Diagrammatic Process Theory as a Logic for Social Interaction

Bob Coecke
University of Oxford

Diagrammatic Process Theory as a Logic for Social Interaction

Bob Coecke
University of Oxford

Towards model independent compositional reasoning about social behaviours.

Diagrammatic Process Theory as a Logic for Social Interaction

Bob Coecke
University of Oxford

Towards model independent compositional reasoning about social behaviours. Candidate models:

- **plain statistical data**
- **theoretical models**

Our starting point is the common structure of:

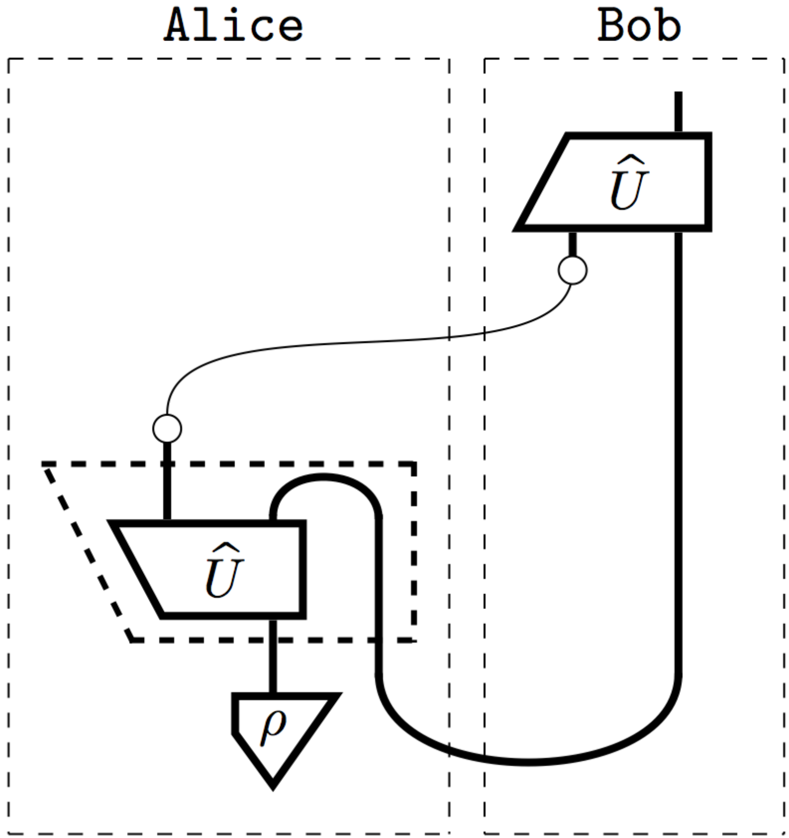
Our starting point is the common structure of:

- how quantum systems interact
- how meanings in natural language interact

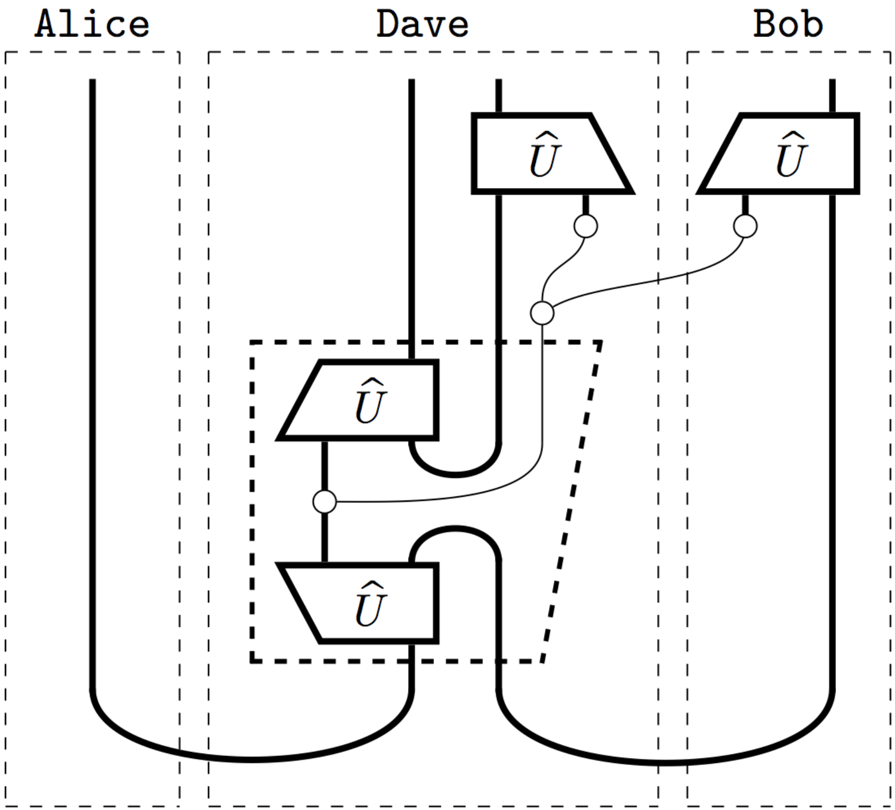
Exploring connections:

- BC (2012) The logic of quantum mechanics - Take II. arXiv:1204.3458
- S. Clark, BC, E. Grefenstette, S. Pulman & M. Sadrzadeh (2013) *A quantum teleportation inspired algorithm produces sentence meaning from word meaning and grammatical structure*. arXiv:1305.0556

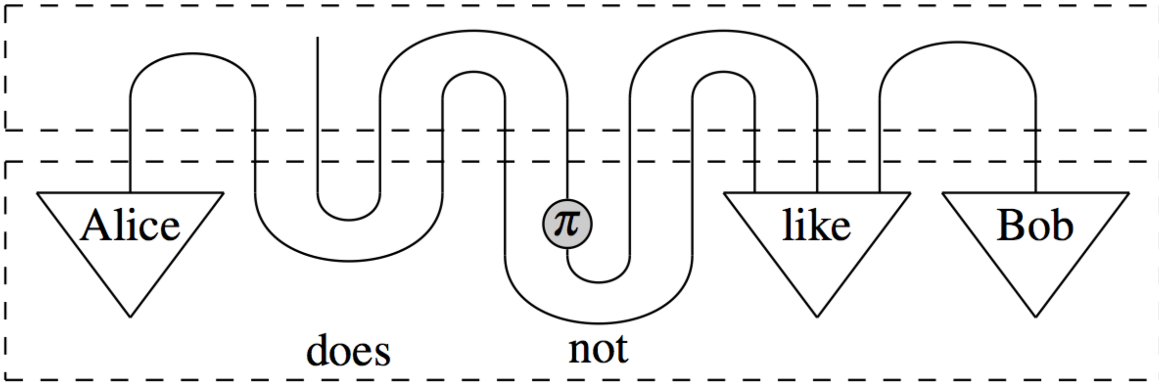
– e.g. quantum teleportation –

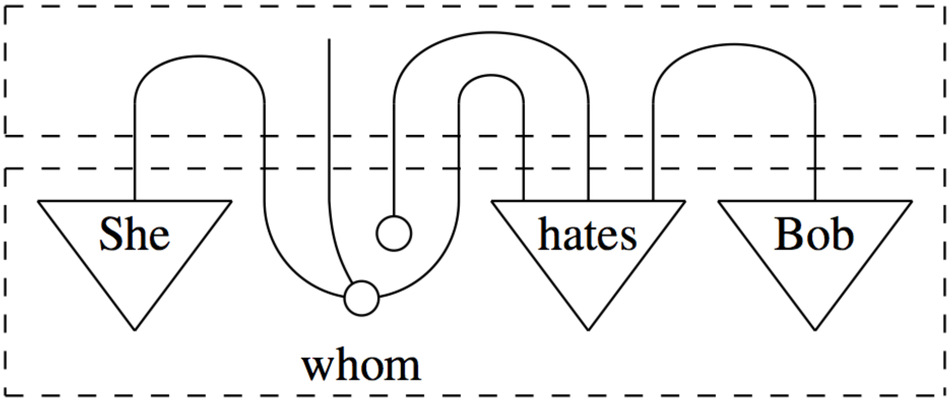


– e.g. entanglement swapping –



– e.g. negation and relative pronouns –





SCIENTIFIC AMERICAN™

Sign In / Register

Search ScientificAmerican.com

Quantum Mechanical Words and Mathematical Organisms

By Joselle Kehoe | May 16, 2013 | 10

FQXI ARTICLE

September 29, 2013

Video Article: The Quantum Linguist

Bob Coecke has developed a new visual language that could be used to spell out a theory of quantum gravity—and help us understand human speech.

by Sophie Hebden

QUANTUM LINGUISTICS Leap forward for artificial intelligence

NewScientist

WEEKLY 11 December 2010

The overarching framework:

The overarching framework:

- **process theories**
- **purely diagrammatic reasoning**

Forthcoming book (750 pp):

- BC & Aleks Kissinger
  Picturing Quantum Processes
  Cambridge University Press, spring 2015

Also in the scope of the framework:

Also in the scope of the framework:

- **animal behaviour and evolution**

Forthcoming paper:

- BC (2014) *In the beginning God created* $\otimes$. In: The Incomputable, S. B. Cooper & S. Soskova, Eds. Springer.

Also in the scope of the framework:

- **animal behaviour and evolution**

Forthcoming paper:

- BC (2014) *In the beginning God created* $\otimes$. In: The Incomputable, S. B. Cooper & S. Soskova, Eds. Springer.

. . . evidently exactly the same:

- **social behaviour and development**

Initial question (quant-ph/0510032):

Initial question (quant-ph/0510032):

$\boxed{\textbf{Can QM be formulated in pictures?}}$

Initial question (quant-ph/0510032):

$$\boxed{\textbf{Can QM be formulated in pictures?}}$$

Same question, put differently:

- **Can QM be formulated in terms of $\otimes$?**

Initial question (quant-ph/0510032):

$\boxed{\textbf{Can QM be formulated in pictures?}}$

Same question, put differently:

- **Can QM be formulated in terms of $\otimes$?**

  (contra: $\mathbb{C}$, $+$, matrices, ...)

Initial question (quant-ph/0510032):

$$\boxed{\textbf{Can QM be formulated in pictures?}}$$

Same question, put differently:

- **Can QM be formulated in terms of $\otimes$?**

  (contra: $\mathbb{C}$, $+$, matrices, ...)

- **Can QM be formulated in terms of processes?**

Initial question (quant-ph/0510032):

$\boxed{\textbf{Can QM be formulated in pictures?}}$

Same question, put differently:

- **Can QM be formulated in terms of $\otimes$?**

  (contra: $\mathbb{C}$, +, matrices, ...)

- **Can QM be formulated in terms of processes?**

  (contra: states, numbers)

Initial question (quant-ph/0510032):

$\boxed{\textbf{Can QM be formulated in pictures?}}$

Same question, put differently:

- **Can QM be formulated in terms of $\otimes$?**
  (contra: $\mathbb{C}$, +, matrices, ...)

- **Can QM be formulated in terms of processes?**
  (contra: states, numbers)

- **Does QM have logic features?**

Initial question (quant-ph/0510032):

$\boxed{\textbf{Can QM be formulated in pictures?}}$

Same question, put differently:

- **Can QM be formulated in terms of $\otimes$?**
  (contra: $\mathbb{C}$, $+$, matrices, ...)

- **Can QM be formulated in terms of processes?**
  (contra: states, numbers)

- **Does QM have logic features?**
  (contra: failures)

Category-theoretic underpinning:

Abramsky, S., and Coecke, B. (2004) A categorical semantics of quantum protocols. LICS. arXiv:quant-ph/0402130.

Selinger, P. (2007) Dagger compact closed categories and completely positive maps. ENTCS.

Coecke, B., and **Pavlovic**, D. (2007) Quantum measurements without sums. In: Mathematics of Quantum Computing and Technology. Taylor and Francis. arXiv:quant-ph/0608035

Coecke, B., and **Duncan**, R. (2008) Interacting quantum observables. ICALP'08 & NJP'10. arXiv:quant- ph/09064725

Coecke, B., **Paquette**, E. O., and Pavlovic, D. (2010) Classical and quantum structuralism. In: Semantic Techniques in Quantum Computation. CUP. arXiv:0904.1997

Chiribella, G., **D'Ariano**, G. M., and **Perinotti**, P. (2010) Probabilistic theories with purification. Physical Review. arXiv:0908.1583

. . . mainly borrowing from Australians:

Kelly, M. (1972) Many-variable functorial calculus I. LNM.

Carboni, A., and **Walters**, R. F. C. (1980) Cartesian bicategories I. JPAA.

Joyal, A., and **Street**, R. (1991) The geometry of tensor calculus I. AM.

Lack, S. (2004) Composing PROPs. TAC.

New structural theorems:

Selinger, P. (2011) Finite dimensional Hilbert spaces are complete for dagger compact closed categories. ENTCS.

Coecke, B., **Pavlovic**, D., and **Vicary**, J. (2011) A new description of orthogonal bases. MSCS. arXiv:quant-ph/0810.1037

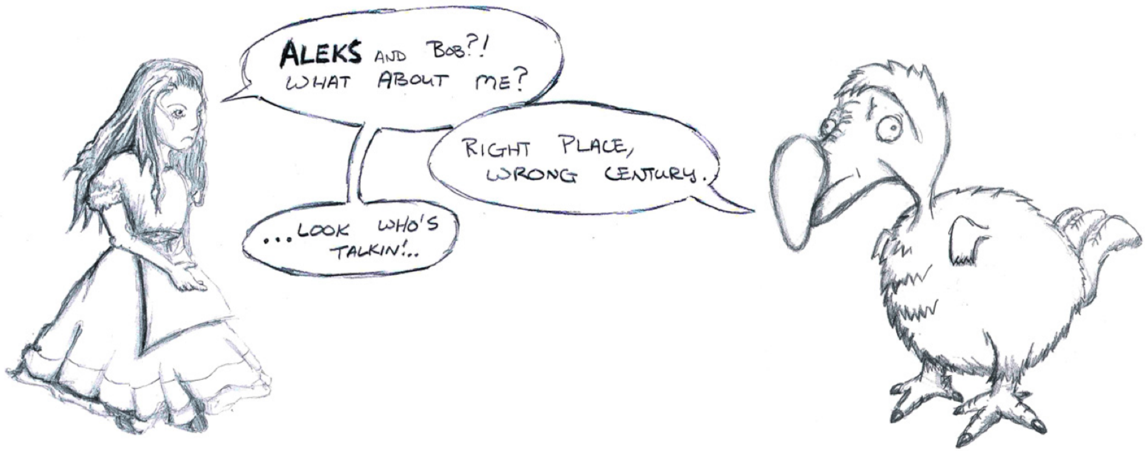
Backens, M. (2013) The ZX-calculus is complete for stabilizer quantum mechanics. arXiv:1307.7025.

Kissinger, A. (2014) Finite matrices are complete for (dagger-)multigraph categories. arXiv:1406.5942.

BC & Aleks Kissinger

Picturing Quantum Processes

Cambridge University Press, spring 2015.



— Ch. 1 – Processes as diagrams —

Philosophy [i.e. physics] is written in this grand book—I mean the universe—which stands continually open to our gaze, but it cannot be understood unless one first learns to comprehend the language and interpret the characters in which it is written. It is written in the language of mathematics, and its characters are triangles, circles, and other geometrical figures, without which it is humanly impossible to understand a single word of it; without these, one is wandering around in a dark labyrinth.

— Galileo Galilei, "Il Saggiatore", 1623.

Here we introduce:

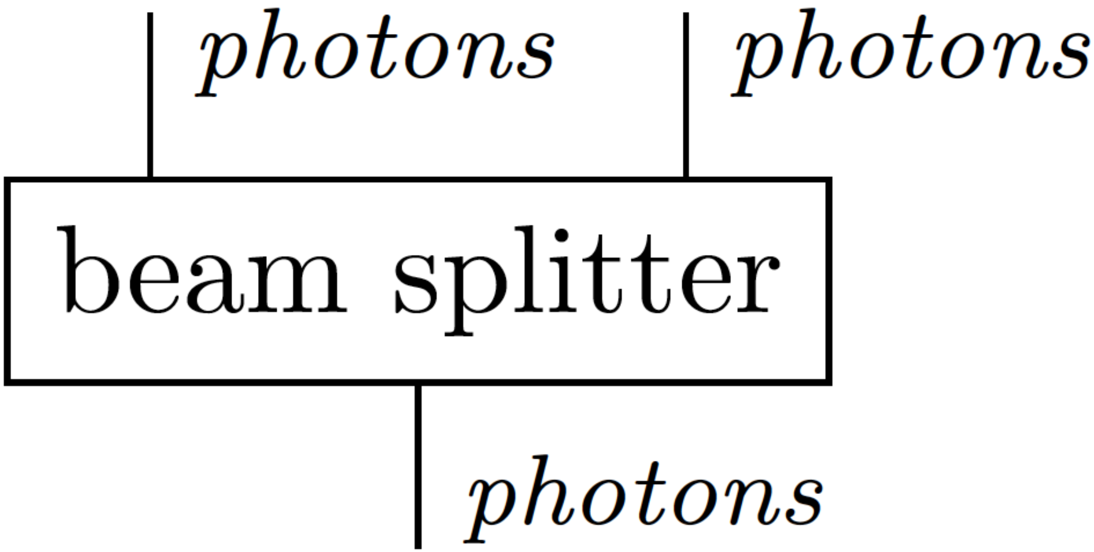
- diagrams
- process theories
- (boring) circuit diagrams

— Ch. 1 – Processes as diagrams —

– processes as boxes and systems as wires –

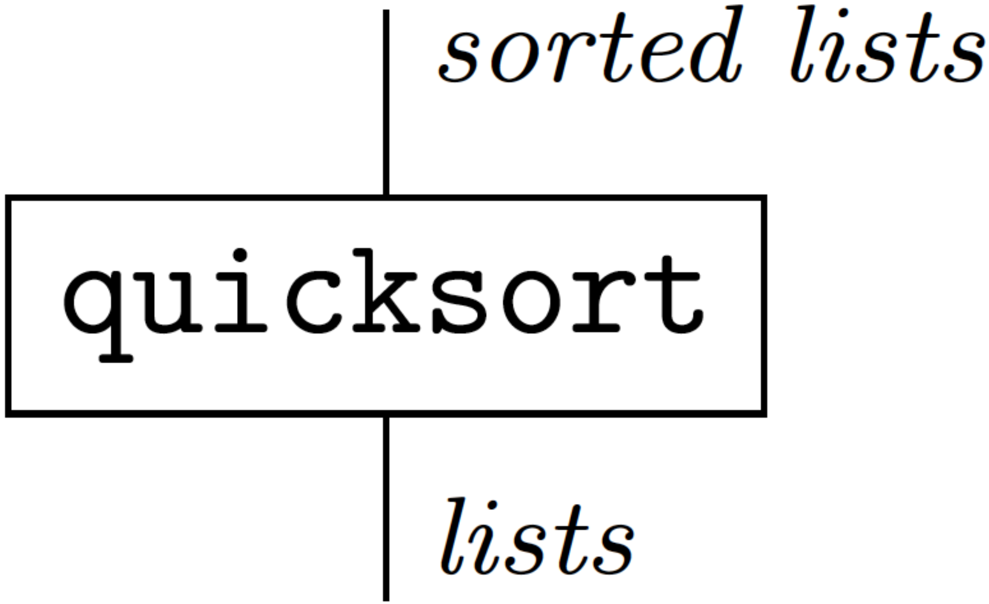
— Ch. 1 – Processes as diagrams —

– processes as boxes and systems as wires –



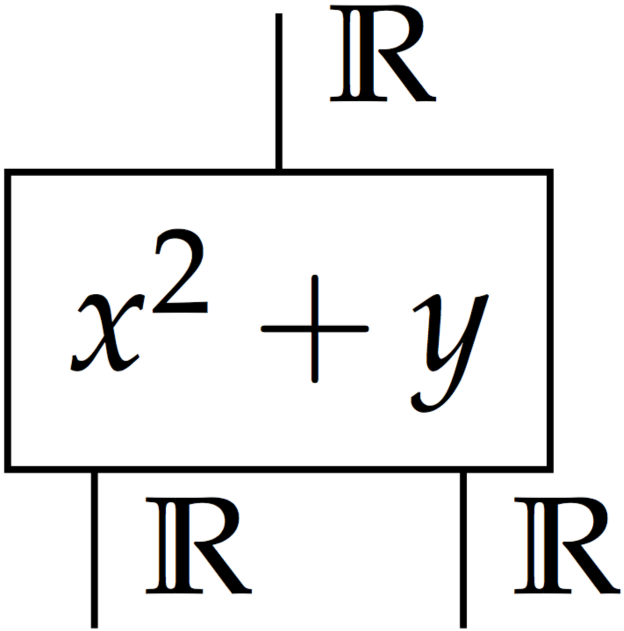
— Ch. 1 – Processes as diagrams —

– processes as boxes and systems as wires –



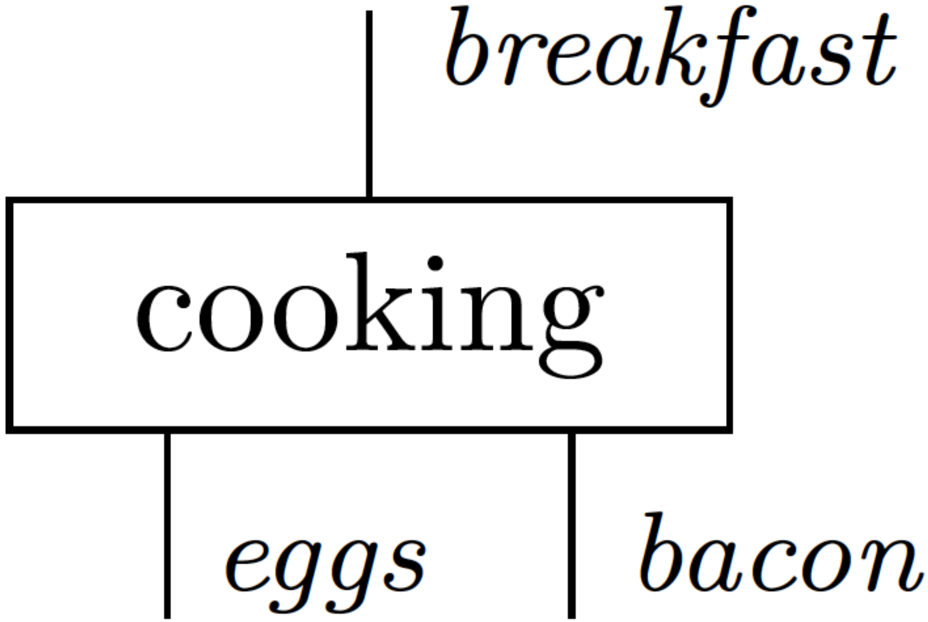
— Ch. 1 – Processes as diagrams —

– processes as boxes and systems as wires –



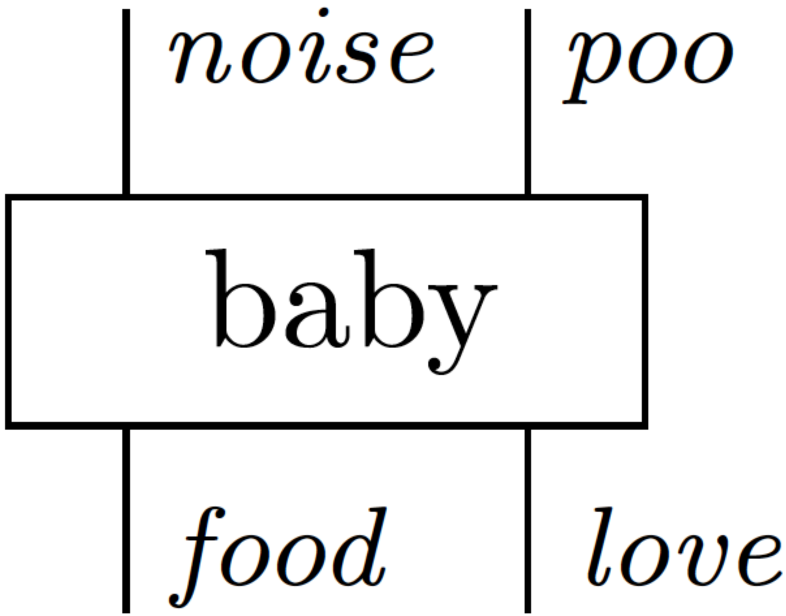
— Ch. 1 – Processes as diagrams —

– processes as boxes and systems as wires –



— Ch. 1 – Processes as diagrams —

– processes as boxes and systems as wires –

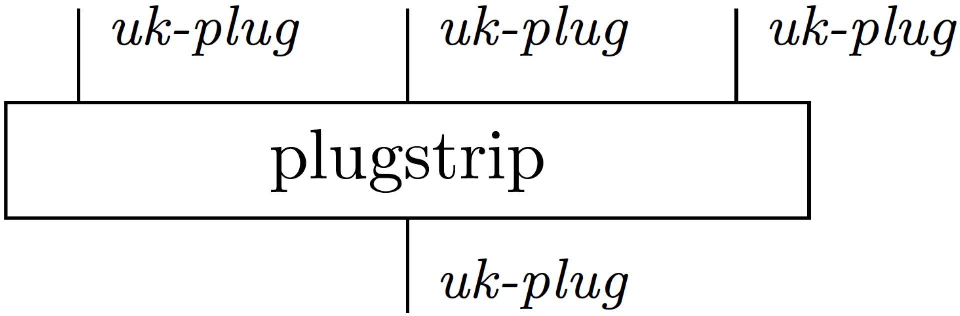


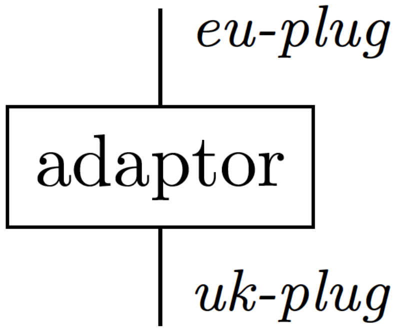
— Ch. 1 – Processes as diagrams —

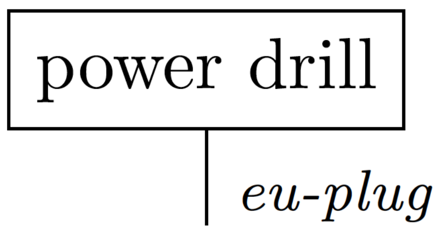
– composing processes –

— Ch. 1 – Processes as diagrams —

– composing processes –

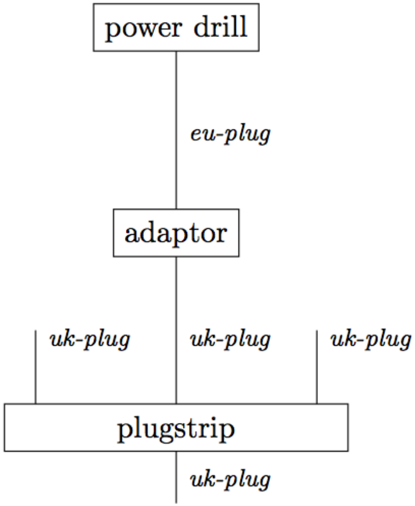




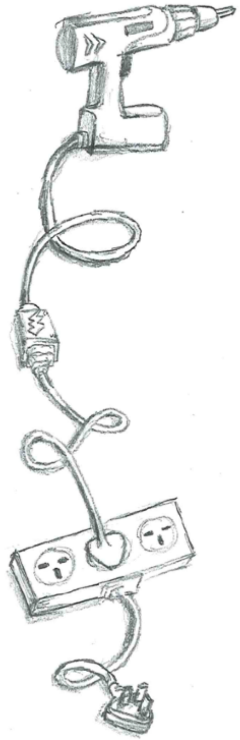


— Ch. 1 – Processes as diagrams —

– composing processes –

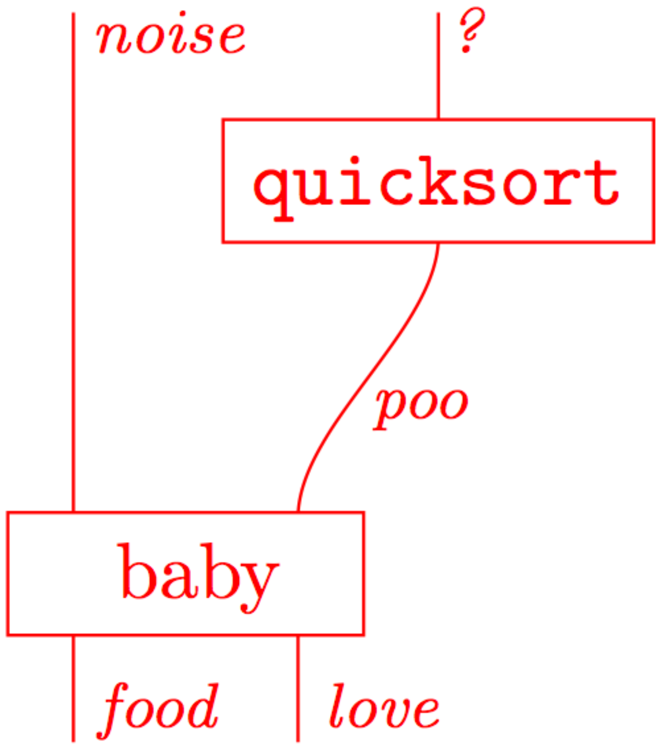


:=



— Ch. 1 – Processes as diagrams —

– composing processes –



— Ch. 1 – Processes as diagrams —

– process theory –

— Ch. 1 – Processes as diagrams —

– process theory –

... consists of:

- **set of systems** S
- **set of processes** P**,** with ins and outs in S**,**

— Ch. 1 – Processes as diagrams —

– process theory –

... consists of:

- **set of systems** S
- **set of processes** P**,** with ins and outs in S**,**

which are:

- **closed under forming diagrams.**

— Ch. 1 – Processes as diagrams —

– process theory –

... consists of:

- **set of systems** S
- **set of processes** P**,** with ins and outs in S**,**

which are:

- **closed under forming diagrams.**

It tells us:

- how to *interpret* boxes and wires,

— Ch. 1 – Processes as diagrams —

– process theory –

... consists of:

- **set of systems** S
- **set of processes** P**,** with ins and outs in S**,**

which are:

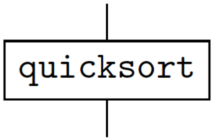
- **closed under forming diagrams.**

It tells us:

- how to *interpret* boxes and wires,
- and hence, when two diagrams are equal.

— Ch. 1 – Processes as diagrams —

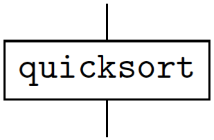
– process theory –



$$:= \begin{cases} \texttt{qs [] = []} \\ \texttt{qs (x :: xs) =} \\ \quad \texttt{qs [y | y <- xs; y < x] ++ [x] ++} \\ \quad \texttt{qs [y | y <- xs; y >= x]} \end{cases}$$

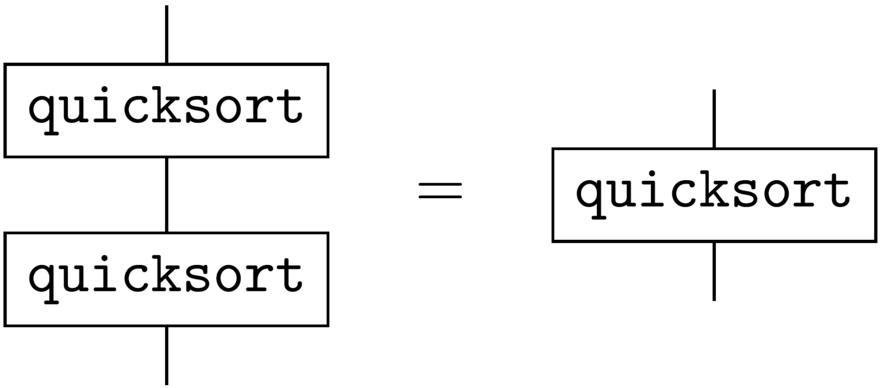
— Ch. 1 – Processes as diagrams —

– *process theory* –



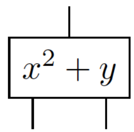
$:=$

```
qs [] = []
qs (x :: xs) =
    qs [y | y <- xs; y < x] ++ [x] ++
    qs [y | y <- xs; y >= x]
```

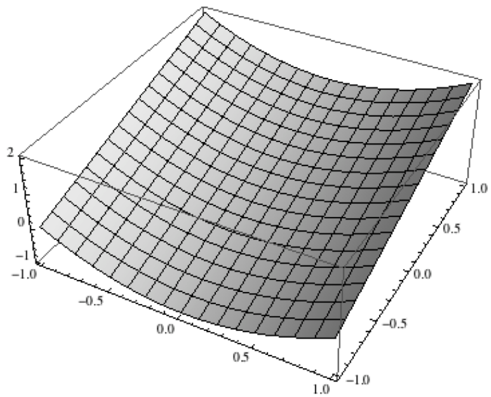


— Ch. 1 – Processes as diagrams —

– process theory –

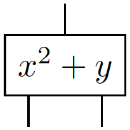


$:=$

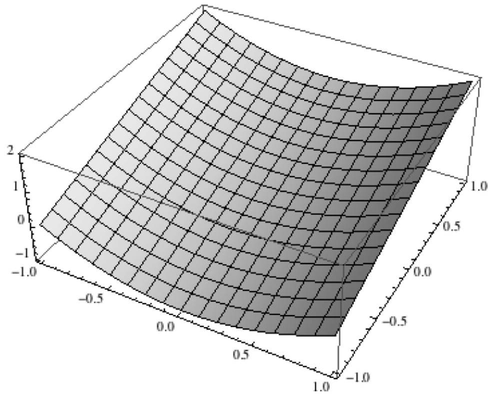


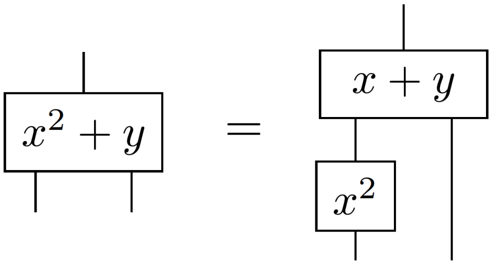
— Ch. 1 – Processes as diagrams —

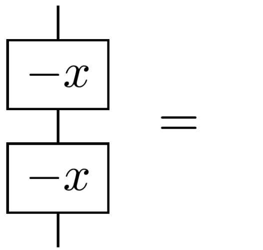
– process theory –



$:=$







— Ch. 1 – Processes as diagrams —

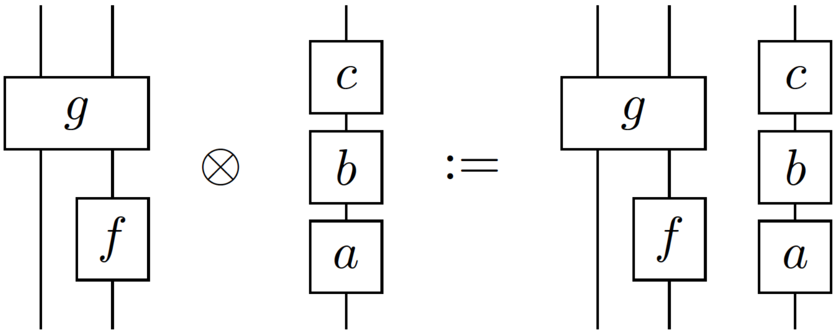
– diagrams algebraically –

— Ch. 1 – Processes as diagrams —

– diagrams algebraically –

Two operations:

$$\text{“}f \otimes g\text{”} := \text{“}f \textbf{ while } g\text{”}$$

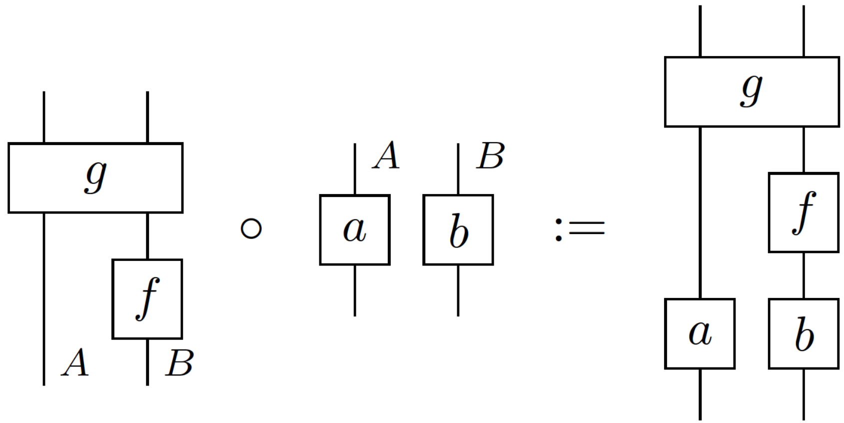


— Ch. 1 – Processes as diagrams —

– *diagrams algebraically* –

Two operations:

$$\text{“}f \circ g\text{”} \;:=\; \text{“}f \textbf{ after } g\text{”}$$



— Ch. 1 – Processes as diagrams —

– why diagrams? –

— Ch. 1 – Processes as diagrams —

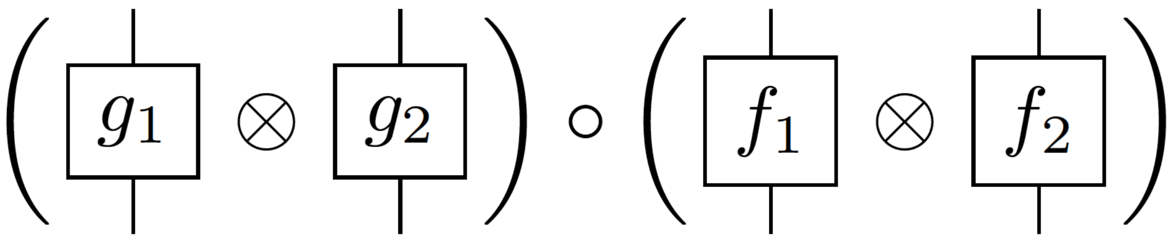
– why diagrams? –

Since all equations come for free!

— Ch. 1 – Processes as diagrams —

– why diagrams? –

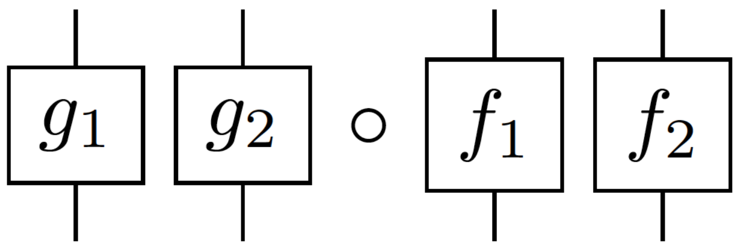
Since all equations come for free!



— Ch. 1 – Processes as diagrams —

– why diagrams? –

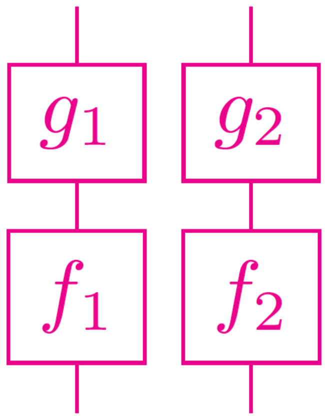
Since all equations come for free!



— Ch. 1 – Processes as diagrams —

– why diagrams? –

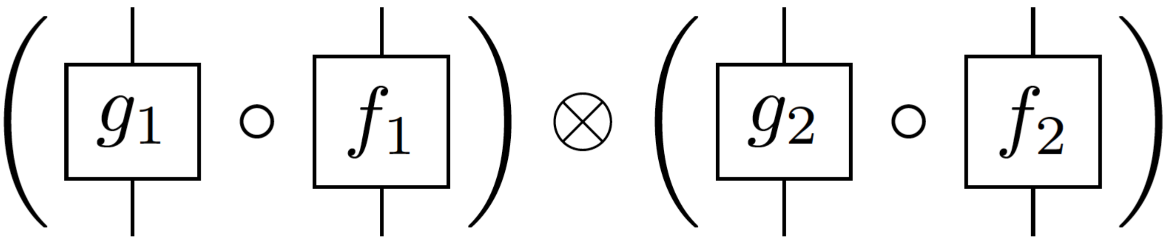
Since all equations come for free!



— Ch. 1 – Processes as diagrams —

– why diagrams? –

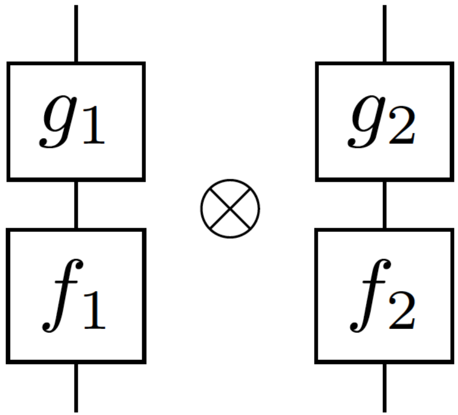
Since all equations come for free!



— Ch. 1 – Processes as diagrams —

– why diagrams? –

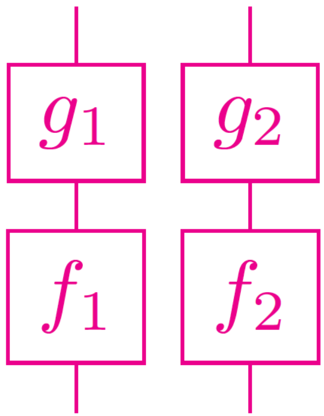
Since all equations come for free!



— Ch. 1 – Processes as diagrams —

– why diagrams? –

Since all equations come for free!



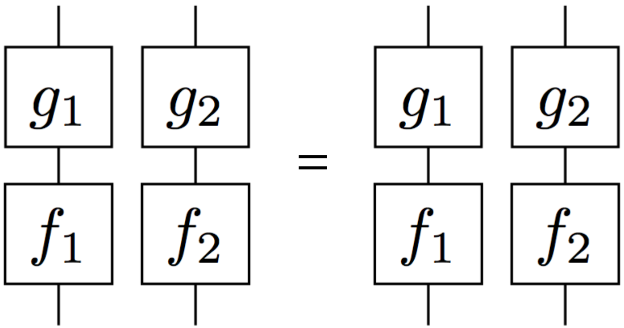
— Ch. 1 – Processes as diagrams —

– why diagrams? –

In formulas:

$$\left(\boxed{g_1}\otimes\boxed{g_2}\right)\circ\left(\boxed{f_1}\otimes\boxed{f_2}\right) = \left(\boxed{g_1}\circ\boxed{f_1}\right)\otimes\left(\boxed{g_2}\circ\boxed{f_2}\right)$$

In diagrams:



— Ch. 1 – Processes as diagrams —

– circuits –

— Ch. 1 – Processes as diagrams —

– circuits –

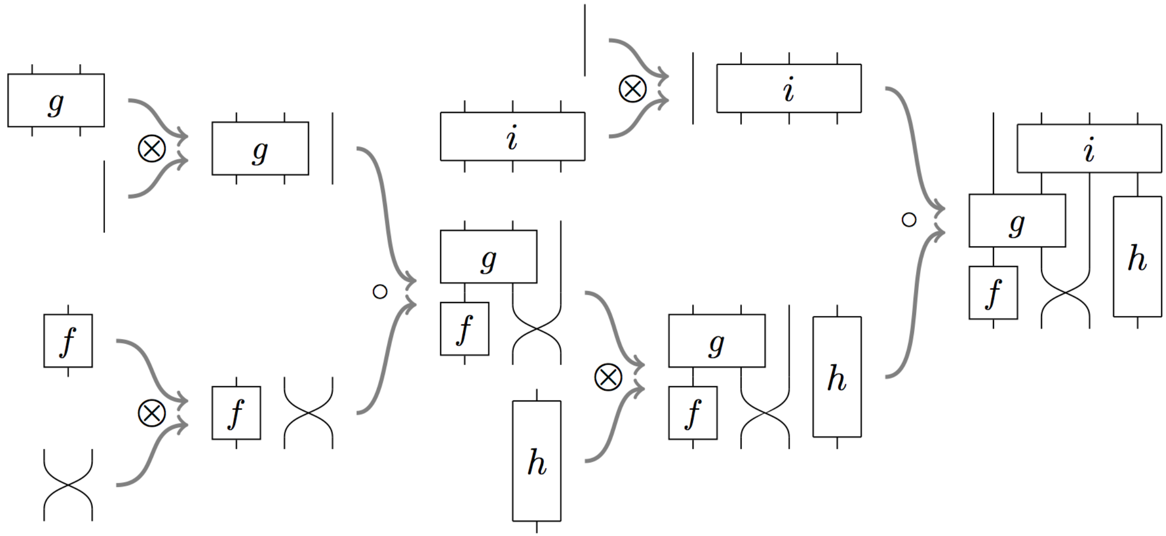
Defn. ... := can be build with $\otimes$ and $\circ$.

— Ch. 1 – Processes as diagrams —

– *circuits* –

Defn. ... := can be build with $\otimes$ and $\circ$.

E.g.:

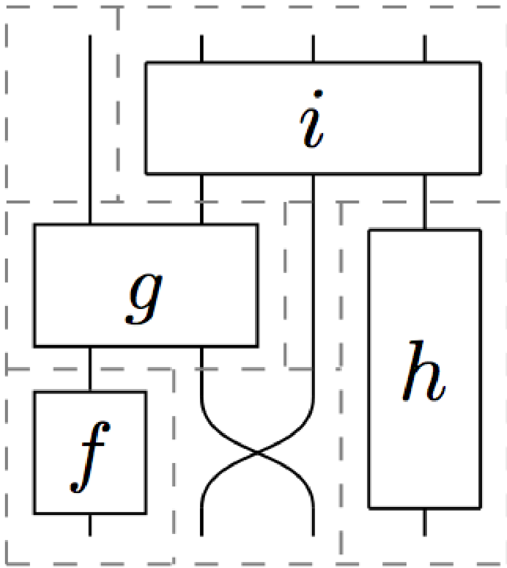


— Ch. 1 – Processes as diagrams —

– *circuits* –

Defn. ... := can be build with $\otimes$ and $\circ$.

E.g.:

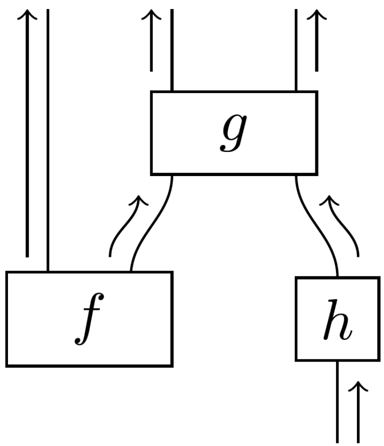


— Ch. 1 – Processes as diagrams —

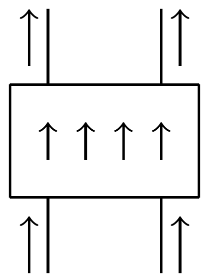
– circuits –

Defn. ... := can be build with $\otimes$ and $\circ$.

Thm. Diagram is circuit $\Leftrightarrow$ is 'causal' e.g.:



with



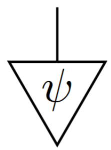
— Ch. 1 – Processes as diagrams —

– special processes/diagrams –

— Ch. 1 – Processes as diagrams —

– special processes/diagrams –

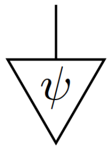
State :=



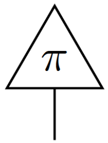
— Ch. 1 – Processes as diagrams —

– special processes/diagrams –

State :=



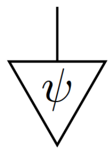
Effect / Test :=



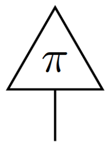
— Ch. 1 – Processes as diagrams —

– special processes/diagrams –

State :=



Effect / Test :=

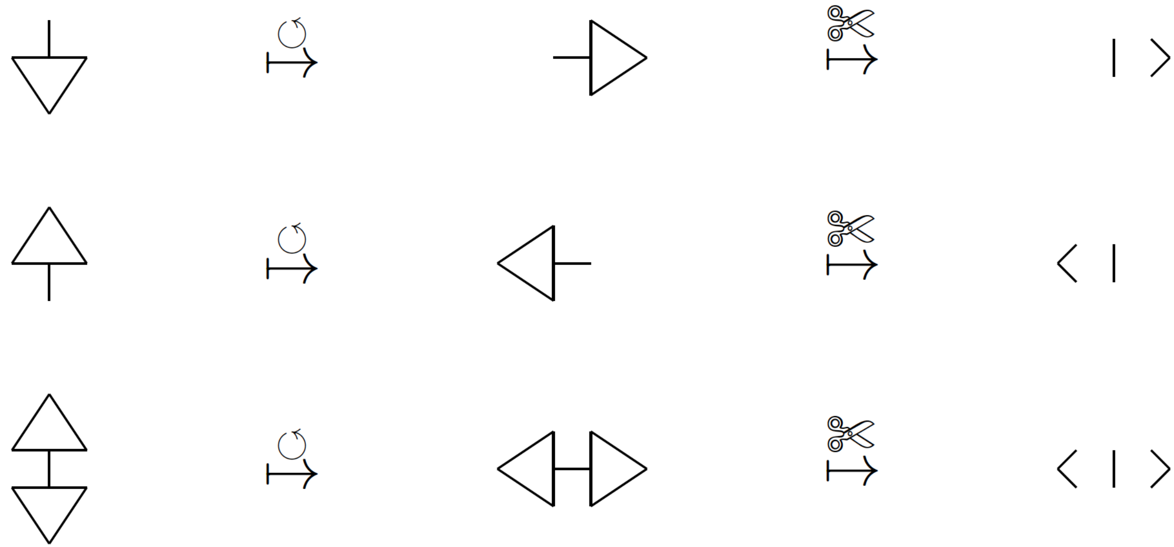


Number :=

— Ch. 1 – Processes as diagrams —

– special processes/diagrams –

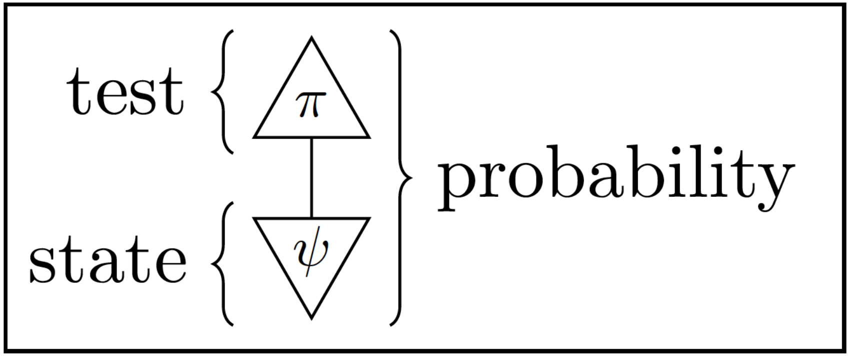
Dirac notation :=



— Ch. 1 – Processes as diagrams —

– special processes/diagrams –

Born rule :=



Candidate systems:

Candidate systems:

- vector space with inner-product:
  - pure (or closed) quantum states (complex)
  - standard natural language processing (real)

Candidate systems:

- vector space with inner-product:
  - pure (or closed) quantum states (complex)
  - standard natural language processing (real)
- density matrices with trace:
  - mixed (or open) quantum states
  - neo natural language processing

Candidate systems:

- vector space with inner-product:
  - pure (or closed) quantum states (complex)
  - standard natural language processing (real)
- density matrices with trace:
  - mixed (or open) quantum states
  - neo natural language processing
- more abstract models and constructions

Vector space model of word meaning in NLP:

Vector space model of word meaning in NLP:

- vector space spanned by context words

Vector space model of word meaning in NLP:

- vector space spanned by context words
- meaning vectors from relative occurrences

Vector space model of word meaning in NLP:

- vector space spanned by context words
- meaning vectors from relative occurrences
- similarity from inner product

Vector space model of word meaning in NLP:

- vector space spanned by context words
- meaning vectors from relative occurrences
- similarity from inner product

Source: huge corpus

Pioneer:

- H. Schuetze (1998) *Automatic word sense discrimination*. Computational Linguistics, **24**, 97123.

Vector space model of social properties:

- vector space spanned by context words
- meaning vectors from relative occurrences
- similarity from inner product

Source: Facebook, personal page, ...

Pioneer:

- H. Schuetze (1998) *Automatic word sense discrimination*. Computational Linguistics, **24**, 97123.

— Ch. 2 – String diagrams —

When two systems, of which we know the states by their respective representatives, enter into temporary physical interaction due to known forces between them, and when after a time of mutual influence the systems separate again, then they can no longer be described in the same way as before, viz. by endowing each of them with a representative of its own. ***I would <u>not</u> call that <u>one</u> <u>but</u> rather <u>the</u> characteristic trait of quantum mechanics****, the one that enforces its entire departure from classical lines of thought.*

— Erwin Schrödinger, 1935.

Here we introduce:

- string diagrams
- transposes and adjoints
- quantum phenomena in great generality

— Ch. 2 – String diagrams —

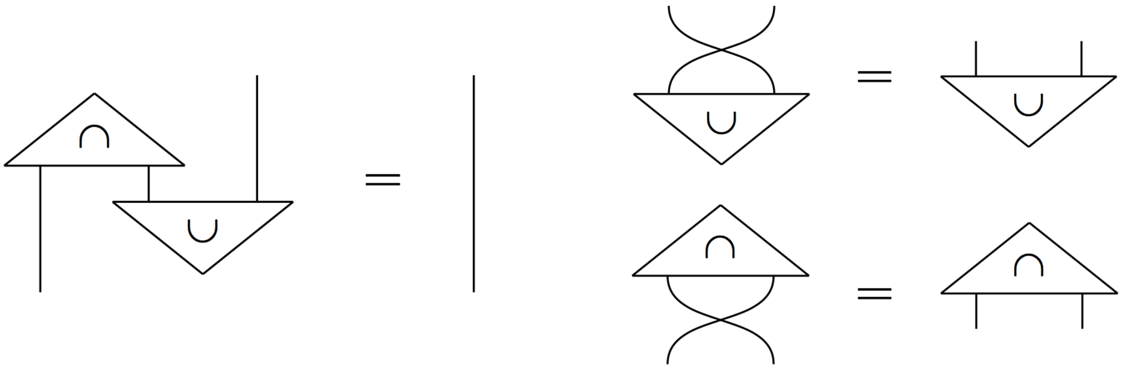
– *TFAE* –

— Ch. 2 – String diagrams —

– *TFAE* –

1. 'Circuits' with cup-state and cup-effect:

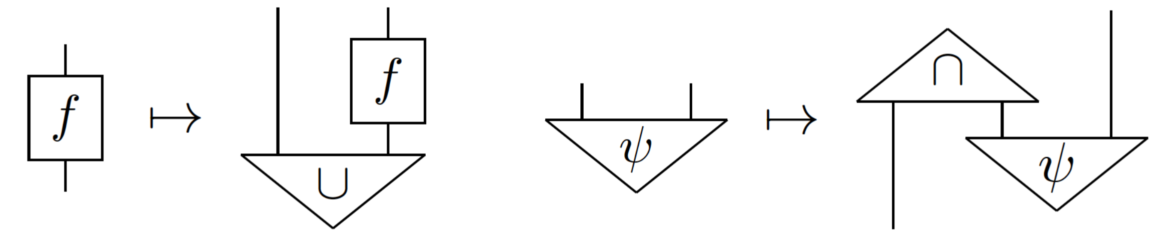
which satisfy:



— Ch. 2 – String diagrams —

– TFAE –

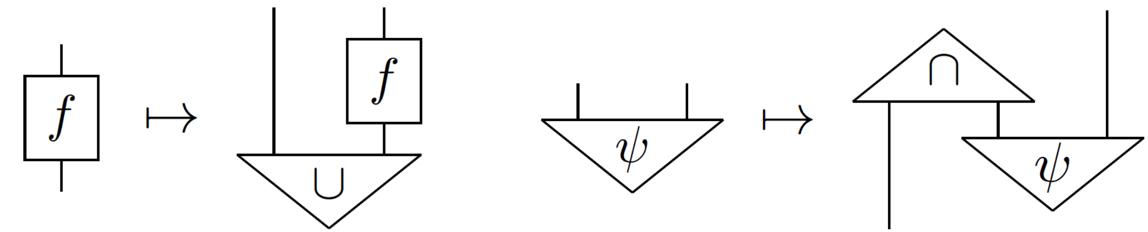
I.e. ‘constructive’ CJ-isomorphism via Bell-state/effect:



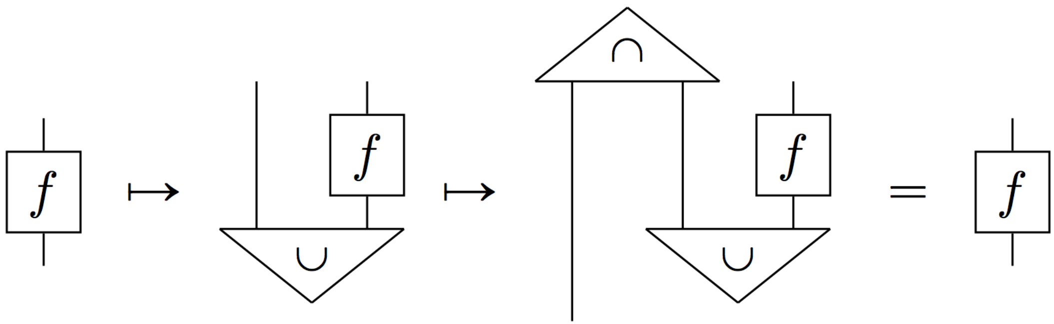
— Ch. 2 – String diagrams —

– *TFAE* –

I.e. 'constructive' CJ-isomorphism via Bell-state/effect:



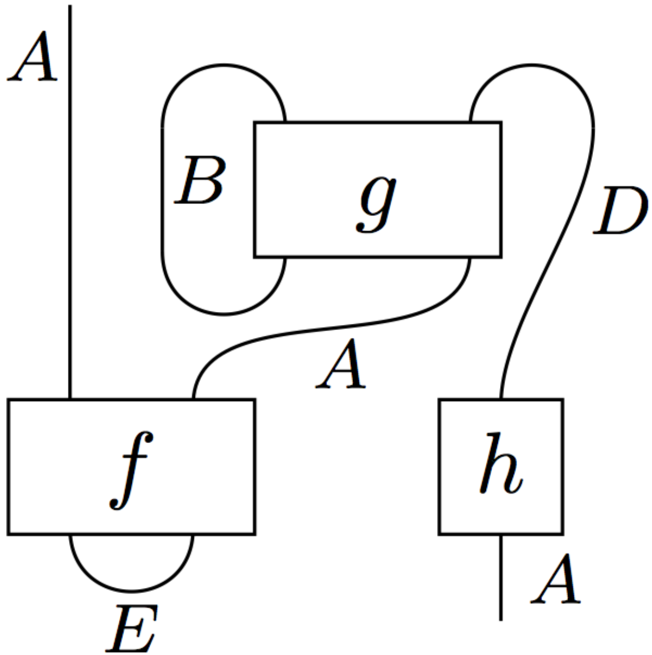
Pf.



— Ch. 2 – String diagrams —

– *TFAE* –

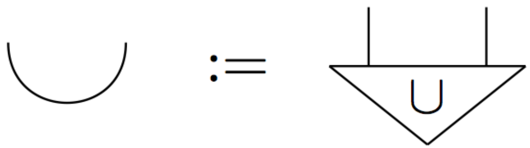
2. diagrams allowing in-in, out-out and out-in wiring:

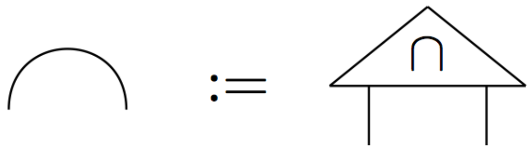


— Ch. 2 – String diagrams —

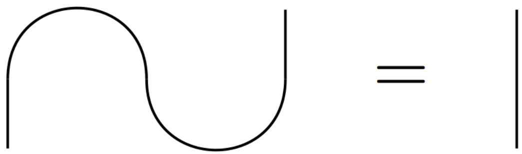
– TFAE –

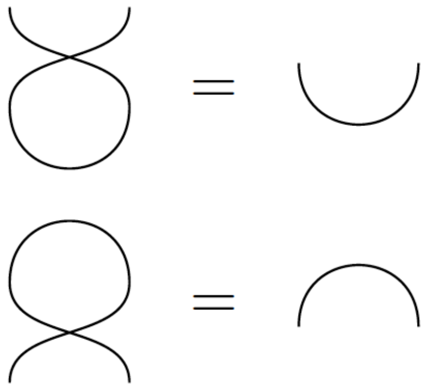
From **1.** to **2.**:





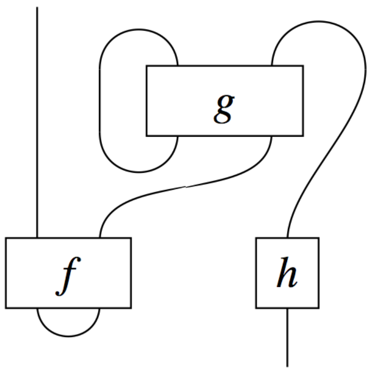
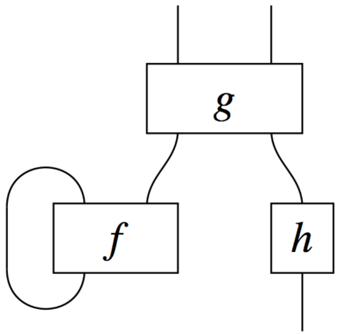
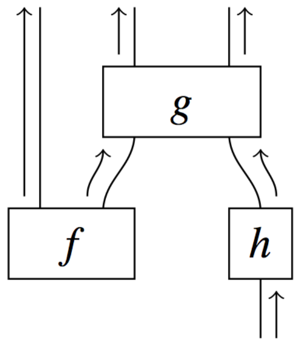
so that:





Symmetric monoidal categories as diagrams:

Symmetric monoidal categories as diagrams:

compact closed	traced	plain
		
string diagrams	diagrams	cicuits
no ins/outs	outs to ins	causal structure

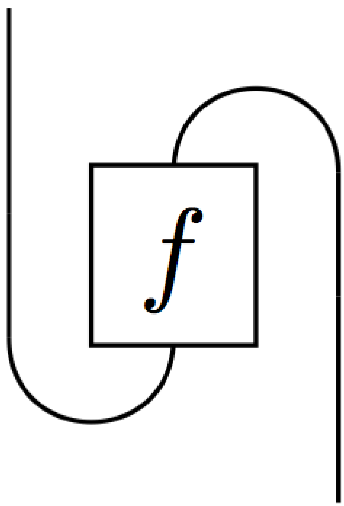
— Ch. 2 – String diagrams —

– transpose –

— Ch. 2 – String diagrams —

– transpose –

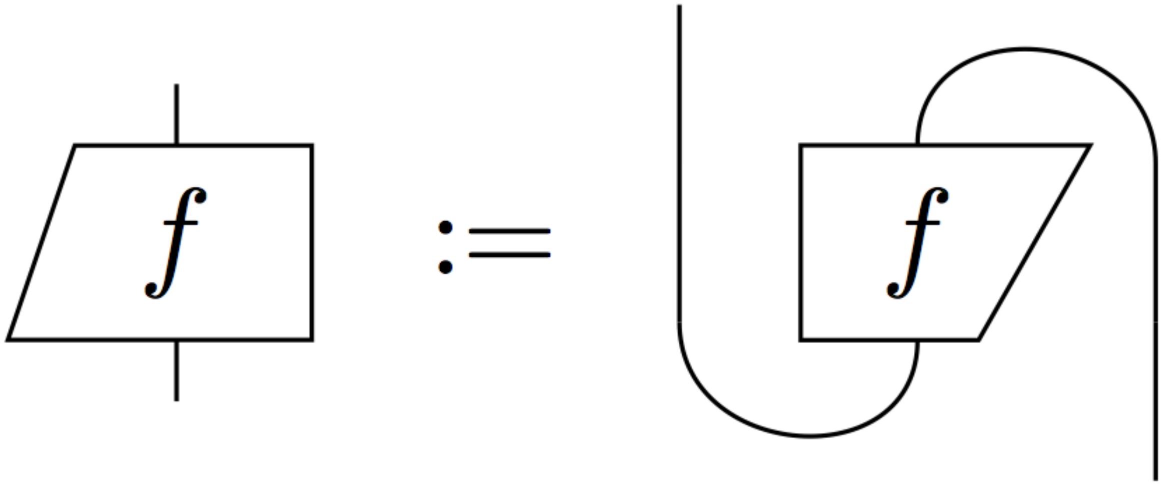
... :=



— Ch. 2 – String diagrams —

– transpose –

Clever new notation:

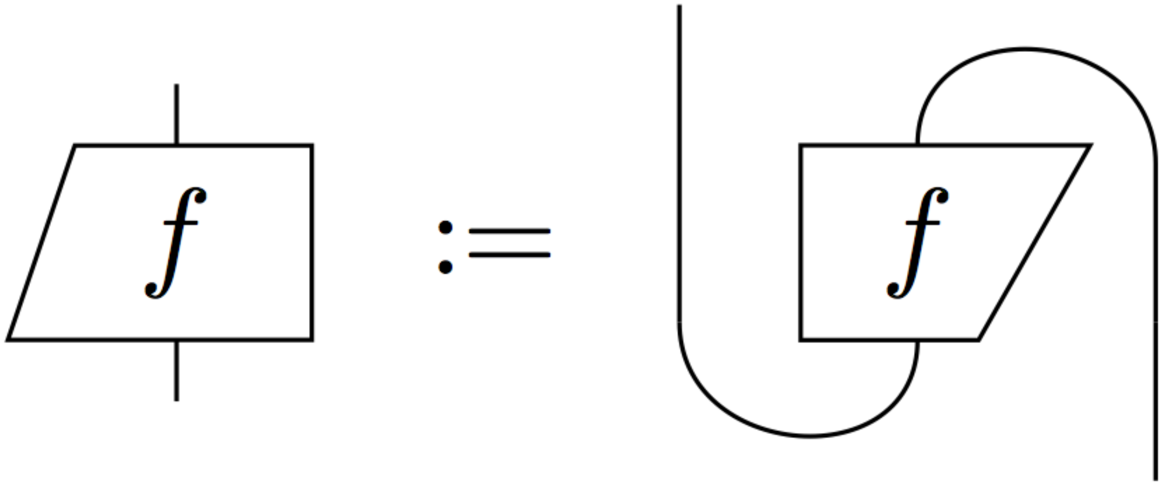


(intuition:)

— Ch. 2 – String diagrams —

– transpose –

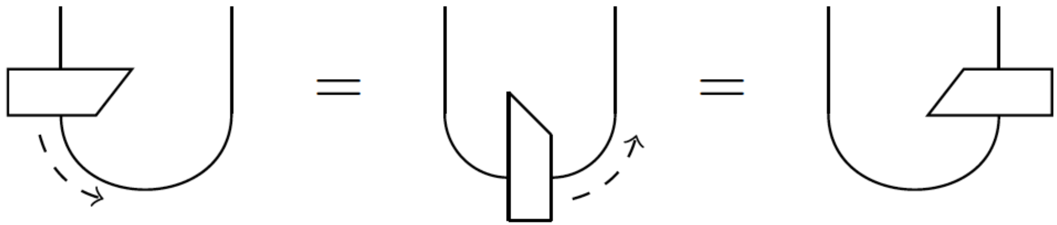
Clever new notation:



(intuition: again yanking the wire)

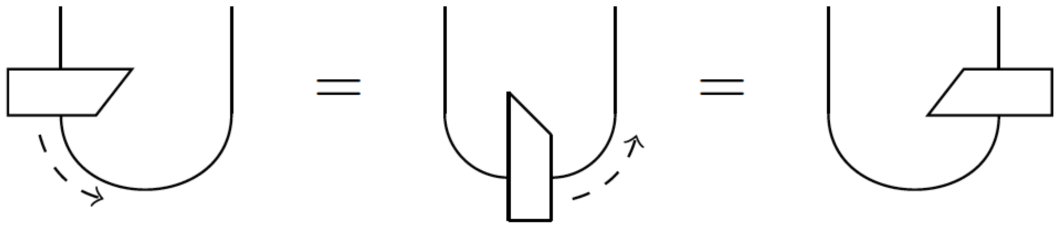
– transpose –

Prop. Sliding:

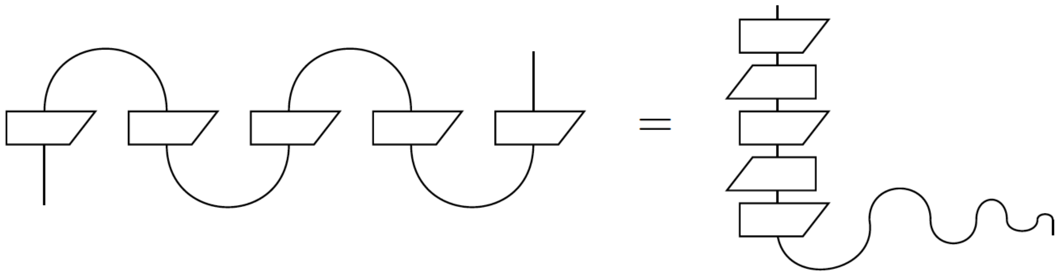


– transpose –

Prop. Sliding:



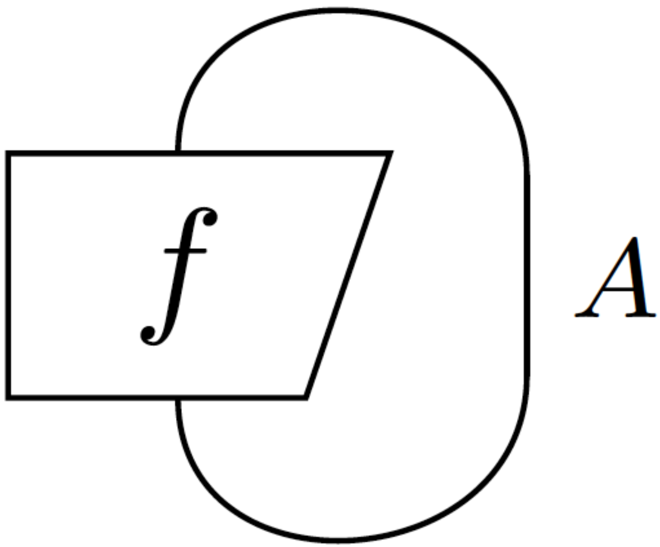
... so this is a mathematical equation:



— Ch. 2 – String diagrams —

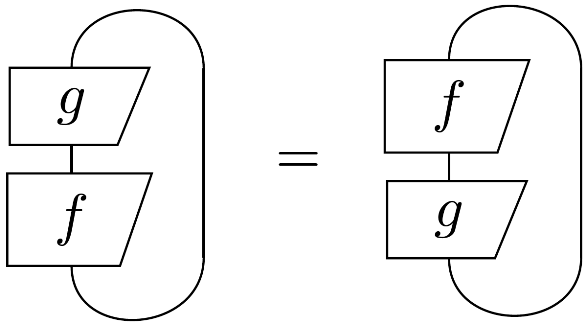
– trace –

... :=

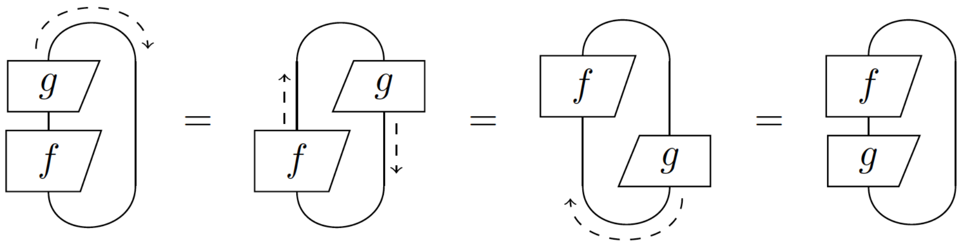


– trace –

Prop. Cyclicity:



Fun but redundant 'ferris wheel' proof:



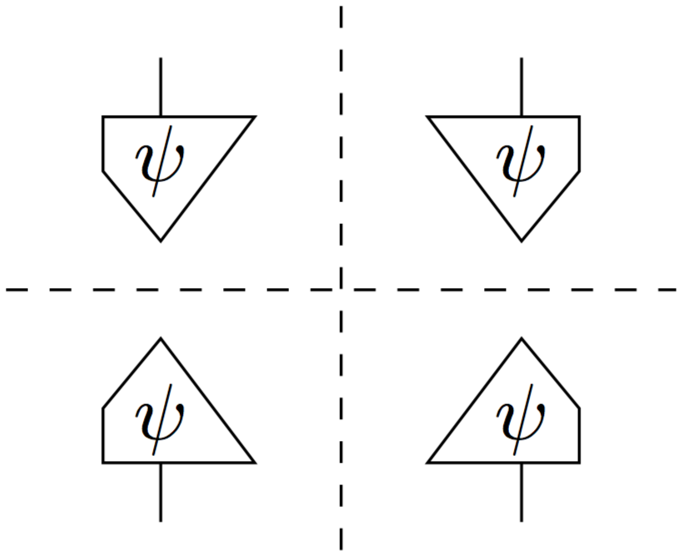
— Ch. 2 – String diagrams —

– adjoint & conjugate –

— Ch. 2 – String diagrams —

– adjoint & conjugate –

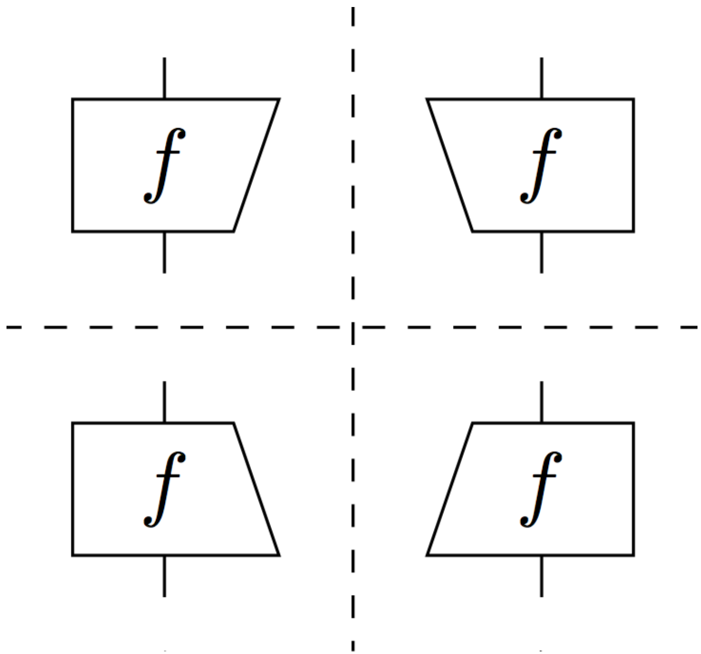
From a state to its test:



— Ch. 2 – String diagrams —

– adjoint & conjugate –

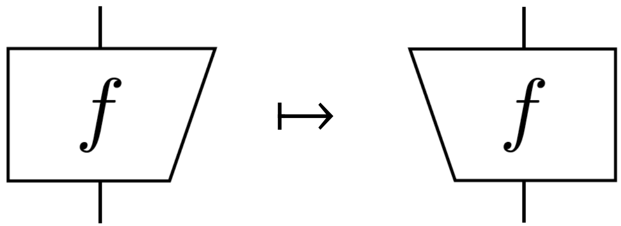
From a state to its test:



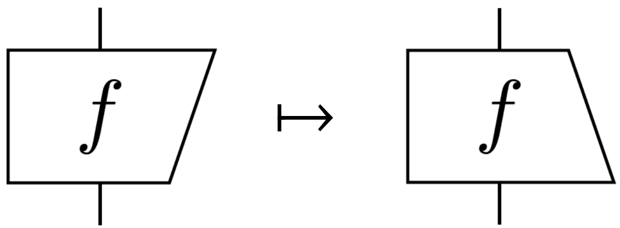
— Ch. 2 – String diagrams —

– adjoint & conjugate –

Conjugate :=



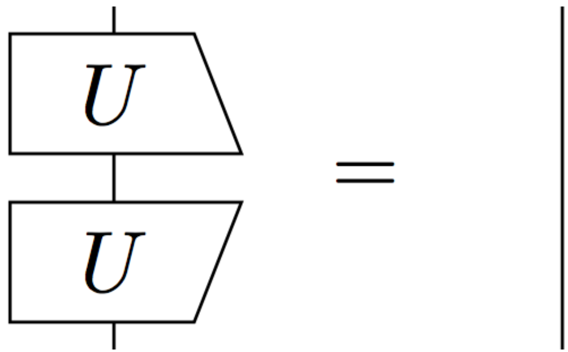
Adjoint :=



— Ch. 2 – String diagrams —

– adjoint & conjugate –

Unitarity/isometry :=



— Ch. 2 – String diagrams —

– sets and relations –

— Ch. 2 – String diagrams —

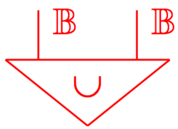
– sets and relations –

- wires := sets
- two wires := cartesian product
- boxes := relations

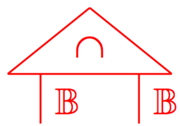
— Ch. 2 – String diagrams —

– sets and relations –

- wires := sets
- two wires := cartesian product
- boxes := relations
- transpose = adjoint := converse
- cups and caps for $\mathbb{B}$:=



$$:: \begin{cases} * \mapsto (0,0) \\ * \mapsto (1,1) \end{cases}$$

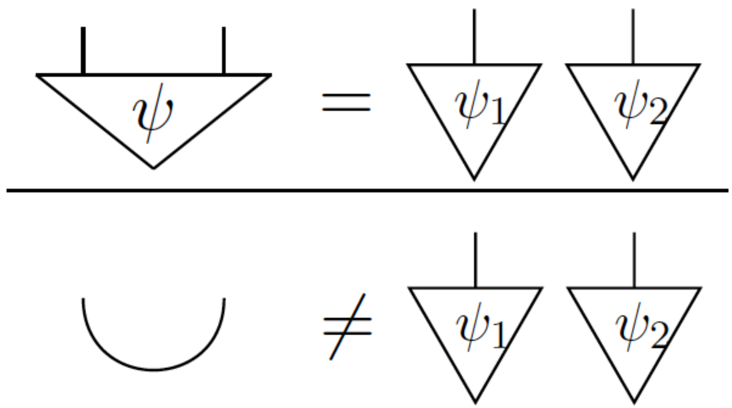


$$:: \begin{cases} (0,0) \mapsto * \\ (1,1) \mapsto * \end{cases}$$

— Ch. 2 – String diagrams —

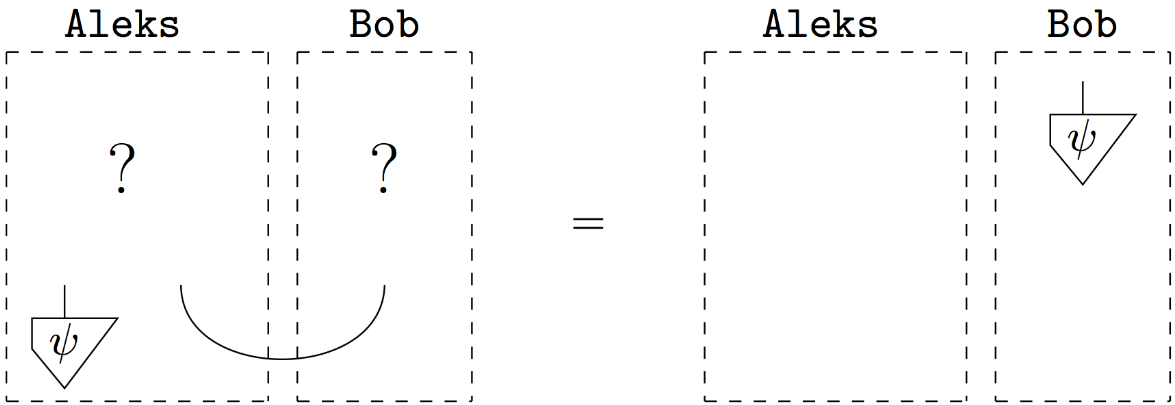
– ‘quantum’-like features –

$$\frac{\text{classical}}{\text{quantum}} =$$



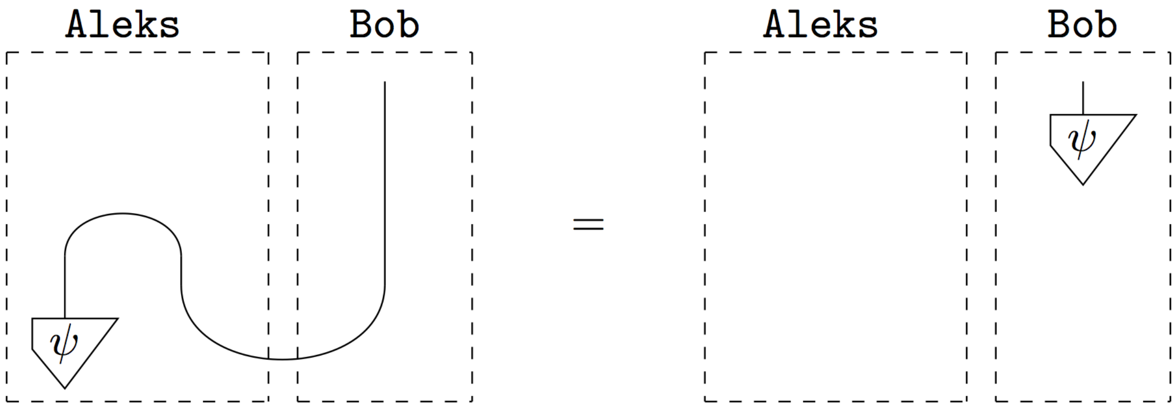
— Ch. 2 – String diagrams —

– quantum teleportation –



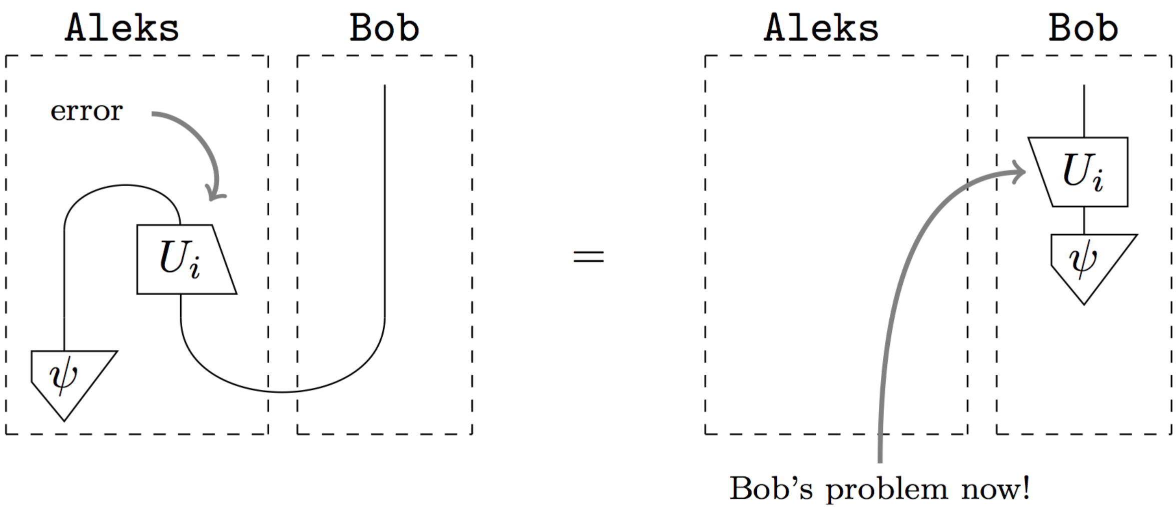
— Ch. 2 – String diagrams —

– quantum teleportation –



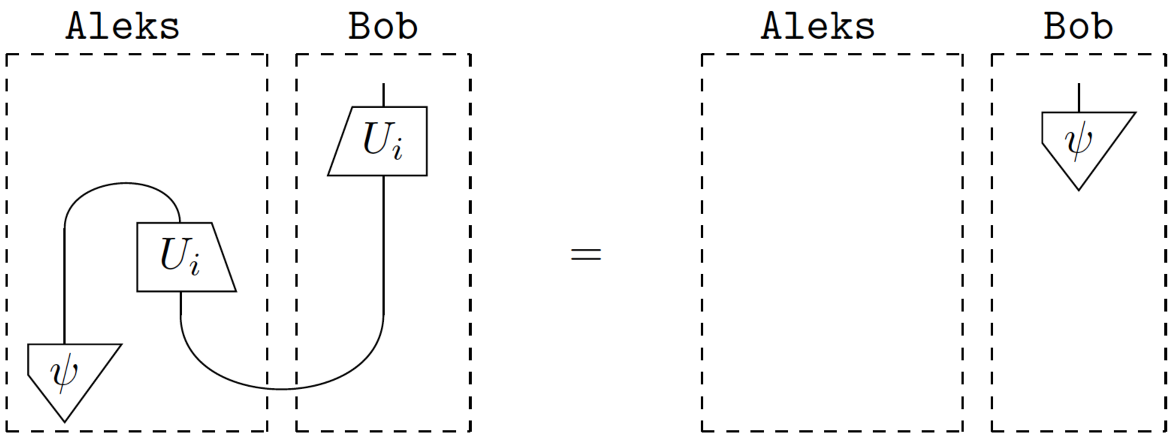
— Ch. 2 – String diagrams —

– quantum teleportation –



— Ch. 2 – String diagrams —

– quantum teleportation –



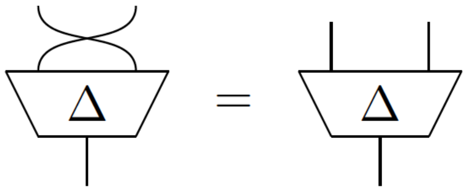
— Ch. 2 – String diagrams —

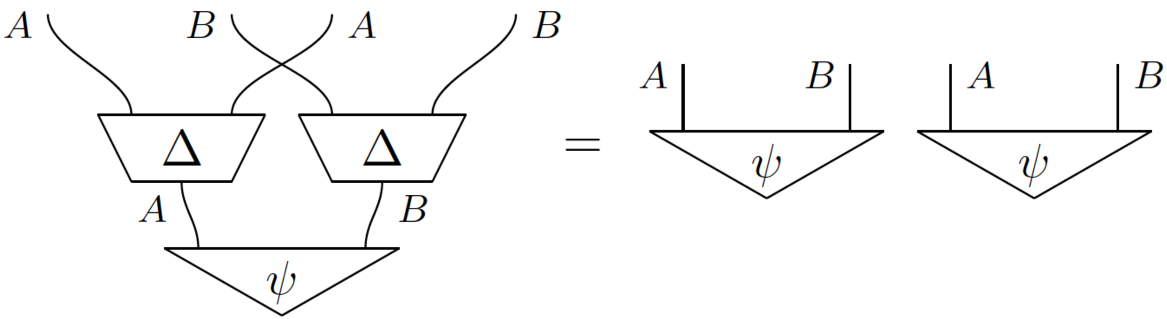
– linearity –

— Ch. 2 – String diagrams —

– linearity –

Thm. No-cloning from assumptions:

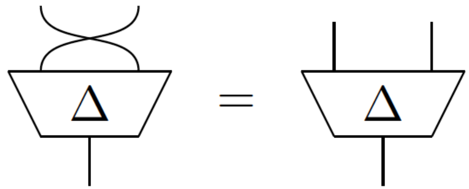


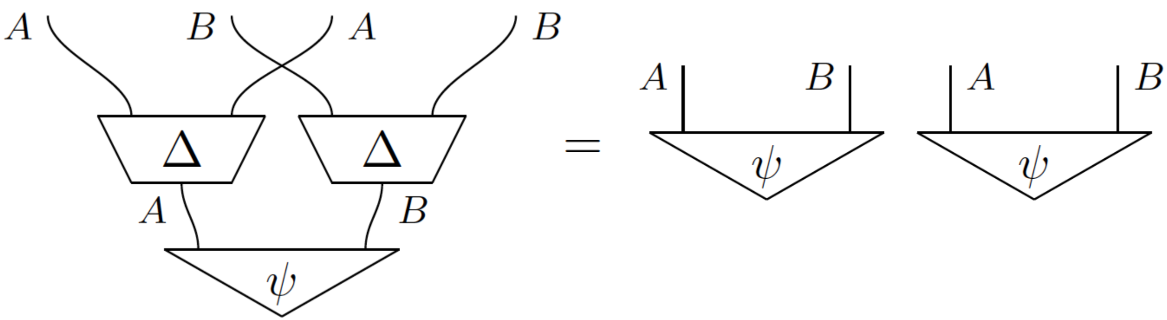


— Ch. 2 – String diagrams —

– linearity –

Thm. No-cloning from assumptions:

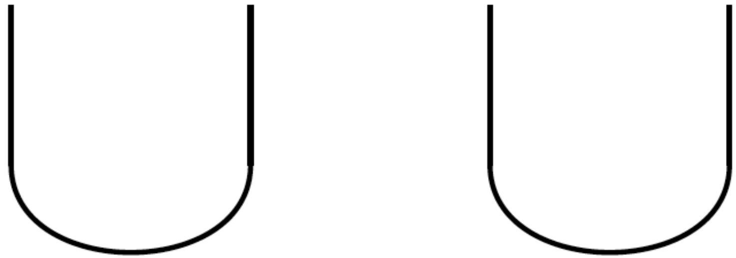




(Categorically:= cartesian ⊥ compact closed)

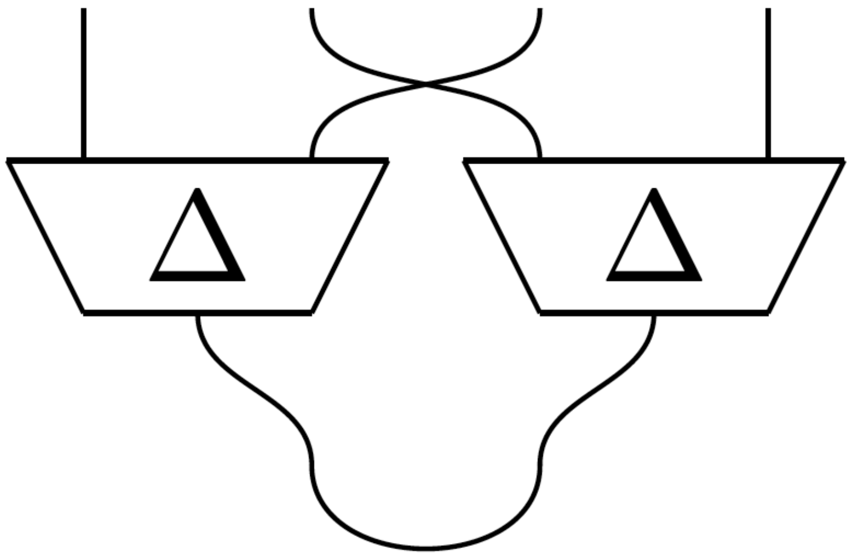
– no-cloning –

Pf.



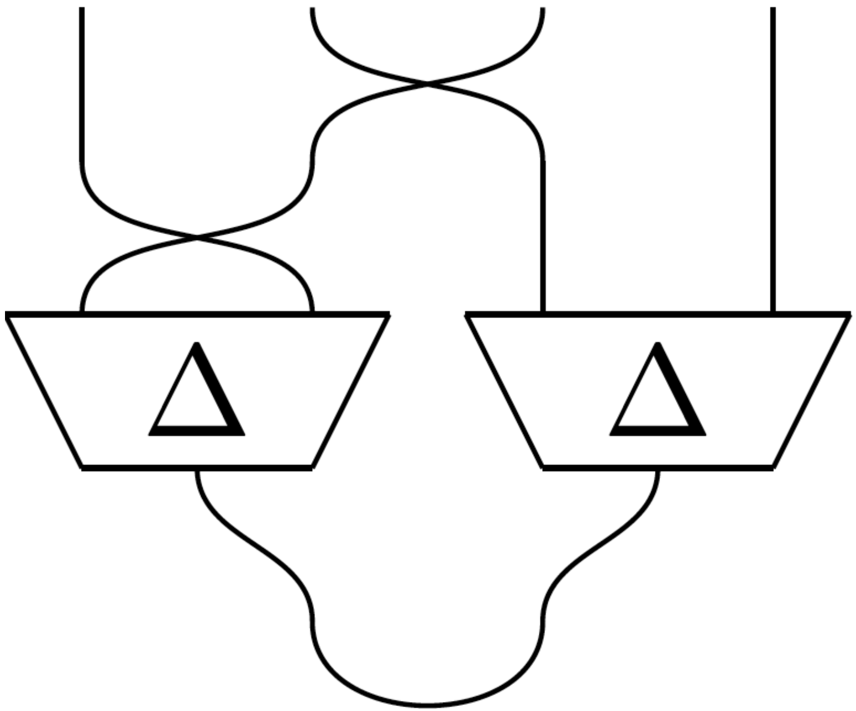
– no-cloning –

Pf.



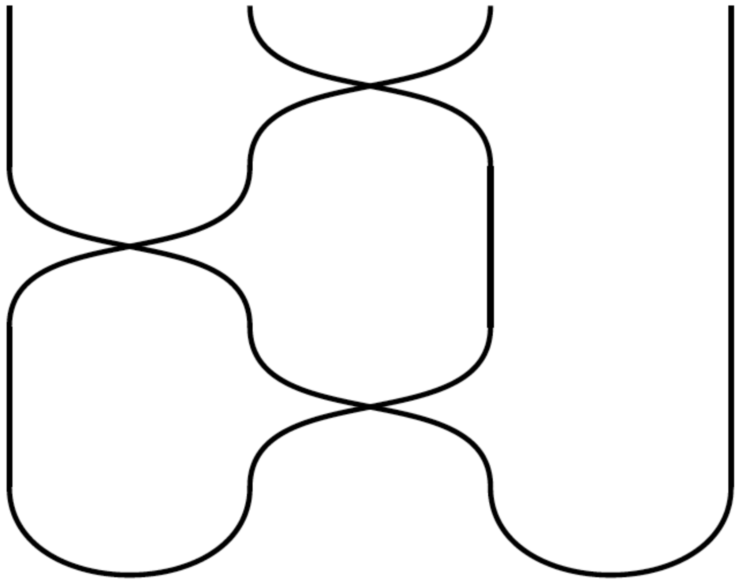
– no-cloning –

Pf.



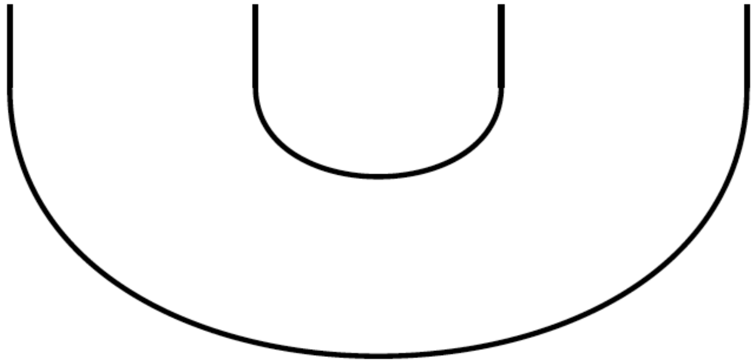
– no-cloning –

Pf.



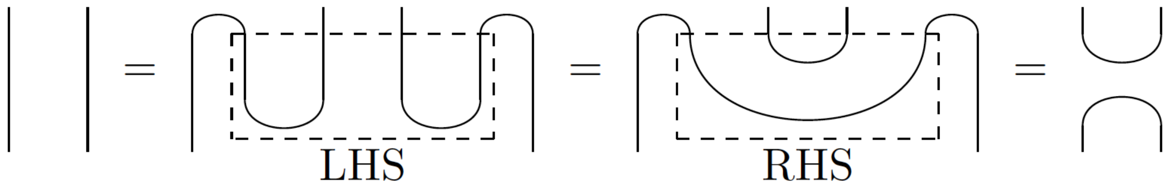
– no-cloning –

Pf.



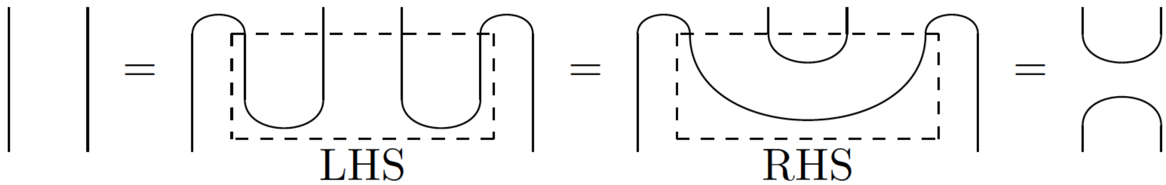
— Ch. 2 – String diagrams —

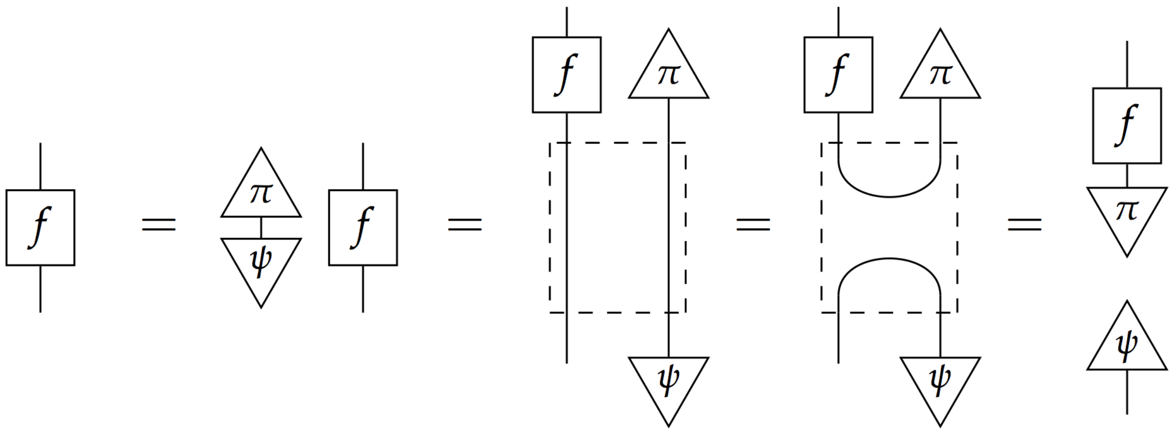
– no-cloning –



— Ch. 2 – String diagrams —

– no-cloning –





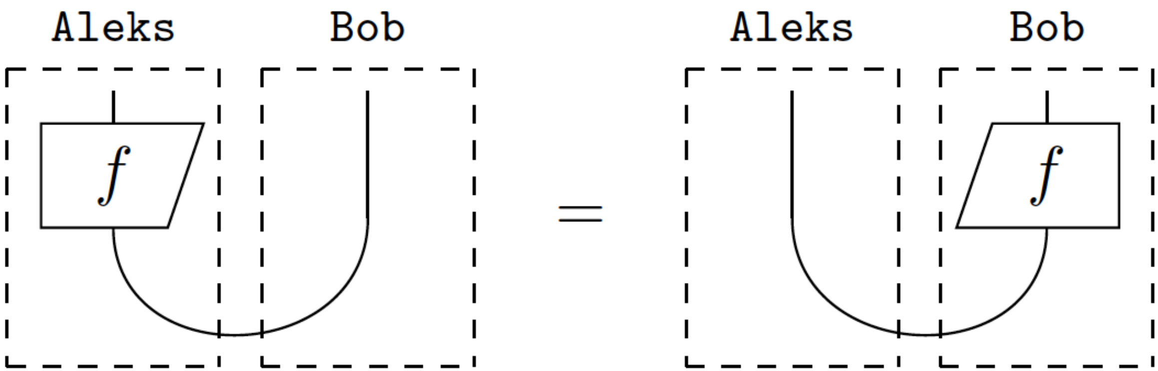
— Ch. 2 – String diagrams —

– correlations –

— Ch. 2 – String diagrams —

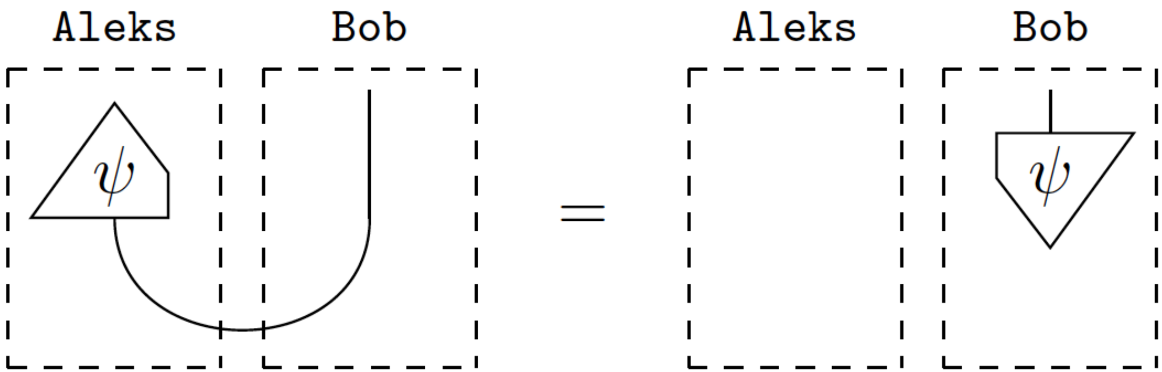
– correlations –

Transpose with agents:



– correlations –

Perfect correlations:

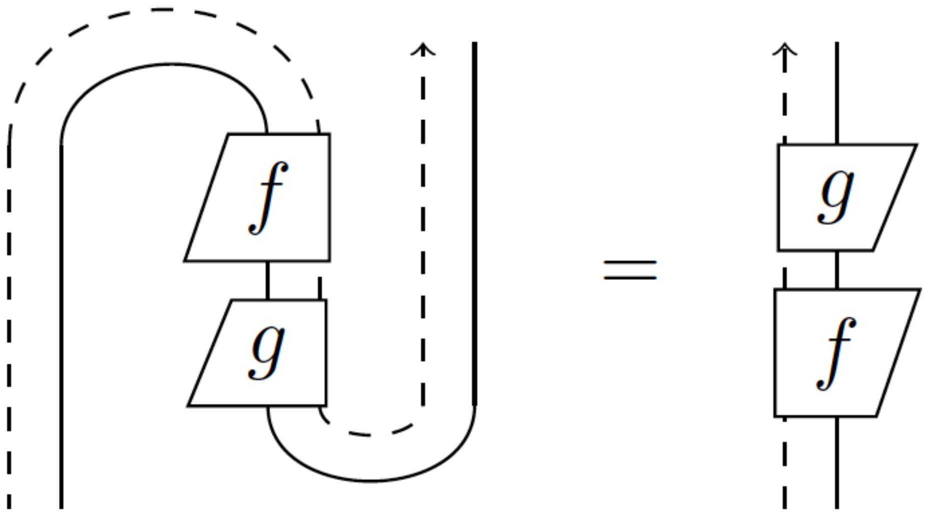


— Ch. 2 – String diagrams —

– time-reversal –

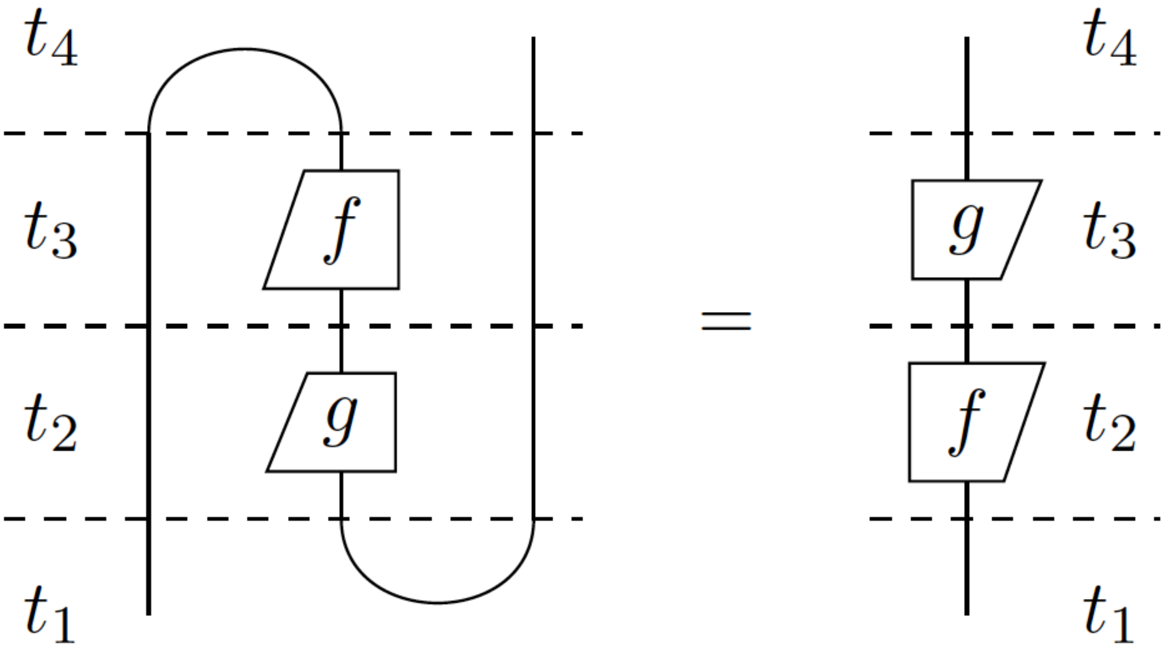
– time-reversal –

Logical reading:



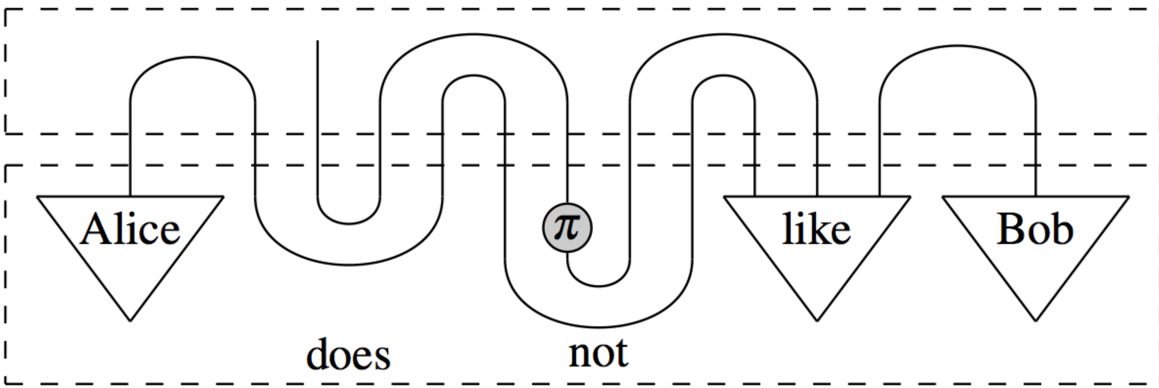
– time-reversal –

Operational reading:



String diagrams for natural language meaning:

String diagrams for natural language meaning:



- Top part: grammar
- Bottom part: meaning vectors

Lambek's Residuated monoids (1950's):

$$b \leq a \multimap c \Leftrightarrow a \cdot b \leq c \Leftrightarrow a \leq c \multimapinv b$$

Lambek's Residuated monoids (1950's):

$$b \leq a \multimap c \Leftrightarrow a \cdot b \leq c \Leftrightarrow a \leq c \multimapinv b$$

or equivalently,

$$a \cdot (a \multimap c) \leq c \leq a \multimap (a \cdot c)$$

$$(c \multimapinv b) \cdot b \leq c \leq (c \cdot b) \multimapinv b$$

Lambek's Residuated monoids (1950's):

$$b \leq a \multimap c \Leftrightarrow a \cdot b \leq c \Leftrightarrow a \leq c \multimapinv b$$

or equivalently,

$$a \cdot (a \multimap c) \leq c \leq a \multimap (a \cdot c)$$

$$(c \multimapinv b) \cdot b \leq c \leq (c \cdot b) \multimapinv b$$

Lambek's Pregroups (2000's):

$$a \cdot {}^{-1}a \leq 1 \leq {}^{-1}a \cdot a$$

$$b^{-1} \cdot b \leq 1 \leq b \cdot b^{-1}$$

From grammar to meaning:

For noun type n, verb type is ${}^{-1}n \cdot s \cdot n^{-1}$, so:

From grammar to meaning:

For noun type n, verb type is ${}^{-1}n \cdot s \cdot n^{-1}$, so:

$$n \cdot {}^{-1}n \cdot s \cdot n^{-1} \cdot n$$

From grammar to meaning:

For noun type n, verb type is ${}^{-1}n \cdot s \cdot n^{-1}$, so:

$$n \cdot {}^{-1}n \cdot s \cdot n^{-1} \cdot n \leq 1 \cdot s \cdot 1$$

From grammar to meaning:

For noun type n, verb type is ${}^{-1}n \cdot s \cdot n^{-1}$, so:

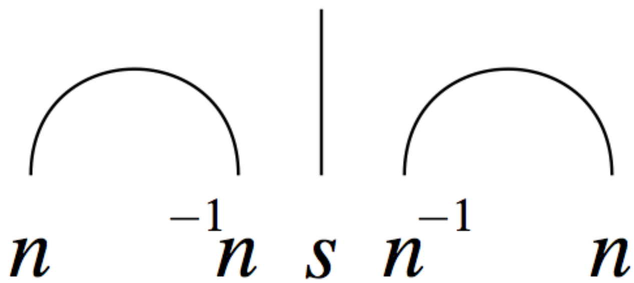
$$n \cdot {}^{-1}n \cdot s \cdot n^{-1} \cdot n \leq 1 \cdot s \cdot 1 \leq s$$

From grammar to meaning:

For noun type n, verb type is ${}^{-1}n \cdot s \cdot n^{-1}$, so:

$$n \cdot {}^{-1}n \cdot s \cdot n^{-1} \cdot n \leq 1 \cdot s \cdot 1 \leq s$$

Diagrammatic type reduction:

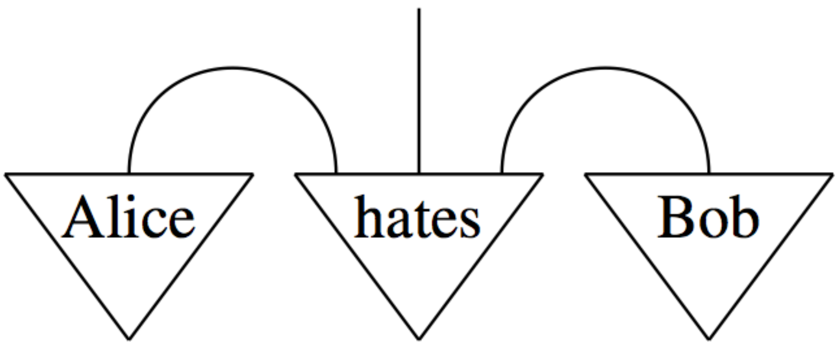


From grammar to meaning:

For noun type n, verb type is ${}^{-1}n \cdot s \cdot n^{-1}$, so:

$$n \cdot {}^{-1}n \cdot s \cdot n^{-1} \cdot n \leq 1 \cdot s \cdot 1 \leq s$$

Diagrammatic type reduction:



Algorithm for meaning composition:

Algorithm for meaning composition:

1. Perform grammatical type reduction:

$$(\textit{word type } 1)\ldots(\textit{word type } n) \quad \rightsquigarrow \quad \textit{sentence type}$$

2. Interpret diagrammatic type reduction as linear map:

$f ::$  $\mapsto \left(\sum_i \langle ii|\right) \otimes \mathrm{id} \otimes \left(\sum_i \langle ii|\right)$

3. Apply this map to tensor of word meaning vectors:

$$f\left(\overrightarrow{v}_1 \otimes \ldots \otimes \overrightarrow{v}_n\right)$$

Algorithm for meaning composition:

Model	ρ with cos	ρ with Eucl.
Verbs only	0.329	0.138
Additive	0.234	0.142
Multiplicative	0.095	0.024
Relational	0.400	0.149
Rank-1 approx. of relational	0.402	0.149
Separable	0.401	0.090
Copy-subject	0.379	0.115
Copy-object	0.381	0.094
Frobenius additive	**0.405**	0.125
Frobenius multiplicative	0.338	0.034
Frobenius tensored	**0.415**	0.010
Human agreement	0.60	

Dimitri Kartsaklis & Mehrnoosh Sadrzadeh (2013) *Prior Disambiguation of Word Tensors for Constructing Sentence Vectors*. In EMNLP'13.

Algorithm for meaning composition:

1. Perform grammatical type reduction:

$$(\textit{word type } 1)\ldots(\textit{word type } n) \ \rightsquigarrow \ \textit{sentence type}$$

2. Interpret diagrammatic type reduction as linear map:

f ::  $\mapsto \left(\sum_i \langle ii|\right) \otimes \mathrm{id} \otimes \left(\sum_i \langle ii|\right)$

3. Apply this map to tensor of word meaning vectors:

$$f\left(\overrightarrow{v}_1 \otimes \ldots \otimes \overrightarrow{v}_n\right)$$

Algorithm for social behaviour composition:

1. Perform social type reduction:

$$(\textit{person type } 1)\ldots(\textit{person type } n) \ \rightsquigarrow \ \textit{group type}$$

2. Interpret diagrammatic type reduction as linear map:

$f ::$  $\mapsto \left(\sum_i \langle ii|\right) \otimes \mathrm{id} \otimes \left(\sum_i \langle ii|\right)$

3. Apply this map to tensor of word meaning vectors:

$$f\left(\overrightarrow{v}_1 \otimes \ldots \otimes \overrightarrow{v}_n\right)$$

— Ch. 3 – Hilbert space from diagrams —

I would like to make a confession which may seem immoral: ***I do not believe absolutely in Hilbert space any more.***

— John von Neumann, letter to Garrett Birkhoff, 1935.

Here we define for string diagrams:

- ONBs, matrices and sums
- (multi-)linear maps & Hilbert spaces

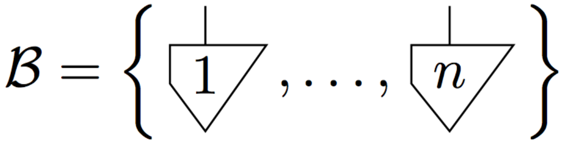
— Ch. 3 – Hilbert space from diagrams —

– orthonormal basis –

— Ch. 3 – Hilbert space from diagrams —

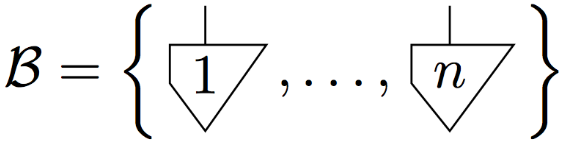
– *basis* –

... := (minimal) set:

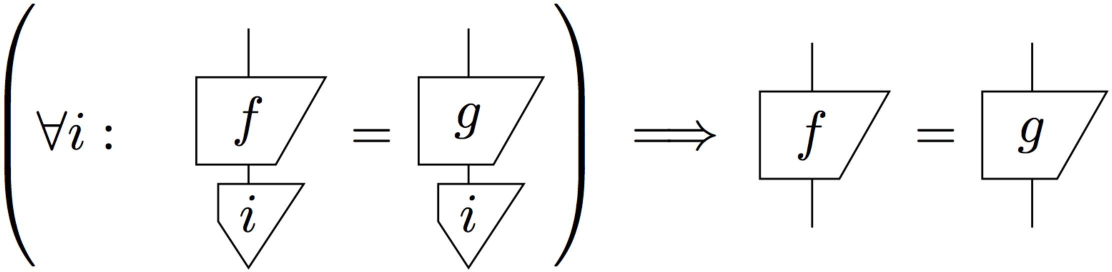


– *basis* –

... := (minimal) set:



such that:



— Ch. 3 – Hilbert space from diagrams —

– orthonormal –

For:

- unit number := ‘empty’ diagram
- zero number := ‘black hole’ diagram

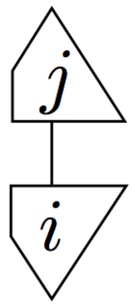
— Ch. 3 – Hilbert space from diagrams —

– orthonormal –

For:

- unit number := ‘empty’ diagram
- zero number := ‘black hole’ diagram

we set:

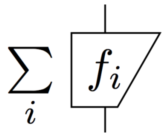


$$= \delta_{ij}$$

— Ch. 3 – Hilbert space from diagrams —

– sum –

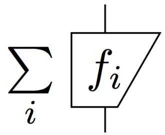
... := for processes of ‘same type’ there exists:



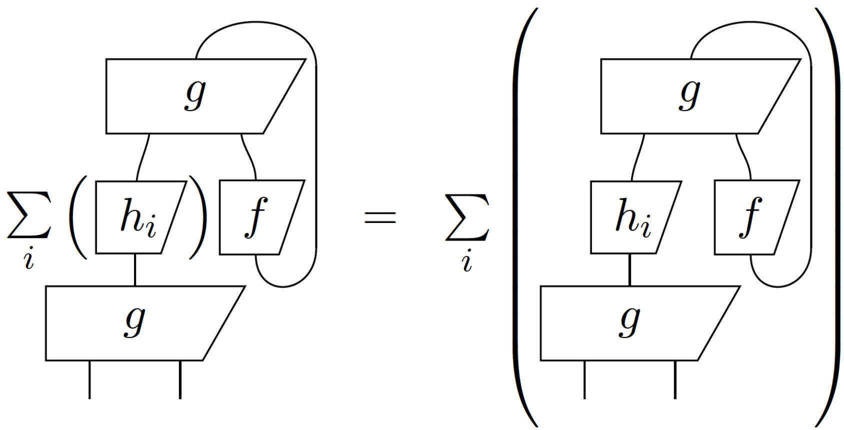
— Ch. 3 – Hilbert space from diagrams —

– sum –

... := for processes of 'same type' there exists:



which 'moves around':



— Ch. 3 – Hilbert space from diagrams —

– definition –

Thm.

(multi) linear maps := string diagrams s.t.:

- each system has ONB
- $\exists$ sums
- numbers are $\mathbb{C}$

— Ch. 3 – Hilbert space from diagrams —

– definition –

Thm.

(multi) linear maps := string diagrams s.t.:

- each system has ONB
- $\exists$ sums
- numbers are $\mathbb{C}$

Hilbert space := states for a system with Born-rule.

— Ch. 3 – Hilbert space from diagrams —

– definition –

Thm.

(multi) linear maps := string diagrams s.t.:

- each system has ONB
- $\exists$ sums
- numbers are $\mathbb{C}$

Hilbert space := states for a system with Born-rule.

(note: tensor product comes for free)

— Ch. 3 – Hilbert space from diagrams —

– completeness –

— Ch. 3 – Hilbert space from diagrams —

– completeness –

THM. (Selinger, 2008)

An equation between string diagrams holds, if and only if it holds for Hilbert spaces and linear maps.

— Ch. 3 – Hilbert space from diagrams —

– completeness –

THM. (Selinger, 2008)

An equation between string diagrams holds, if and only if it holds for Hilbert spaces and linear maps.

I.e. defining Hilbert spaces and linear maps in this manner is a 'conservative extension' of string diagrams.

— Ch. 4 – Quantum processes —

The art of progress is to preserve order amid change, and to preserve change amid order.

— Alfred North Whitehead, Process and Reality, 1929.

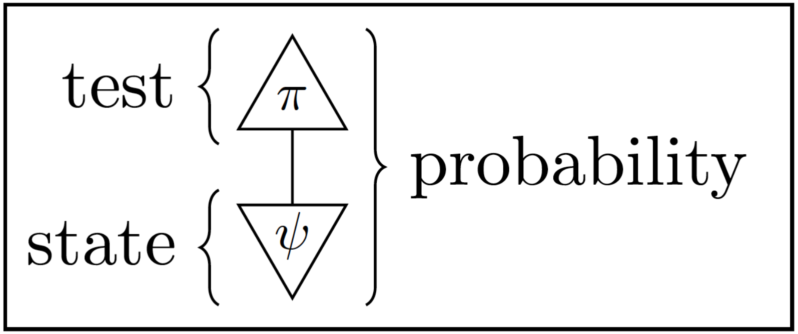
Here we introduce:

- pure quantum maps
- general quantum maps
- causality, no-signalling & Stinespring dilation

— Ch. 4 – Quantum processes —

– pure quantum maps –

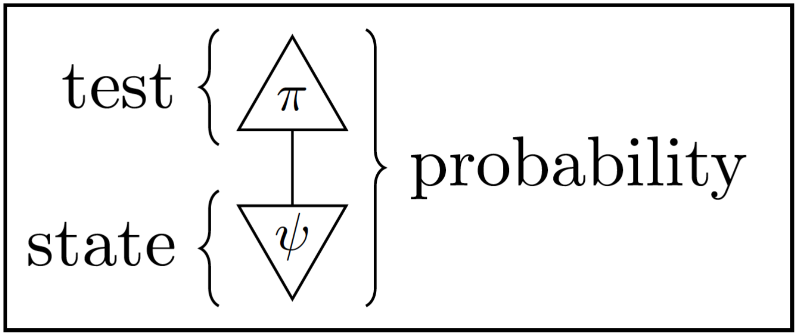
Goal 1:



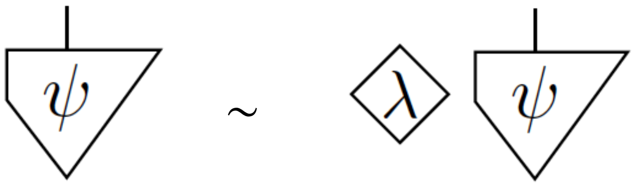
— Ch. 4 – Quantum processes —

– pure quantum maps –

Goal 1:



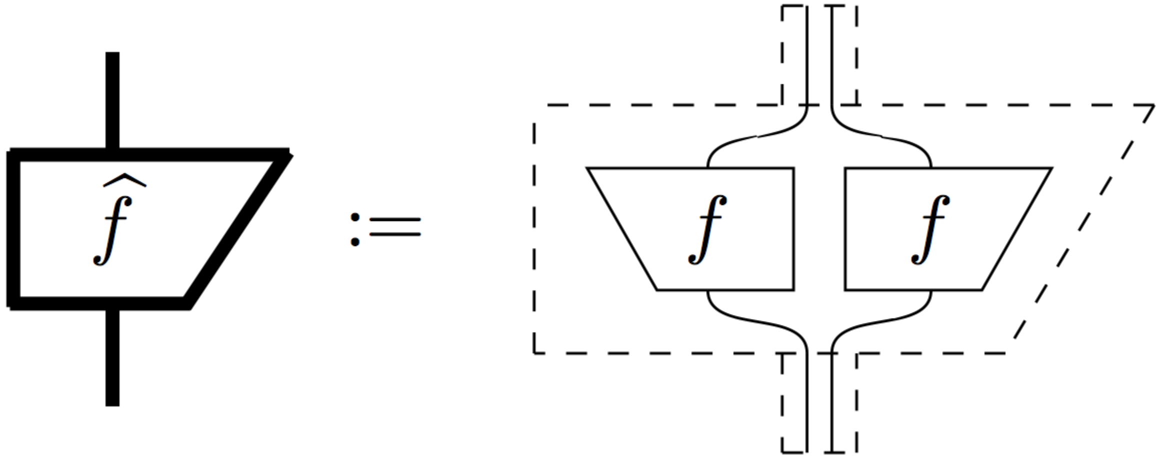
Goal 2:



— Ch. 4 – Quantum processes —

– pure quantum maps –

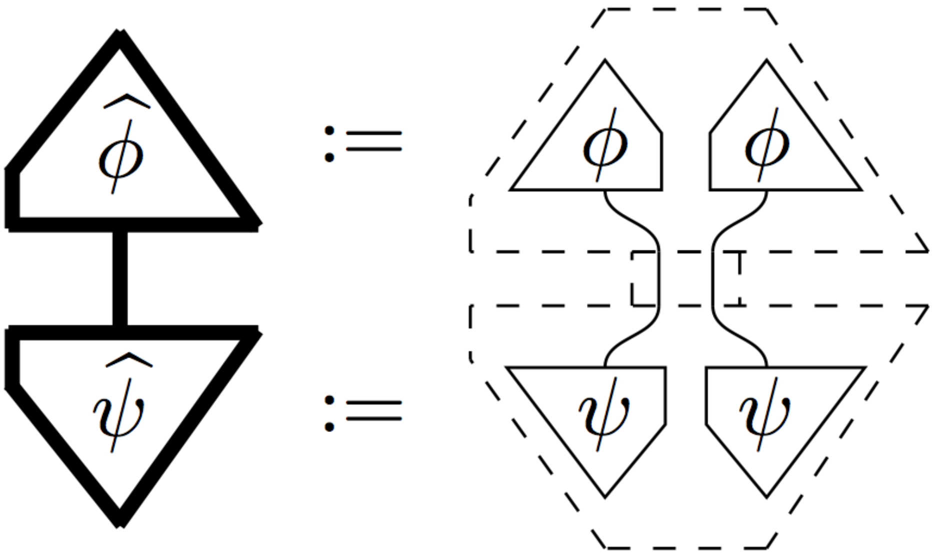
... :=



— Ch. 4 – Quantum processes —

– pure quantum maps –

Born-rule :=



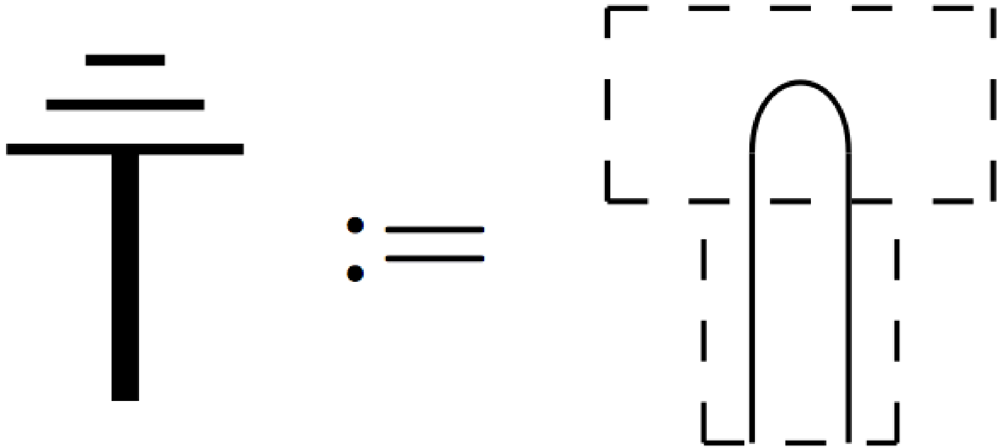
— Ch. 4 – Quantum processes —

– quantum maps –

— Ch. 4 – Quantum processes —

– quantum maps –

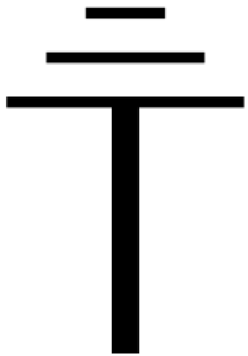
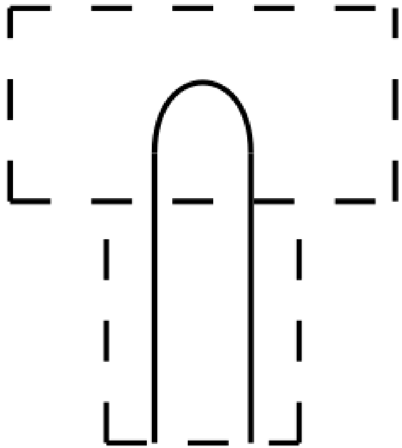
Discarding :=



— Ch. 4 – Quantum processes —

– quantum maps –

Discarding :=

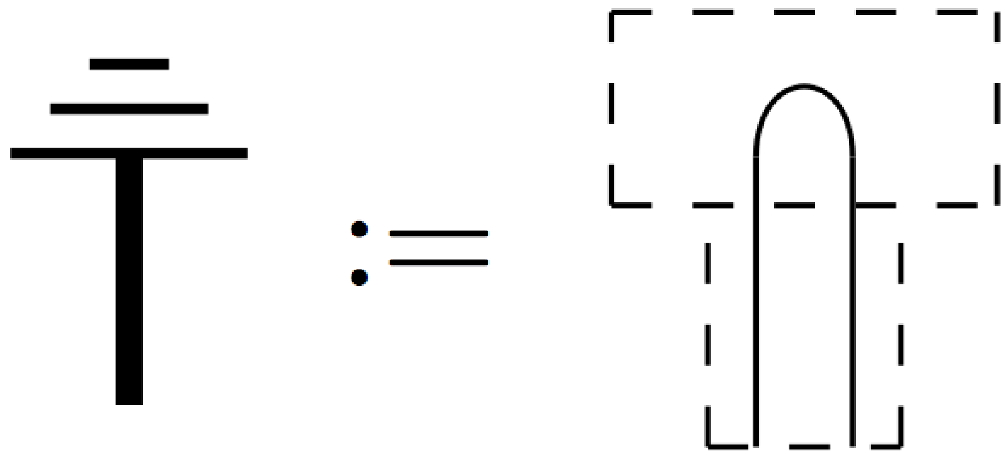
 $:=$ 

Thm. Discarding is not a pure quantum map.

— Ch. 4 – Quantum processes —

– quantum maps –

Discarding :=



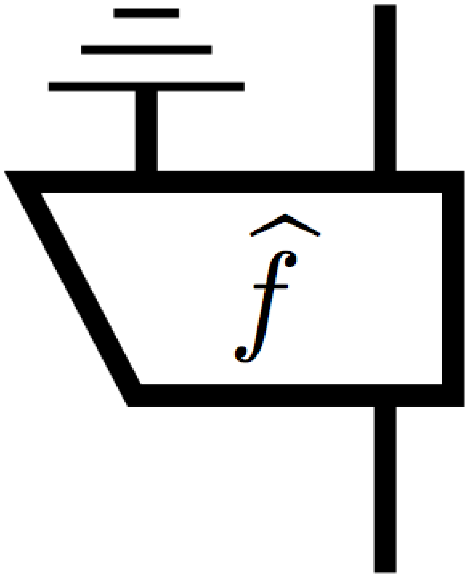
Thm. Discarding is not a pure quantum map.

Pf. Something connected $\neq$ something disconnected.

— Ch. 4 – Quantum processes —

– quantum maps –

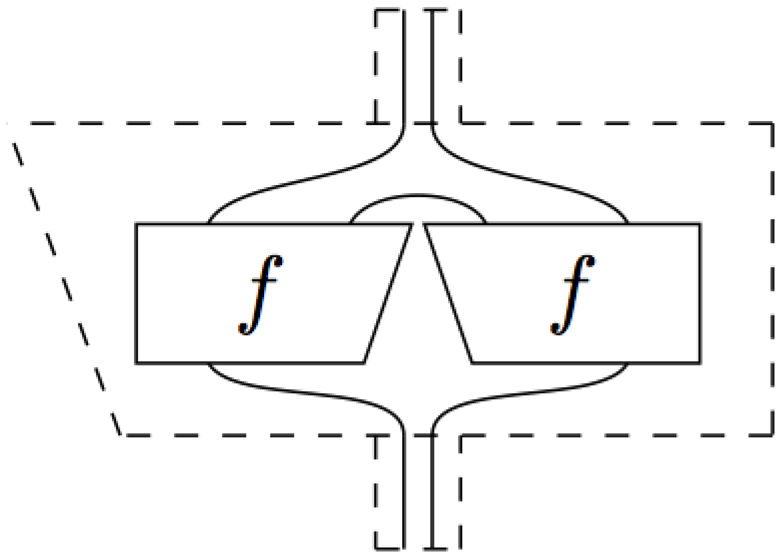
... := pure quantum maps + discarding



— Ch. 4 – Quantum processes —

– quantum maps –

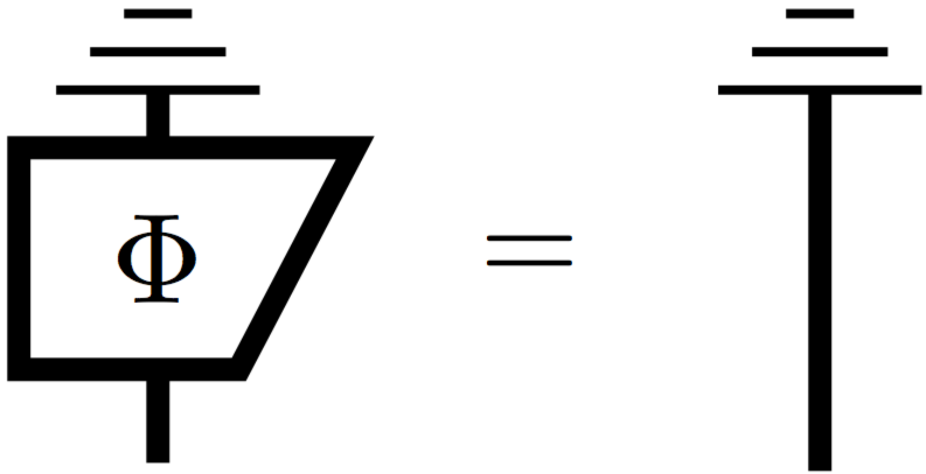
... := pure quantum maps + discarding



(cf. Krauss form of CP-map)

— Ch. 4 – Quantum processes —

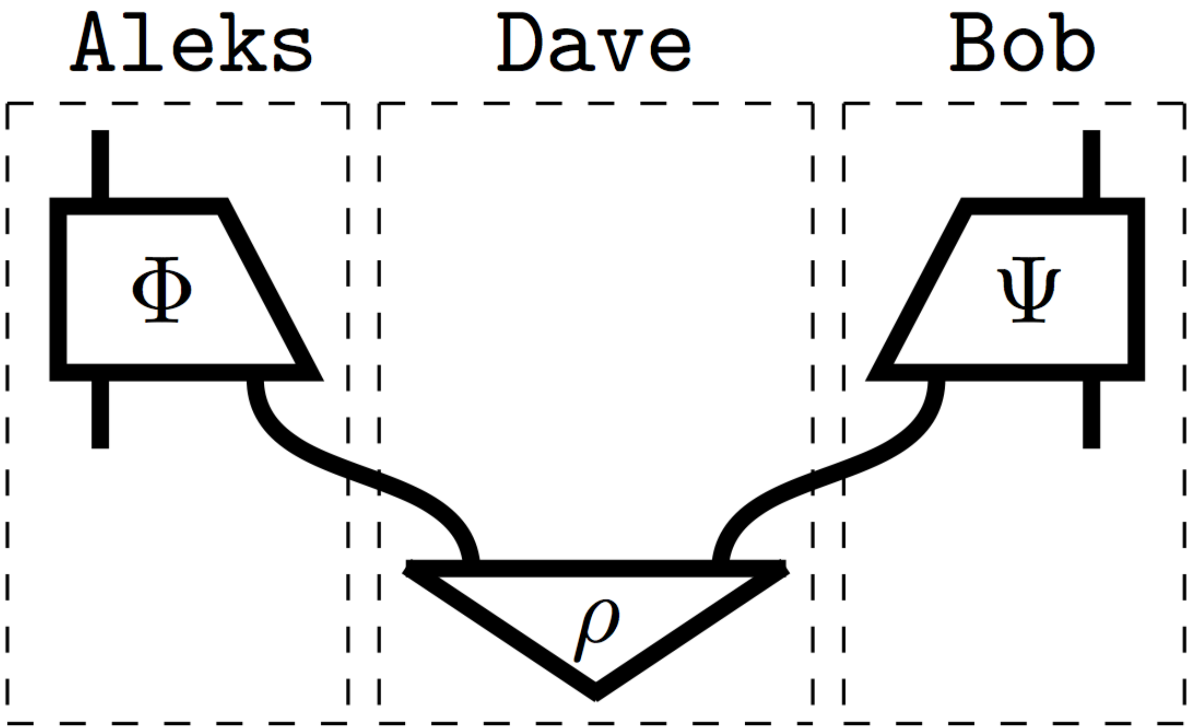
– axiom: causality (= terminality of I) –



(a surprising plethora of things follows: an arrow of time, non-signalling, relativistic covariance, ...)

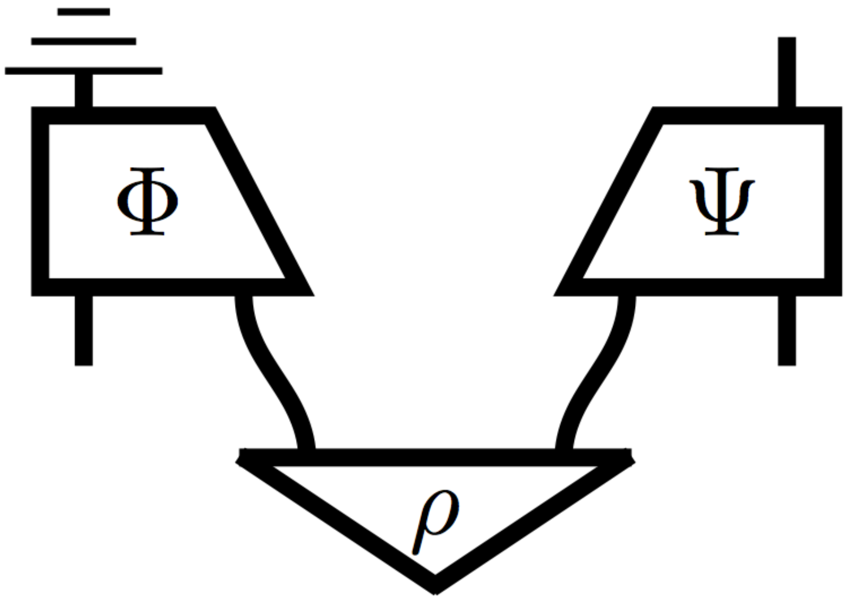
— Ch. 4 – Quantum processes —

– causality –



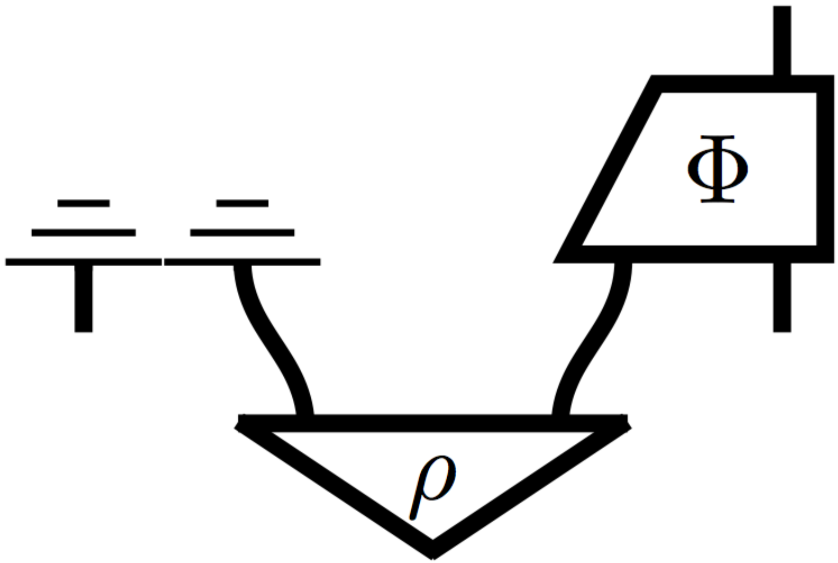
— Ch. 4 – Quantum processes —

– causality –



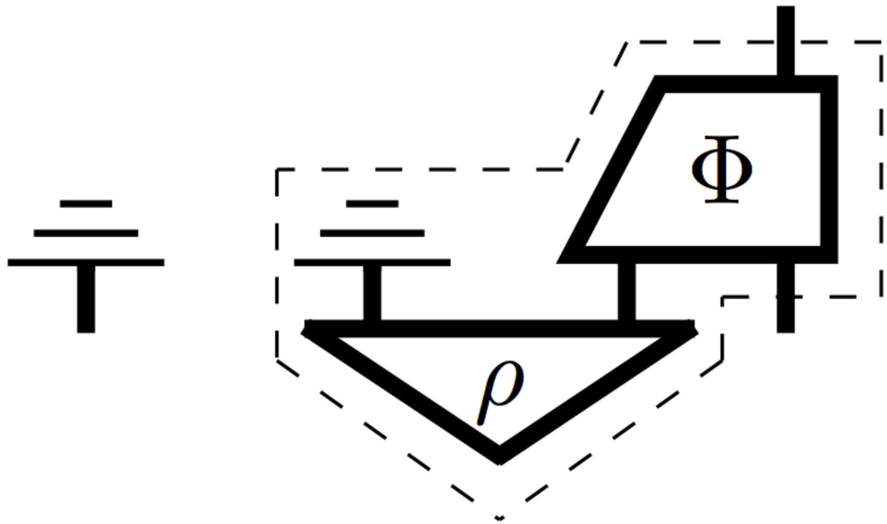
— Ch. 4 – Quantum processes —

– causality –



— Ch. 4 – Quantum processes —

– causality –



— Ch. 4 – Quantum processes —

– axiom: causality (= terminality of I) –

Prop. For pure quantum maps:

$$\boxed{\text{causality} \iff \text{isometry}}$$

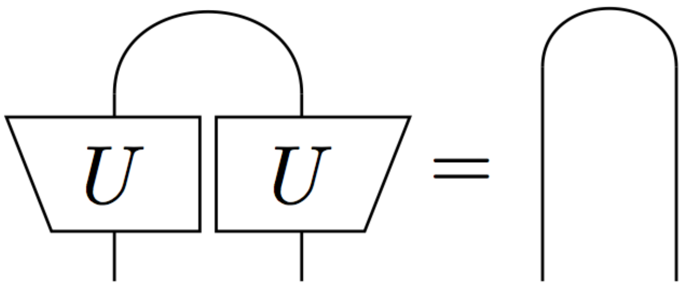
— Ch. 4 – Quantum processes —

– axiom: causality (= terminality of I) –

Prop. For <u>pure</u> quantum maps:

$$\boxed{\text{causality} \iff \text{isometry}}$$

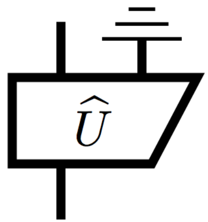
Pf.



— Ch. 4 – Quantum processes —

– axiom: causality (= terminality of I) –

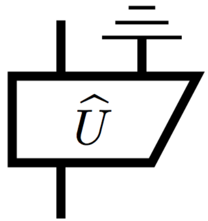
Prop. For general quantum maps:

causality $\overset{\text{Stinespring}}{\Longleftrightarrow}$ of the form 

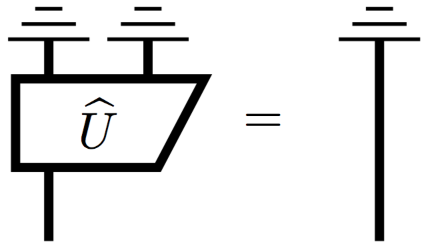
— Ch. 4 – Quantum processes —

– axiom: causality (= terminality of I) –

Prop. For general quantum maps:

causality $\overset{\text{Stinespring}}{\Longleftrightarrow}$ of the form 

Pf.

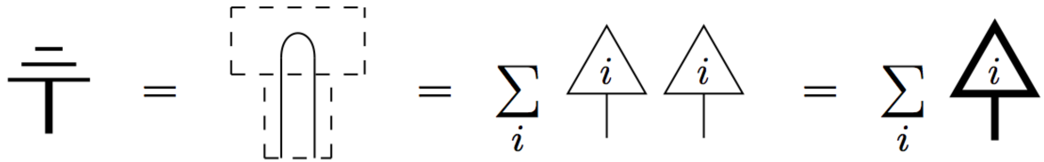


Candidate systems:

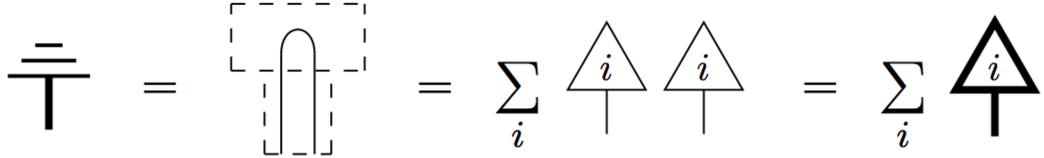
- *vector space with inner-product*:
  - *pure (or closed) quantum states (complex)*
  - *standard natural language processing (real)*
- **density matrices with trace:**
  - **mixed (or open) quantum states**
  - **neo natural language processing**
- *more abstract models and constructions*

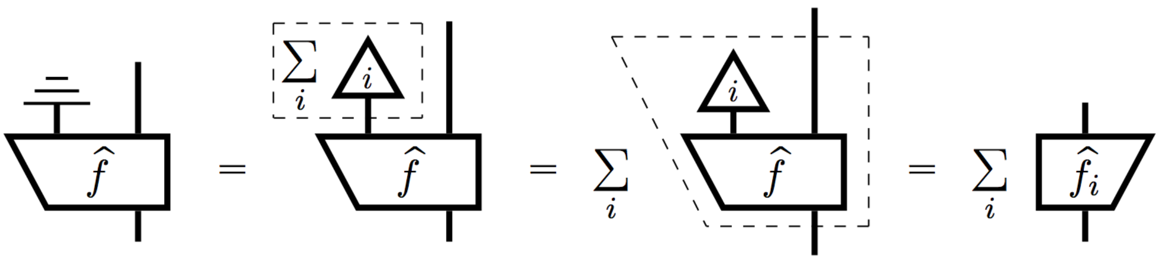
Mixing:

Mixing:

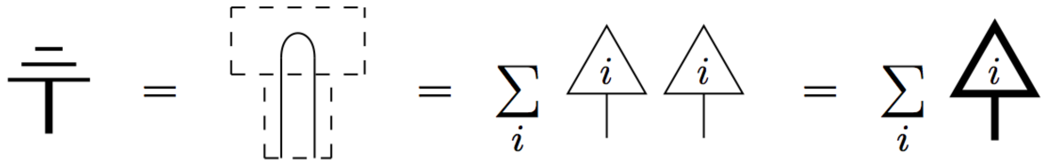


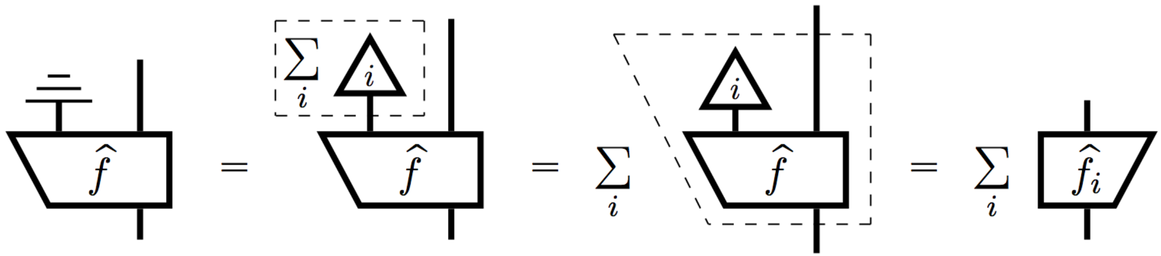
Mixing:



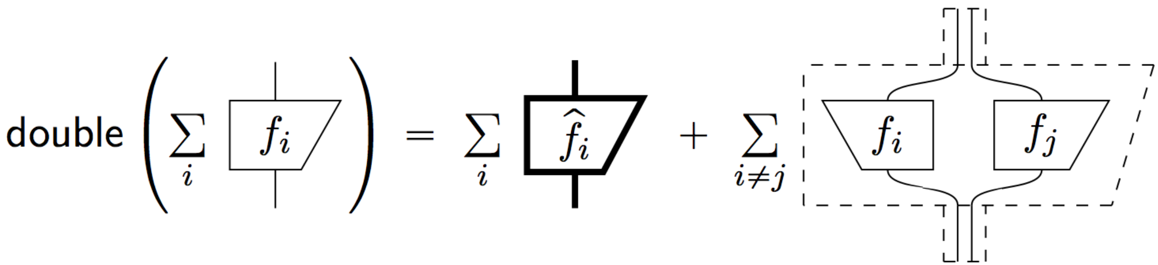


Mixing:





Two distinct sums:



Advantages over vector spaces of meaning:

Advantages over vector spaces of meaning:

- Ambiguity:
  - Robin Piedeleu's MSc thesis (2014)
  - Dimitri Kartsaklis's PhD thesis (2014)

Advantages over vector spaces of meaning:

- Ambiguity:
  - Robin Piedeleu's MSc thesis (2014)
  - Dimitri Kartsaklis's PhD thesis (2014)
- Information/propositional content:
  - Esma Balkir's MSc thesis (2014)

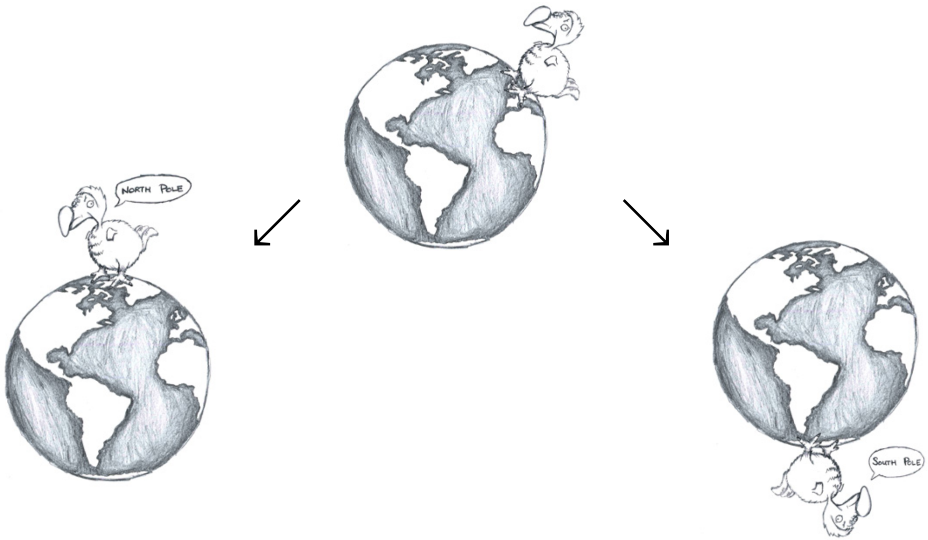
Advantages over vector spaces of meaning:

- Ambiguity:
  - Robin Piedeleu's MSc thesis (2014)
  - Dimitri Kartsaklis's PhD thesis (2014)
- Information/propositional content:
  - Esma Balkir's MSc thesis (2014)
- Construction can be iterated

— Ch. 5 – Quantum measurement —

The bureaucratic mentality is the only constant in the universe.

— Dr. McCoy, Star Trek IV: The Voyage Home, 2286.



— Ch. 5 – Quantum measurement —

– is quantum measurement weird? –

— Ch. 5 – Quantum measurement —

– is quantum measurement weird? –

Heisenberg-Bohr:

any attempt to observe is bound to disturb

— Ch. 5 – Quantum measurement —

– is quantum measurement weird? –

Heisenberg-Bohr:

any attempt to observe is bound to disturb

Newtonian equivalent:

locating a baloon by mechanical means

— Ch. 5 – Quantum measurement —

– is quantum measurement weird? –

Heisenberg-Bohr:

any attempt to observe is bound to disturb

Newtonian equivalent:

locating a baloon by mechanical means

Resulting diagnosis:

we suffer from quantum-blindness

— Ch. 6 – Picturing classical processes —

Damn it! I knew she was a monster! John! Amy! Listen! Guard your buttholes.

— David Wong, This Book Is Full of Spiders, 2012.

Here we fully diagrammatically describe:

- classical-quantum processes
- classical data as spiders
- fully diagrammatic protocols

— Ch. 6 – Picturing classical processes —

– classical-quantum maps –

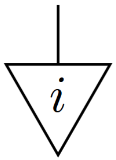
Main idea:

$$\frac{\text{classical system}}{\textbf{quantum system}} = \frac{\text{single wire}}{\textbf{double wire}}$$

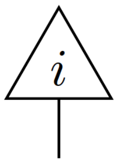
— Ch. 6 – Picturing classical processes —

– classical-quantum maps –

Fix ONB and set:



:= “providing classical value *i*”

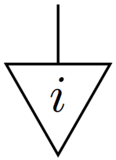


:= “testing for classical value *i*”

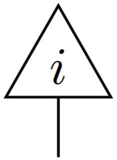
— Ch. 6 – Picturing classical processes —

– classical-quantum maps –

Fix ONB and set:

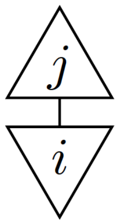


:= “providing classical value i”



:= “testing for classical value i”

Sanity check:



$$= \delta_{ij}$$

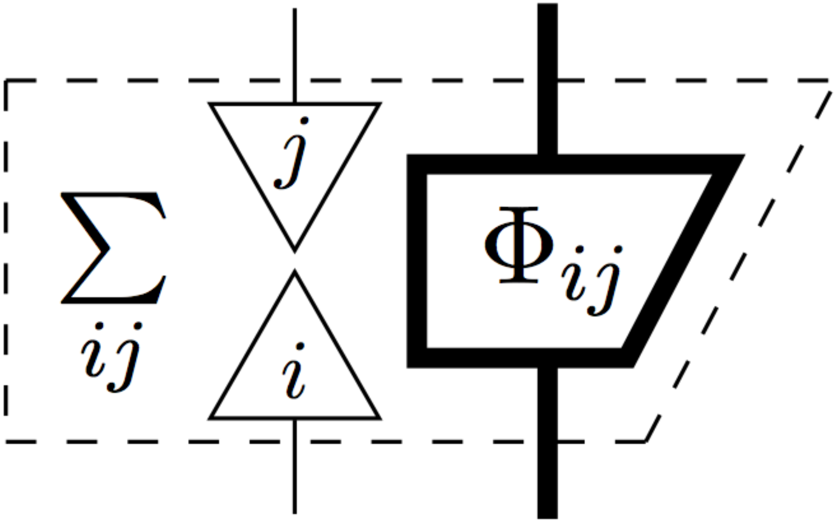
— Ch. 6 – Picturing classical processes —

– classical-quantum maps –

— Ch. 6 – Picturing classical processes —

– classical-quantum maps –

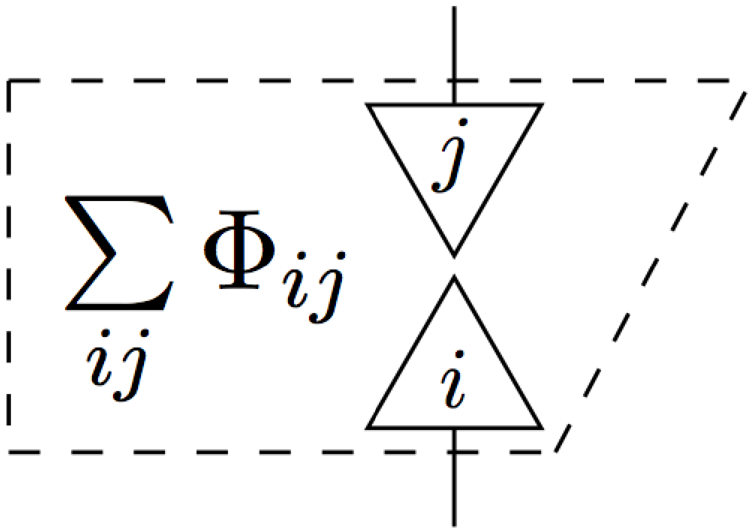
... :=



— Ch. 6 – Picturing classical processes —

– classical-quantum maps –

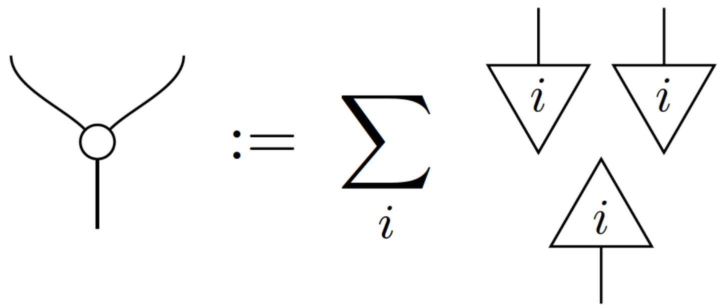
Classical map :=



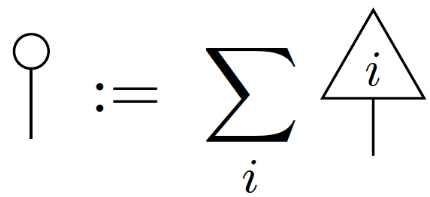
— Ch. 6 – Picturing classical processes —

– classical-quantum maps –

copy :=



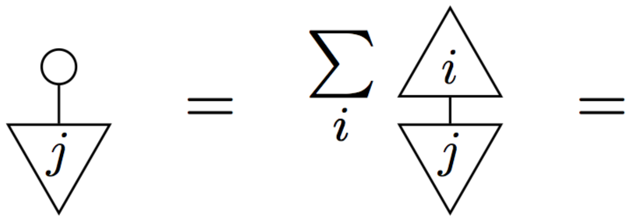
delete :=

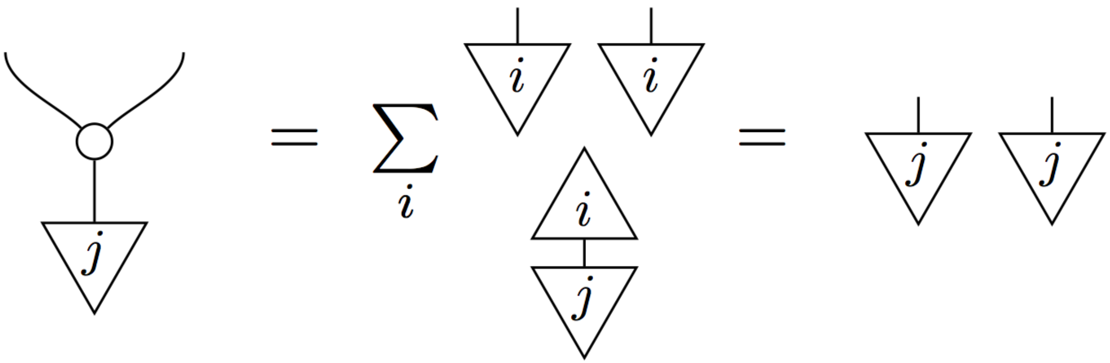


— Ch. 6 – Picturing classical processes —

– classical-quantum maps –

Indeed:

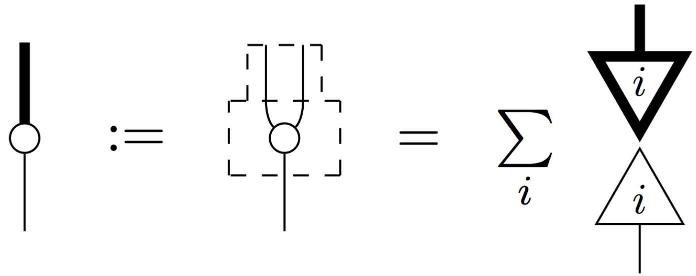




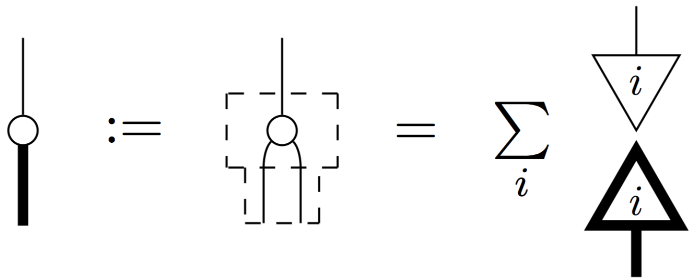
— Ch. 6 – Picturing classical processes —

– classical-quantum maps –

encode :=



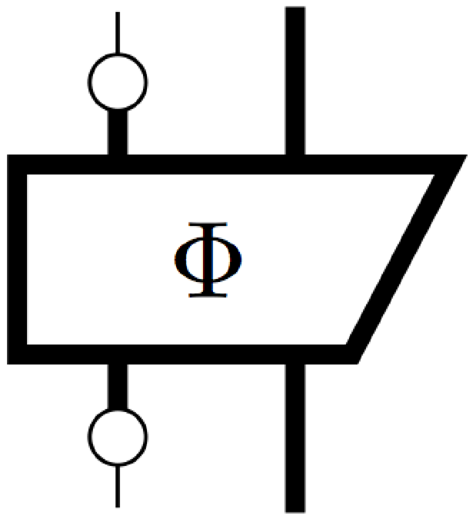
measure :=



— Ch. 6 – Picturing classical processes —

– classical-quantum maps –

Thm. ... are always of the form:

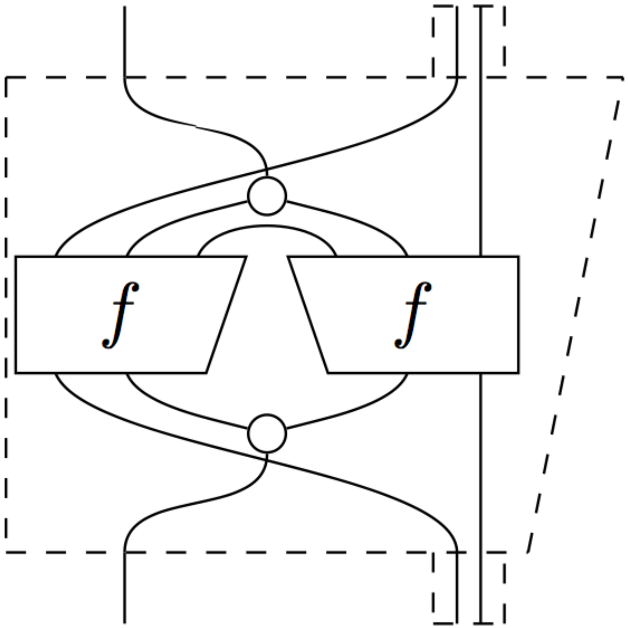


where **Φ** is a quantum map.

— Ch. 6 – Picturing classical processes —

– classical-quantum maps –

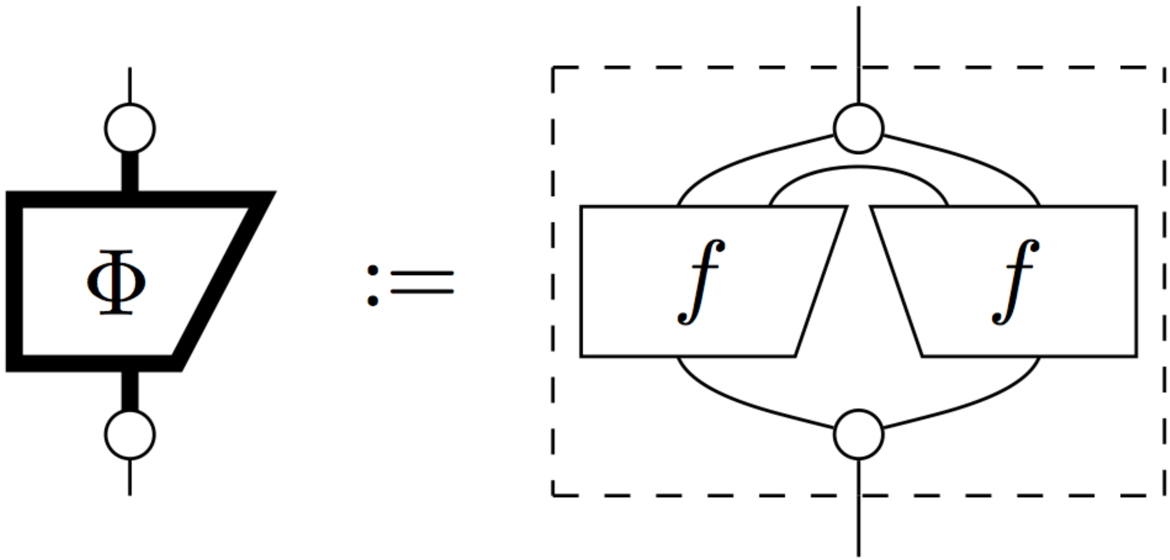
Thm. ... are always of the form:



— Ch. 6 – Picturing classical processes —

– classical maps –

Thm. ... are always of the form:

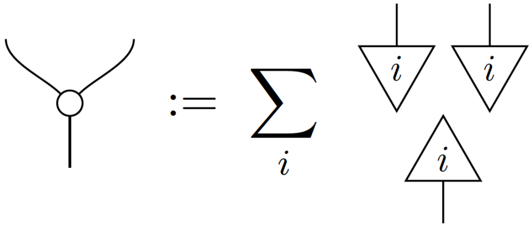


— Ch. 6 – Picturing classical processes —

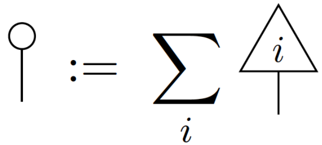
– spiders –

— Ch. 6 – Picturing classical processes —

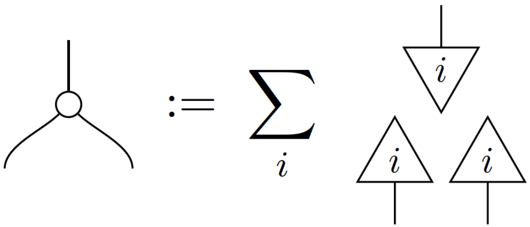
– spiders –



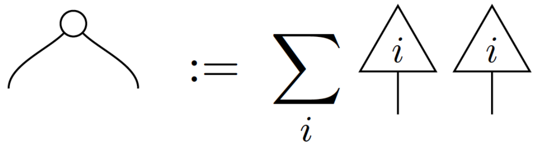
copy



delete



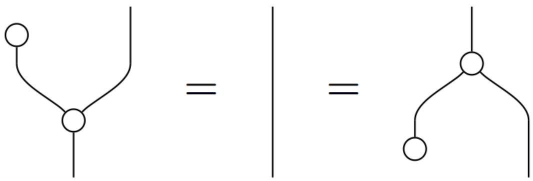
match

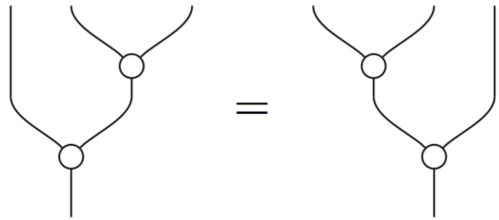


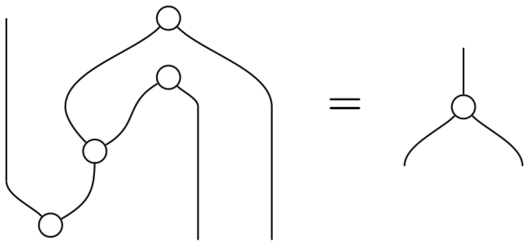
compare

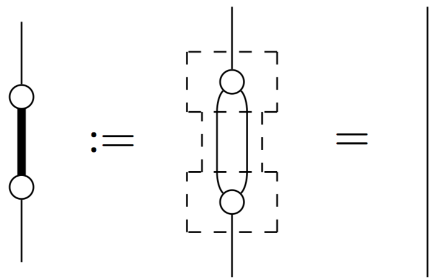
— Ch. 6 – Picturing classical processes —

– *spiders* –





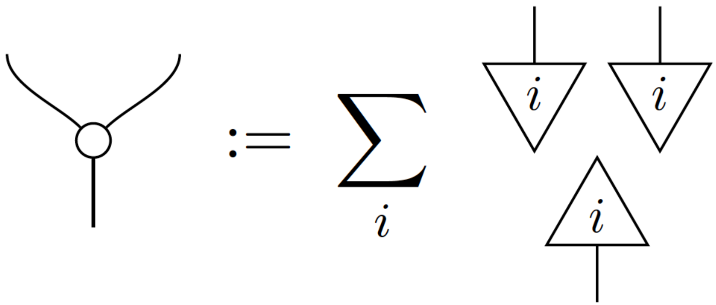




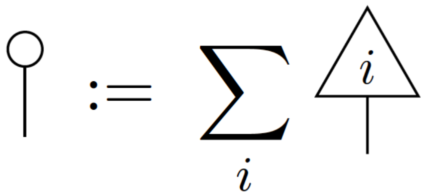
— Ch. 6 – Picturing classical processes —

– spiders –

copy :=



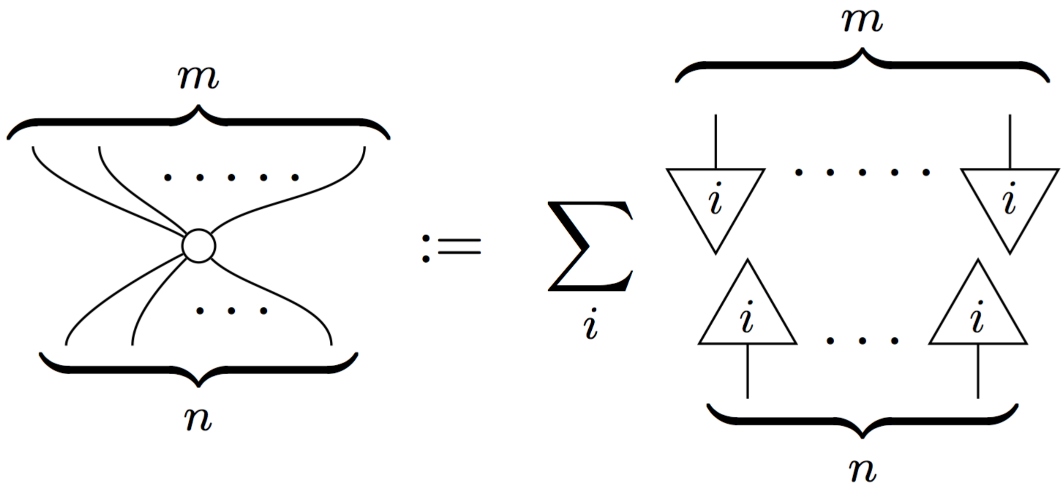
delete :=



— Ch. 6 – Picturing classical processes —

– spiders –

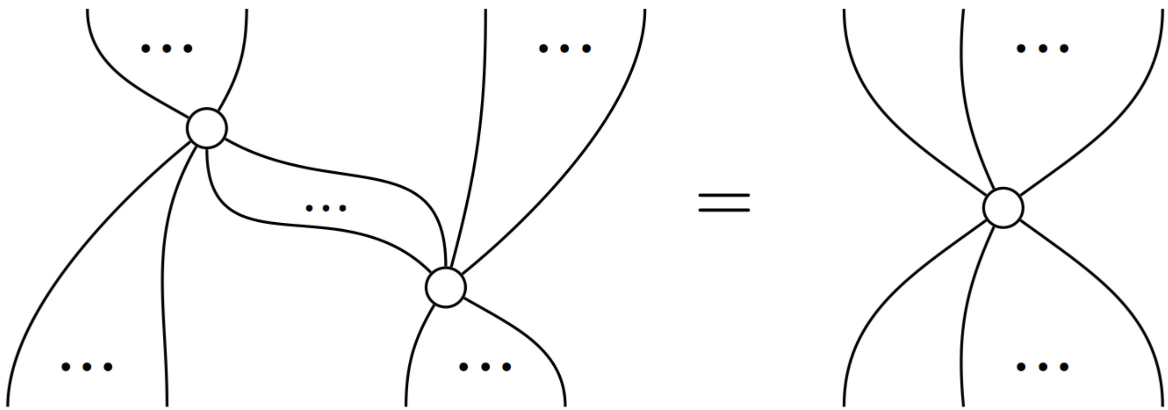
... :=



— Ch. 6 – Picturing classical processes —

– spiders –

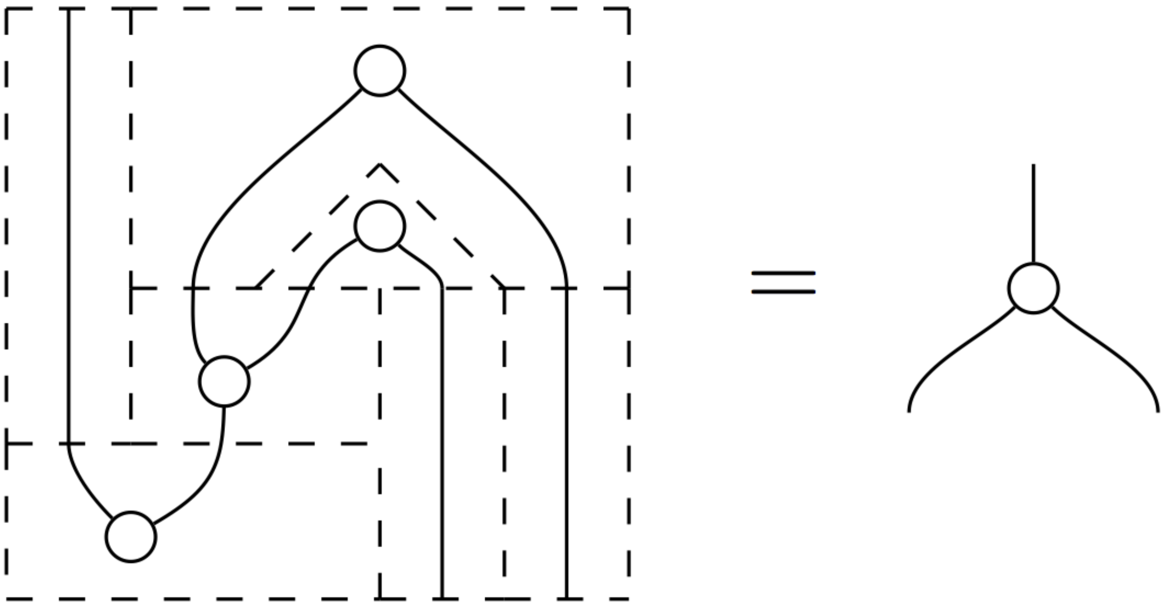
Prop.



— Ch. 6 – Picturing classical processes —

– spiders –

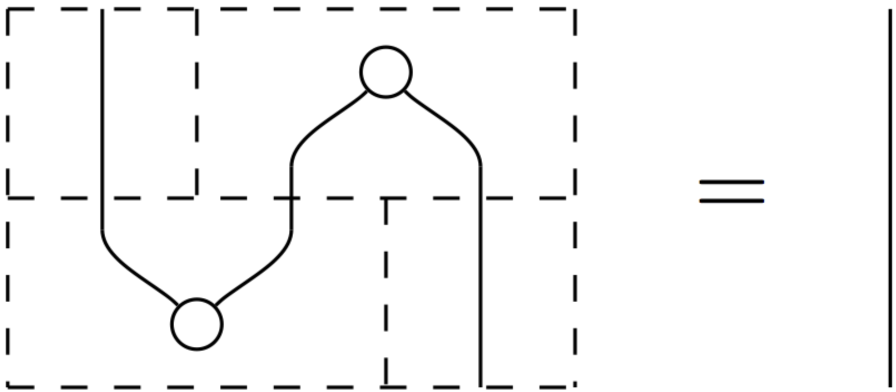
E.g.:



— Ch. 6 – Picturing classical processes —

– spiders –

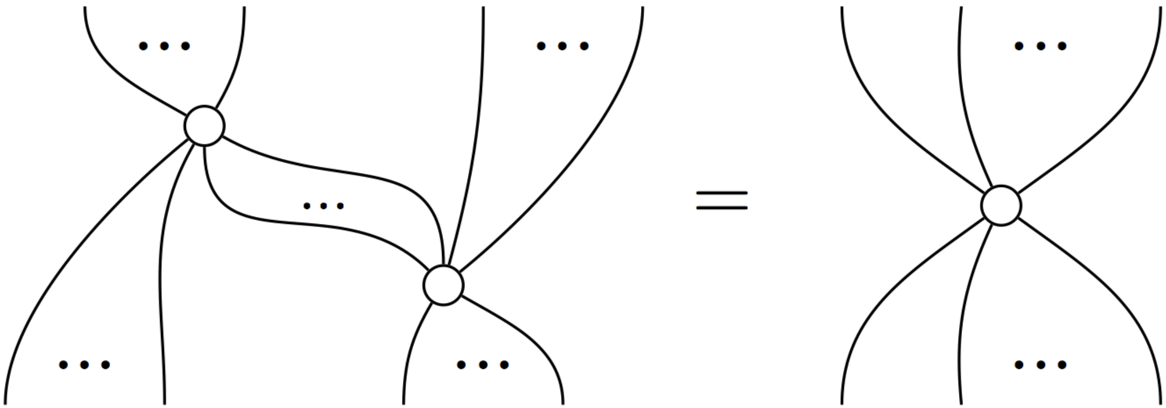
... and in particular:



— Ch. 6 – Picturing classical processes —

– spiders –

THM.



These equations imply ONB for linear maps!

— Ch. 3 – Hilbert space from diagrams —

– completeness –

THM. (Kissinger, 2014)

An equation between dot diagrams holds, if and only if it holds for Hilbert spaces with a fixed basis and linear maps, that is, for matrices of complex numbers.

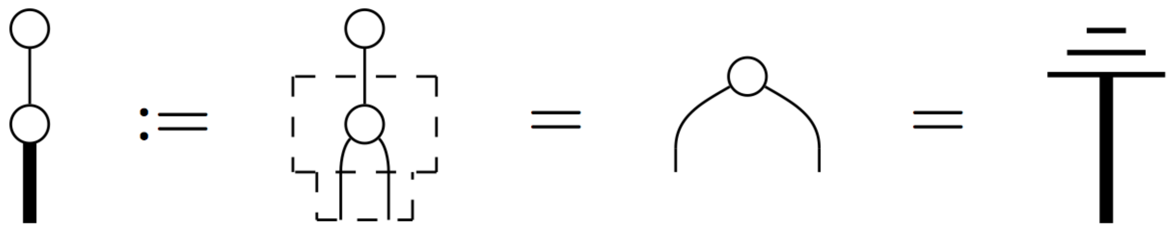
— Ch. 6 – Picturing classical processes —

– causality –

— Ch. 6 – Picturing classical processes —

– causality –

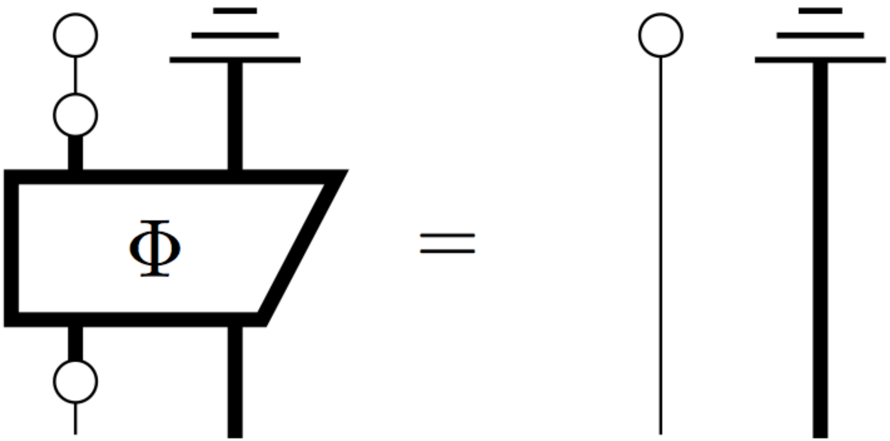
Lem.



— Ch. 6 – Picturing classical processes —

– causality –

Thm. ... :=

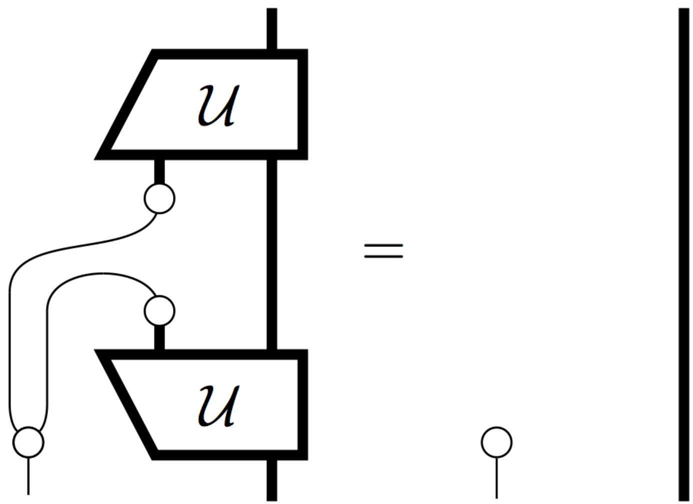


— Ch. 6 – Picturing classical processes —

– teleportation diagrammatically –

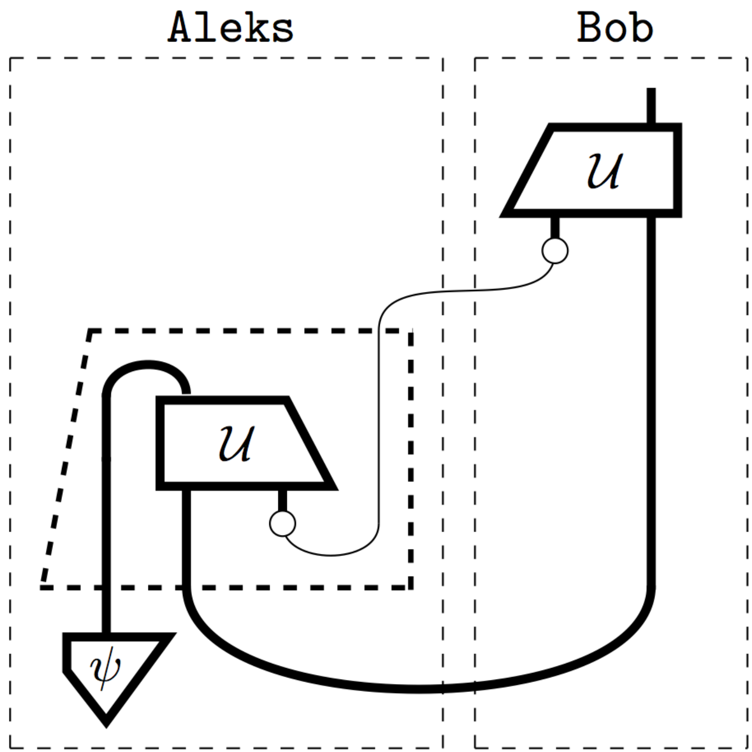
– teleportation diagrammatically –

Prop. Controlled isometry:



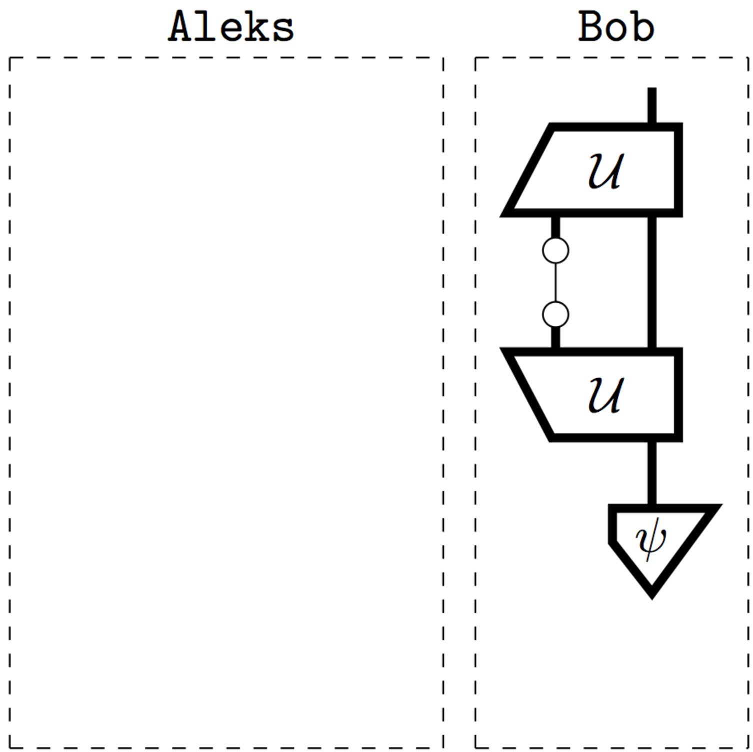
— Ch. 6 – Picturing classical processes —

– teleportation diagrammatically –



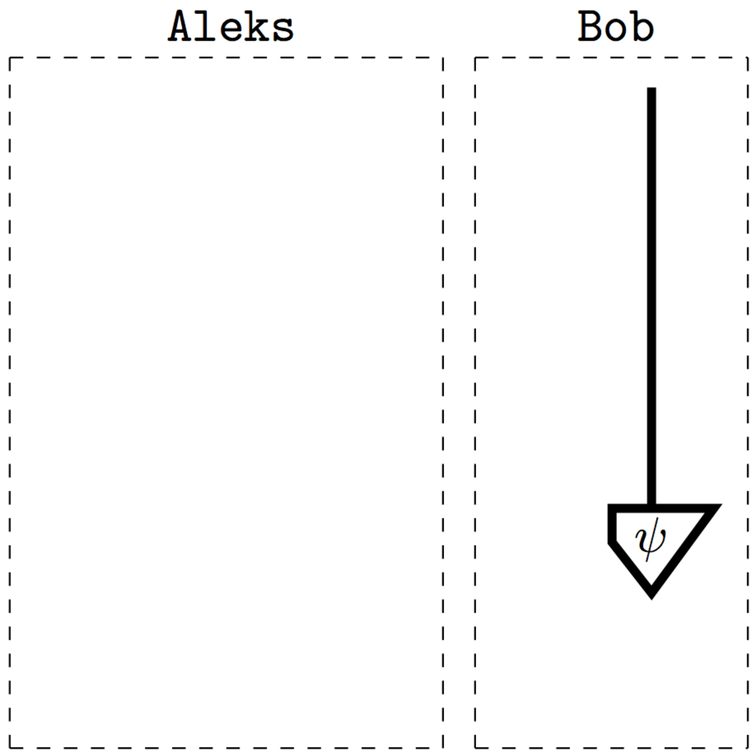
— Ch. 6 – Picturing classical processes —

– teleportation diagrammatically –



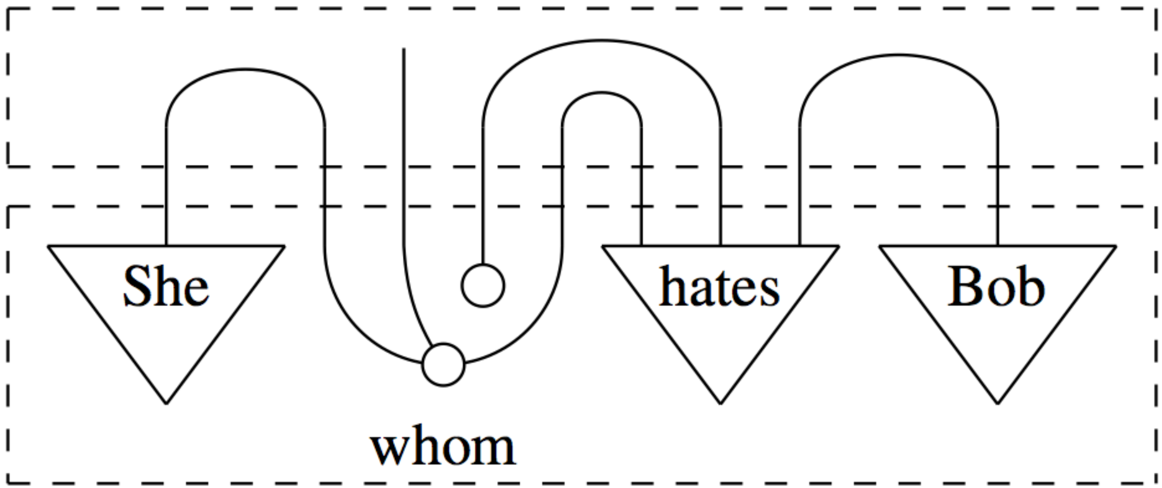
— Ch. 6 – Picturing classical processes —

– teleportation diagrammatically –



Dot diagrams for natural language meaning:

Dot diagrams for natural language meaning:



- Top part: grammar
- Bottom part: meaning vectors

Dual type social dynamic-epistemic paradigm:

Dual type social dynamic-epistemic paradigm:

- Thick wires := social types
- Thin wires := information types
- Measurement := share information
- Encode := being affected by information

Dual type social dynamic-epistemic paradigm:

- Thick wires := social types
- Thin wires := information types
- Measurement := share information
- Encode := being affected by information

Not so crazy:

- A. Baltag, L. S. Moss, S. Solecki (1999) *The logic of public announcements, common knowledge, and private suspicions*. TARK'98.
- A. Baltag, BC & M. Sadrzadeh (2006) *Epistemic actions as resources*. arXiv:math/0608166.
- A. Carboni & R. F. C. Walters (1987) *Cartesian bicategories I*. JPAA.
- BC, E. O. Paquette & D. Pavlovic (2009) *Classical and quantum structuralism*. arXiv:0904.1997

— Ch. 7 – Picturing phases & complementarity —

When spider webs unite, they can tie up a lion.

— Ethiopian proverb.

Here we identify in terms of spiders:

- phases
- complementarity
- strong complementarity

— Ch. 7 – Picturing phases & complementarity —

– complementary spiders –

Thm.