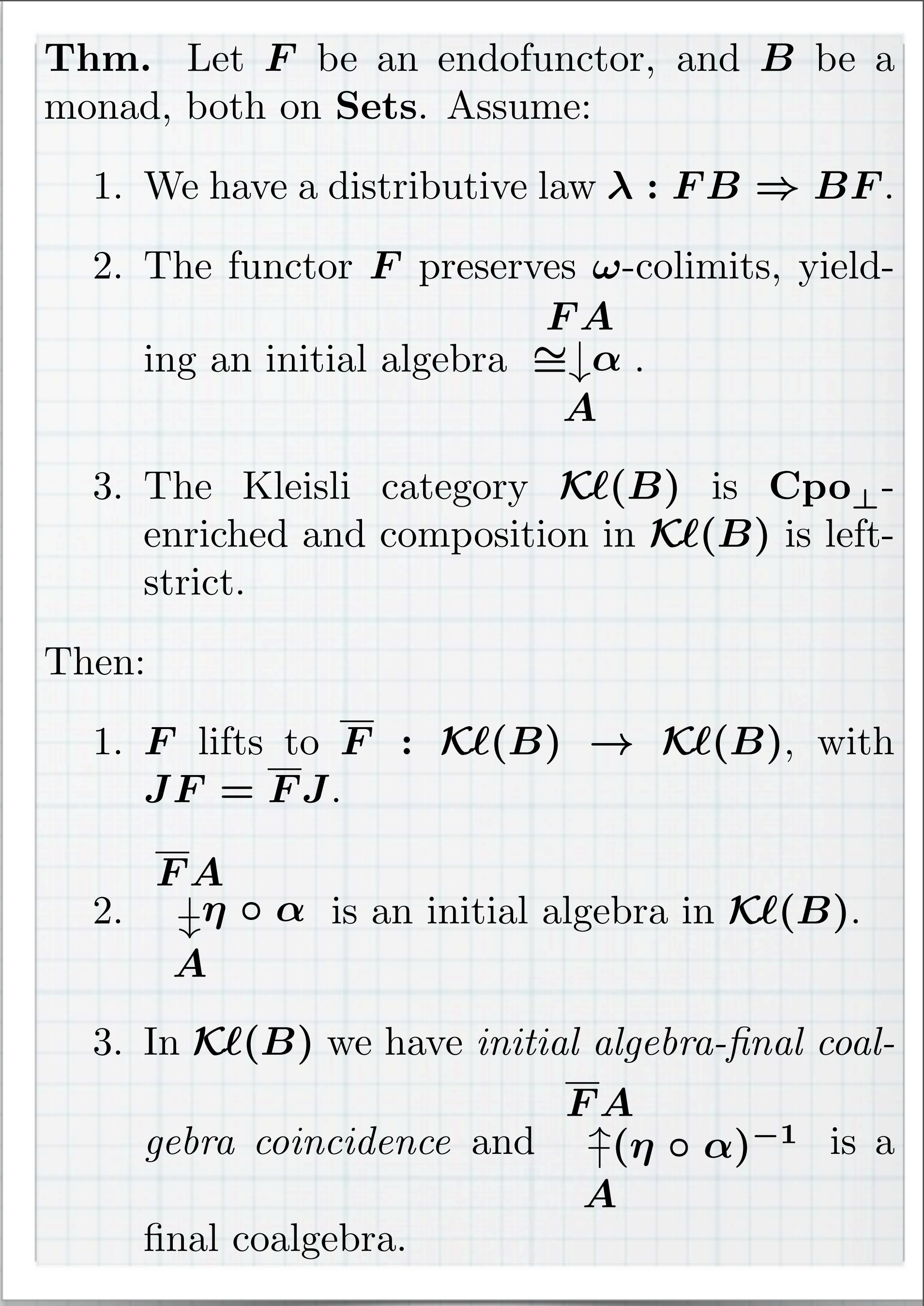
1. $\boldsymbol{F}$ lifts to $\overline{\boldsymbol{F}} : \mathcal{K}\ell(\boldsymbol{B}) \to \mathcal{K}\ell(\boldsymbol{B})$, with $\boldsymbol{JF = \overline{F}J}$.
2. $\begin{tikzcd} \overline{\boldsymbol{F}}\boldsymbol{A} \arrow[d, "\boldsymbol{\eta \circ \alpha}", "+"' marking] \\ \boldsymbol{A} \end{tikzcd}$ is an initial algebra in $\mathcal{K}\ell(\boldsymbol{B})$.
3. In $\mathcal{K}\ell(\boldsymbol{B})$ we have *initial algebra-final coalgebra coincidence* and $\begin{tikzcd} \overline{\boldsymbol{F}}\boldsymbol{A} \\ \boldsymbol{A} \arrow[u, "(\boldsymbol{\eta \circ \alpha})^{-1}", "+"' marking] \end{tikzcd}$ is a final coalgebra.

Coinduction in a Kleisli Category

[IH, Jacobs, Sokolova, '07]

$$\begin{array}{c} X \relbar\joinrel\mapstochar\joinrel\rightarrow Y \quad \text{in } \mathcal{K}\ell(B) \\ \hline\hline X \longrightarrow BY \quad \text{in } \mathbf{Sets} \end{array}$$

- Initial algebra lifts from Sets to $Kl(B)$
  - diagram chasing [Johnstone]



Thm. Let F be an endofunctor, and B be a monad, both on **Sets**. Assume:

1. We have a distributive law $\lambda : FB \Rightarrow BF$.
2. The functor F preserves ω-colimits, yielding an initial algebra $\begin{tikzcd} FA \arrow[d, "\cong"', "\alpha"] \\ A \end{tikzcd}$.
3. The Kleisli category $\mathcal{K}\ell(B)$ is $\mathbf{Cpo}_\perp$-enriched and composition in $\mathcal{K}\ell(B)$ is left-strict.

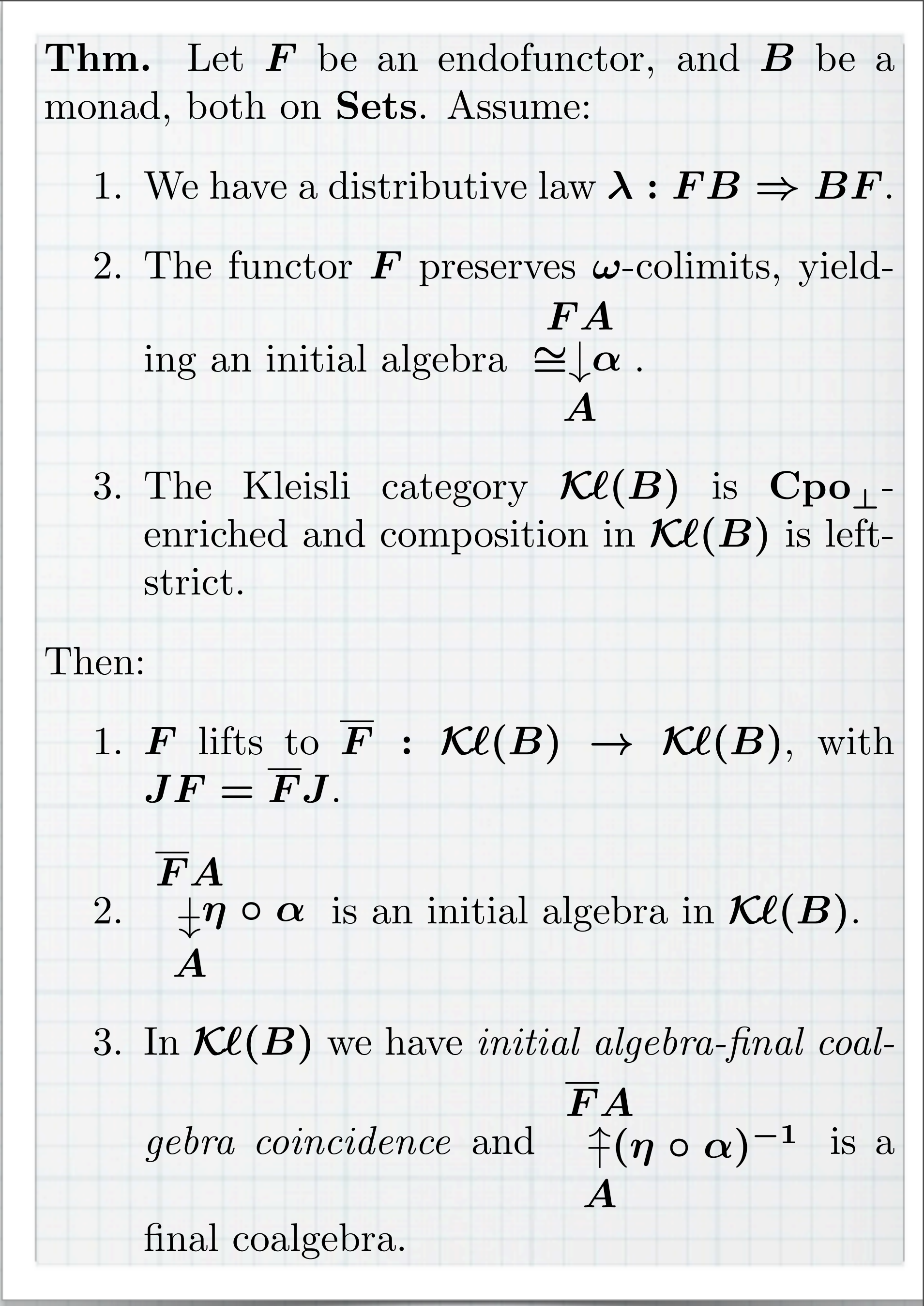
Then:

1. F lifts to $\overline{F} : \mathcal{K}\ell(B) \to \mathcal{K}\ell(B)$, with $JF = \overline{F}J$.
2. $\begin{tikzcd} \overline{F}A \arrow[d, "|" marking, "\eta \circ \alpha"] \\ A \end{tikzcd}$ is an initial algebra in $\mathcal{K}\ell(B)$.
3. In $\mathcal{K}\ell(B)$ we have *initial algebra-final coalgebra coincidence* and $\begin{tikzcd} \overline{F}A \\ A \arrow[u, "|" marking, "(\eta \circ \alpha)^{-1}"'] \end{tikzcd}$ is a final coalgebra.

Coinduction in a Kleisli Category

[IH, Jacobs, Sokolova, '07]

$$\begin{array}{c} X \mathrel{-\!\!\!+\!\!\!\rightarrow} Y \quad \text{in } \mathcal{K}\ell(B) \\ \hline\hline X \longrightarrow BY \quad \text{in } \mathbf{Sets} \end{array}$$



Thm. Let F be an endofunctor, and B be a monad, both on $\mathbf{Sets}$. Assume:

1. We have a distributive law $\lambda : FB \Rightarrow BF$.
2. The functor F preserves ω-colimits, yielding an initial algebra $\begin{tikzcd} FA \arrow[d, "\alpha", "\cong"'] \\ A \end{tikzcd}$.
3. The Kleisli category $\mathcal{K}\ell(B)$ is $\mathbf{Cpo}_{\perp}$-enriched and composition in $\mathcal{K}\ell(B)$ is left-strict.

Then:

1. F lifts to $\overline{F} : \mathcal{K}\ell(B) \rightarrow \mathcal{K}\ell(B)$, with $JF = \overline{F}J$.
2. $\begin{tikzcd} \overline{F}A \arrow[d, "\eta \circ \alpha", "+"' marking] \\ A \end{tikzcd}$ is an initial algebra in $\mathcal{K}\ell(B)$.
3. In $\mathcal{K}\ell(B)$ we have *initial algebra-final coalgebra coincidence* and $\begin{tikzcd} \overline{F}A \\ A \arrow[u, "(\eta \circ \alpha)^{-1}", "+"' marking] \end{tikzcd}$ is a final coalgebra.

- Initial algebra lifts from **Sets** to $Kl(B)$
  - diagram chasing [Johnstone]
- In $Kl(B)$ we have **IA-FC coincidence**
  - typical of "domain-theoretic" categories
  - "Algebraically compact" [Freyd]

Coinduction in a Kleisli Category

- E.g. $B = \mathcal{P},\ F = 1 + \Sigma \times (\_\,)$

$$\begin{tikzcd}
1+\Sigma\times X \arrow[r, dashed, "1+\Sigma\times \mathsf{tr}(c)"] & 1+\Sigma\times\Sigma^* & \text{in } \mathcal{K}\ell(\mathcal{P}) \\
X \arrow[u, "c"] \arrow[r, dashed, "\mathsf{tr}(c)"'] & \Sigma^* \arrow[u, "\text{final}"'] &
\end{tikzcd}$$

- Separation between B and F

Coinduction in a Kleisli Category

- E.g. $B = \mathcal{P},\ F = 1 + \Sigma \times (\_\,)$

$$\begin{tikzcd}
1 + \Sigma \times X \arrow[r, dashed, "1 + \Sigma \times \mathsf{tr}(c)", "\shortmid" marking] & 1 + \Sigma \times \Sigma^* \\
X \arrow[u, "c", "\shortmid" marking] \arrow[r, dashed, "\mathsf{tr}(c)"', "\shortmid" marking] & \Sigma^* \arrow[u, "\text{final}"', "\shortmid" marking]
\end{tikzcd} \quad \text{in } \mathcal{K}\ell(\mathcal{P})$$

$$\begin{tikzcd}
\mathcal{P}(1 + \Sigma \times X) \\
X \arrow[u, "c"]
\end{tikzcd} \quad \text{in Sets}$$

- Separation between B and F

Coinduction in a Kleisli Category

- E.g. $B = \mathcal{P}$, $F = 1 + \Sigma \times (\_)$

$$\begin{tikzcd}
1 + \Sigma \times X \arrow[r, dashed, "|", "1 + \Sigma \times \mathrm{tr}(c)"] & 1 + \Sigma \times \Sigma^* \\
X \arrow[u, "|" marking, "c"] \arrow[r, dashed, "|" marking, "\mathrm{tr}(c)"'] & \Sigma^* \arrow[u, "|" marking, "\mathrm{final}"']
\end{tikzcd}$$

in $\mathcal{K}\ell(\mathcal{P})$

induced by

$$\begin{tikzcd}
1 + \Sigma \times \Sigma^* \arrow[d, "\mathrm{initial}"] \\
\Sigma^*
\end{tikzcd}$$

in **Sets**

$$\begin{tikzcd}
\mathcal{P}(1 + \Sigma \times X) \\
X \arrow[u, "c"]
\end{tikzcd}$$

in **Sets**

- Separation between B and F

Coinduction in a Kleisli Category

- E.g. $B = \mathcal{P}$, $F = 1 + \Sigma \times (\_)$

$$\begin{tikzcd}
1 + \Sigma \times X \arrow[r, dashed, "1 + \Sigma \times \mathsf{tr}(c)", "\shortmid" marking] & 1 + \Sigma \times \Sigma^* \\
X \arrow[u, "c", "\shortmid" marking] \arrow[r, dashed, "\mathsf{tr}(c)"', "\shortmid" marking] & \Sigma^* \arrow[u, "\text{final}"', "\shortmid" marking]
\end{tikzcd}$$

in $\mathcal{K}\ell(\mathcal{P})$

induced by

$$\begin{tikzcd}
1 + \Sigma \times \Sigma^* \arrow[d, "\text{initial}"] \\
\Sigma^*
\end{tikzcd}$$

in **Sets**

$$\begin{tikzcd}
\mathcal{P}(1 + \Sigma \times X) \\
X \arrow[u, "c"]
\end{tikzcd}$$

in **Sets**

$$X \xrightarrow{\mathsf{tr}(c)} \mathcal{P}(\Sigma^*)$$

- Separation between B and F

Coinduction in a Kleisli Category

- E.g. $B = \mathcal{P}$, $F = 1 + \Sigma \times (\_)$

$$\begin{tikzcd}
1+\Sigma\times X \arrow[r, dashed, "1+\Sigma\times\mathbf{tr}(c)", "+" marking] & 1+\Sigma\times\Sigma^* \\
X \arrow[u, "c"] \arrow[r, dashed, "\mathbf{tr}(c)"', "+" marking] & \Sigma^* \arrow[u, "\text{final}"']
\end{tikzcd}$$

in $\mathcal{K}\ell(\mathcal{P})$

induced by

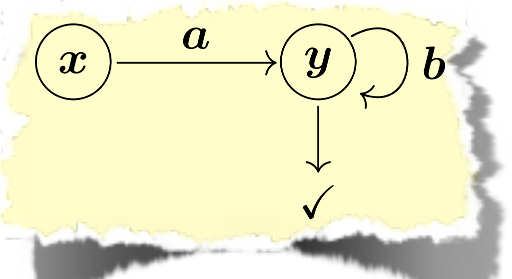
$$\begin{tikzcd}
1+\Sigma\times\Sigma^* \arrow[d, "\text{initial}"] \\
\Sigma^*
\end{tikzcd}$$

in Sets

$$\begin{tikzcd}
\mathcal{P}(1+\Sigma\times X) \\
X \arrow[u, "c"]
\end{tikzcd}$$

in Sets

$$X \xrightarrow{\mathbf{tr}(c)} \mathcal{P}(\Sigma^*)$$



$\mathbf{tr}(x) = \{a, ab, abb, \dots\} = ab^*$

- Separation between B and F

Coinduction in a Kleisli Category

- E.g. $B = \mathcal{P}$, $F = 1 + \Sigma \times (\_)$

$$\begin{tikzcd}
1+\Sigma\times X \arrow[r, dashed, "1+\Sigma\times\mathbf{tr}(c)", "+" marking] & 1+\Sigma\times\Sigma^{*} \\
X \arrow[u, "c", "+" marking] \arrow[r, dashed, "\mathbf{tr}(c)"', "+" marking] & \Sigma^{*} \arrow[u, "\text{final}"', "+" marking]
\end{tikzcd}$$

in $\mathcal{K}\ell(\mathcal{P})$

induced by

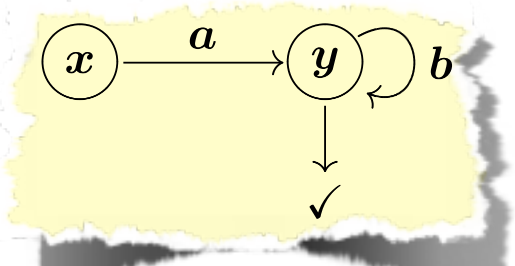
$$\begin{tikzcd}
1+\Sigma\times\Sigma^{*} \arrow[d, "\text{initial}"] \\
\Sigma^{*}
\end{tikzcd}$$

in Sets

$$\begin{tikzcd}
\mathcal{P}(1+\Sigma\times X) \\
X \arrow[u, "c"]
\end{tikzcd}$$

in Sets

$$X \xrightarrow{\mathbf{tr}(c)} \mathcal{P}(\Sigma^{*})$$

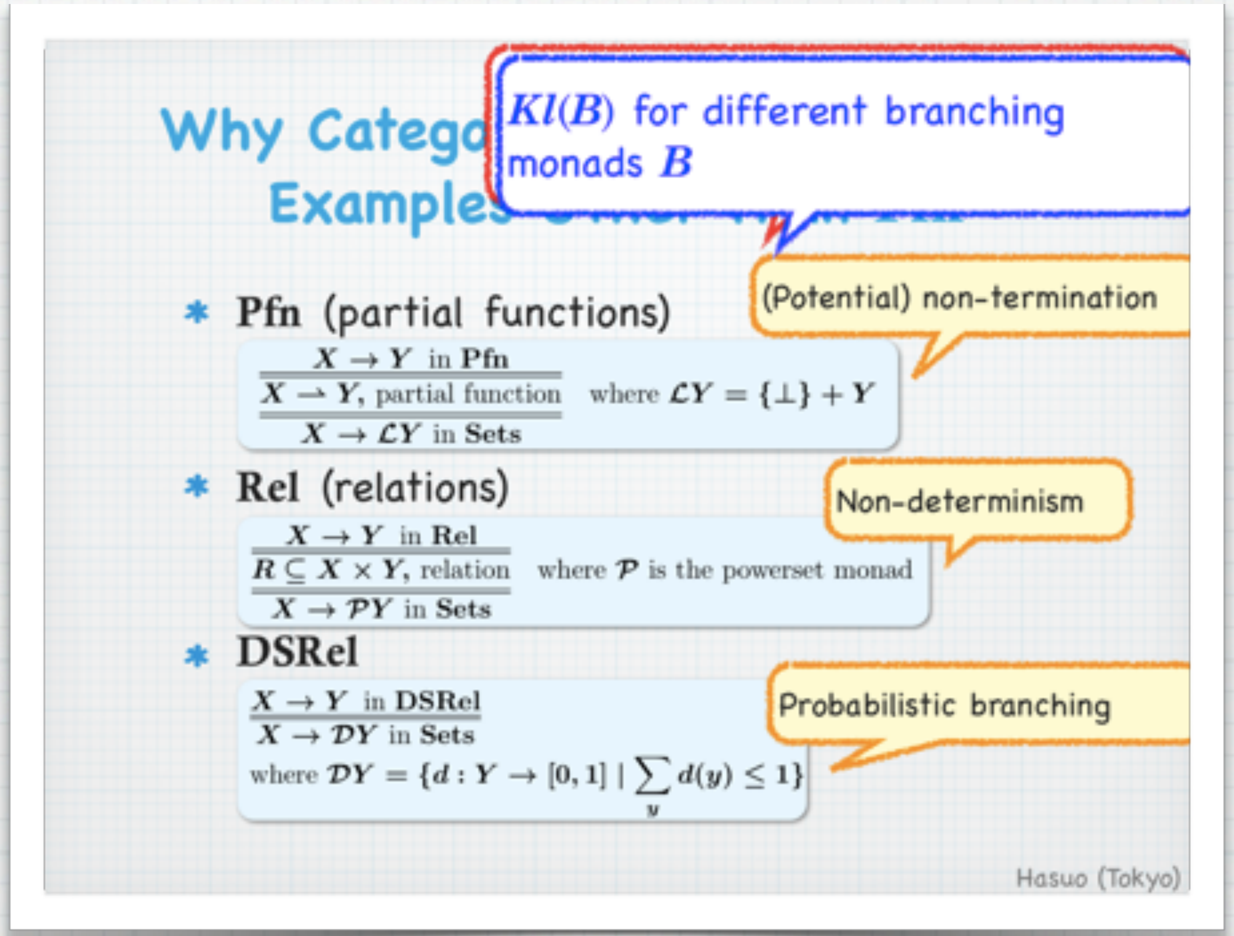


$\mathbf{tr}(x) = \{a, ab, abb, \dots\} = ab^{*}$

- Separation between B and F

Examples

- A branching monad B:
  - **Lift monad** $\mathcal{L} = 1 + (\_)$, **powerset monad** $\mathcal{P}$, **subdistribution monad** $\mathcal{D}$
  - Precisely those in



Why Catego

Examples

$Kl(B)$ for different branching monads B

- **Pfn** (partial functions)

$$\frac{X \to Y \text{ in } \mathbf{Pfn}}{\dfrac{X \rightharpoonup Y, \text{ partial function}}{X \to \mathcal{L}Y \text{ in } \mathbf{Sets}}} \quad \text{where } \mathcal{L}Y = \{\bot\} + Y$$

(Potential) non-termination

- **Rel** (relations)

$$\frac{X \to Y \text{ in } \mathbf{Rel}}{\dfrac{R \subseteq X \times Y, \text{ relation}}{X \to \mathcal{P}Y \text{ in } \mathbf{Sets}}} \quad \text{where } \mathcal{P} \text{ is the powerset monad}$$

Non-determinism

- **DSRel**

$$\frac{X \to Y \text{ in } \mathbf{DSRel}}{X \to \mathcal{D}Y \text{ in } \mathbf{Sets}}$$

$$\text{where } \mathcal{D}Y = \{d : Y \to [0,1] \mid \sum_{y} d(y) \le 1\}$$

Probabilistic branching

Hasuo (Tokyo)

- A functor F: polynomial functors

The Coauthor

- Naohiko Hoshino
  - DSc (Kyoto, 2011)
    - Supervisor: Masahito "Hassei" Hasegawa
  - Currently at RIMS, Kyoto U.
    - `http://www.kurims.kyoto-u.ac.jp/~naophiko/`

From Coalgebraic Trace to Monoidal Trace

Thm. ([Jacobs,CMCS10])
Given a "branching monad" B on **Sets**, the monoidal category

$$(\mathcal{K}\ell(\boldsymbol{B}), +, \mathbf{0})$$

is a traced symmetric monoidal category.

Cor.
$\big(\,(\mathcal{K}\ell(\boldsymbol{B}), +, \mathbf{0}),\ \mathbb{N}\cdot\_\,,\ \mathbb{N}\,\big)$ is a GoI situation.

From Coalgebraic Trace to Monoidal Trace

Thm. ([Jacobs,CMCS10])
Given a "branching monad" B on **Sets**, the monoidal category

$$(\mathcal{K}\ell(B), +, \mathbf{0})$$

is a traced symmetric monoidal category.

Cor.
$\big((\mathcal{K}\ell(B), +, \mathbf{0}),\ \mathbb{N}\cdot\_\ ,\ \mathbb{N} \big)$ is a GoI situation.

From Coalgebraic Trace to Monoidal Trace

Thm. ([Jacobs,CMCS10])
Given a "branching monad" B on **Sets**, the monoidal category

$$(\mathcal{K}\ell(B), +, 0)$$

is a traced symmetric monoidal category.

Cor.
$\big(\,(\mathcal{K}\ell(B), +, 0),\ \mathbb{N}\cdot\_\,,\ \mathbb{N}\,\big)$ is a GoI situation.

Proof. We need

$$\frac{X + Z \overset{f}{\longrightarrow} Y + Z \quad \text{in } \mathcal{K}\ell(T)}{X \overset{\mathsf{tr}(f)}{\longrightarrow} Y \quad \text{in } \mathcal{K}\ell(T)}$$

- $X + Z \overset{f}{\longrightarrow} Y + Z \overset{\kappa}{\longrightarrow} Y + (X + Z)$ is a $Y + (\_)$-coalgebra
- $$\begin{tikzcd} Y + \mathbb{N}\cdot Y \arrow[d, "\cong", "\alpha"'] \\ \mathbb{N}\cdot Y \end{tikzcd}$$ is an initial algebra in **Sets**
- Therefore in $\mathcal{K}\ell(T)$:

$$\begin{tikzcd}
Y + (X + Z) \arrow[r, dashed] & Y + \mathbb{N}\cdot Y \\
X + Z \arrow[u, "\kappa \circ f"] \arrow[r, dashed, "\mathsf{tr}(c)"'] & \mathbb{N}\cdot Y \arrow[u, "\text{final}"'] \arrow[d, "\nabla"'] \\
X \arrow[u, "\kappa_1"] & Y
\end{tikzcd}$$

From Coalgebraic Trace to Monoidal Trace

Thm. ([Jacobs,CMCS10])
Given a "branching monad" B on **Sets**, the monoidal category

$$(\mathcal{K}\ell(B), +, 0)$$

is a traced symmetric monoidal category.

Cor.
$\big(\, (\mathcal{K}\ell(B), +, 0),\ \mathbb{N}\cdot\_\,,\ \mathbb{N} \,\big)$ is a GoI situation.

Proof. We need

$$\frac{X + Z \xrightarrow{f} Y + Z \quad \text{in } \mathcal{K}\ell(T)}{X \xrightarrow{\mathsf{tr}(f)} Y \quad \text{in } \mathcal{K}\ell(T)}$$

- $X + Z \xrightarrow{f} Y + Z \xrightarrow{\kappa} Y + (X + Z)$
  is a $Y + (\_)$-coalgebra
- $$\begin{tikzcd} Y + \mathbb{N}\cdot Y \arrow[d, "\alpha", "\cong"'] \\ \mathbb{N}\cdot Y \end{tikzcd}$$ is an initial algebra in **Sets**
- Therefore in $\mathcal{K}\ell(T)$:

$$\begin{tikzcd}
Y + (X + Z) \arrow[r, dashed] & Y + \mathbb{N}\cdot Y \\
X + Z \arrow[u, "\kappa \circ f"] \arrow[r, dashed, "\mathsf{tr}(c)"'] & \mathbb{N}\cdot Y \arrow[u, "\text{final}"'] \arrow[d, "\nabla"] \\
X \arrow[u, "\kappa_1"] & Y
\end{tikzcd}$$

The Categorical GoI Workflow

Traced monoidal category $\mathbb{C}$

+ other constructs ➜ "GoI situation" [AHS02]

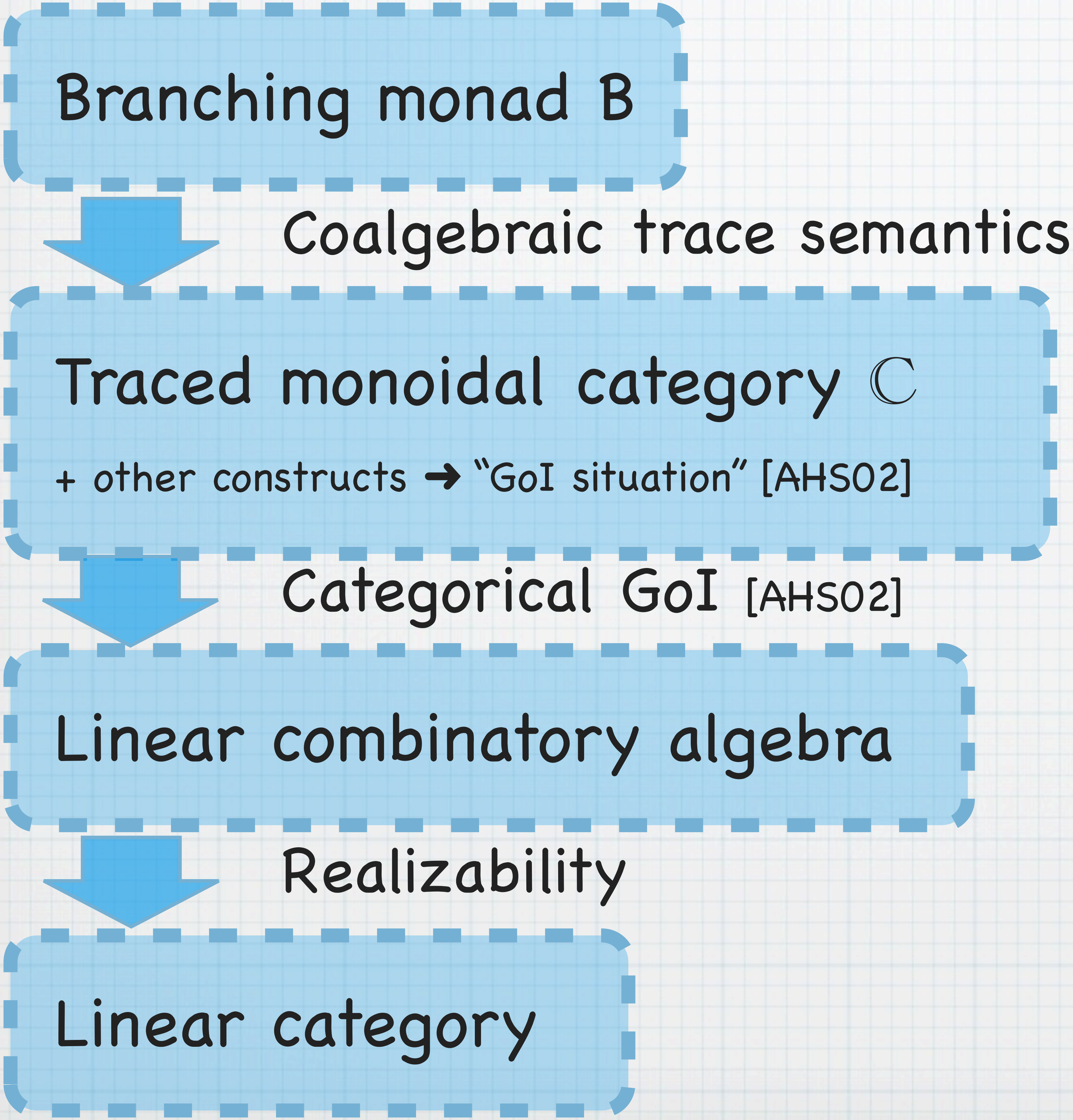
Categorical GoI [AHS02]

Linear combinatory algebra

Realizability

Linear category

The Categorical GoI Workflow



Branching monad B

Coalgebraic trace semantics

Traced monoidal category $\mathbb{C}$

+ other constructs ➜ “GoI situation” [AHS02]

Categorical GoI [AHS02]

Linear combinatory algebra

Realizability

Linear category

The Categorical GoI Workflow

Branching monad B

Coalgebraic trace semantics

Traced monoidal category $\mathbb{C}$

+ other constructs ➜ "GoI situation" [AHS02]

Categorical GoI [AHS02]

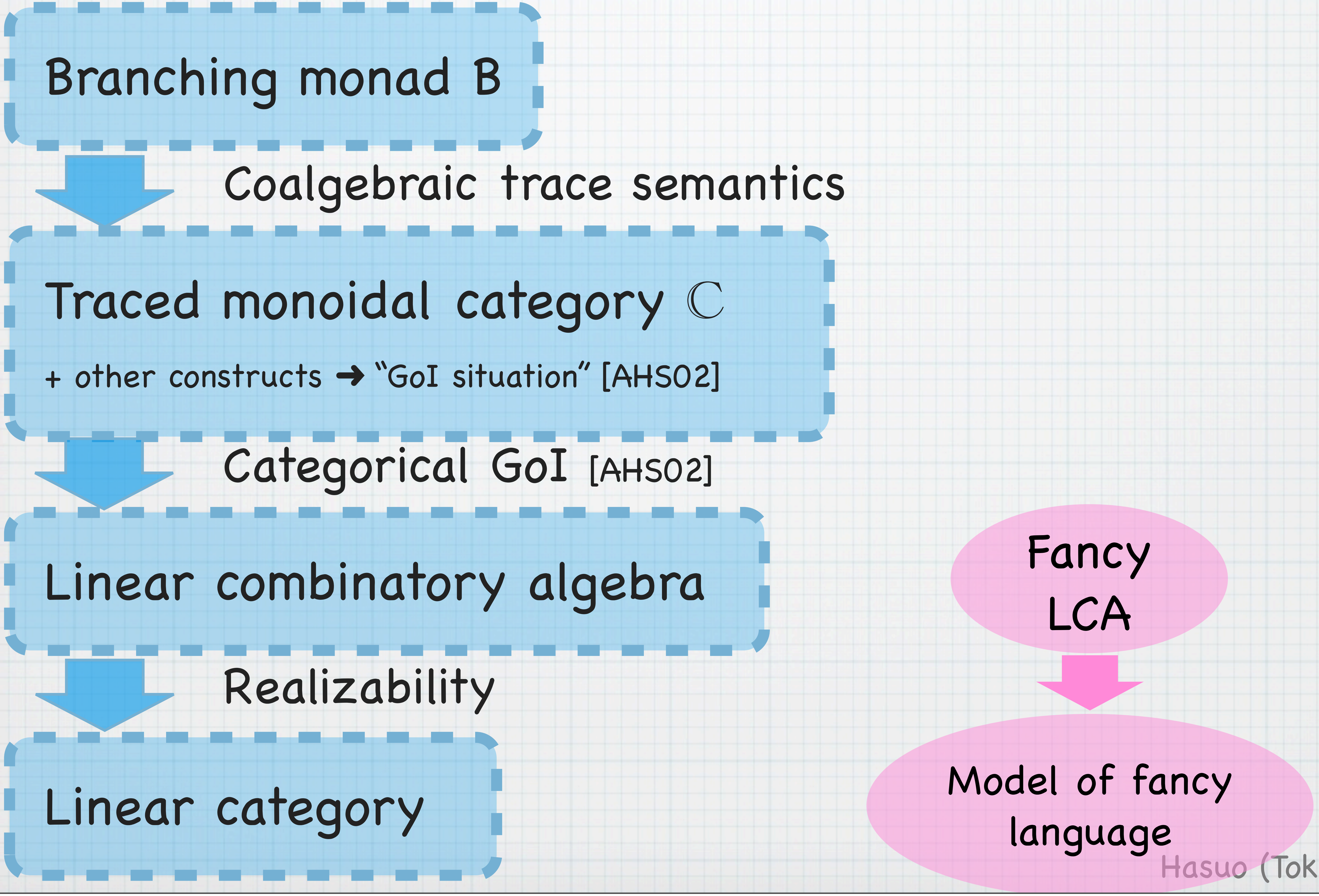
Linear combinatory algebra

Realizability

Linear category

Model of fancy language

The Categorical GoI Workflow



Branching monad B

Coalgebraic trace semantics

Traced monoidal category $\mathbb{C}$

+ other constructs ➜ "GoI situation" [AHS02]

Categorical GoI [AHS02]

Linear combinatory algebra

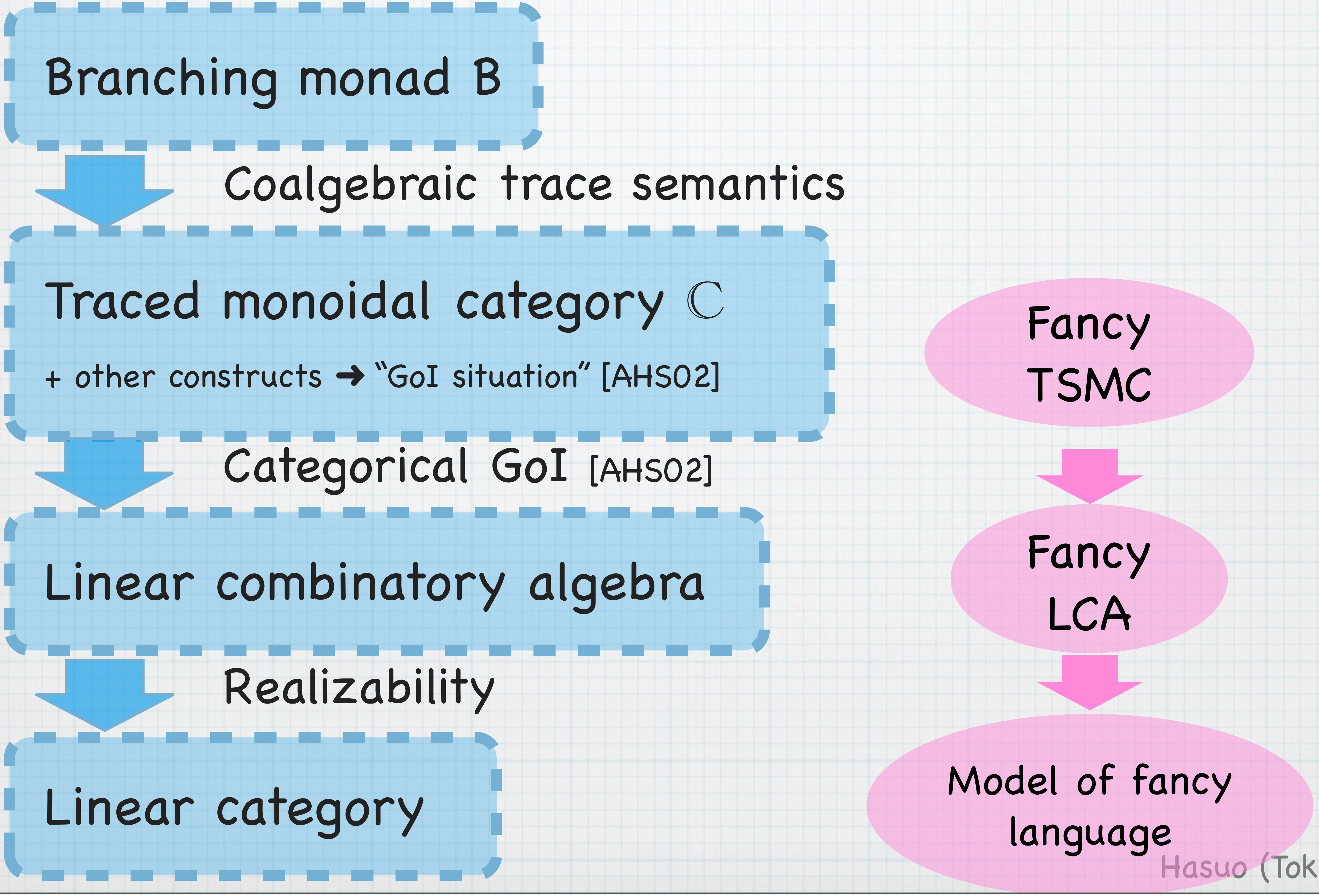
Realizability

Linear category

Fancy LCA

Model of fancy language

The Categorical GoI Workflow



Branching monad B

Coalgebraic trace semantics

Traced monoidal category $\mathbb{C}$

+ other constructs ➔ "GoI situation" [AHS02]

Categorical GoI [AHS02]

Linear combinatory algebra

Realizability

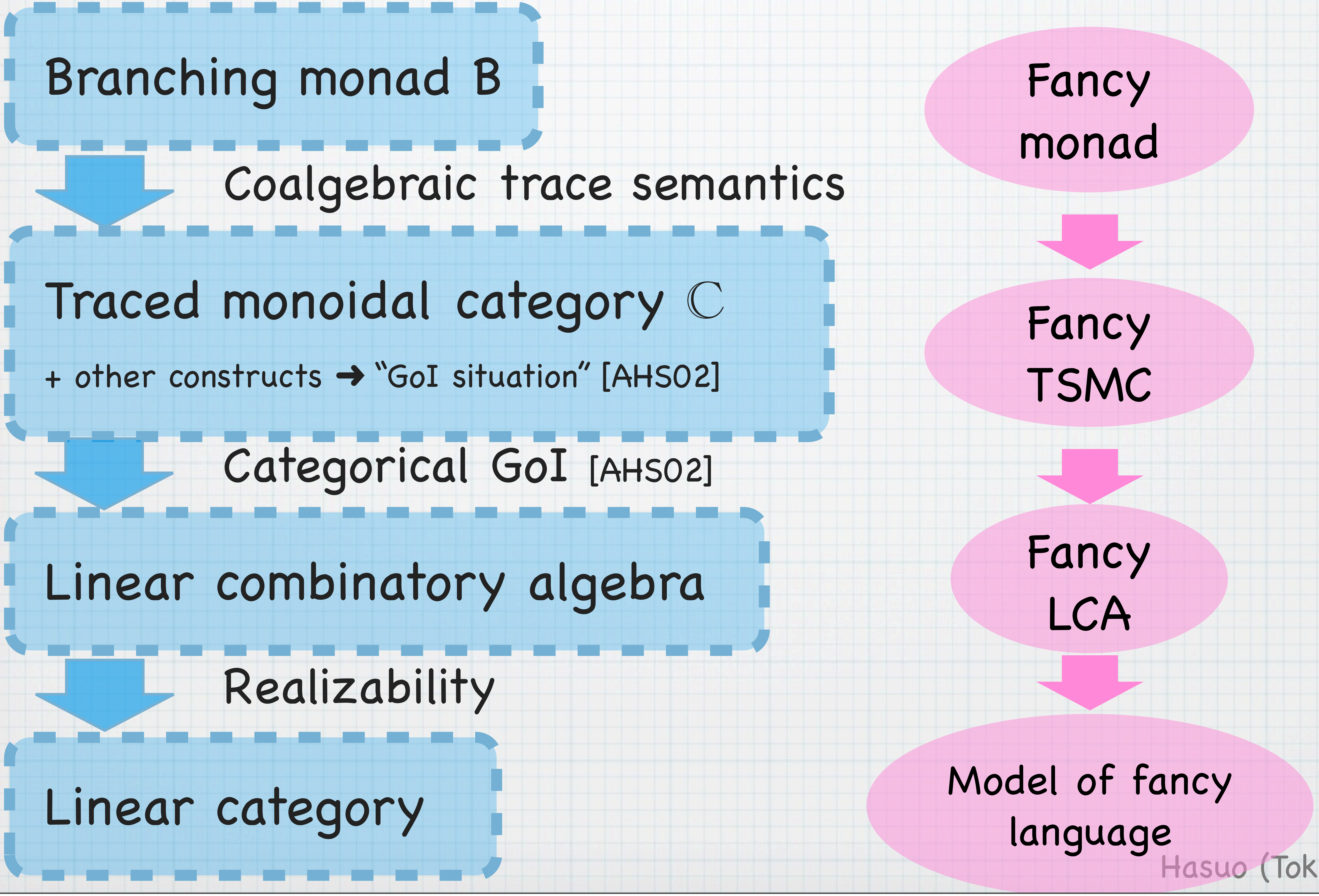
Linear category

Fancy TSMC

Fancy LCA

Model of fancy language

The Categorical GoI Workflow



Branching monad B

Coalgebraic trace semantics

Traced monoidal category $\mathbb{C}$

+ other constructs ➜ "GoI situation" [AHS02]

Categorical GoI [AHS02]

Linear combinatory algebra

Realizability

Linear category

Fancy monad

Fancy TSMC

Fancy LCA

Model of fancy language

What is Fancy, Nowadays?

What is Fancy, Nowadays?

- Biology?

What is Fancy, Nowadays?

- Biology?
- Hybrid systems?
  - Both discrete and continuous data, typically in **cyber-physical systems** (CPS)
  - → Our approach via **non-standard analysis**
    [Suenaga, IH, ICALP'11] [IH, Suenaga, CAV'12]
    [Suenaga, Sekine, IH, POPL'13]

What is Fancy, Nowadays?

- Biology?
- Hybrid systems?
  - Both discrete and continuous data, typically in **cyber-physical systems** (CPS)
  - → Our approach via **non-standard analysis**
    [Suenaga, IH, ICALP'11] [IH, Suenaga, CAV'12]
    [Suenaga, Sekine, IH, POPL'13]
- Quantum?
  - Yes this worked!

Part 3

Phil Scott.
Tutorial on Geometry of
Interaction, FMCS 2004.
Page 47/47

Future Directions

- GoI 2 : Non-converging algms (untyped λ-calc / PCF)
  - uses more topological info on operatn algs
- GoI 3: uses additives & additive proof nets —
- GoI 4 (last month): von Neumann algebras: $Ex(f, \tau)$ for f arb (not necessarily coming from proof)
- Quantum GoI ?

Part 3

Future Directions

- GoI 2: Non-converging algms (untyped λ-calc / PCF)
  - uses more topological info on operatn algs
- GoI 3: uses additives & additive proof nets —
- GoI 4 (last month): von Neumann algebras: $Ex(f, \tau)$ for f arb (not necessarily coming from proof)
- Quantum GoI ?

Phil Scott.
Tutorial on Geometry of
Interaction, FMCS 2004.
Page 47/47

The Categorical GoI Workflow

Branching monad B

Coalgebraic trace semantics

Traced monoidal category $\mathbb{C}$

+ other constructs ➜ "GoI situation" [AHS02]

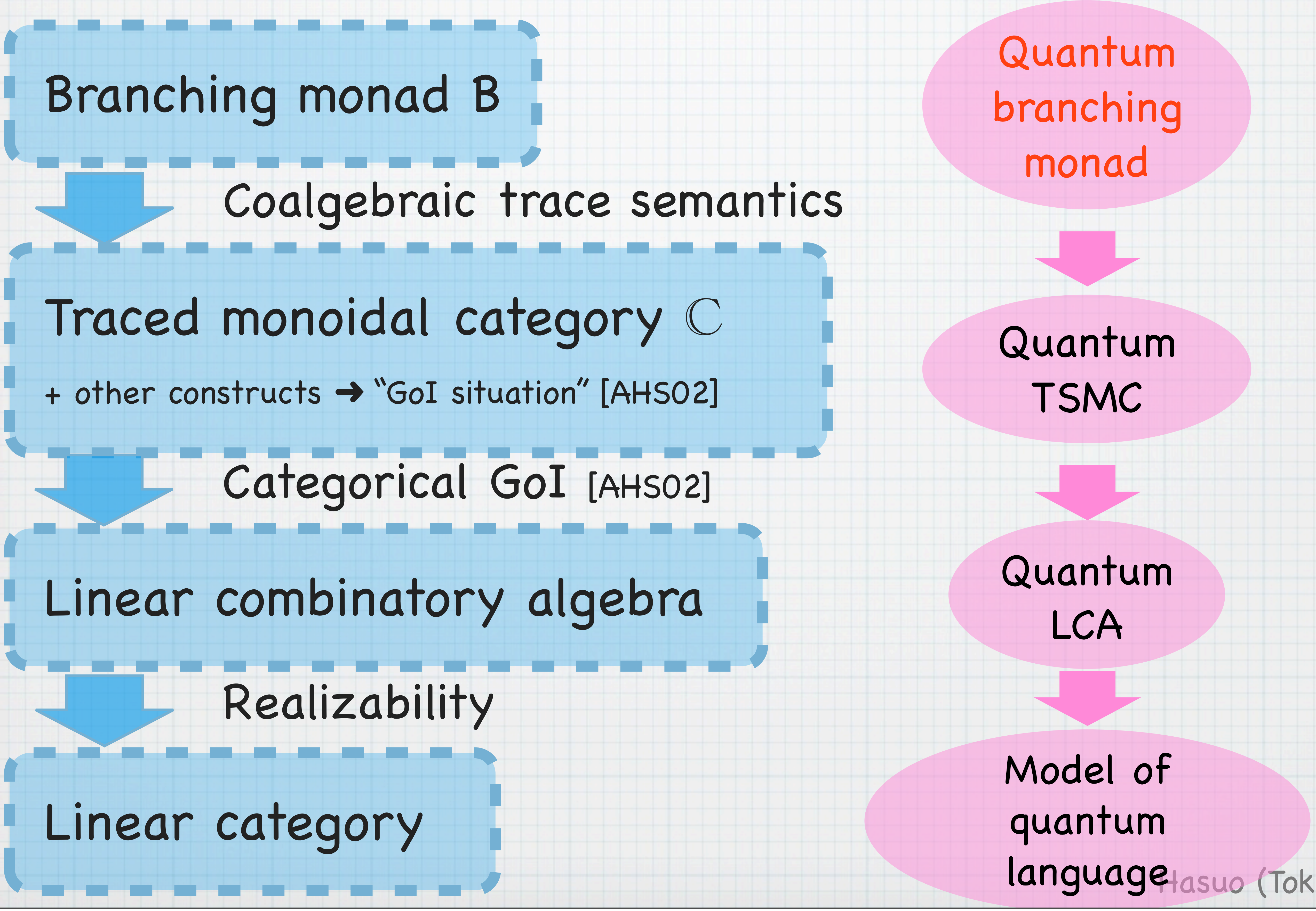
Categorical GoI [AHS02]

Linear combinatory algebra

Realizability

Linear category

The Categorical GoI Workflow



Branching monad B

Coalgebraic trace semantics

Traced monoidal category $\mathbb{C}$

+ other constructs ➜ "GoI situation" [AHS02]

Categorical GoI [AHS02]

Linear combinatory algebra

Realizability

Linear category

Quantum branching monad

Quantum TSMC

Quantum LCA

Model of quantum language

The Quantum Branching Monad

$$\boldsymbol{\mathcal{Q}Y} = \left\{ \boldsymbol{c : Y \to \prod_{m,n\in\mathbb{N}} \mathbf{QO}_{m,n}} \;\middle|\; \text{the trace condition} \right\}$$

The Quantum Branching M

$$\mathbf{QO}_{m,n} := \left\{ \begin{array}{l} \text{quantum operations,} \\ \text{from dim. } m \text{ to dim. } n \end{array} \right\}$$

$$\mathcal{Q}Y = \left\{ c : Y \to \prod_{m,n \in \mathbb{N}} \underline{\mathbf{QO}_{m,n}} \;\middle|\; \text{the trace condition} \right\}$$

The Quantum Branching M

$$\mathbf{QO}_{m,n} := \left\{ \begin{array}{l} \text{quantum operations,} \\ \text{from dim. } m \text{ to dim. } n \end{array} \right\}$$

$$QY = \left\{ c : Y \to \prod_{m,n \in \mathbb{N}} \underline{\mathbf{QO}_{m,n}} \;\middle|\; \underline{\text{the trace condition}} \right\}$$

$$\sum_{y \in Y} \sum_{n \in \mathbb{N}} \mathbf{tr}\big[(c(y))_{m,n}(\rho) \big] \le 1 \;,$$

$$\forall m \in \mathbb{N},\ \forall \rho \in D_m .$$

The Quantum Branching M

$$\mathbf{QO}_{m,n} := \left\{ \begin{array}{l} \text{quantum operations,} \\ \text{from dim. } m \text{ to dim. } n \end{array} \right\}$$

$$\mathcal{Q}Y = \left\{ c : Y \to \prod_{m,n\in\mathbb{N}} \underline{\mathbf{QO}_{m,n}} \;\middle|\; \underline{\text{the trace condition}} \right\}$$

$$\sum_{y\in Y} \sum_{n\in\mathbb{N}} \mathbf{tr}\big[\big(c(y)\big)_{m,n}(\rho)\big] \leq 1 \;,$$

$$\forall m \in \mathbb{N},\ \forall \rho \in D_m.$$

- Compare with

$$\mathcal{P}Y = \left\{ c : Y \to 2 \right\}$$

$$\mathcal{D}Y = \left\{ c : Y \to [0,1] \;\middle|\; \sum_{y\in Y} c(y) \leq 1 \right\}$$

The Quantum Branching M

$$\mathbf{QO}_{m,n} := \left\{ \begin{array}{l} \text{quantum operations,} \\ \text{from dim. } m \text{ to dim. } n \end{array} \right\}$$

$$\mathcal{Q}Y = \left\{ c : Y \to \prod_{m,n\in\mathbb{N}} \underline{\mathbf{QO}_{m,n}} \;\middle|\; \underline{\text{the trace condition}} \right\}$$

$$\sum_{y\in Y}\sum_{n\in\mathbb{N}} \mathbf{tr}\big[\big(c(y)\big)_{m,n}(\rho)\big] \leq 1 \;,$$

$$\forall m \in \mathbb{N},\ \forall \rho \in D_m .$$

- Compare with

$$\mathcal{P}Y = \left\{ c : Y \to 2 \right\}$$

$$\mathcal{D}Y = \left\{ c : Y \to [0,1] \;\middle|\; \sum_{y\in Y} c(y) \leq 1 \right\}$$

The Quantum Branching M

$$\mathbf{QO}_{m,n} := \left\{ \begin{array}{l} \text{quantum operations,} \\ \text{from dim. } m \text{ to dim. } n \end{array} \right\}$$

$$\mathcal{Q}Y = \left\{ c : Y \to \prod_{m,n\in\mathbb{N}} \underline{\mathbf{QO}_{m,n}} \;\middle|\; \underline{\text{the trace condition}} \right\}$$

$$\sum_{y\in Y}\sum_{n\in\mathbb{N}} \mathbf{tr}\left[\left(c(y)\right)_{m,n}(\rho) \right] \le 1 \ ,$$

$$\forall m\in\mathbb{N},\ \forall\rho\in D_m .$$

- Compare with

$$\mathcal{P}Y = \left\{ c : Y \to 2 \right\}$$

$$\mathcal{D}Y = \left\{ c : Y \to [0,1] \;\middle|\; \underline{\sum_{y\in Y} c(y) \le 1} \right\}$$

The Quantum Branching Monad

$$\mathcal{Q}Y = \left\{ c : Y \to \prod_{m,n\in\mathbb{N}} \mathbf{QO}_{m,n} \;\middle|\; \underline{\text{the trace condition}} \right\}$$

$$\sum_{y\in Y}\sum_{n\in\mathbb{N}} \mathbf{tr}\left[\left(c(y)\right)_{m,n}(\rho)\right] \le 1 \; , \qquad \forall m \in \mathbb{N},\ \forall \rho \in D_m .$$

$$\begin{array}{c} X \xrightarrow{f} Y \ \text{ in } \mathcal{K}\ell(\mathcal{Q}) \\ \hline\hline X \xrightarrow{f} \mathcal{Q}Y \ \text{ in } \mathbf{Sets} \end{array}$$

- Given $x \in X$, $y \in Y$, $m \in \mathbb{N}$, $n \in \mathbb{N}$ determines a quantum operation

$$\Big(f(x)(y) \Big)_{m,n} \; : \; D_m \to D_n$$

- Subject to the **trace condition**

The Quantum Branching Monad

$$\mathcal{Q}Y = \left\{ c : Y \to \prod_{m,n\in\mathbb{N}} \mathbf{QO}_{m,n} \;\middle|\; \text{the trace condition} \right\}$$

$$\sum_{y\in Y}\sum_{n\in\mathbb{N}} \mathbf{tr}\big[(c(y))_{m,n}(\rho)\big] \le 1\ ,$$

$$\forall m\in\mathbb{N},\ \forall\rho\in D_m .$$

$$\begin{array}{c} X \xrightarrow{f} Y \ \text{ in } \mathcal{K}\ell(\mathcal{Q}) \\ \hline\hline X \xrightarrow{f} \mathcal{Q}Y \ \text{ in } \mathbf{Sets} \end{array}$$

- Given $x \in X$, $y \in Y$, $m \in \mathbb{N}$, $n \in \mathbb{N}$ determines a quantum operation

$$\Big(\, f(x)(y) \,\Big)_{m,n} \ :\ D_m \to D_n$$

- Subject to the **trace condition**

Any opr. on quantum data:

combination of

- preparation
- unitary transf.
- measurement

The Quantum Branching Monad

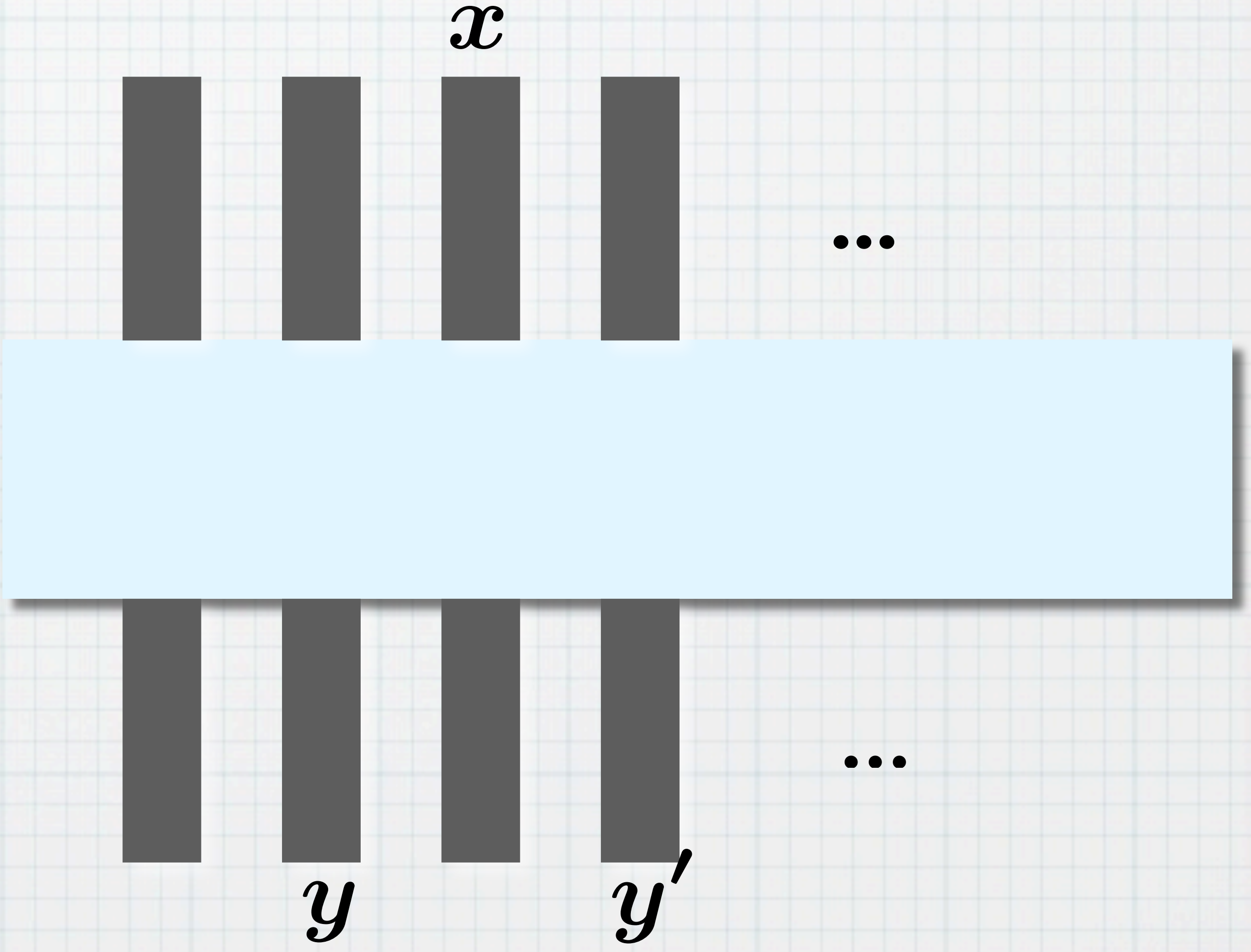
$$\frac{X \xrightarrow{f} Y \text{ in } \mathcal{K}\ell(\mathcal{Q})}{X \xrightarrow{f} \mathcal{Q}Y \text{ in } \mathbf{Sets}}$$

$$\mathcal{Q}Y = \left\{ c : Y \to \prod_{m,n \in \mathbb{N}} \mathbf{QO}_{m,n} \;\middle|\; \text{the trace condition} \right\}$$

$$\sum_{y \in Y} \sum_{n \in \mathbb{N}} \mathbf{tr}\left[\big(c(y)\big)_{m,n}(\rho) \right] \leq 1 \ ,$$

$$\forall m \in \mathbb{N},\ \forall \rho \in D_m.$$

- Given $x \in X$, $y \in Y$, $m \in \mathbb{N}$, $n \in \mathbb{N}$ determines a quantum operation $\Big(f(x)(y) \Big)_{m,n}$
- **trace cond.:**



The Quantum Branching Monad

$$\mathcal{Q}Y = \Big\{ c : Y \to \prod_{m,n\in\mathbb{N}} \mathbf{QO}_{m,n} \;\Big|\; \text{the trace condition} \Big\}$$

$$\sum_{y\in Y}\sum_{n\in\mathbb{N}} \mathbf{tr}\big[\big(c(y)\big)_{m,n}(\rho)\big] \le 1 \ ,$$

$$\forall m \in \mathbb{N},\ \forall \rho \in D_m .$$

$$\begin{array}{c} X \xrightarrow{f} Y \ \text{ in } \mathcal{K}\ell(\mathcal{Q}) \\ \hline\hline X \xrightarrow{f} \mathcal{Q}Y \ \text{ in Sets} \end{array}$$

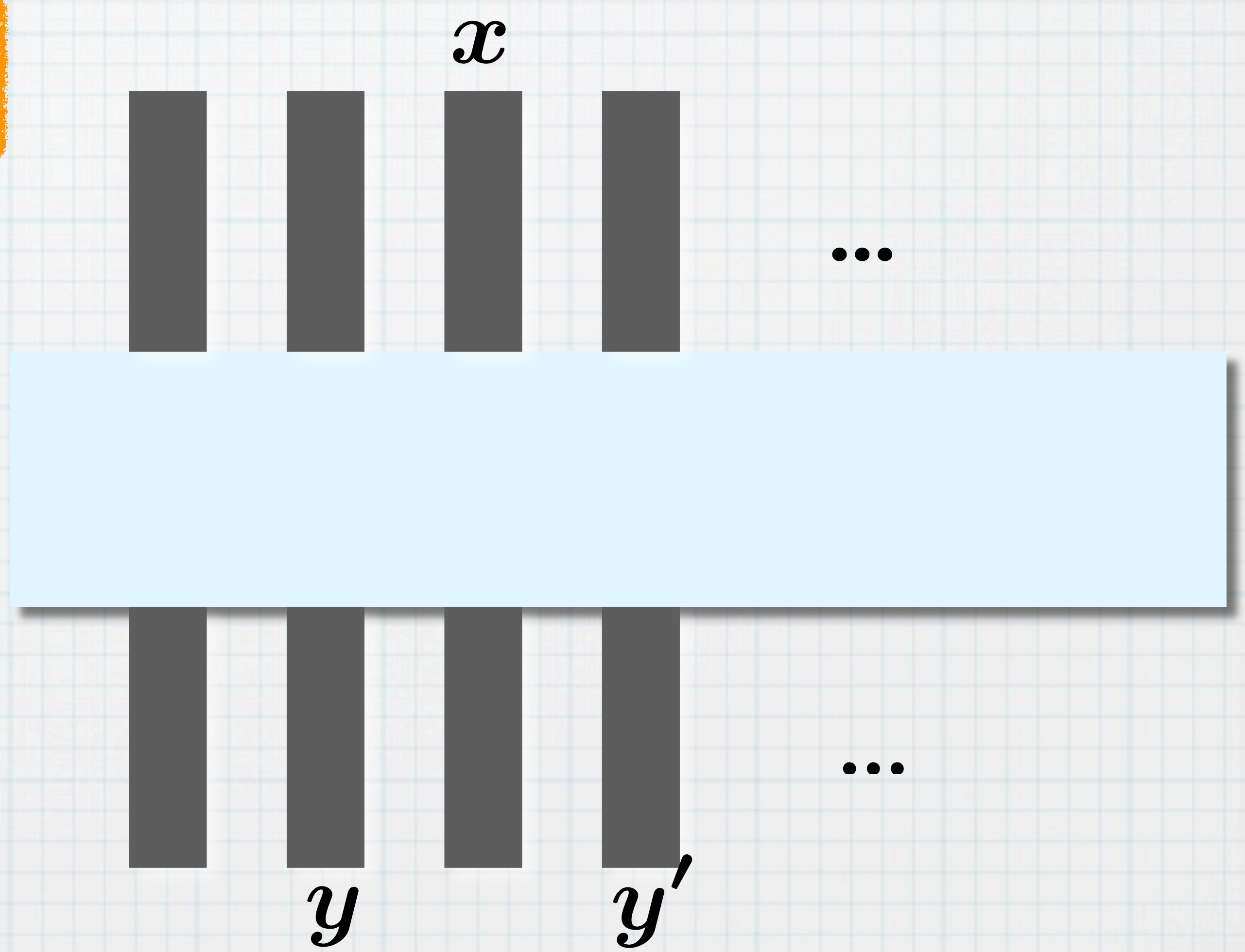
entrance

exit

in dim.

out dim.

- Given $x \in X$, $y \in Y$, $m \in \mathbb{N}$, $n \in \mathbb{N}$ determines a quantum operation $\Big(f(x)(y)\Big)_{m,n}$
- **trace cond.:**



The Quantum Branching Monad

$$\mathcal{Q}Y = \left\{ c : Y \to \prod_{m,n\in\mathbb{N}} \mathbf{QO}_{m,n} \;\middle|\; \text{the trace condition} \right\}$$

$$\sum_{y\in Y}\sum_{n\in\mathbb{N}} \mathbf{tr}\bigl[\bigl(c(y)\bigr)_{m,n}(\rho)\bigr] \le 1 \ , \qquad \forall m\in\mathbb{N},\ \forall \rho\in D_m .$$

$$\begin{array}{c} X \xrightarrow{f} Y \ \text{ in } \mathcal{K}\ell(\mathcal{Q}) \\ \hline\hline X \xrightarrow{f} \mathcal{Q}Y \ \text{ in Sets} \end{array}$$

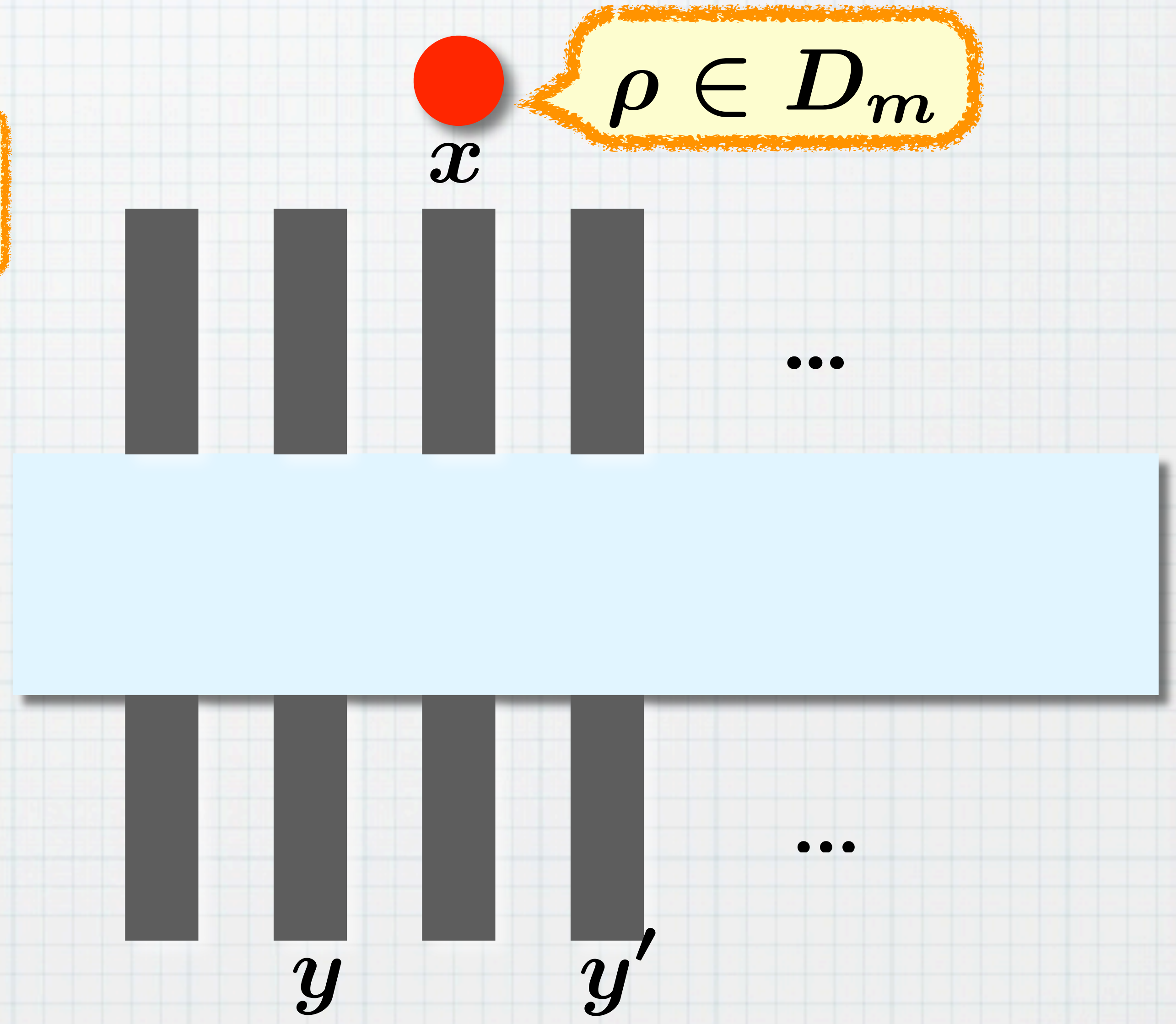
entrance

exit

in dim.

out dim.

- Given $x \in X,\ y \in Y,\ m \in \mathbb{N},\ n \in \mathbb{N}$ determines a quantum operation $\bigl(f(x)(y)\bigr)_{m,n}$
- **trace cond.:**



$\rho \in D_m$

The Quantum Branching Monad

$$\mathcal{Q}Y = \left\{ c : Y \to \prod_{m,n\in\mathbb{N}} \mathbf{QO}_{m,n} \;\middle|\; \text{the trace condition} \right\}$$

$$\sum_{y\in Y}\sum_{n\in\mathbb{N}} \mathbf{tr}\big[\big(c(y)\big)_{m,n}(\rho)\big] \le 1 \ ,$$

$$\forall m\in\mathbb{N},\ \forall\rho\in D_m .$$

$$\begin{array}{c} X \xrightarrow{f} Y \ \text{ in } \mathcal{K}\ell(\mathcal{Q}) \\ \hline\hline X \xrightarrow{f} \mathcal{Q}Y \ \text{ in Sets} \end{array}$$

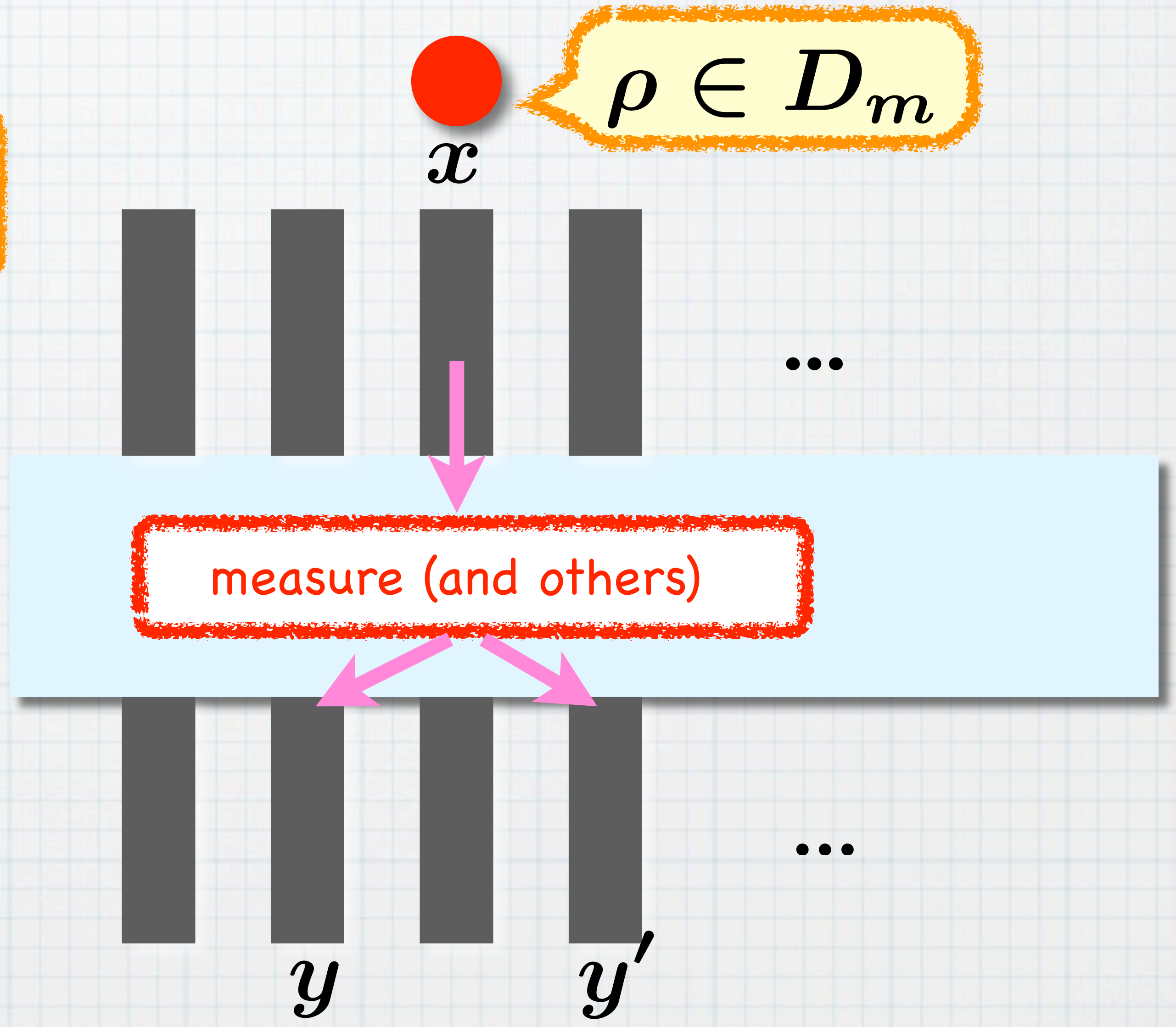
entrance

exit

in dim.

out dim.

- Given $x\in X,\ y\in Y,\ m\in\mathbb{N},\ n\in\mathbb{N}$ determines a quantum operation $\Big(f(x)(y)\Big)_{m,n}$
- **trace cond.:**



The Quantum Branching Monad

$$\mathcal{Q}Y = \left\{ c : Y \to \prod_{m,n \in \mathbb{N}} \mathbf{QO}_{m,n} \;\middle|\; \text{the trace condition} \right\}$$

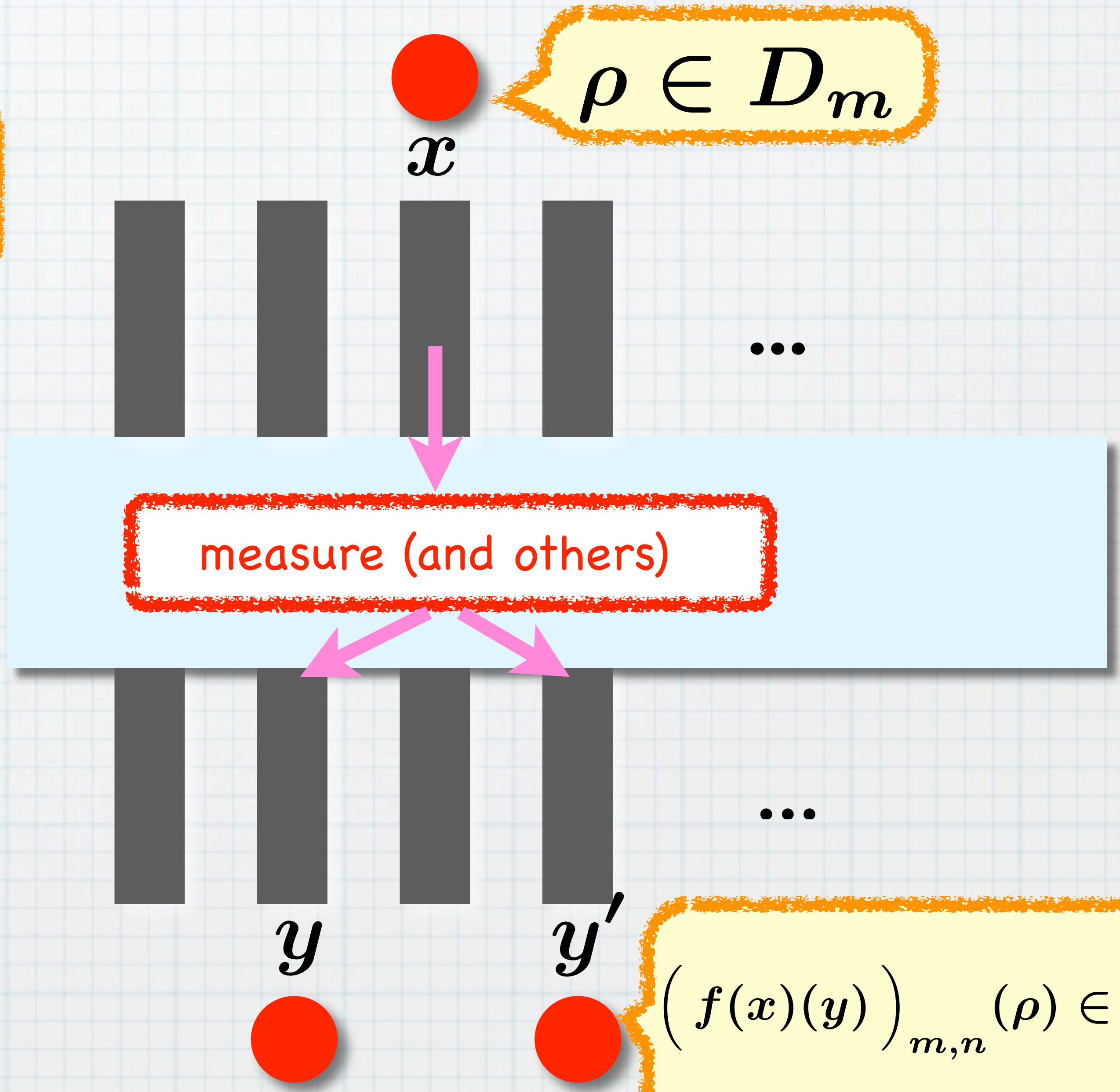
$$\sum_{y \in Y} \sum_{n \in \mathbb{N}} \mathbf{tr}\big[\big(c(y)\big)_{m,n}(\rho)\big] \leq 1 \;,$$

$$\forall m \in \mathbb{N},\ \forall \rho \in D_m .$$

$$\begin{array}{c} X \xrightarrow{f} Y \ \text{ in } \mathcal{K}\ell(\mathcal{Q}) \\ \hline\hline X \xrightarrow{f} \mathcal{Q}Y \ \text{ in Sets} \end{array}$$

entrance exit in dim. out dim.

- Given $x \in X,\ y \in Y,\ m \in \mathbb{N},\ n \in \mathbb{N}$ determines a quantum operation $\big(f(x)(y)\big)_{m,n}$
- **trace cond.:**



$\rho \in D_m$

$\big(f(x)(y)\big)_{m,n}(\rho) \in D_n$

for each n

The Quantum Branching Monad

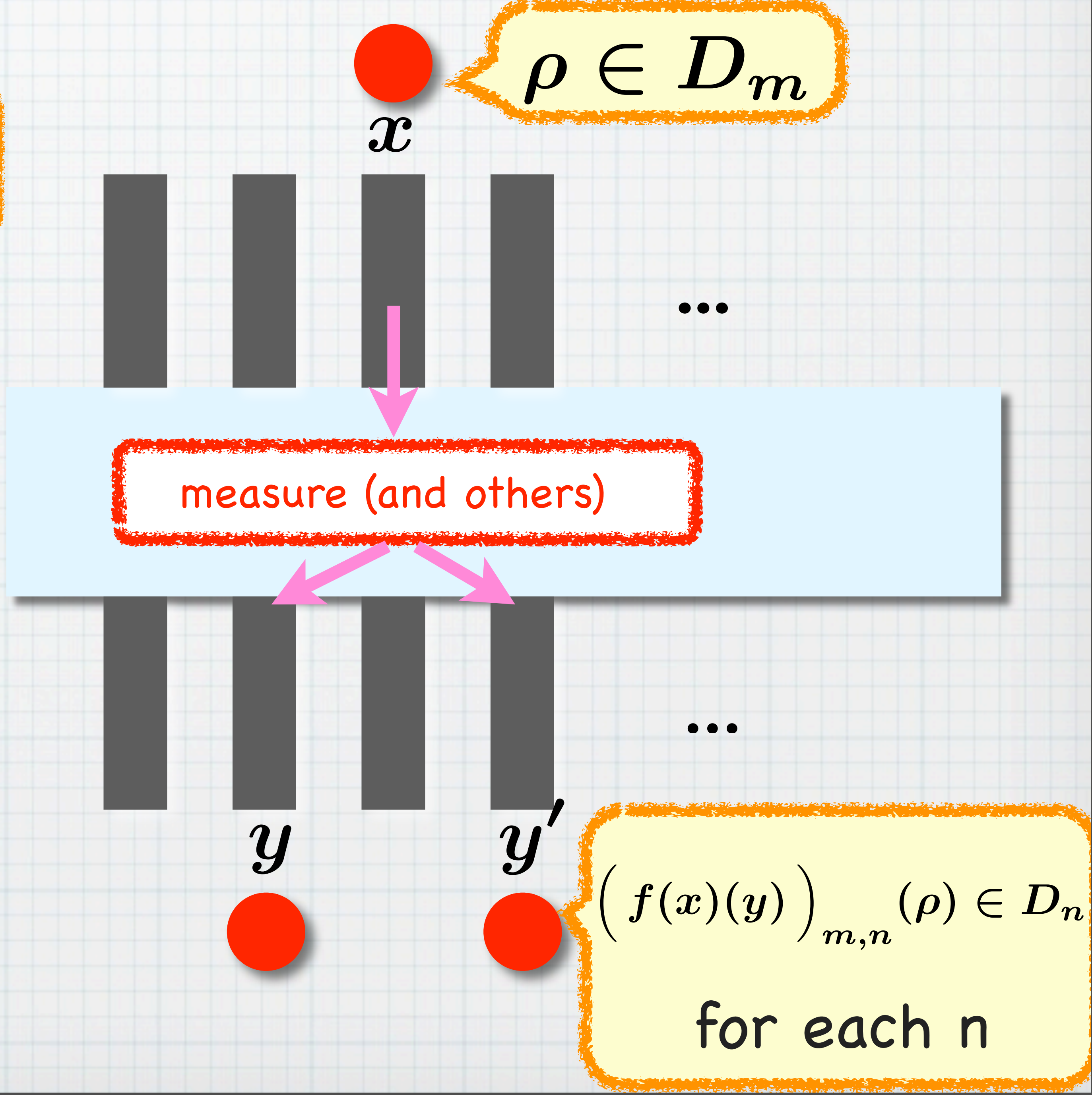
$$\mathcal{Q}Y = \left\{ c : Y \to \prod_{m,n\in\mathbb{N}} \mathbf{QO}_{m,n} \;\middle|\; \text{the trace condition} \right\}$$

$$\sum_{y\in Y}\sum_{n\in\mathbb{N}} \mathbf{tr}\big[(c(y))_{m,n}(\rho)\big] \le 1\ ,$$
$$\forall m\in\mathbb{N},\ \forall \rho\in D_m .$$

$$\begin{array}{c} X \xrightarrow{f} Y \ \text{ in } \mathcal{K}\ell(\mathcal{Q}) \\ \hline\hline X \xrightarrow{f} \mathcal{Q}Y \ \text{ in Sets} \end{array}$$

- Given $\overset{\text{entrance}}{x \in X}$, $\overset{\text{exit}}{y \in Y}$, $\overset{\text{in dim.}}{m \in \mathbb{N}}$, $\overset{\text{out dim.}}{n \in \mathbb{N}}$ determines a quantum operation $\big(f(x)(y)\big)_{m,n}$
- **trace cond.:**

$$\sum_{y,n} \Pr\left(\begin{array}{c}\text{Token led}\\ \text{to } y\\ \text{with dim. } n\end{array}\right) \le 1$$

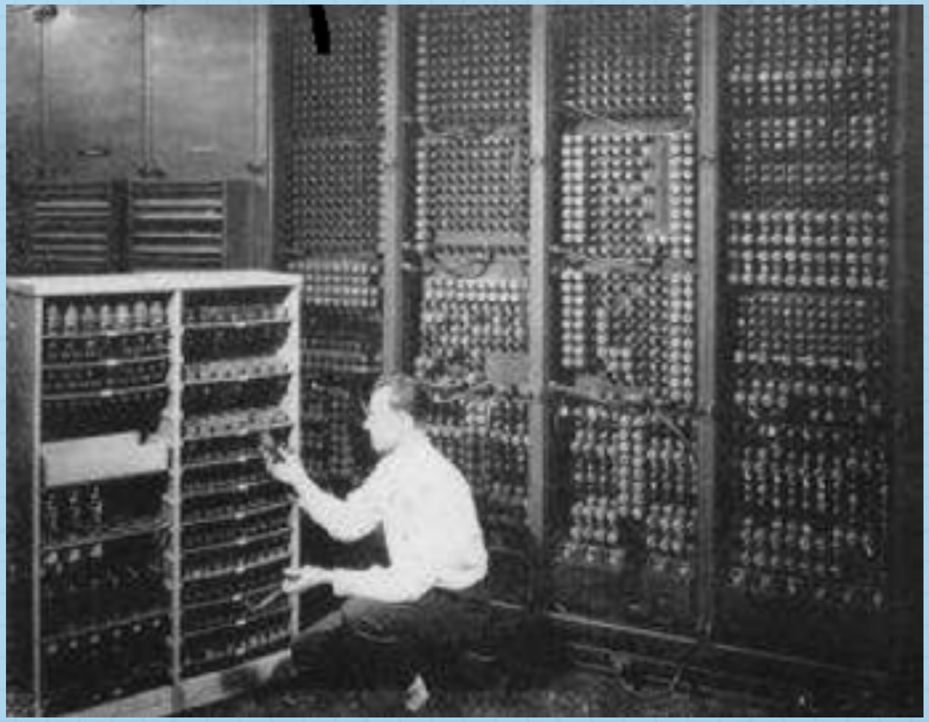


$\big(f(x)(y)\big)_{m,n}(\rho) \in D_n$

for each n

“Quantum Data, Classical Control”

Illustration by N. Hoshino



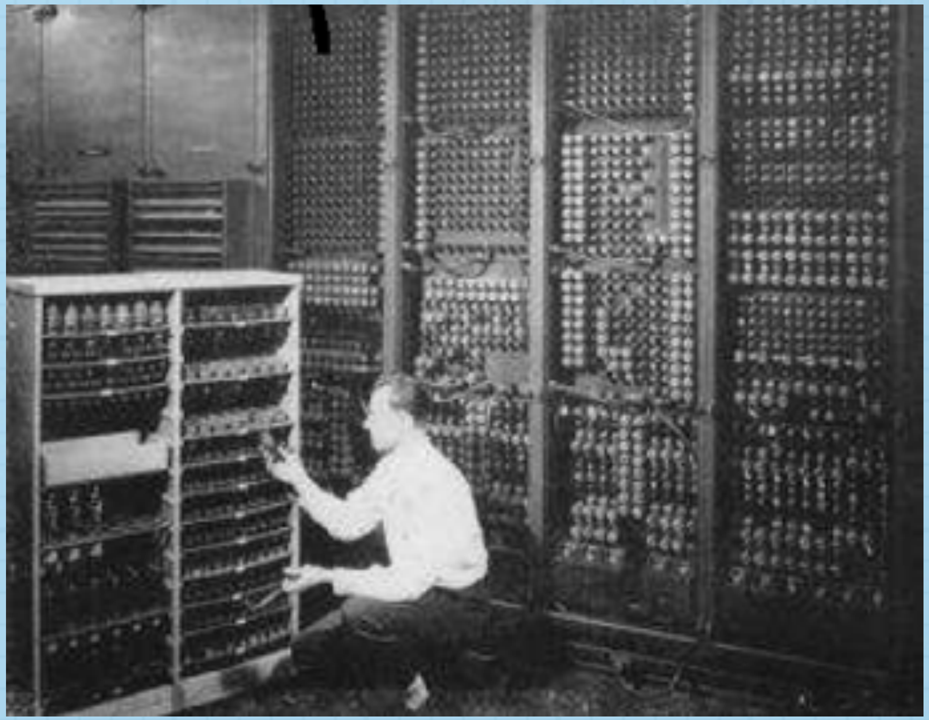
“Quantum Data, Classical Control”

Illustration by N. Hoshino

Quantum data

$$\frac{1}{\sqrt{2}}$$

Classical control



"Quantum Data, Classical Control"

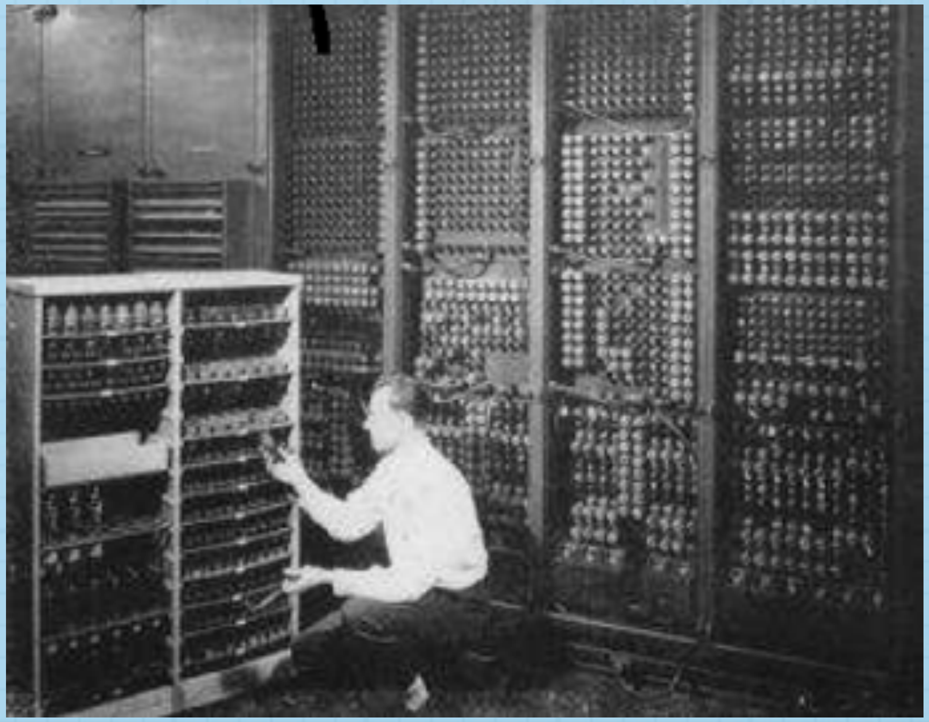
Illustration by N. Hoshino

Quantum data

$\frac{1}{\sqrt{2}}$

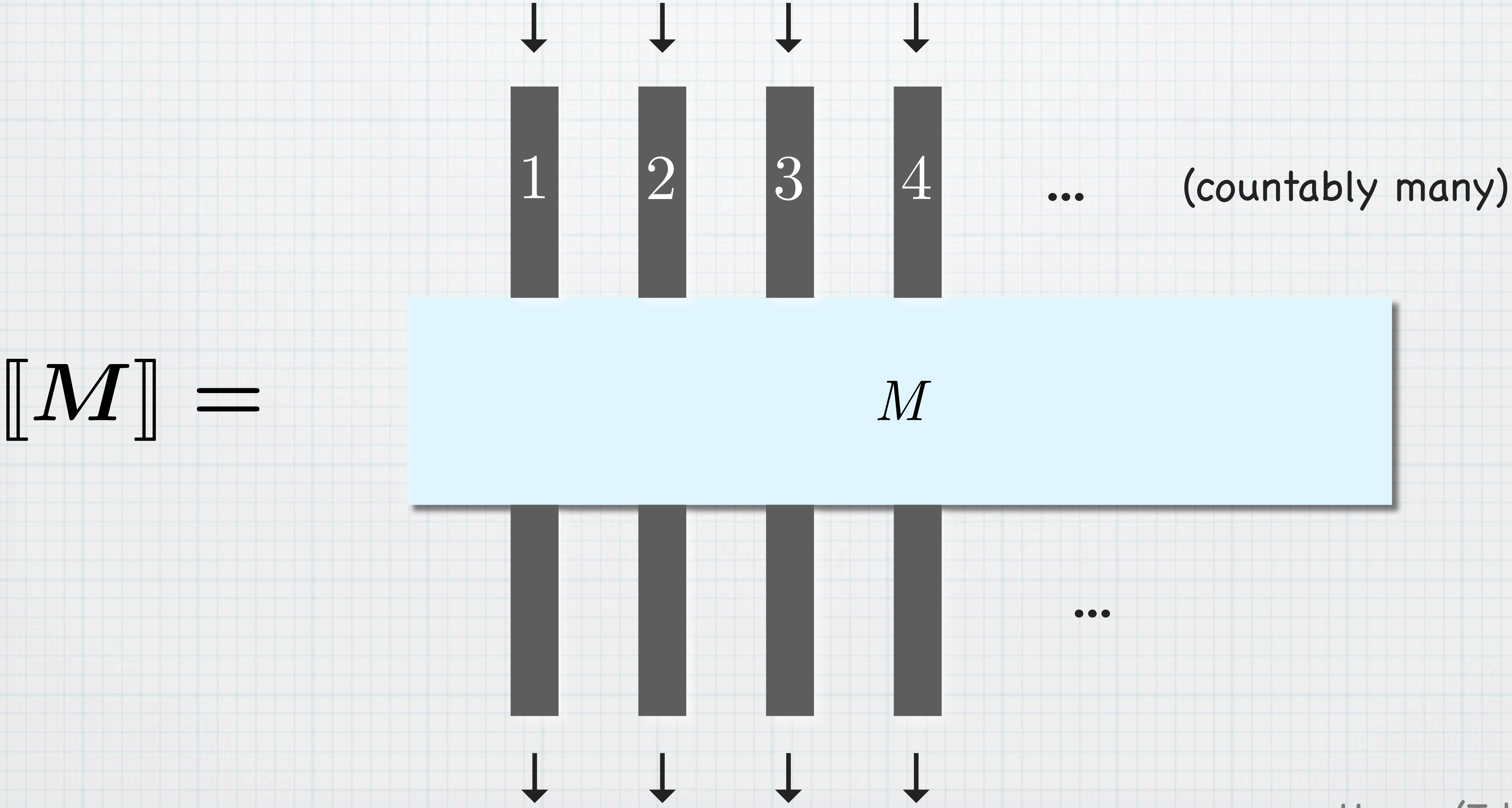
$+\frac{1}{\sqrt{2}}$

Classical control



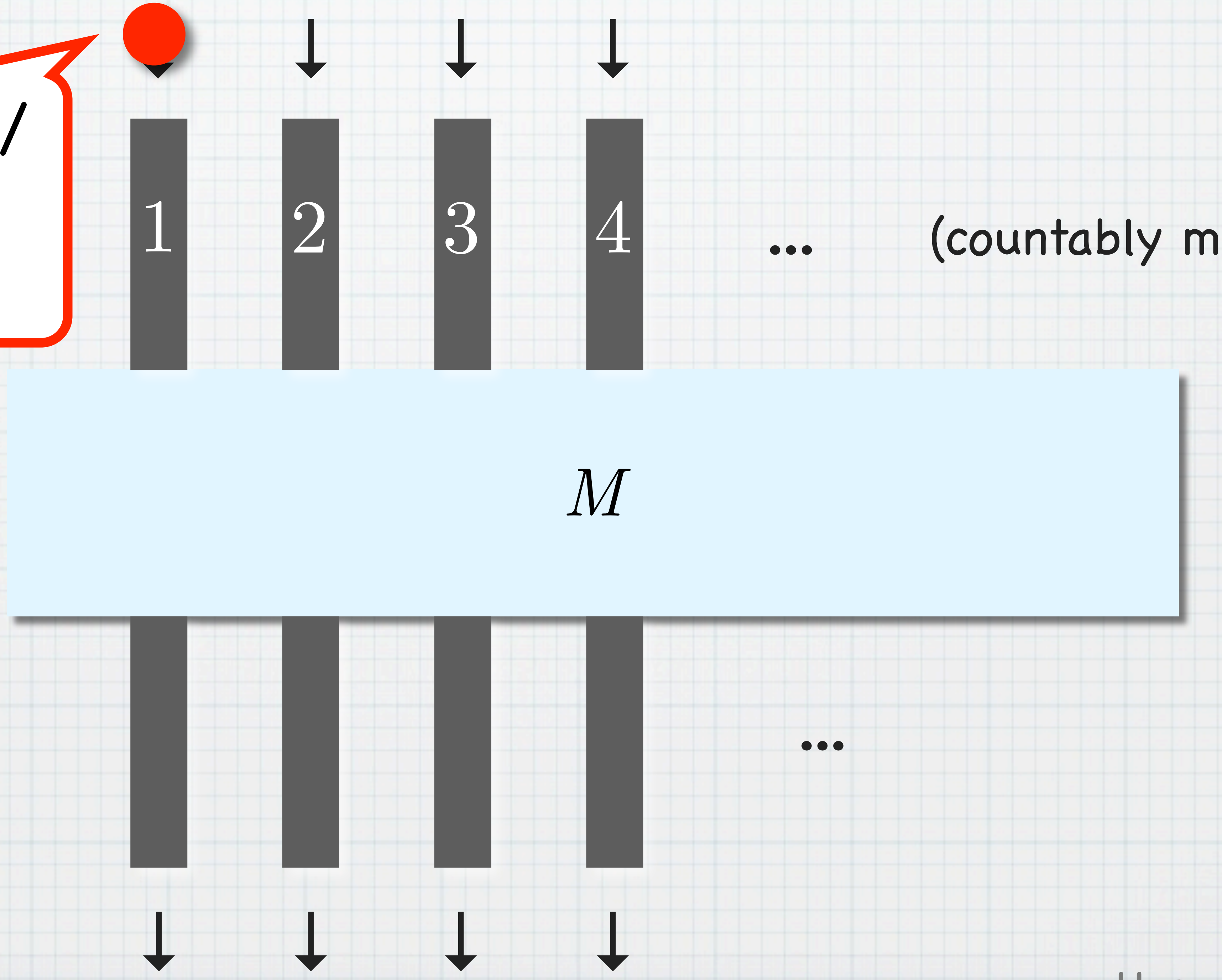
Quantum

Geometry of Interaction



Quantum
Geometry of Interaction

Not just a token/ particle, but quantum state!

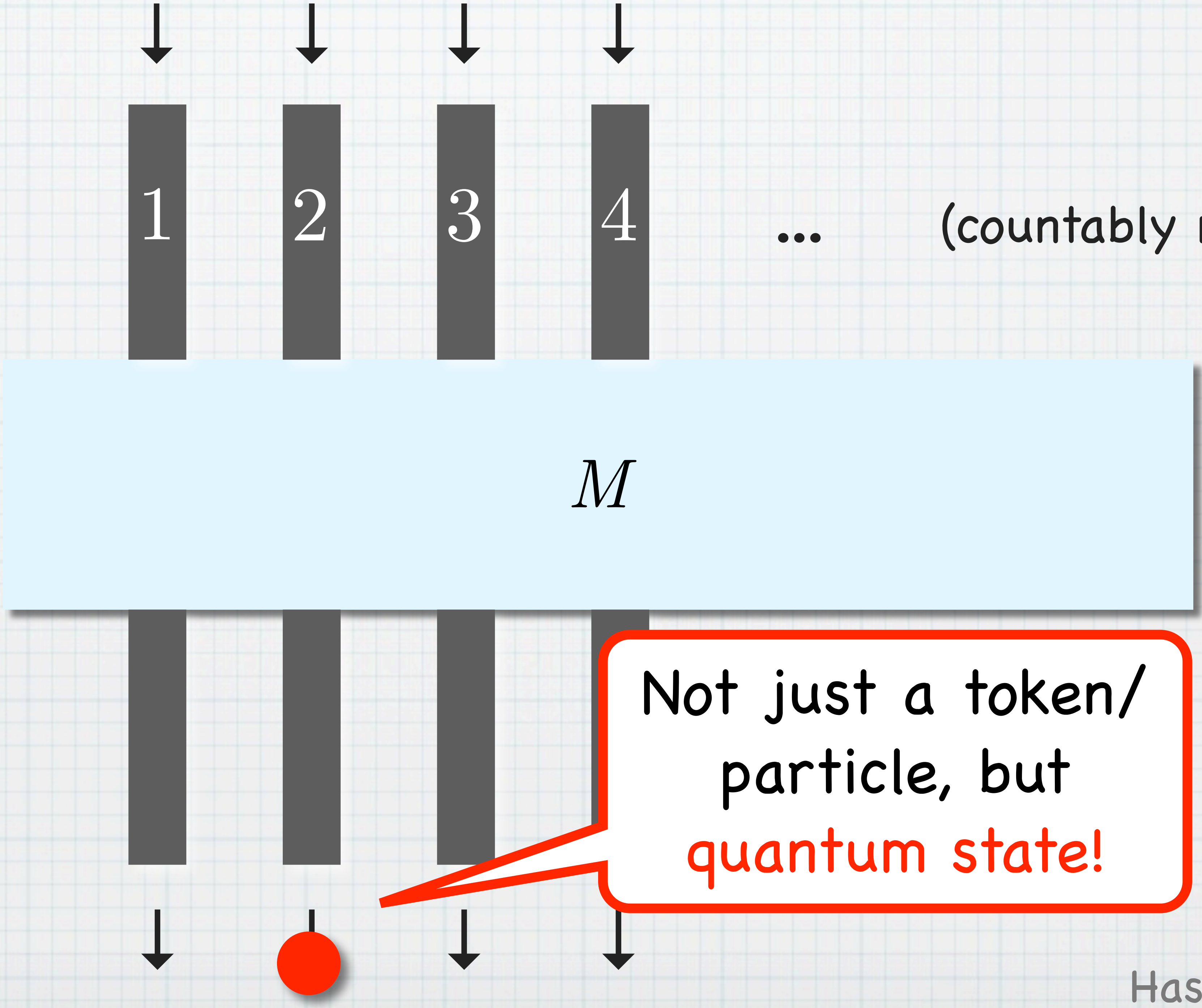


(countably many)

$$[\![M]\!] =$$

Quantum
Geometry of Interaction

$[\![M]\!] =$



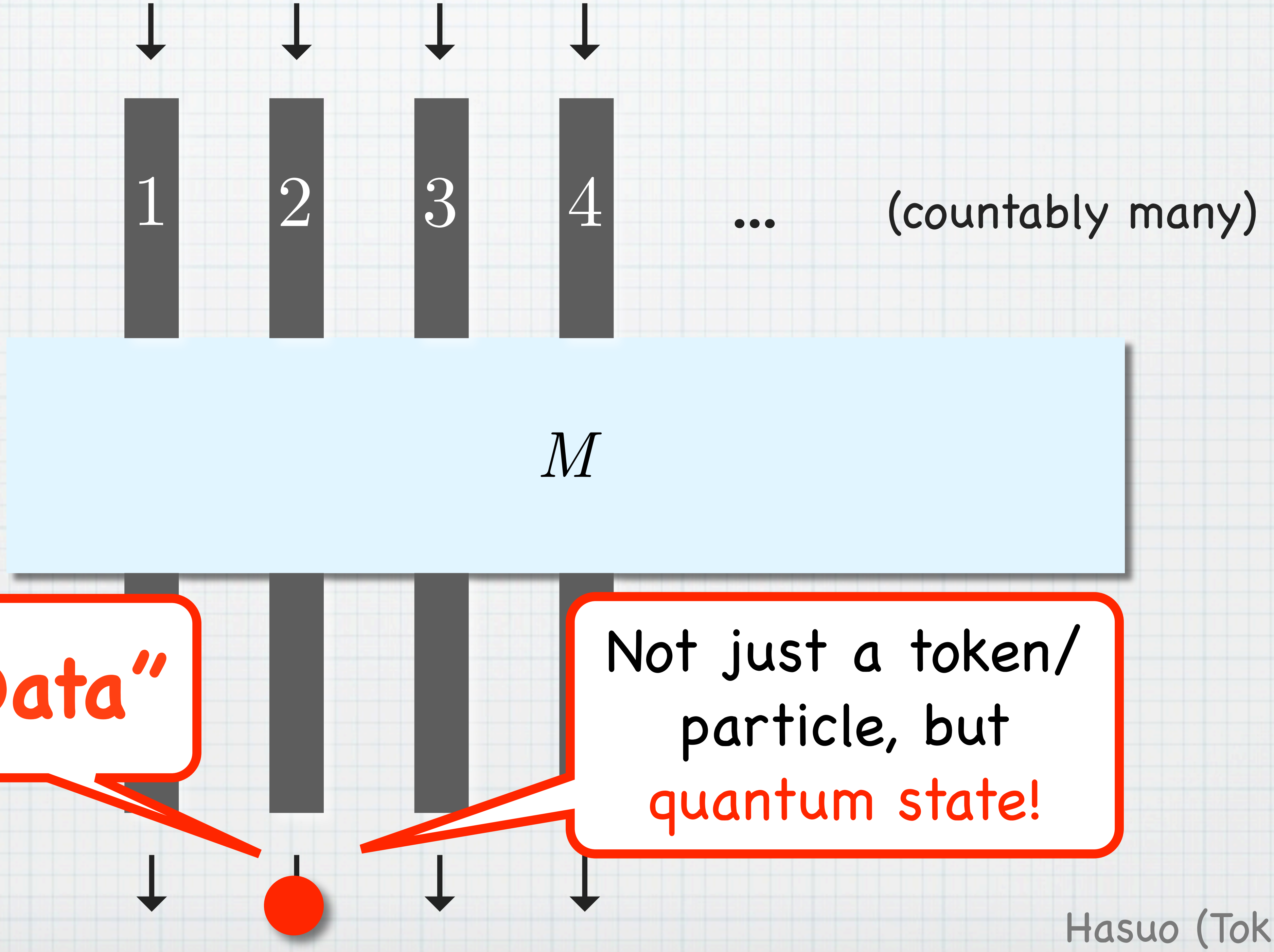
(countably many)

Not just a token/ particle, but quantum state!

Hasuo (Tokyo)

Quantum
Geometry of Interaction

$$[\![M]\!] =$$

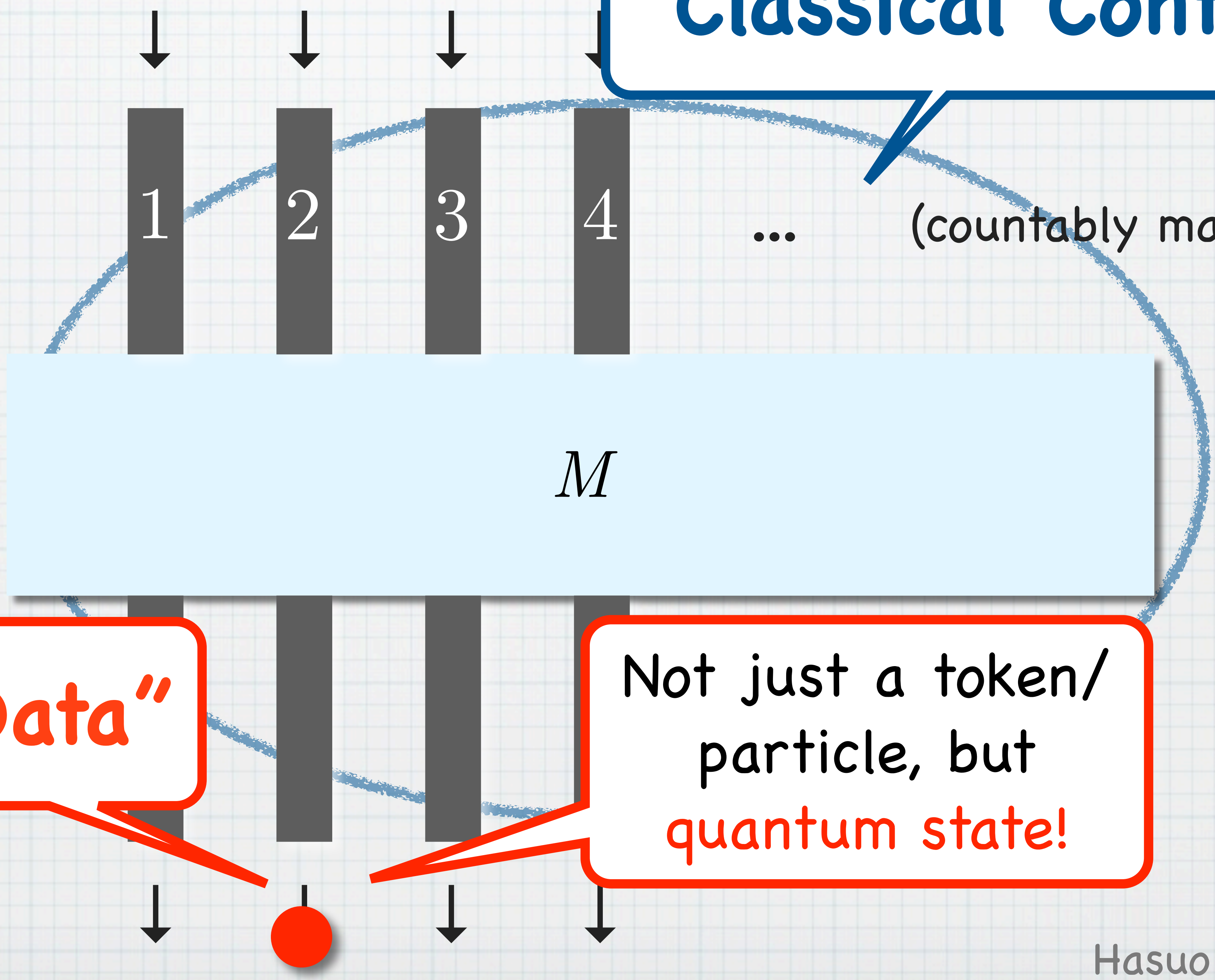


"Quantum Data"

Not just a token/ particle, but quantum state!

Quantum
Geometry of Interaction

$[\![M]\!] =$



"Classical Control"

(countably many)

"Quantum Data"

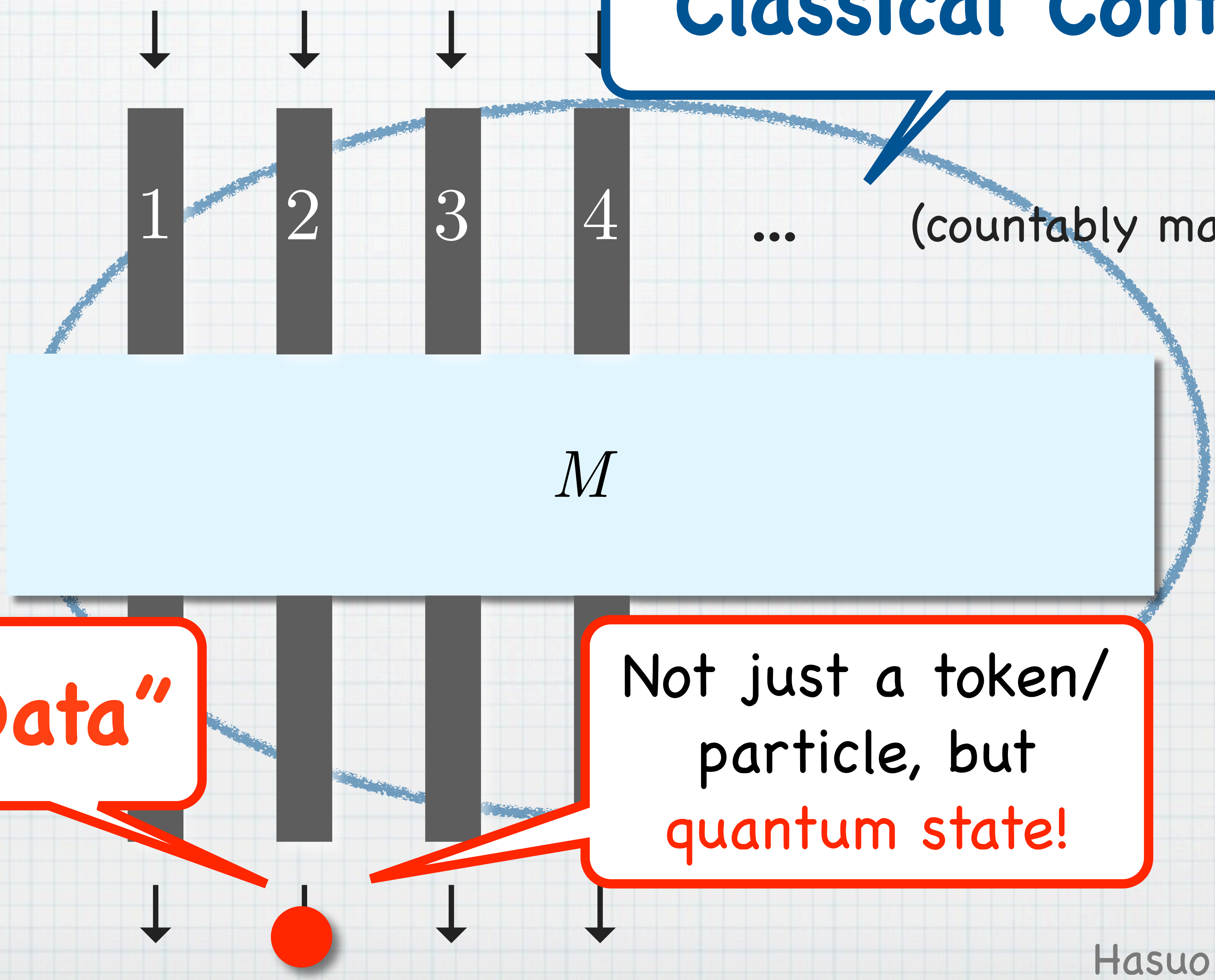
Not just a token/ particle, but quantum state!

Qua
Geometry o

- “in which pipe”
- (measurement ➜ case-distinction) leads a token to different pipes

“Classical Control”

$$[\![M]\!] =$$



(countably many)

“Quantum Data”

Not just a token/ particle, but quantum state!

End of the Story?

- No! All the technicalities are yet to come:
  - CPS-style interpretation (for partial measurement)
  - Result type: a final coalgebra in $\mathbf{PER}_Q$
  - **Admissible PERs** for recursion
  - …
- On the next occasion :-)

Results

- The monad Q qualifies as a "branching monad"
- The quantum GoI workflow leads to a linear category $\mathbf{PER}_Q$
- From which we construct an adequate denotational model for a quantum λ-calculus (a variant of Selinger & Valiron's)

Three "Traces"

$$\begin{tikzcd} \boldsymbol{FX} \arrow[r, dashed, "\boldsymbol{F}\mathbf{beh}(\boldsymbol{c})"] & \boldsymbol{FZ} \\ \boldsymbol{X} \arrow[u, "\boldsymbol{c}"] \arrow[r, dashed, "\mathbf{beh}(\boldsymbol{c})"'] & \boldsymbol{Y} \arrow[u, "\text{final}"'] \end{tikzcd}$$

Coinduction in $\boldsymbol{Kl}(\mathbf{B})$

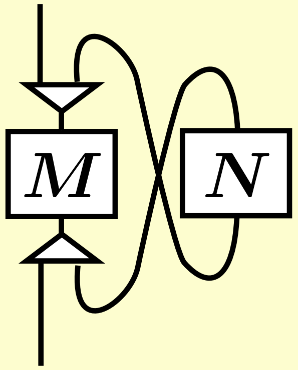
Coalgebraic **Trace** Semantics

appl.

Traced monoidal category

Categorical GoI

[Abramsky, Haghverdi, Scott]



Quantum λ-calculus

Measurements by **tracing out** matrices

Conclusions & Future Work

- Coalgebraic technologies in **interaction-based denotational semantics**
  - GoI, games (AJM/HO), token machines, ...
- Dynamic/operational stuff: not only in concurrency theory!
- Simplifying our model; lang. w/ "quantum store"
  - Ongoing w/ N. Hoshino, T. Roussel, C. Faggian

Thank you for your attention!

Ichiro Hasuo (Dept. CS, U Tokyo)

http://www-mmm.is.s.u-tokyo.ac.jp/~ichiro/

Conclusions & Future Work

- Coalgebraic technologies in **interaction-based denotational semantics**
  - GoI, games (AJM/HO), token machines, ...
- Dynamic/operational stuff: not only in concurrency theory!
- Simplifying our model; lang. w/ "quantum store"
  - Ongoing w/ N. Hoshino, T. Roussel, C. Faggian