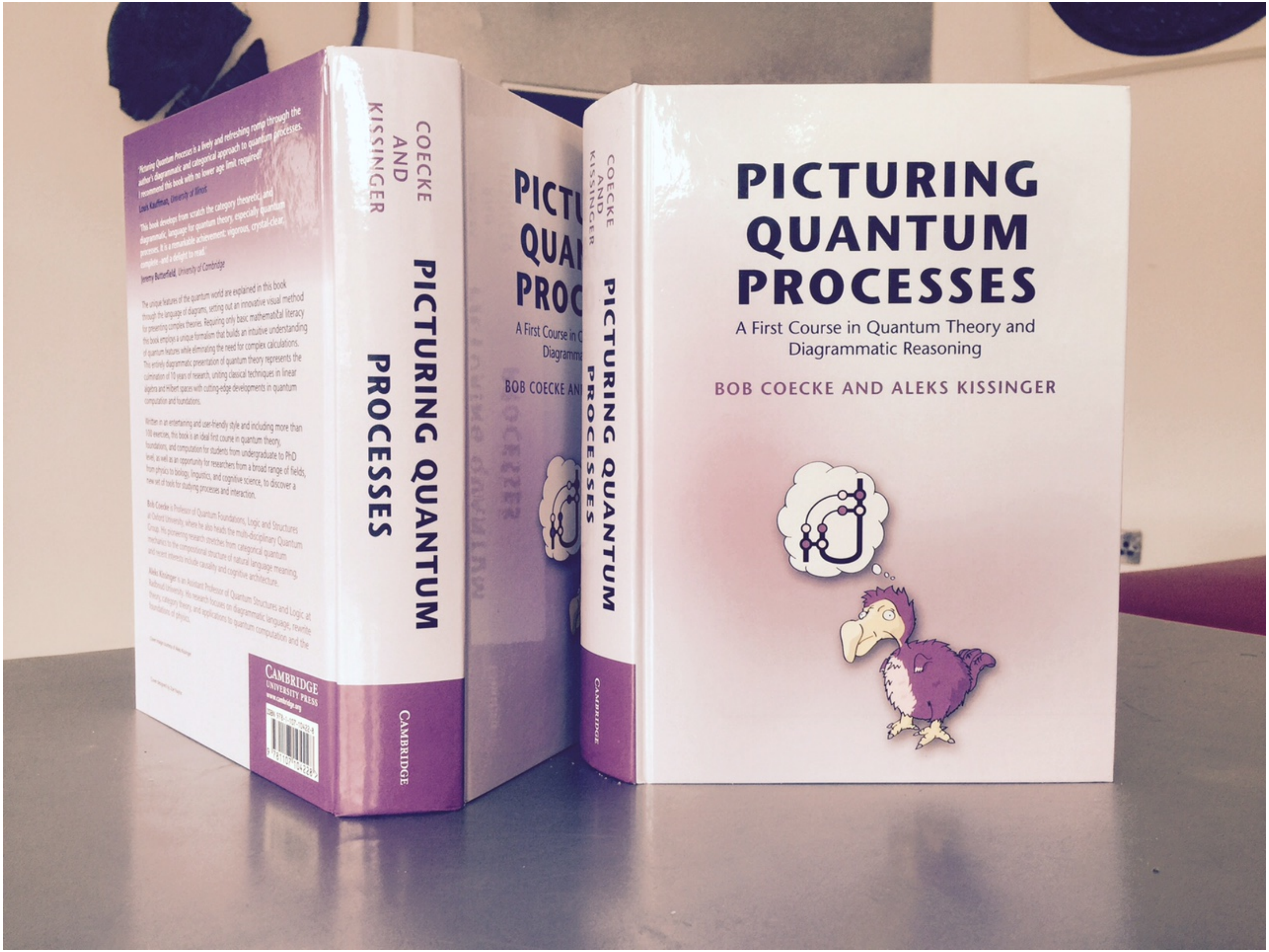
Today’s menu:

- pictorial formalism for **quantum systems**



Today's menu:

- pictorial formalism for **quantum systems**
- theory for **natural language meaning** composition

QUANTUM LINGUISTICS Leap forward for artificial intelligence

NewScientist

WEEKLY 11 December 2010

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

SCIENTIFIC AMERICAN™

Sign In / Register

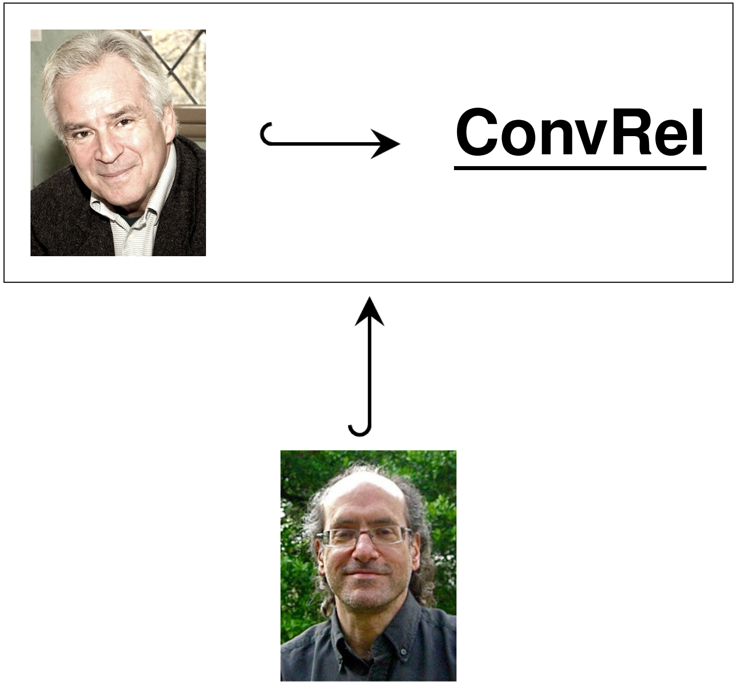
Search ScientificAmerican.com

Quantum Mechanical Words and Mathematical Organisms

By Joselle Kehoe | May 16, 2013 | 10

Today's menu:

- pictorial formalism for **quantum systems**
- theory for **natural language meaning** composition
- theory for **compositional cognition**



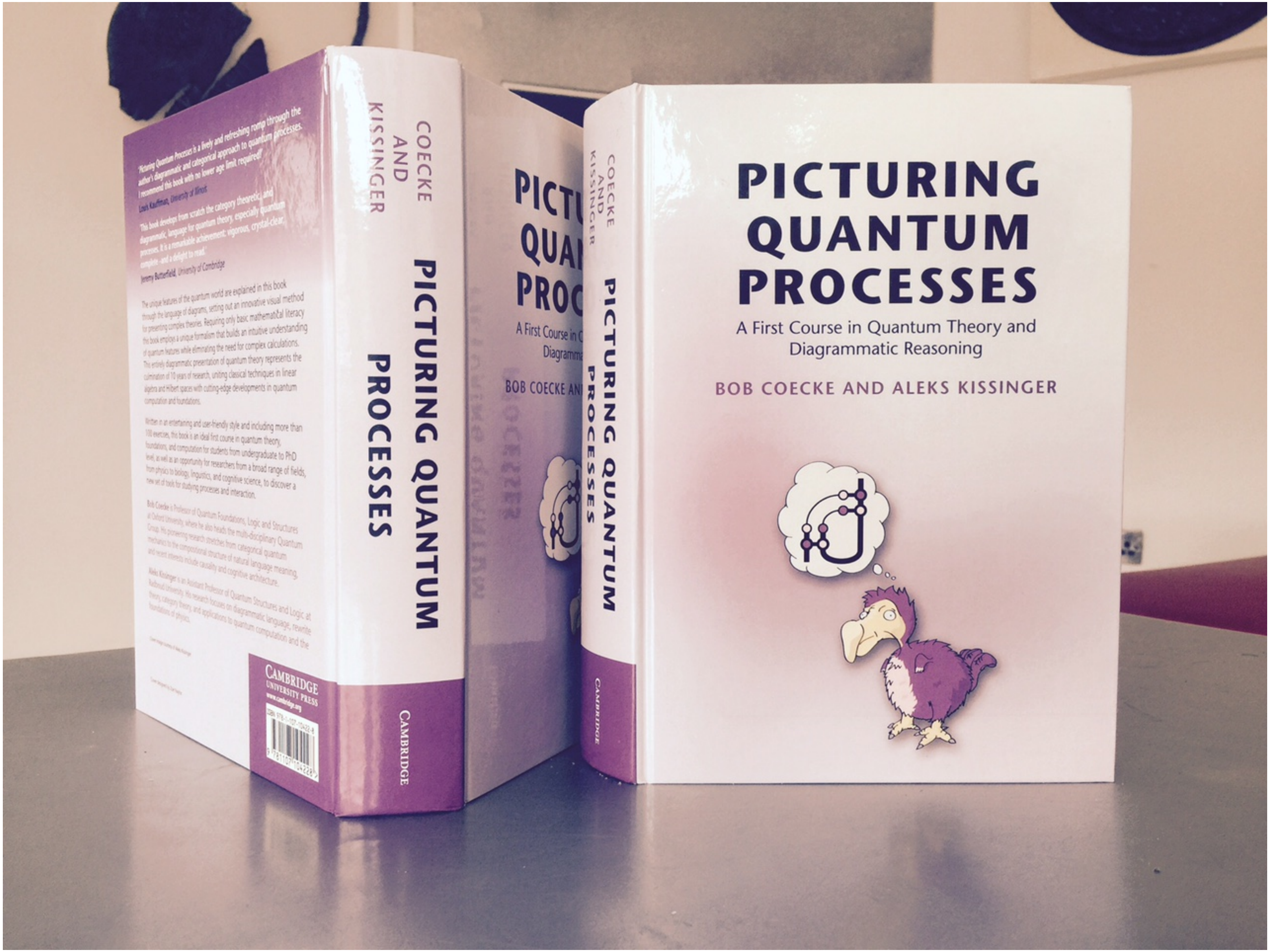
J. Bolt, B. Coecke, F. Genovese, M. Lewis, D. Marsden & R. Piedeleu (2017) *Interacting Conceptual Spaces I : Grammatical Composition of Concepts*. arXiv:1703.08314

Y. Al-Mehairi, B. Coecke & M. Lewis (2016) *Compositional Distributional Cognition*. QI'16.

Can QM be formulated in pictures ?

B. Coecke (2005) Kindergarten quantum mechanics. quant-ph/0510032

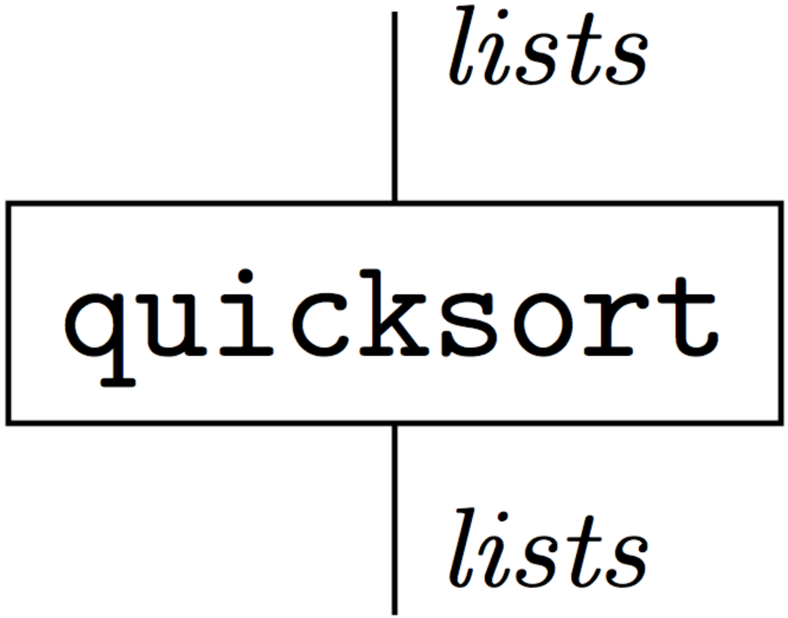
YES !


PICTURING QUANTUM PROCESSES
A First Course in Quantum Theory and Diagrammatic Reasoning
BOB COECKE AND ALEKS KISSINGER
COECKE AND KISSINGER
PICTURING QUANTUM PROCESSES
CAMBRIDGE
CAMBRIDGE UNIVERSITY PRESS

— Ch. 1 – Processes as diagrams —

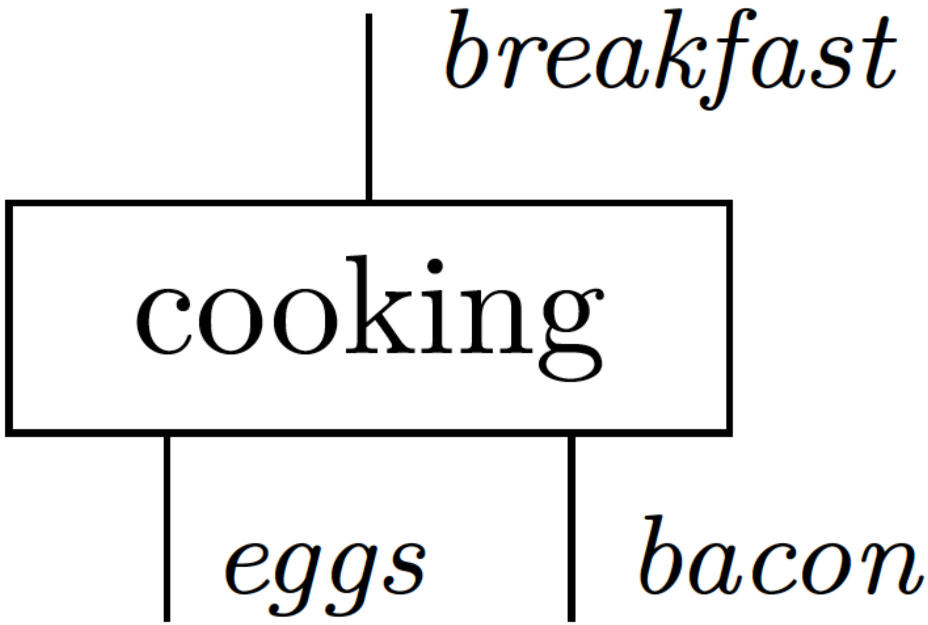
— Ch. 1 – Processes as diagrams —

– processes as boxes and systems as wires –



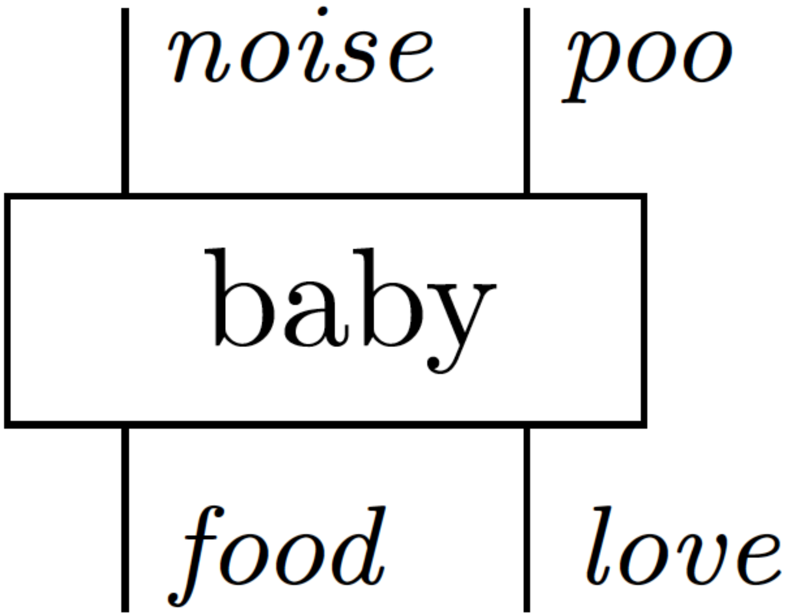
— Ch. 1 – Processes as diagrams —

– processes as boxes and systems as wires –



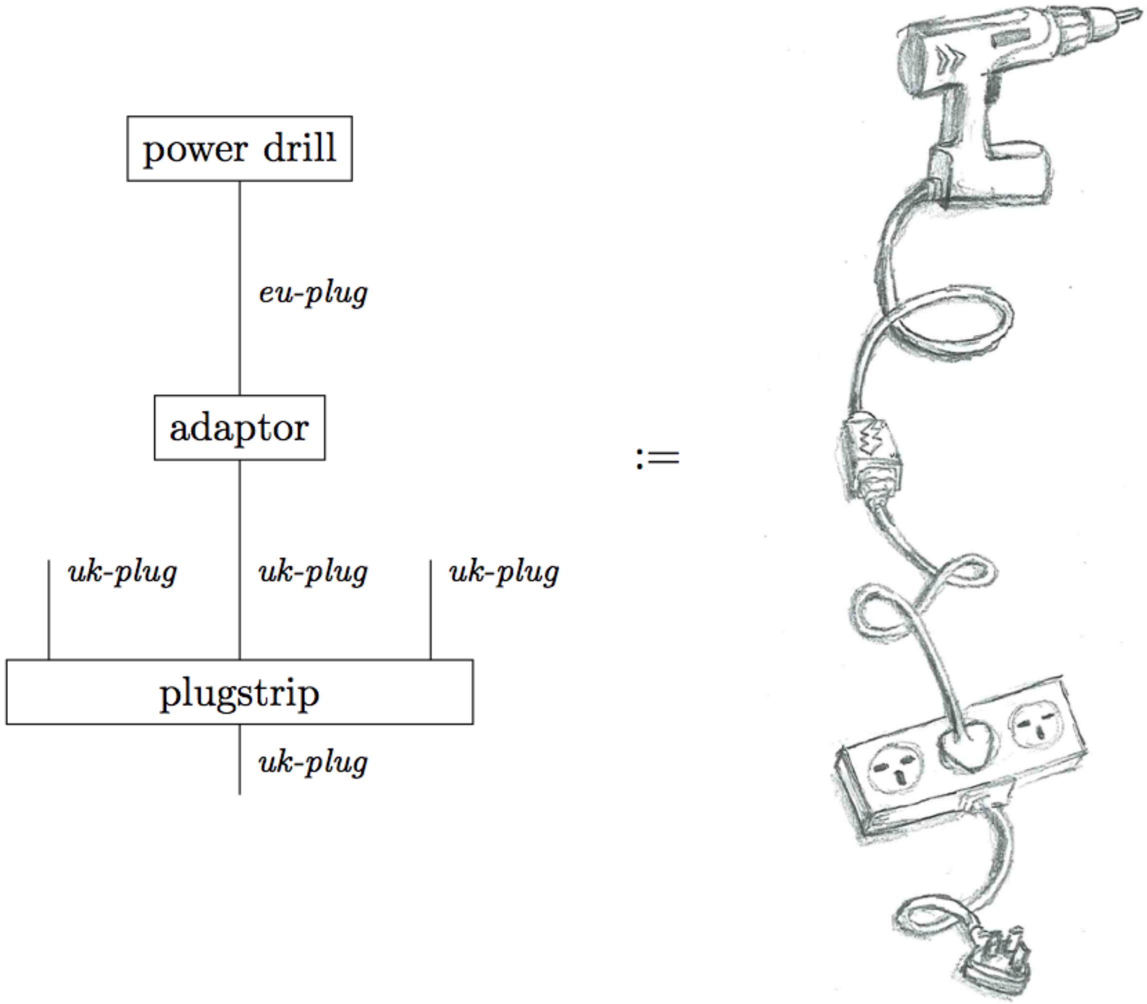
— Ch. 1 – Processes as diagrams —

– processes as boxes and systems as wires –



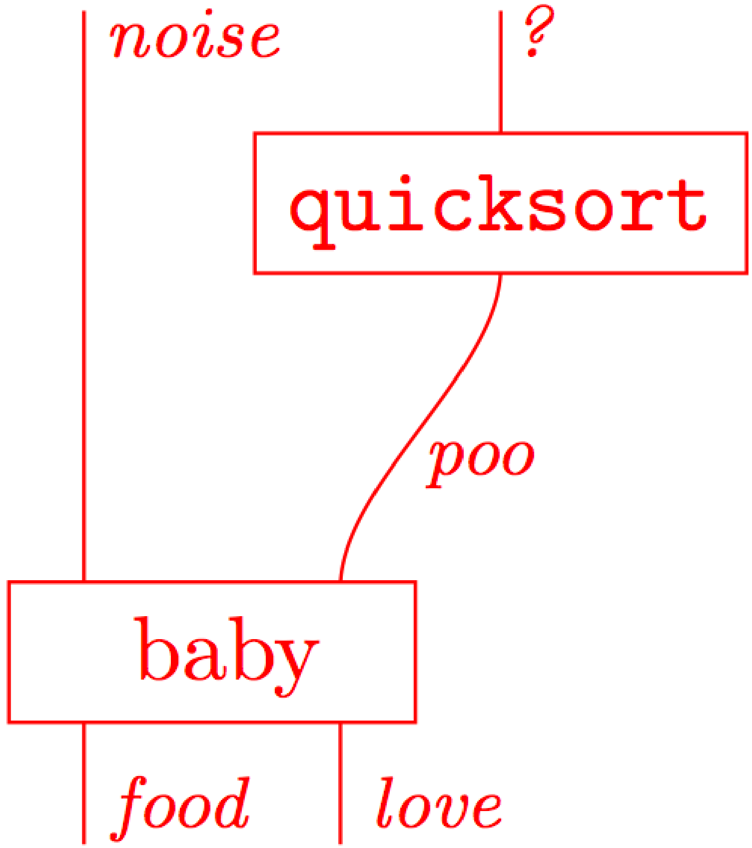
— Ch. 1 – Processes as diagrams —

– composing processes –



— Ch. 1 – Processes as diagrams —

– composing processes –

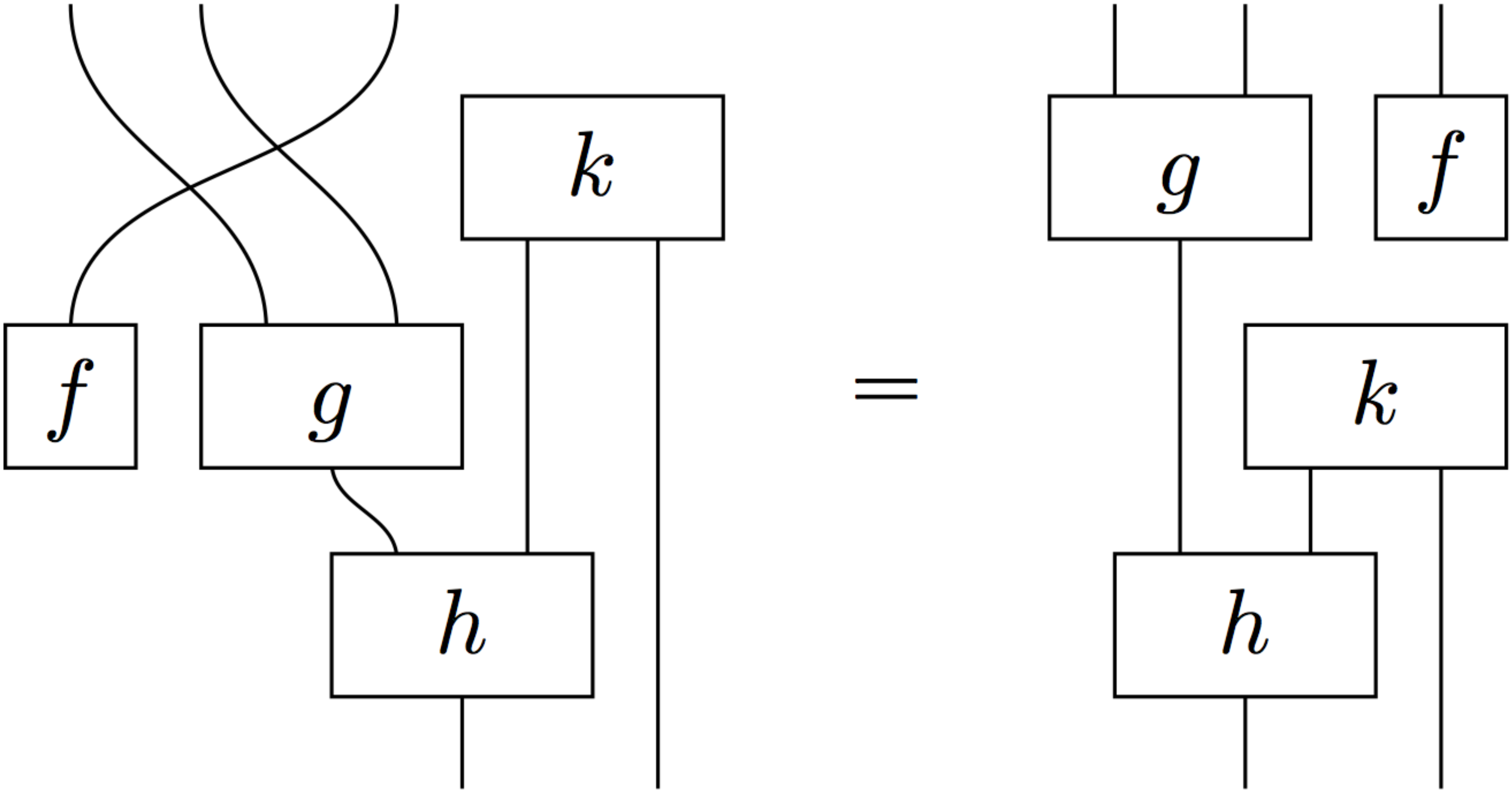


— Ch. 1 – Processes as diagrams —

– diagram equations –

— Ch. 1 – Processes as diagrams —

– diagram equations –



— Ch. 1 – Processes as diagrams —

– process theory –

— Ch. 1 – Processes as diagrams —

– process theory –

... consists of:

- **collection of systems**
- **collection of processes**
- **formalises 'wiring together'**

— Ch. 1 – Processes as diagrams —

– process theory –

... consists of:

- **collection of systems**
- **collection of processes**
- **formalises ‘wiring together’**

so in particular:

- **closed under forming diagrams.**

— Ch. 1 – Processes as diagrams —

– process theory –

... consists of:

- **collection of systems**
- **collection of processes**
- **formalises ‘wiring together’**

so in particular:

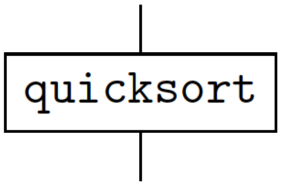
- **closed under forming diagrams.**

and it tells us:

- **when two diagrams are equal.**

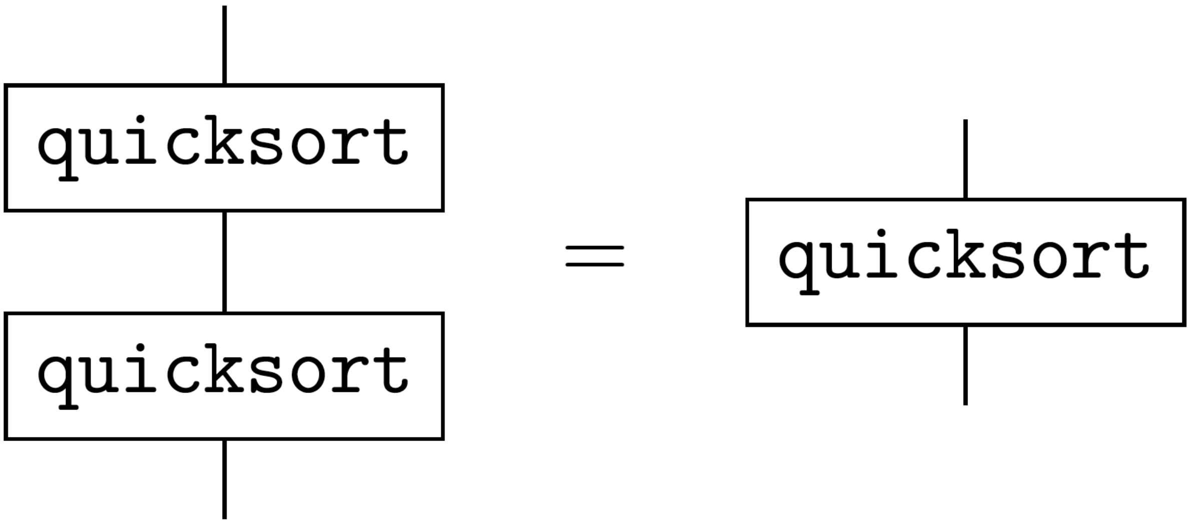
— Ch. 1 – Processes as diagrams —

– process theory –



:=

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



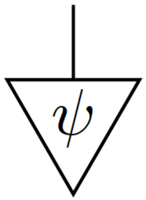
— Ch. 1 – Processes as diagrams —

– special processes/diagrams –

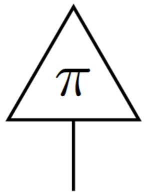
— Ch. 1 – Processes as diagrams —

– special processes/diagrams –

State :=



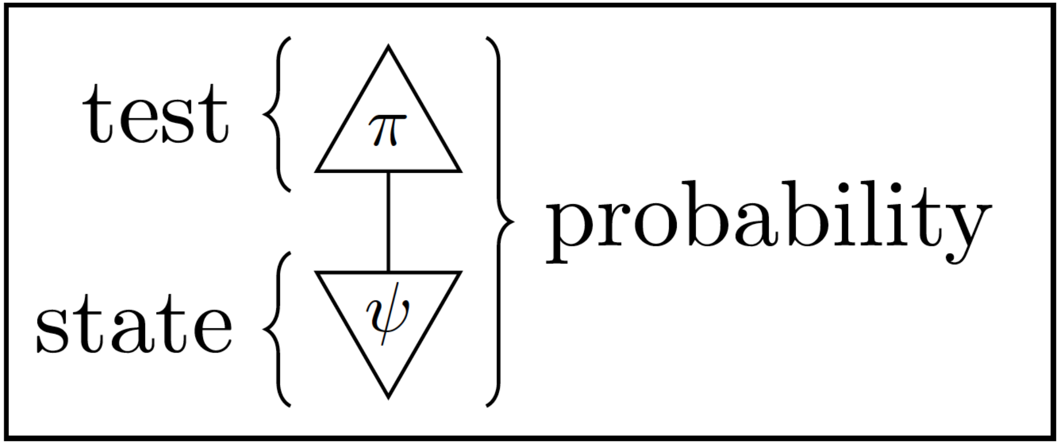
Effect / Test :=



Number :=

– special processes/diagrams –

Born rule :=



— 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.*

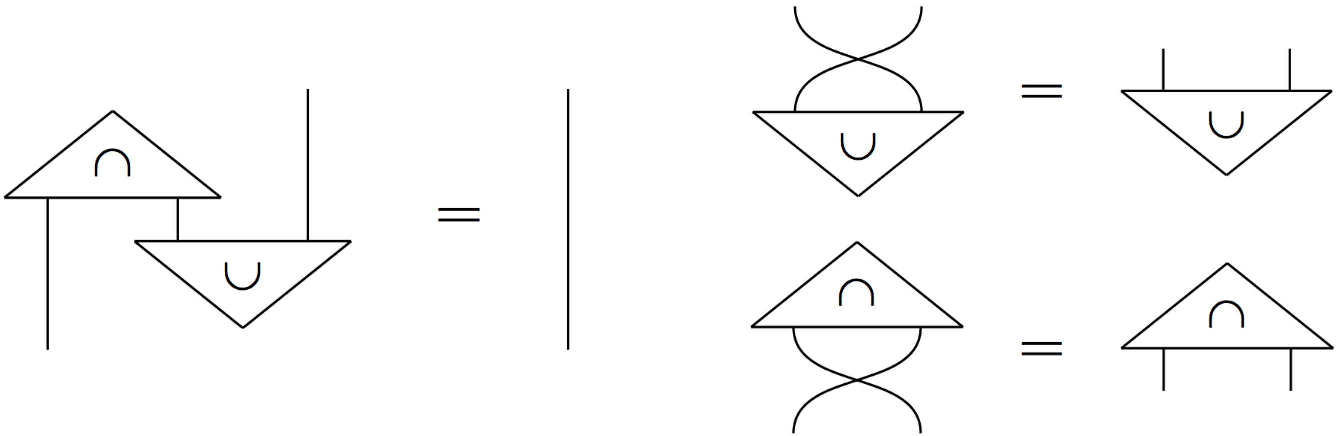
— Erwin Schrödinger, 1935.

— Ch. 2 – String diagrams —

– TFAE –

1. ‘Circuits’ with cup-state and cup-effect:

which satisfy:

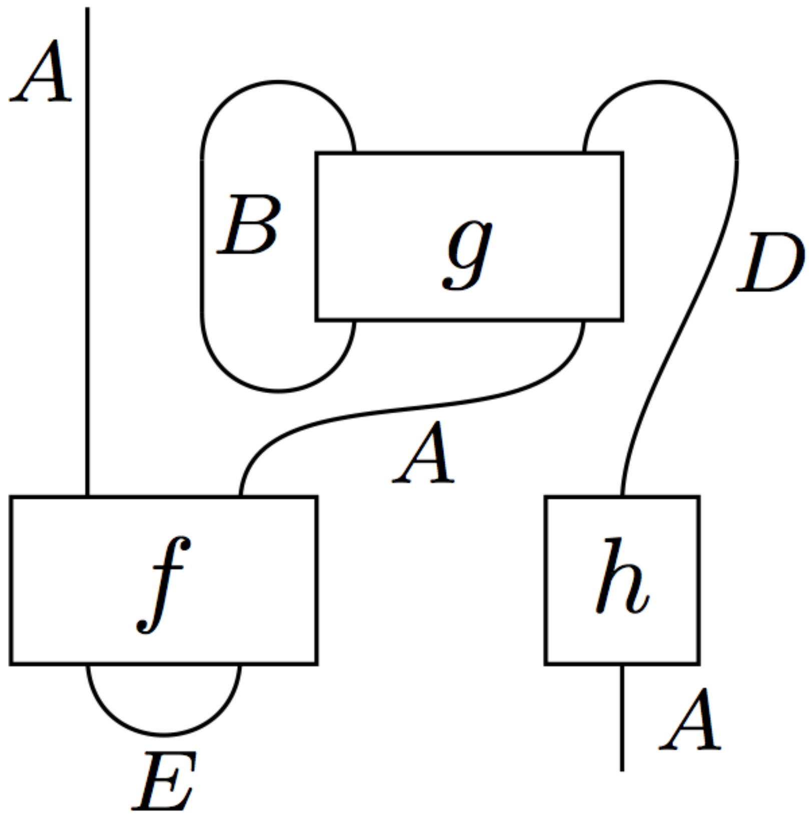


— Ch. 2 – String diagrams —

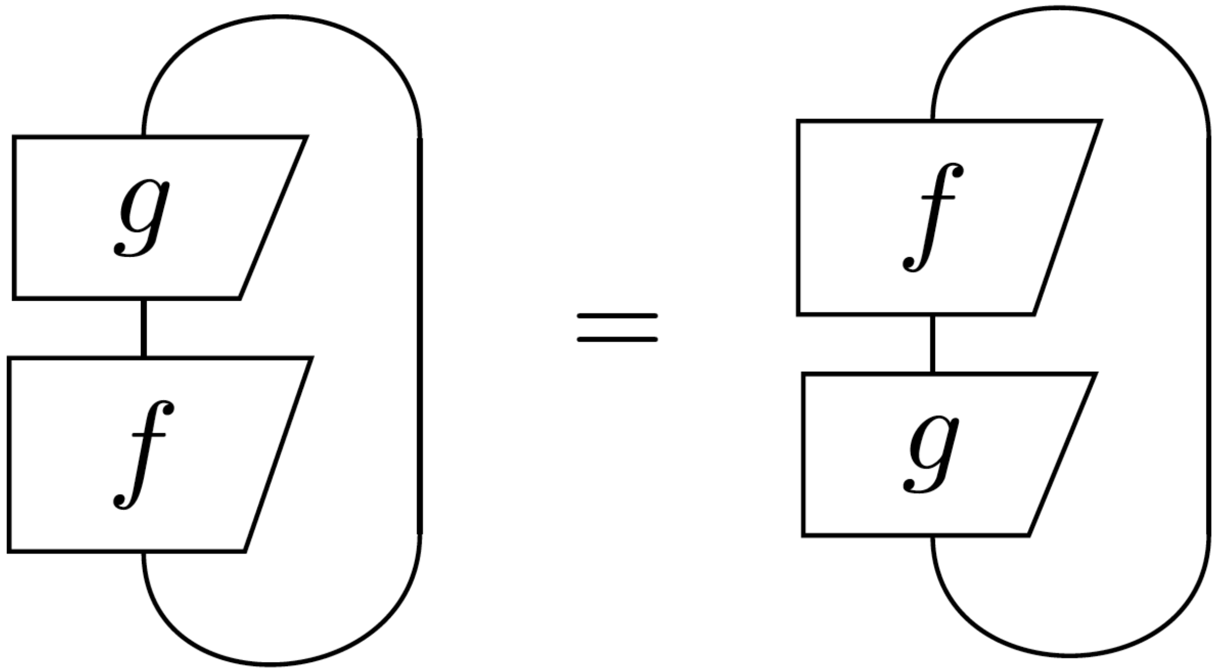
– TFAE –

– TFAE –

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

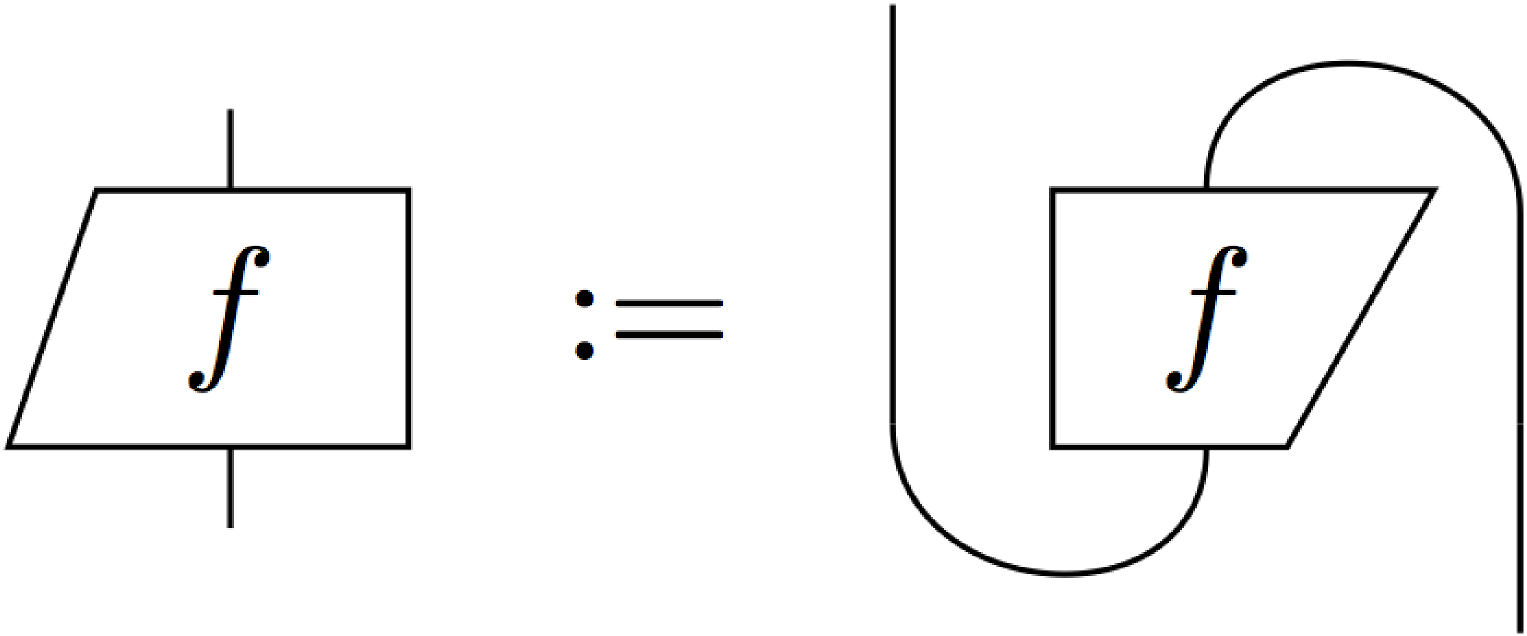


– cyclicity of the trace –



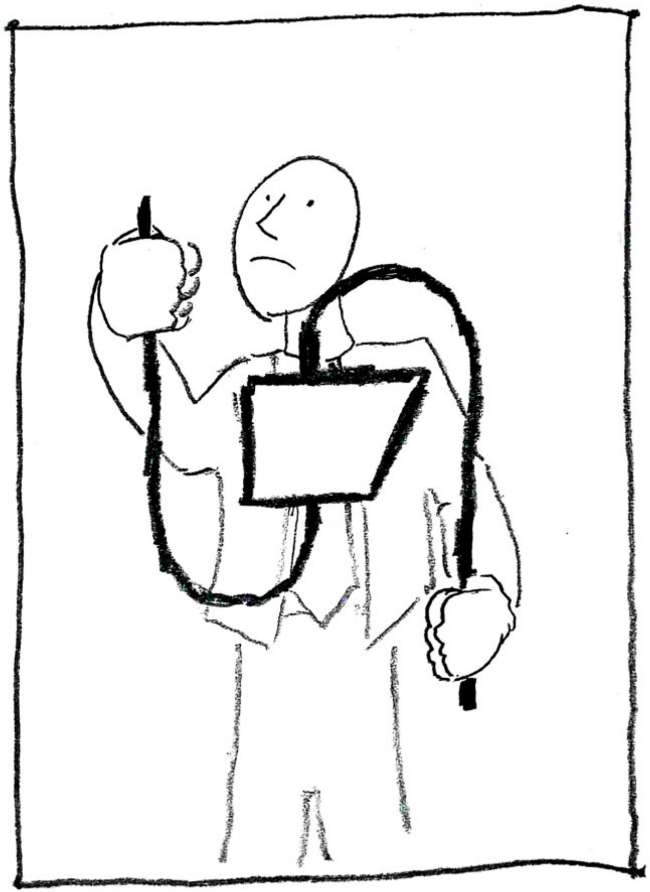
— Ch. 2 – String diagrams —

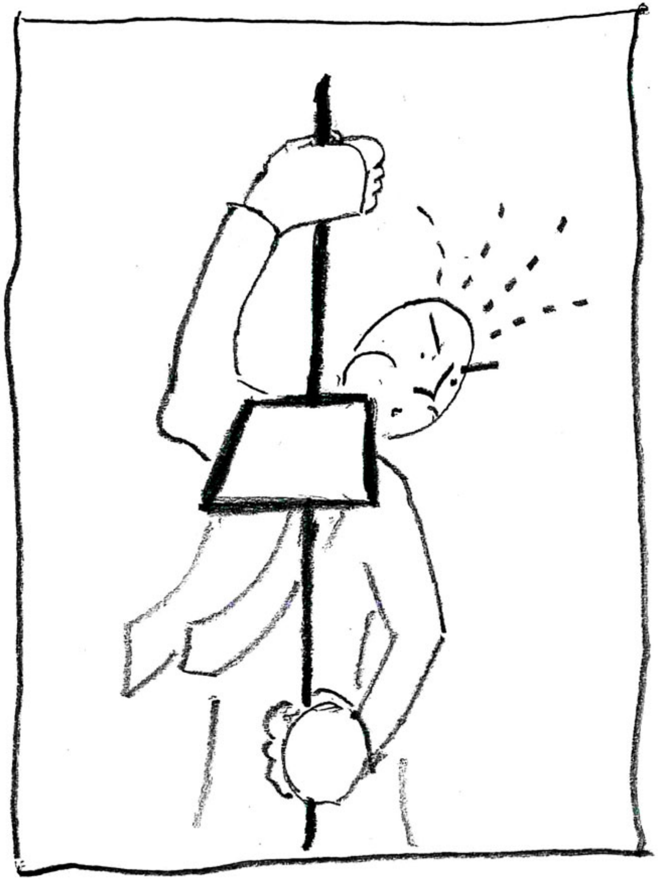
– *transpose* –



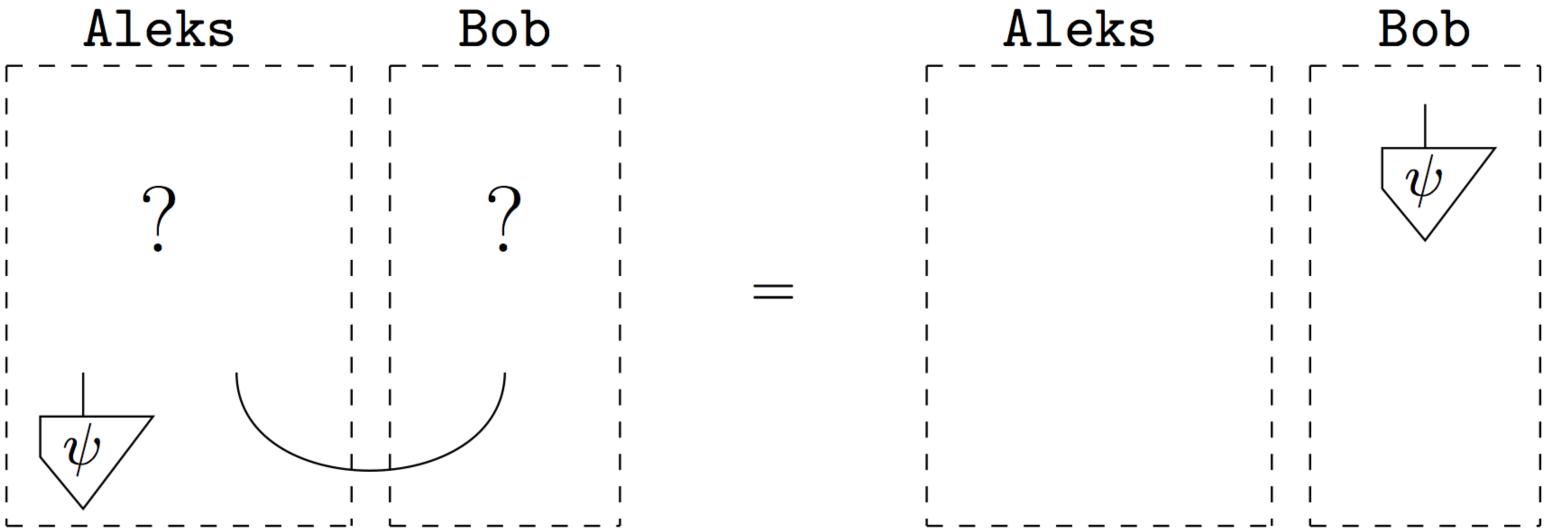
— Ch. 2 – String diagrams —

– transpose –



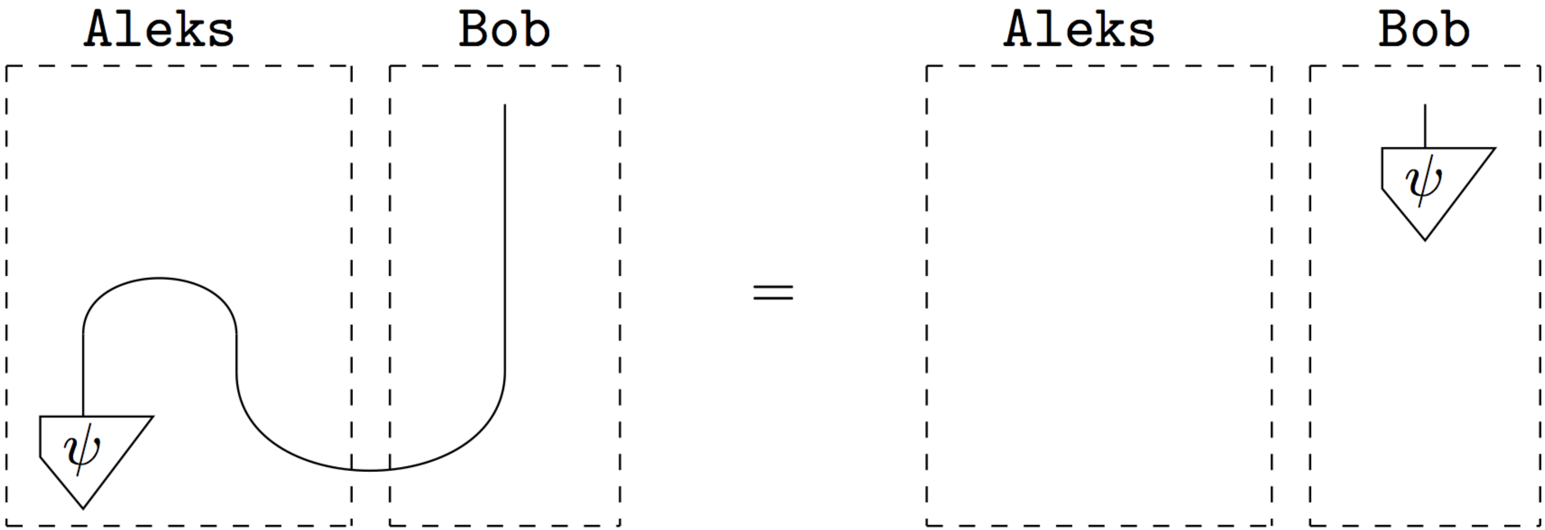


– quantum teleportation –



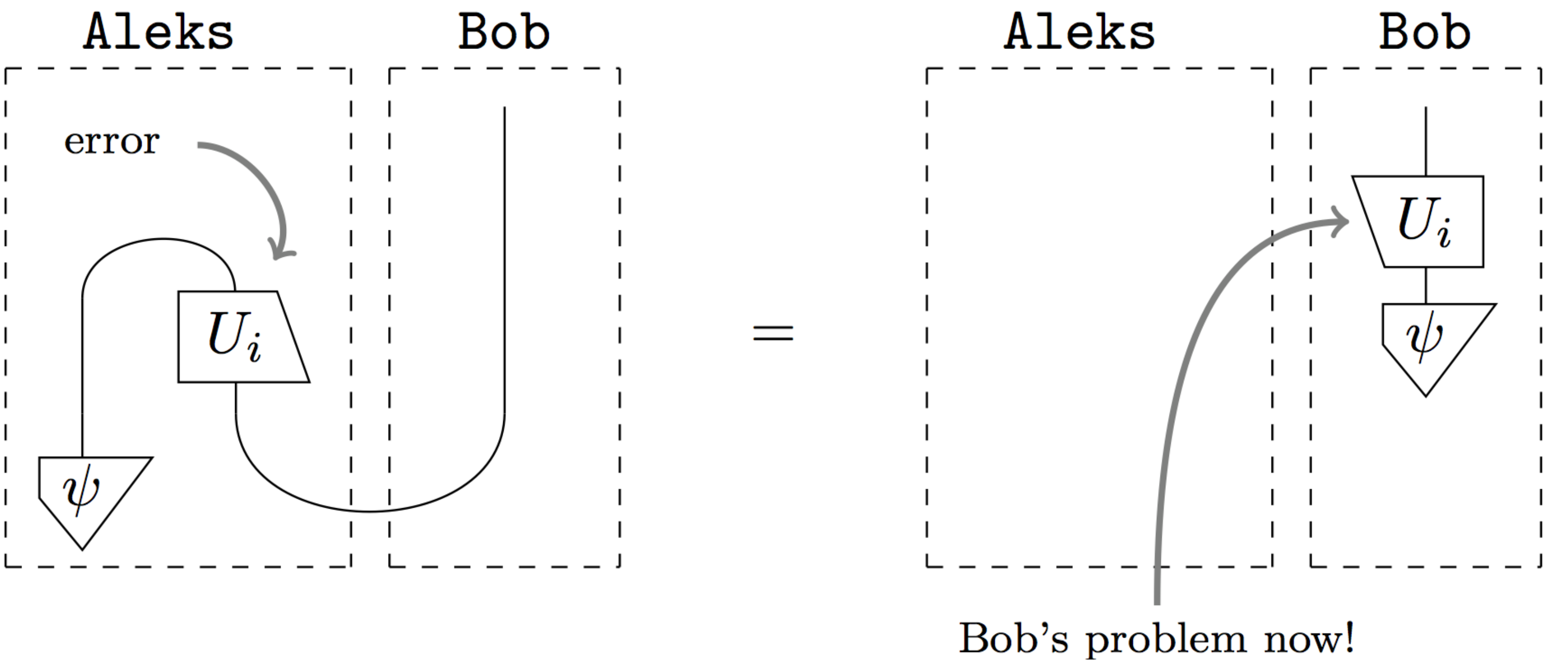
— Ch. 2 – String diagrams —

– quantum teleportation –



— Ch. 2 – String diagrams —

– quantum teleportation –

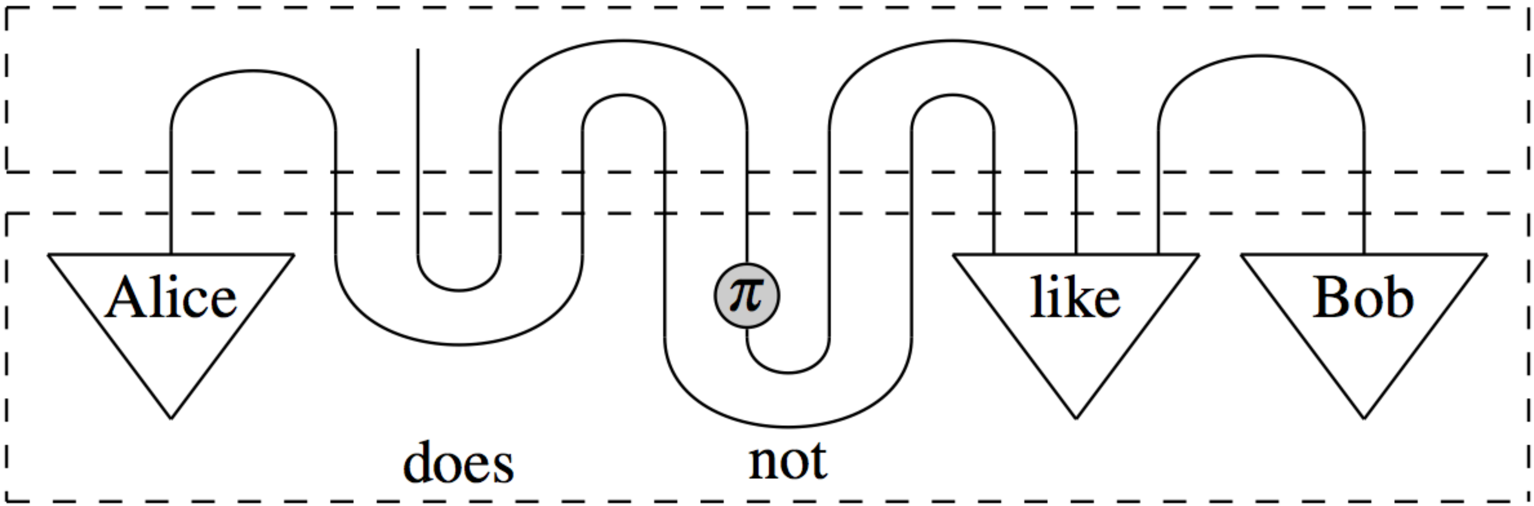


... what about natural language meaning?

... there are dictionaries for words

... why no dictionaries for sentences?

Computing the meaning of a sentence:



- Bottom part: meaning vectors
- Top part: grammar

B. Coecke, M. Sadrzadeh & S. Clark (2010) *Mathematical foundations for a compositional distributional model of meaning*. Lambek Festschrift. arXiv:1003.4394

Mathematics of grammar:

Lambek's Residuated monoids (1950's):

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

so in particular,

$$a \cdot (a \multimap 1) \leq 1 \leq a \multimap (a \cdot 1)$$

$$(1 \multimapinv b) \cdot b \leq 1 \leq (1 \cdot b) \multimapinv b$$

Lambek's Pregroups (2000's):

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

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

Mathematics of grammar:

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

Mathematics of grammar:

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

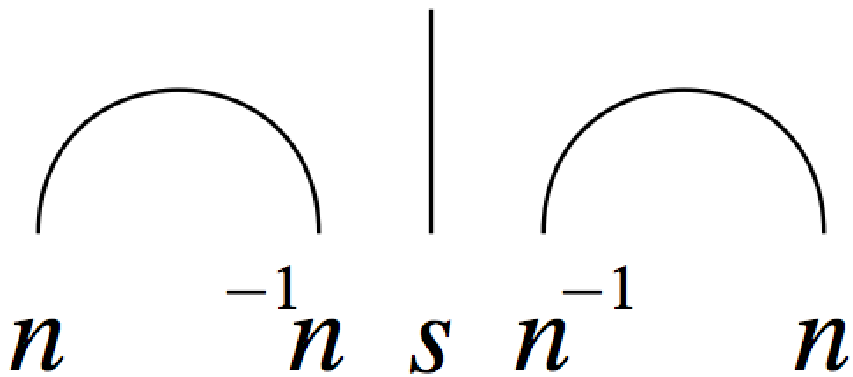
$$n \cdot {}^{-1}n \cdot s \cdot n^{-1} \cdot n \leq 1 \cdot s \cdot 1 \leq s$$

Mathematics of grammar:

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

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

As a diagram:

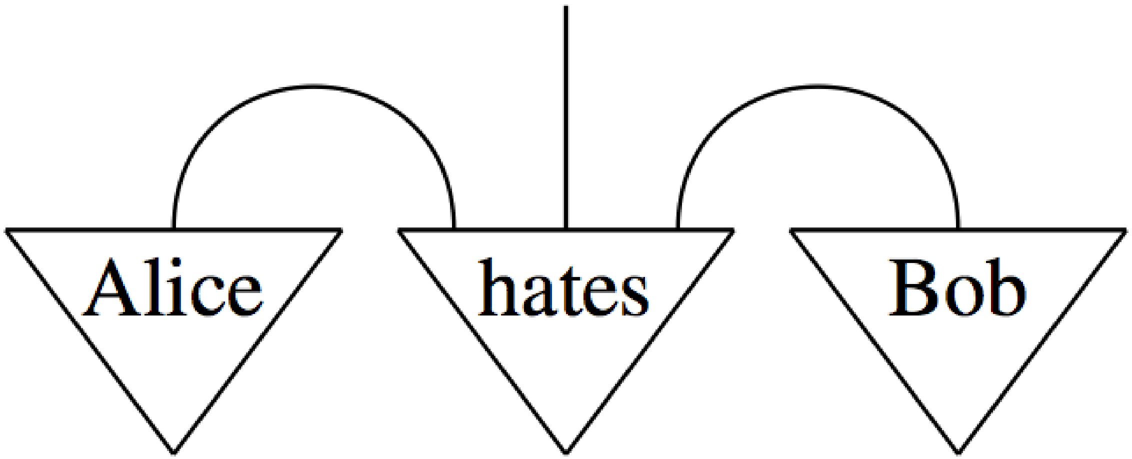


Mathematics of grammar:

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

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

As a diagram:



Algorithm for NLP-meaning composition:

1. Perform grammatical type reduction:

$$(word\ type\ 1)\ldots(word\ type\ n) \rightsquigarrow sentence\ type$$

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:

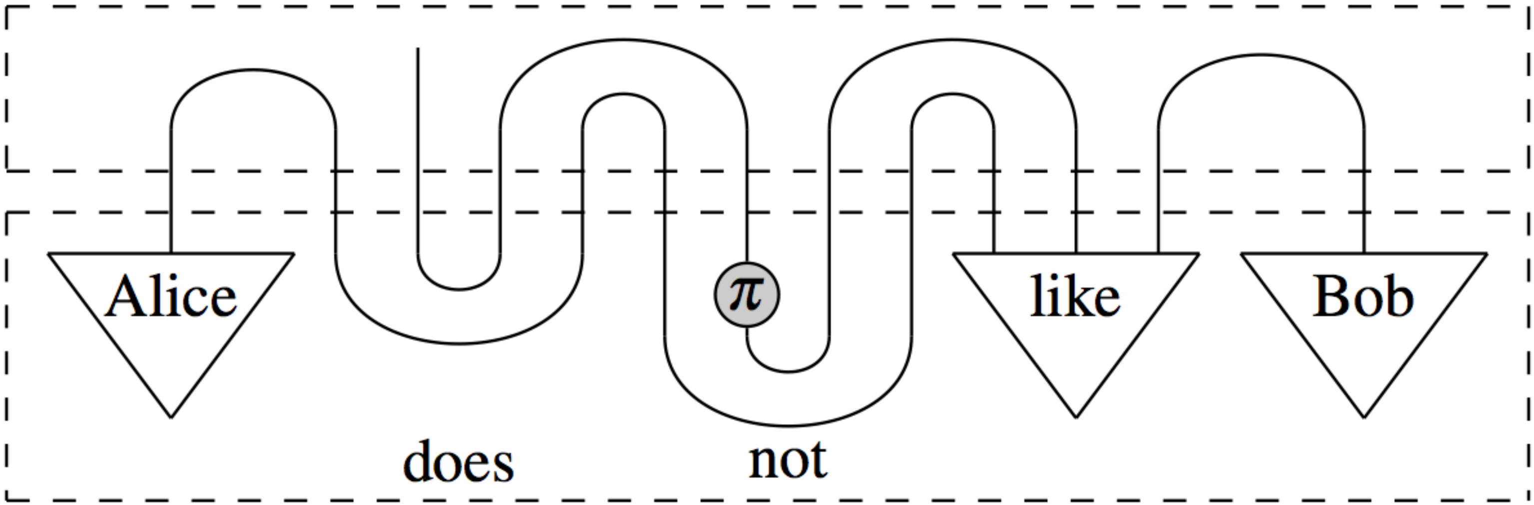
$$f\left(\overrightarrow{v}_1 \otimes \ldots \otimes \overrightarrow{v}_n\right)$$

Experimental evidence:

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	

D. Kartsaklis & M. Sadrzadeh (2013) *Prior disambiguation of word tensors for constructing sentence vectors*. In EMNLP'13.

Logical meanings:



- Bottom part: meaning vectors
- Top part: grammar

B. Coecke, M. Sadrzadeh & S. Clark (2010) *Mathematical foundations for a compositional distributional model of meaning*. Lambek Festschrift. arXiv:1003.4394

Algorithm for NLP-meaning composition:

1. Perform grammatical type reduction:

$$(word\ type\ 1)\ldots(word\ type\ n) \rightsquigarrow sentence\ type$$

2. Interpret diagrammatic type reduction as NLP-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 NLP-states:

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

Algorithm for XYZ-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 XYZ-map:

$f ::$  $\mapsto$ ‘cap’ $\otimes\, \mathrm{id} \otimes$ ‘cap’

3. Apply this map to tensor of word XYZ-states:

$$f\,(v_1 \otimes \ldots \otimes v_n)$$

Examples:

Examples:

1. Boolean matrices $\Rightarrow$ Montague

Examples:

1. Boolean matrices $\Rightarrow$ Montague

2. non-Boolean matrices $\Rightarrow$ logic dies

Examples:

1. Boolean matrices ⇒ Montague

2. non-Boolean matrices ⇒ logic dies

3. density matrices ⇒ ‘some’ logic re-emerges

Examples:

1. Boolean matrices ⇒ Montague

2. non-Boolean matrices ⇒ logic dies

3. density matrices ⇒ 'some' logic re-emerges

- ambiguity

- lexical entailment

R. Piedeleu, D. Kartsaklis, B. Coecke & M. Sadrzadeh (2015) *Open system categorical quantum semantics in natural language processing*. CalCo. arXiv:1502.00831

D. Bankova, B. Coecke, M. Lewis & D. Marsden (2016) *Graded entailment for compositional distributional semantics*. arXiv:1601.04908

... what about cognition?

Algorithm for XYZ-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 XYZ-map:

f ::  $\mapsto$ 'cap' $\otimes\, \mathrm{id} \otimes$ 'cap'

3. Apply this map to tensor of word meaning XYZ-states:

$$f\,(v_1 \otimes \ldots \otimes v_n)$$

Algorithm for **cog.-meaning** composition:

1. Perform grammatical type reduction:

$$(word\ type\ 1)\ldots(word\ type\ n) \;\rightsquigarrow\; sentence\ type$$

2. Interpret diagrammatic type reduction as cog.-map:

$f ::$

$\mapsto$ ‘cap’ $\otimes \mathrm{id} \otimes$ ‘cap’

3. Apply this map to tensor of word meaning cog.-states:

$$f\,(v_1 \otimes \ldots \otimes v_n)$$

General recipe:

General recipe:

1. Pick compositional mechanism **CM** (e.g. grammar)

General recipe:

1. Pick compositional mechanism **CM** (e.g. grammar)

2. Organise meaning/concept/cognitive spaces & maps in tensor-category $\otimes$-**Cat** that matches **CM**.

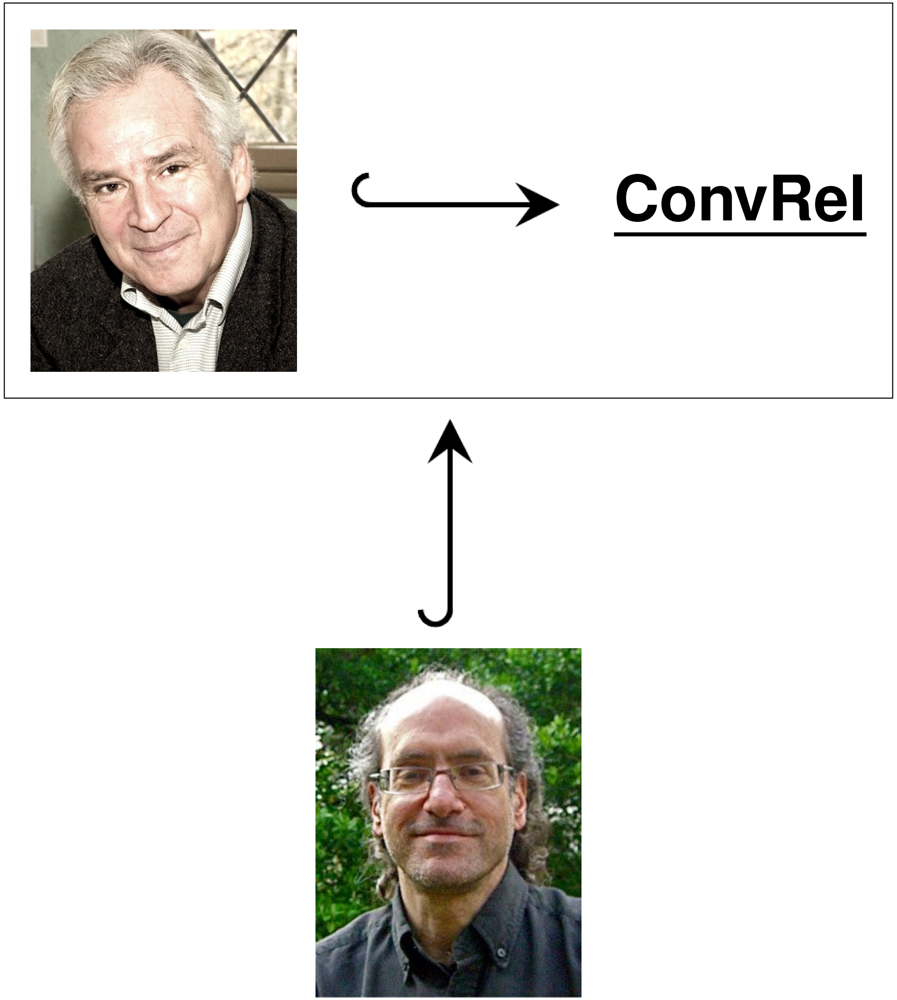
General recipe:

1. Pick compositional mechanism **CM** (e.g. grammar)

2. Organise meaning/concept/cognitive spaces & maps in tensor-category $\otimes$-**Cat** that matches **CM**.

3. Carry over compositional reasoning:

$$\mathbf{CM} \longrightarrow \otimes\text{-}\mathbf{Cat}$$



J. Bolt, B. Coecke, F. Genovese, M. Lewis, D. Marsden & R. Piedeleu (2017) *Interacting Conceptual Spaces I : Grammatical Composition of Concepts*. arXiv:1703.08314

Y. Al-Mehairi, B. Coecke & M. Lewis (2016) *Compositional Distributional Cognition*. QI'16.

A **convex algebra** is set A and 'mixing' function:

$$\alpha : D(A) \to A$$

i.e.:

$$\alpha(|a\rangle) = a$$

$$\alpha\left(\sum_{i,j} p_i q_{i,j} |a_{i,j}\rangle\right) = \alpha\left(\sum_i p_i |\alpha(\sum_j q_{i,j} |a_{i,j}\rangle)\rangle\right)$$

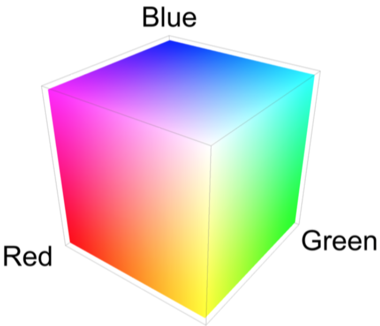
A **convex relation** of type $(A, \alpha) \to (B, \beta)$ is relation:

$$R : A \to B$$

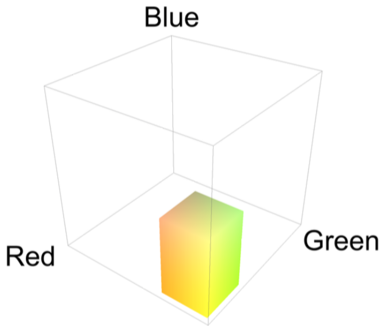
that 'commutes with mixtures':

$$(\forall i. R(a_i, b_i)) \Rightarrow R\left(\sum_i p_i a_i, \sum_i p_i b_i\right)$$

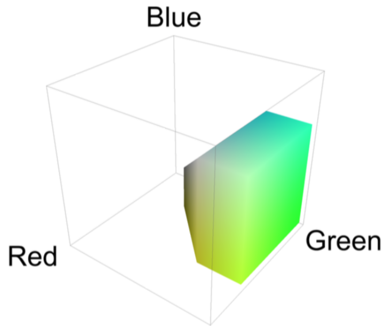
$$N_{food} = N_{colour} \otimes N_{taste} \otimes N_{texture}$$



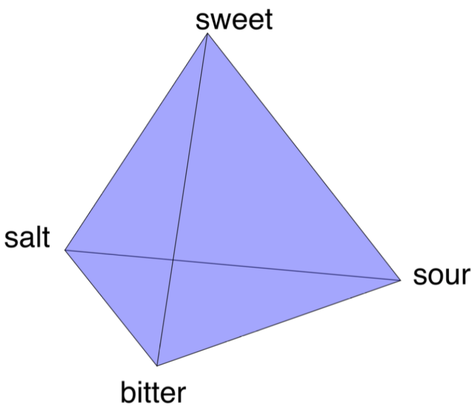
(a) The RGB colour cube



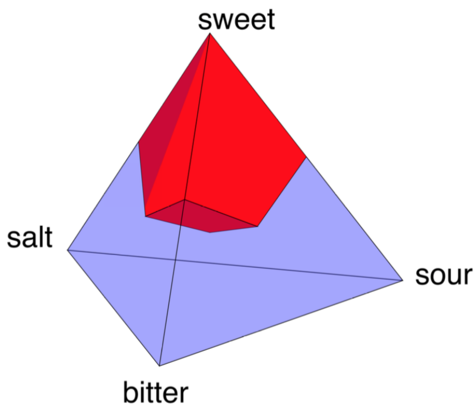
(b) Property p_{yellow}



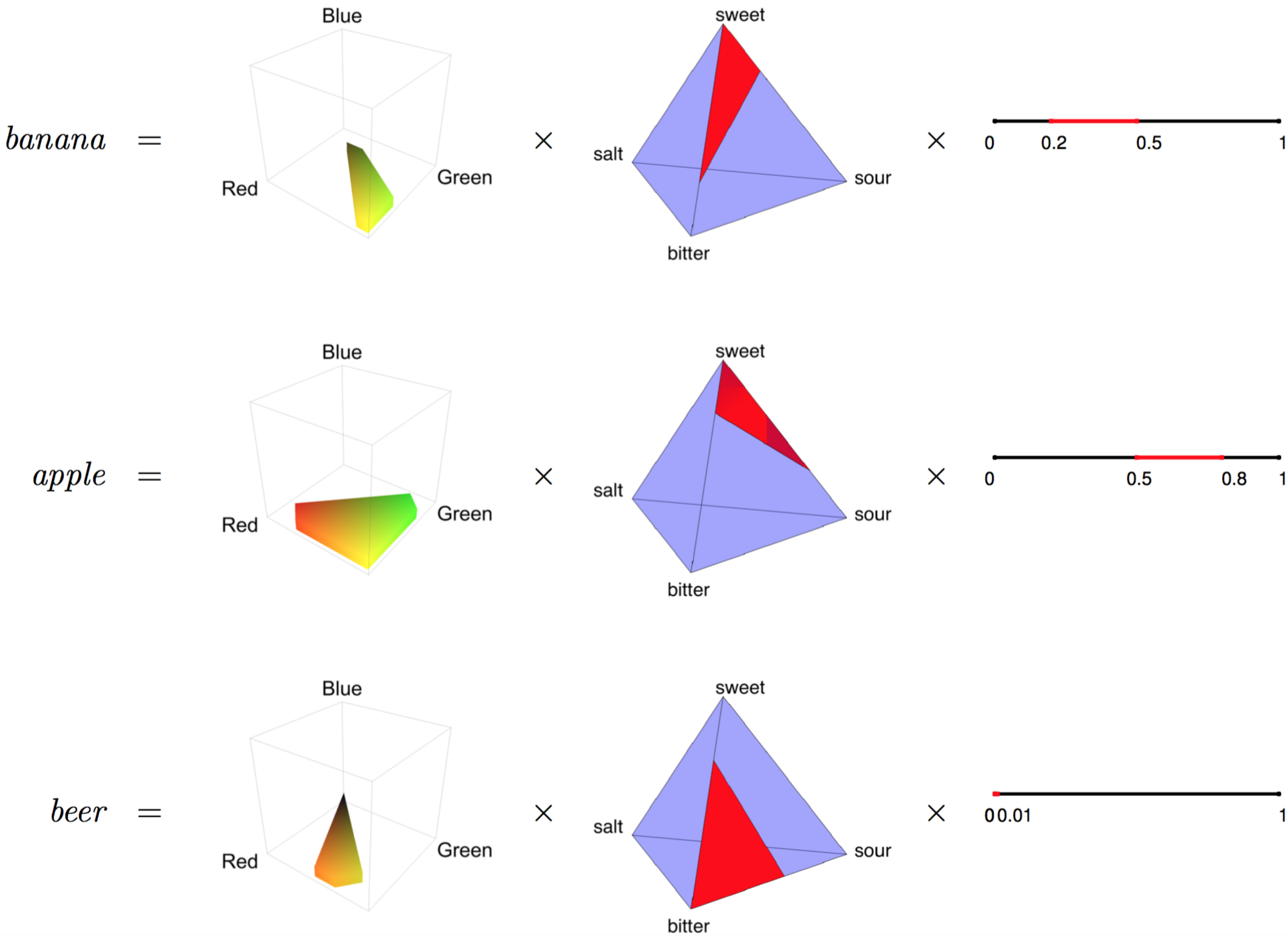
(c) Property p_{green}



(a) The taste tetrahedron



(b) The property p_{sweet}



banana =
Blue
Red
Green
×
sweet
salt
sour
bitter
×
0
0.2
0.5
1
apple =
Blue
Red
Green
×
sweet
salt
sour
bitter
×
0
0.5
0.8
1
beer =
Blue
Red
Green
×
sweet
salt
sour
bitter
×
0 0.01
1

— Ch. 4 – Quantum processes —

– quantum vs. classical –

— Ch. 4 – Quantum processes —

– quantum vs. classical –

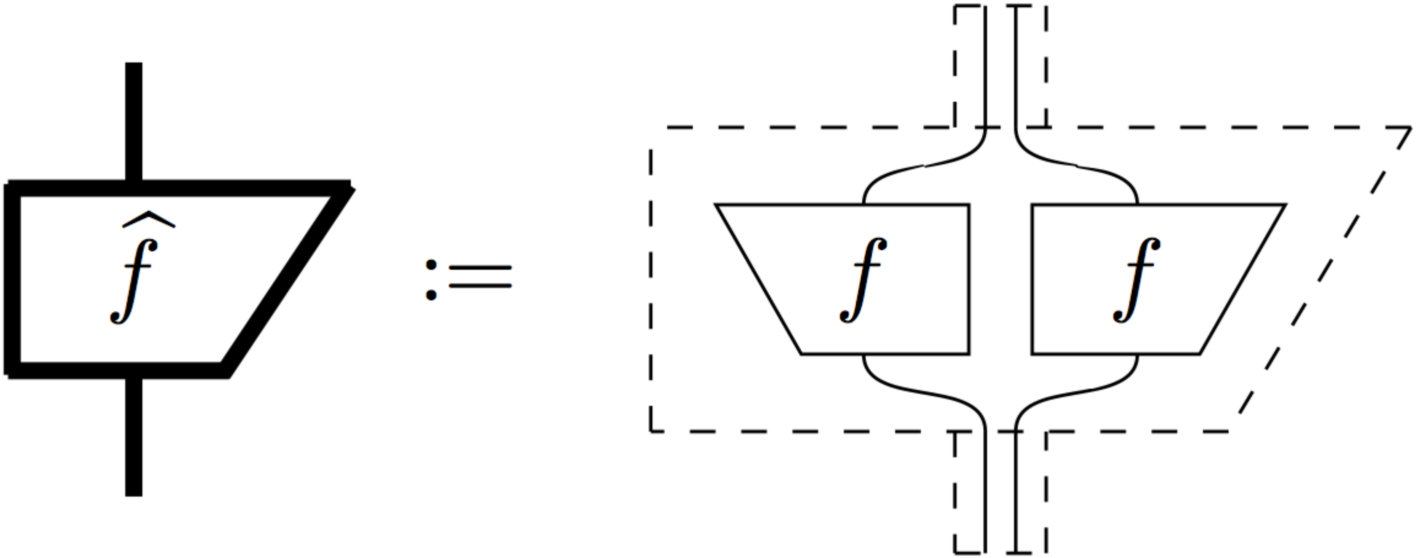
Main idea:

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

— Ch. 4 – Quantum processes —

– pure quantum box –

... :=



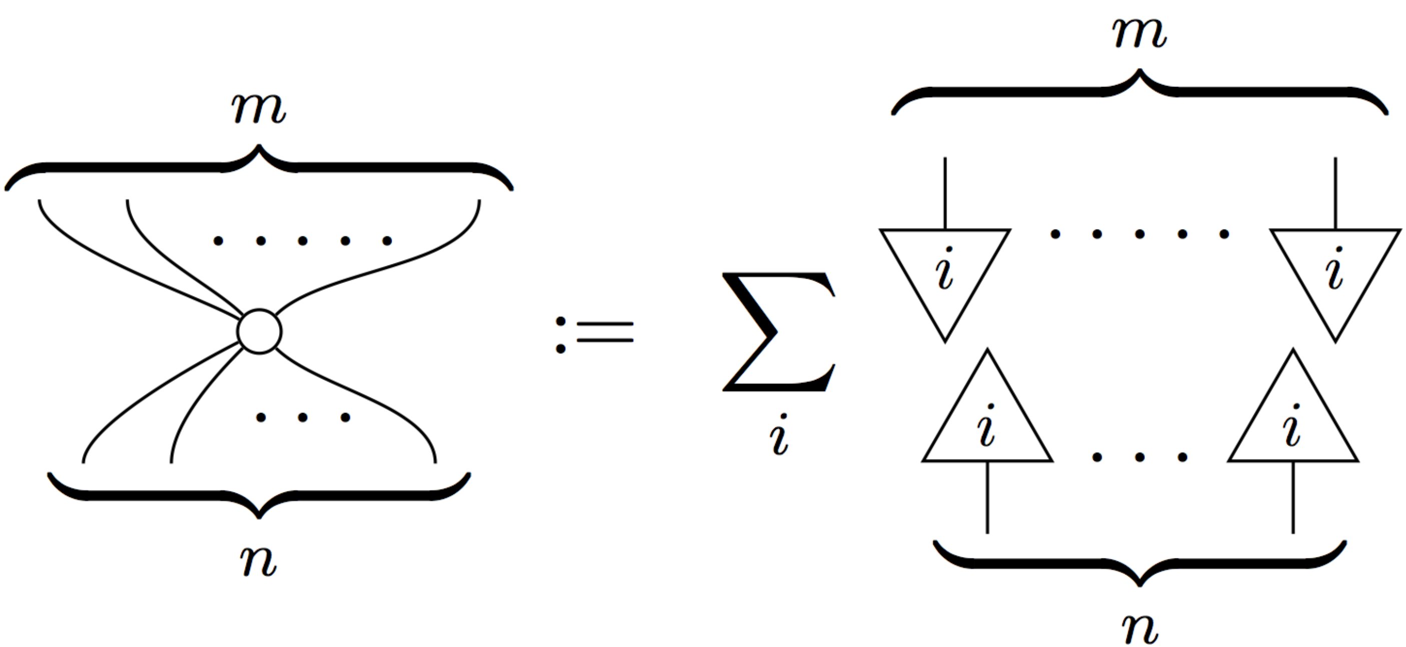
— Ch. 6 – Picturing classical processes —

– classical data diagrammatically –

— Ch. 6 – Picturing classical processes —

– classical data diagrammatically –

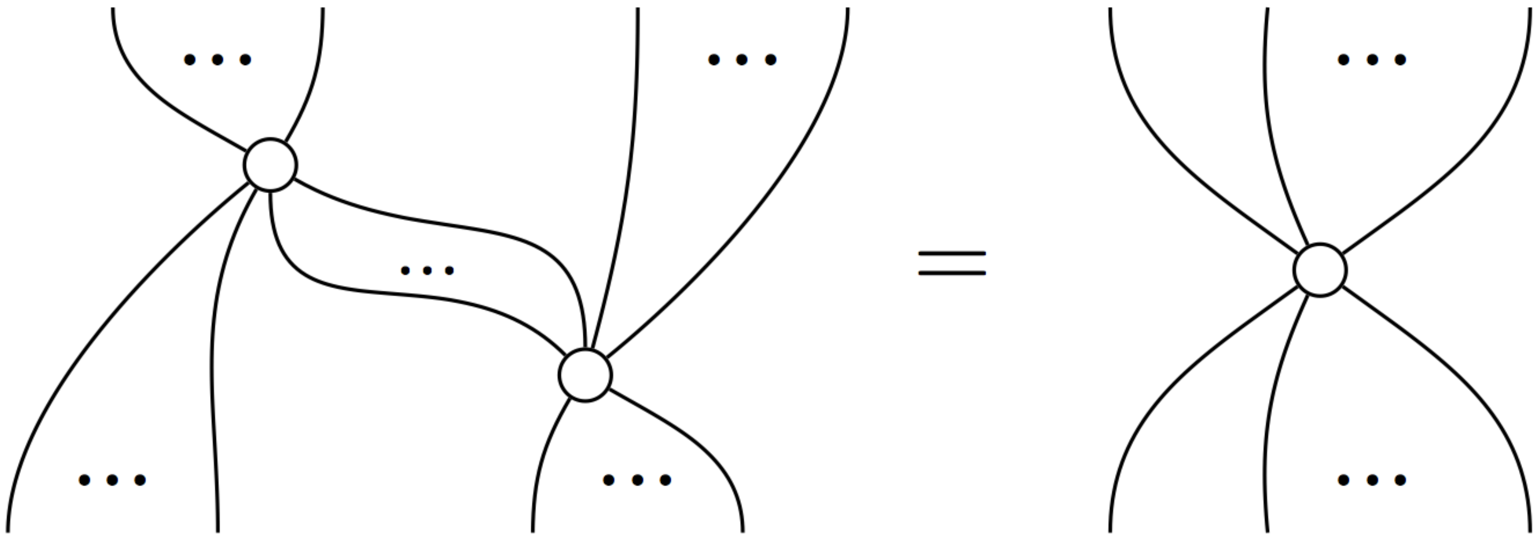
spider :=



— Ch. 6 – Picturing classical processes —

– classical data diagrammatically –

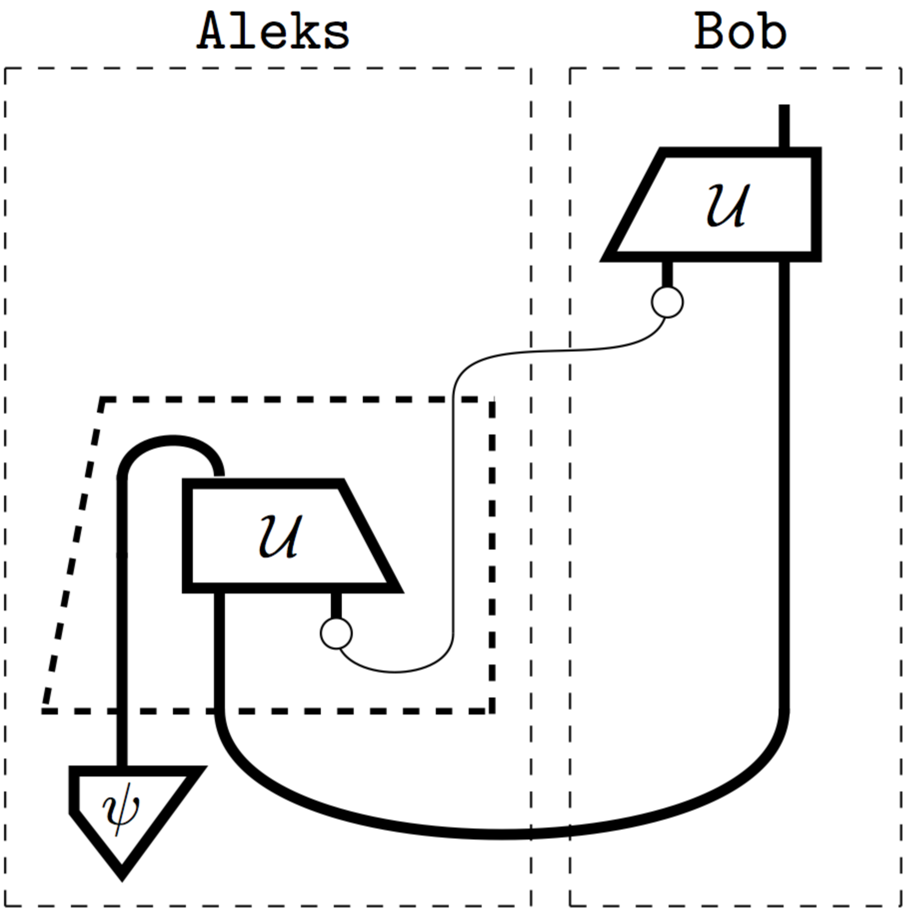
Prop. $\Longrightarrow$



(≡ dagger special commutative Frobenius algebra)

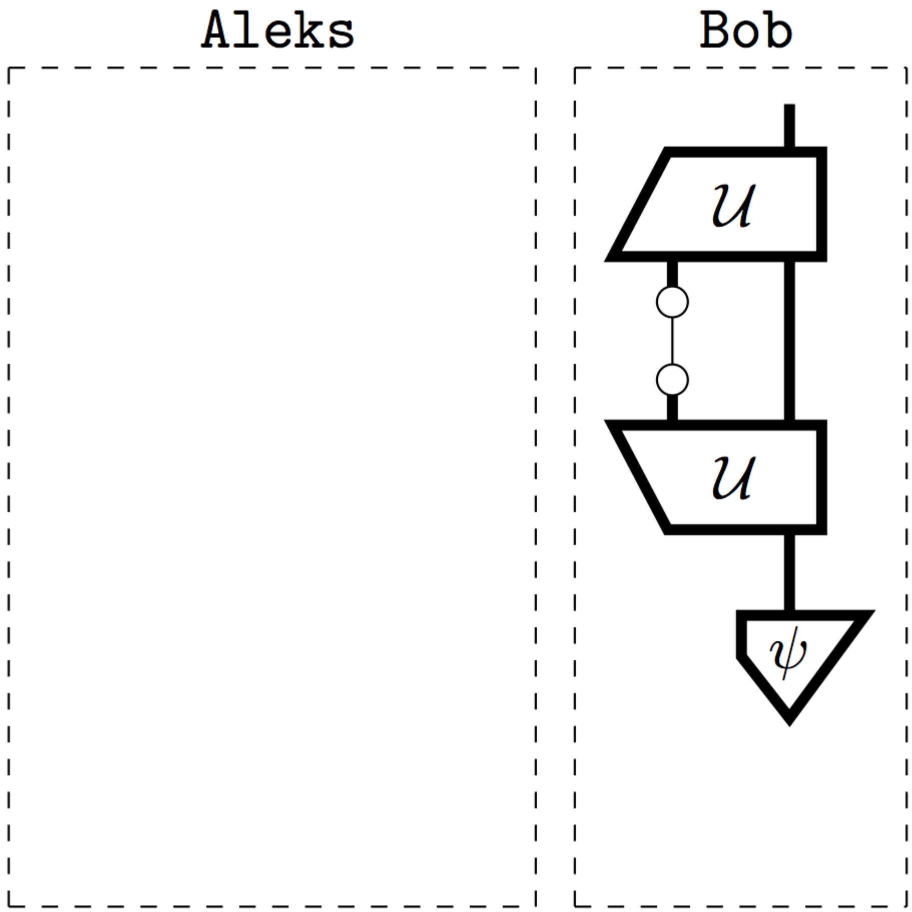
— Ch. 6 – Picturing classical processes —

– teleportation diagrammatically –



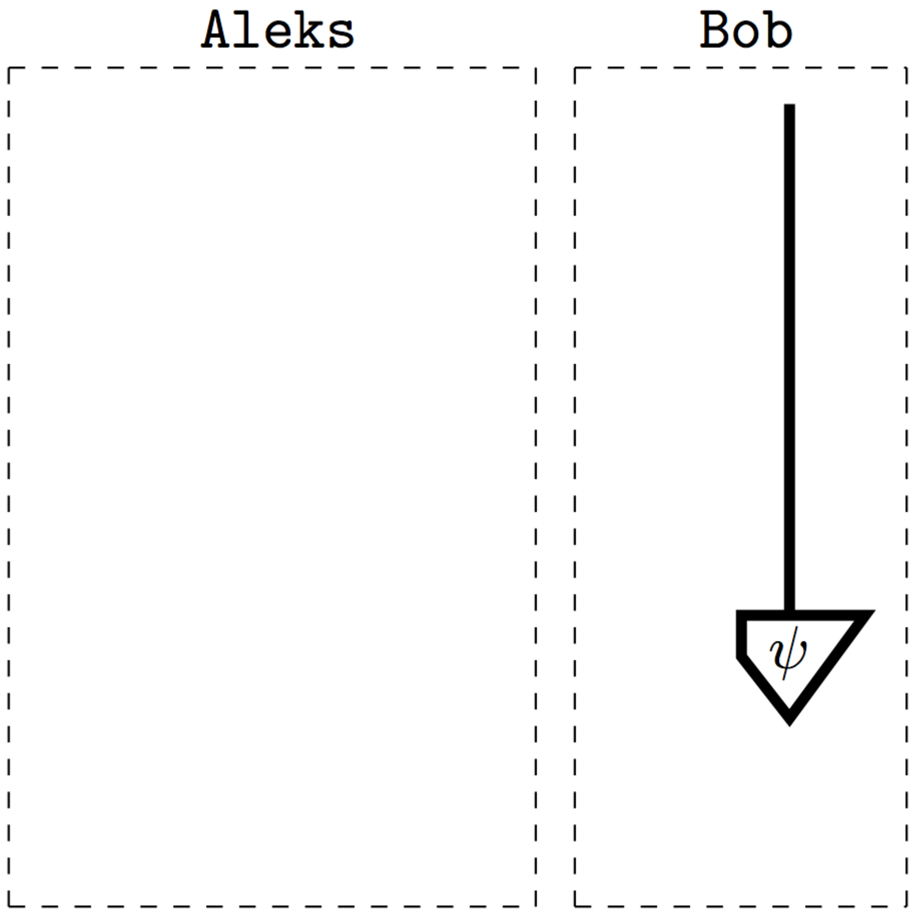
— Ch. 6 – Picturing classical processes —

– teleportation diagrammatically –



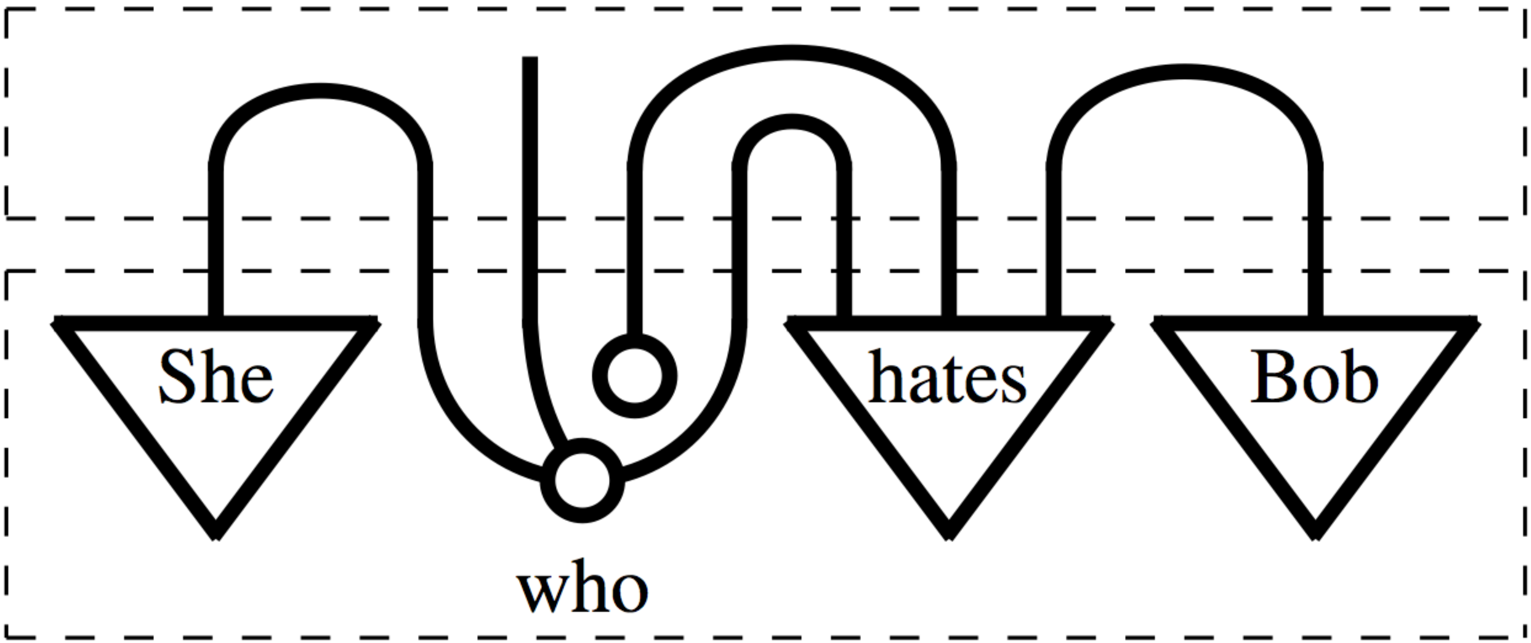
— Ch. 6 – Picturing classical processes —

– teleportation diagrammatically –



... what about language meaning?

Relative pronouns:



M. Sadrzadeh, B. Coecke & S. Clark (2013–2014) *The Frobenius anatomy of word meaning I & II*. Journal of Logic and Computation. arXiv:1404.5278

$$\rho_{she} := \sum \begin{cases} |Alice\rangle\langle Alice| \\ |Beth\rangle\langle Beth| \\ \ldots \end{cases}$$

$$\rho_{hates} := \sum \begin{cases} |Alice\rangle\langle Alice| \otimes \rho' \otimes |Bob\rangle\langle Bob| \\ |Beth\rangle\langle Beth| \otimes \rho'' \otimes |Colin\rangle\langle Colin| \\ \ldots \end{cases}$$

$$\rho_{Bob} := |Bob\rangle\langle Bob|$$

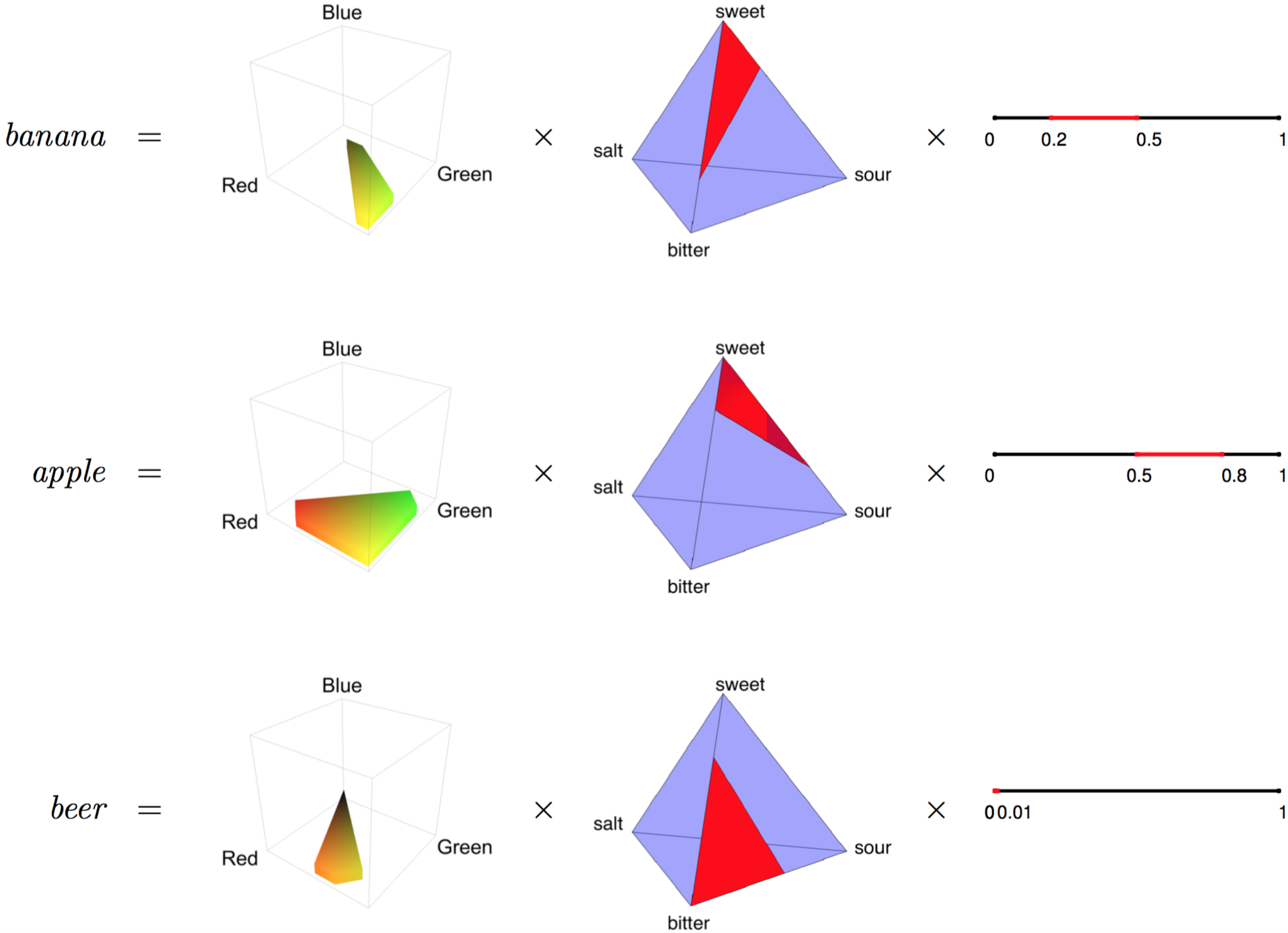
$$\rho_{she} := \sum \begin{cases} |Alice\rangle\langle Alice| \\ |Beth\rangle\langle Beth| \\ \ldots \end{cases}$$

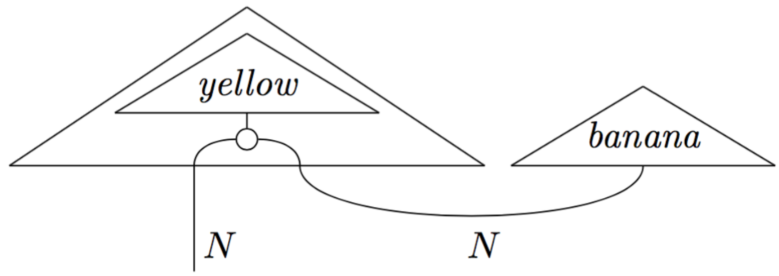
$$\rho_{hates} := \sum \begin{cases} |Alice\rangle\langle Alice| \otimes \rho' \otimes |Bob\rangle\langle Bob| \\ |Beth\rangle\langle Beth| \otimes \rho'' \otimes |Colin\rangle\langle Colin| \\ \ldots \end{cases}$$

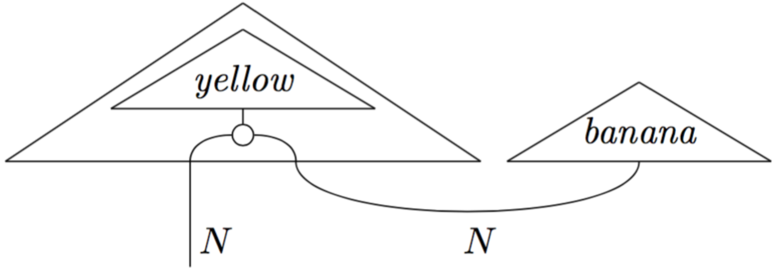
$$\rho_{Bob} := |Bob\rangle\langle Bob|$$

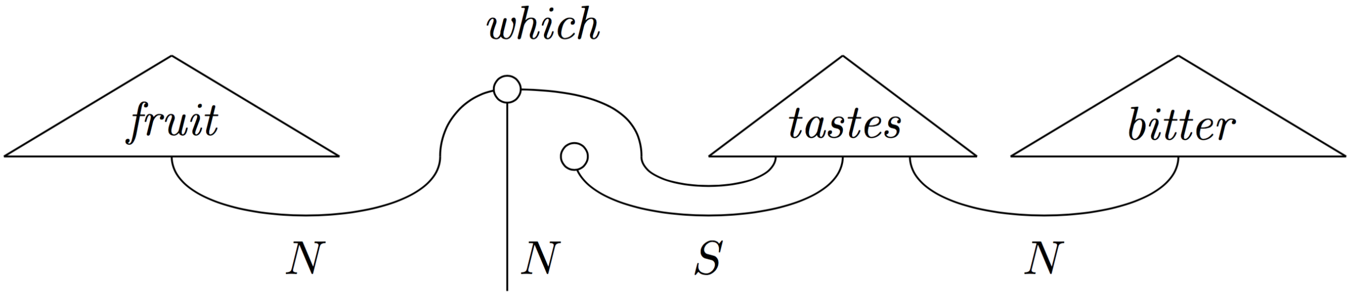
$$\rho_{sentence} := |Alice\rangle\langle Alice|$$

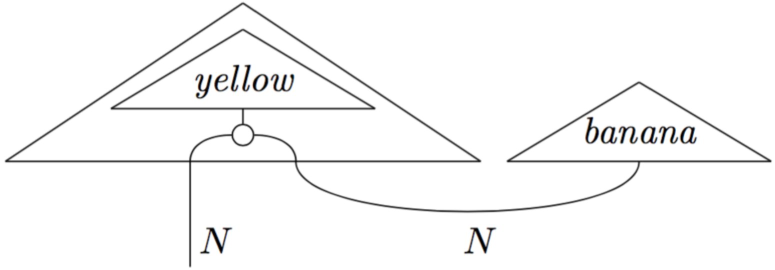
... what about cognition?

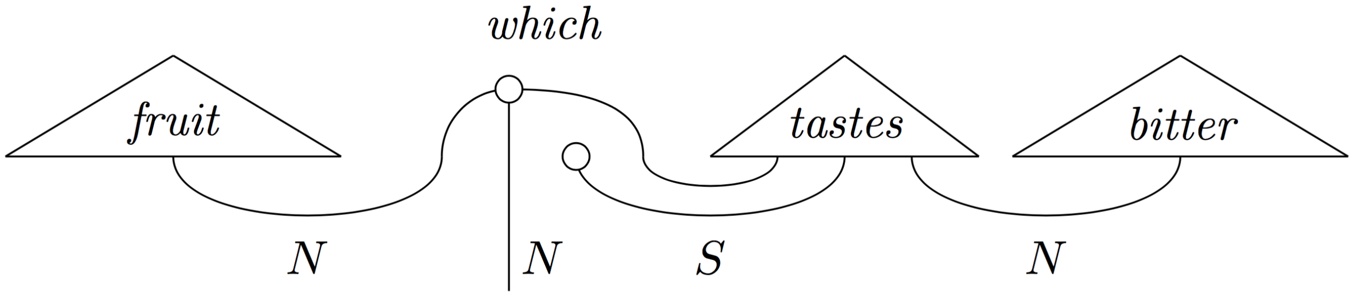

banana =
Blue
Red
Green
×
sweet
salt
sour
bitter
×
0
0.2
0.5
1
apple =
Blue
Red
Green
×
sweet
salt
sour
bitter
×
0
0.5
0.8
1
beer =
Blue
Red
Green
×
sweet
salt
sour
bitter
×
0 0.01
1


yellow
banana
N
N


yellow
banana
N
N


which
fruit
tastes
bitter
N
N
S
N





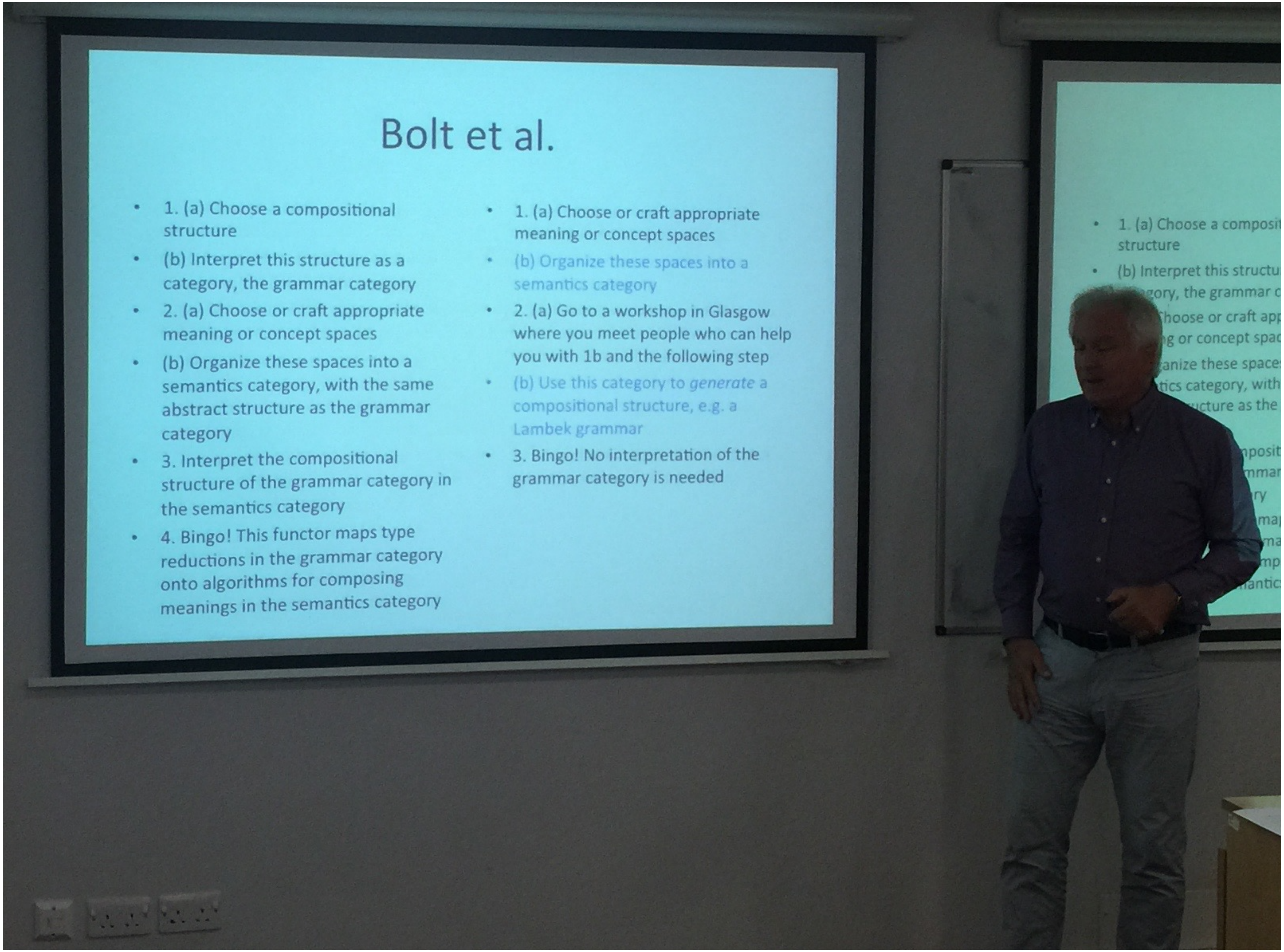
Fruit which tastes bitter

$= (\mu_N \times \iota_S \times \epsilon_N)(Conv(\textit{bananas} \cup \textit{apples}) \times \textit{taste} \times \textit{bitter})$

$= (\mu_N \times \iota_S)(Conv(\textit{bananas} \cup \textit{apples}) \times (\textit{green banana} \times \{(0,0)\}))$

$= \mu_N(Conv(\textit{bananas} \cup \textit{apples}) \times (\textit{green banana}))$

$= \textit{green banana}$

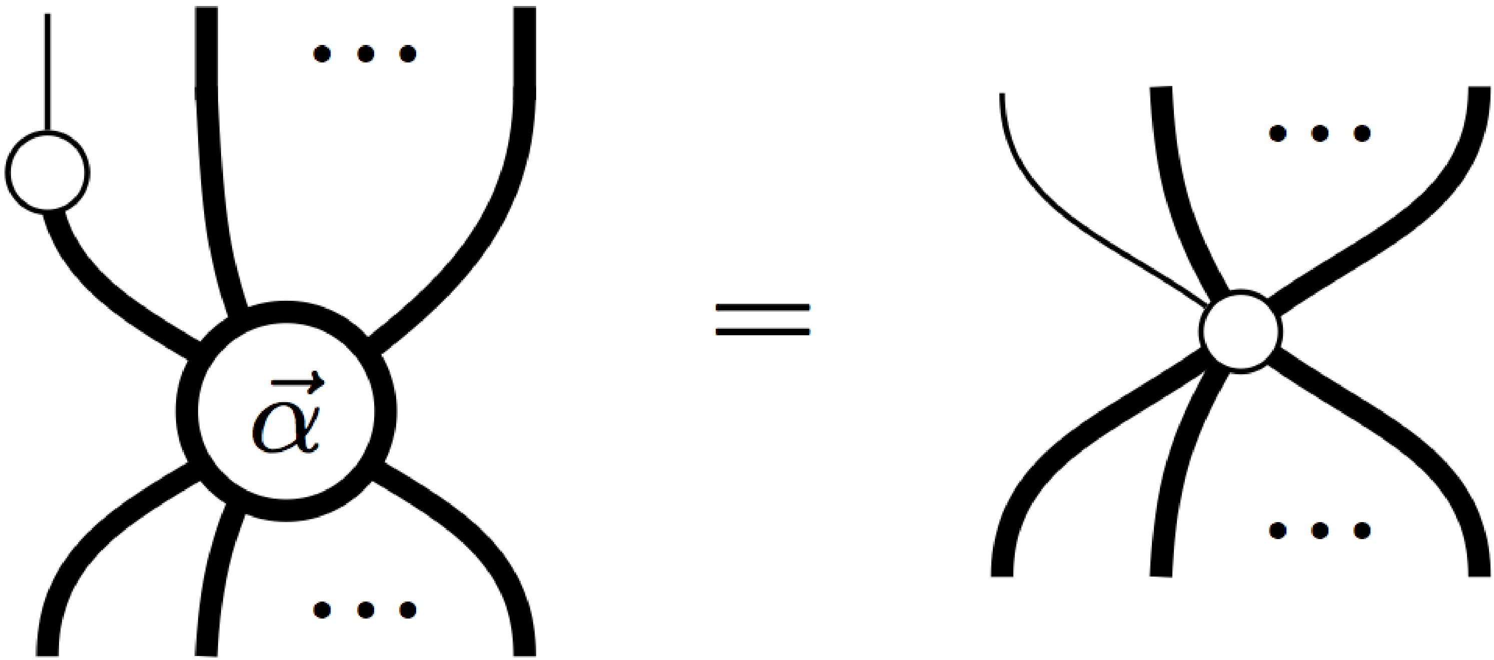

Bolt et al.
1. (a) Choose a compositional structure
(b) Interpret this structure as a category, the grammar category
2. (a) Choose or craft appropriate meaning or concept spaces
(b) Organize these spaces into a semantics category, with the same abstract structure as the grammar category
3. Interpret the compositional structure of the grammar category in the semantics category
4. Bingo! This functor maps type reductions in the grammar category onto algorithms for composing meanings in the semantics category
1. (a) Choose or craft appropriate meaning or concept spaces
(b) Organize these spaces into a semantics category
2. (a) Go to a workshop in Glasgow where you meet people who can help you with 1b and the following step
(b) Use this category to *generate* a compositional structure, e.g. a Lambek grammar
3. Bingo! No interpretation of the grammar category is needed

— Ch. 7 – Picturing phases & complementarity —

phases := purely quantum decoration of spiders

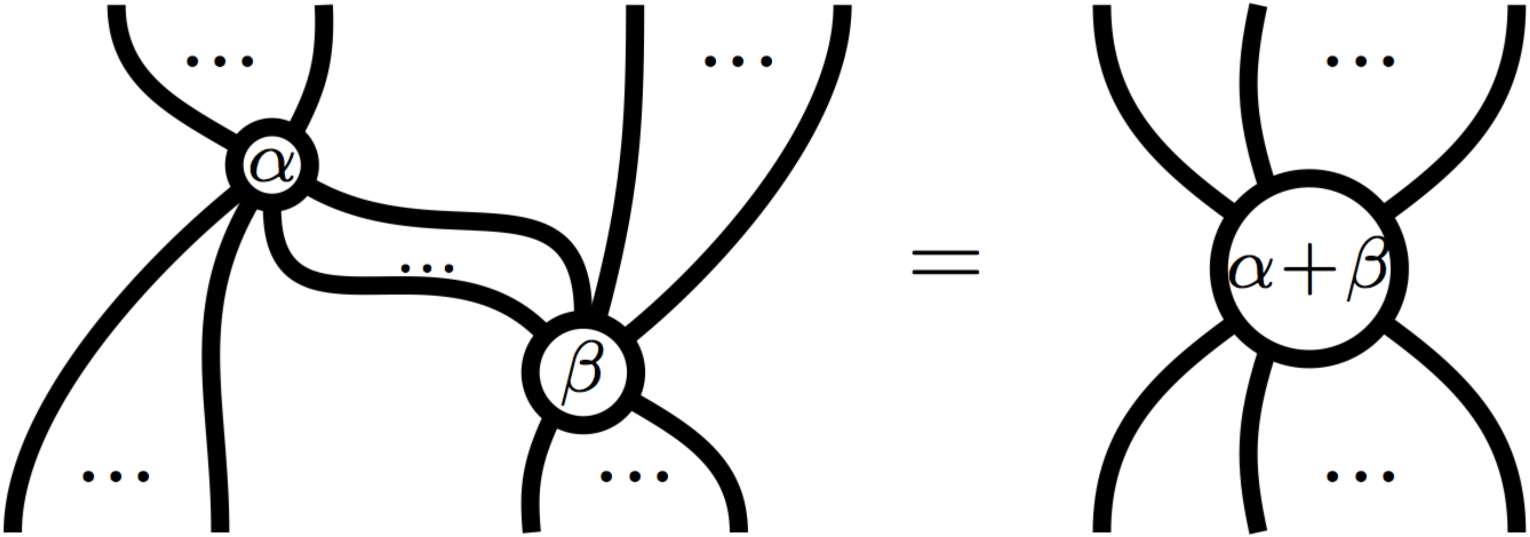
— Ch. 7 – Picturing phases & complementarity —

phases := purely quantum decoration of spiders



phases := purely quantum decoration of spiders

Prop.

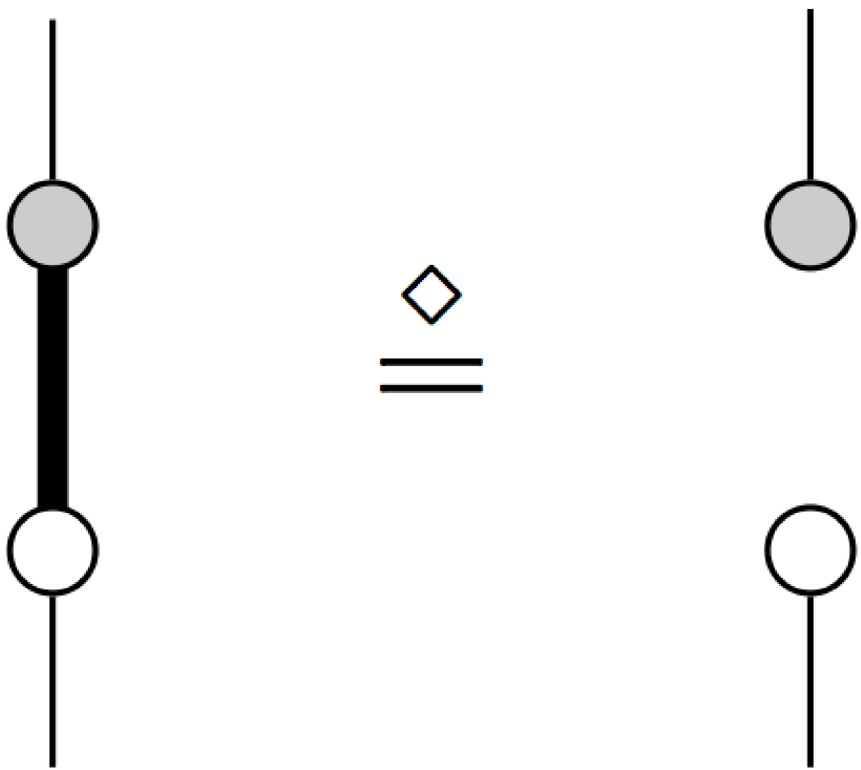


— Ch. 7 – Picturing phases & complementarity —

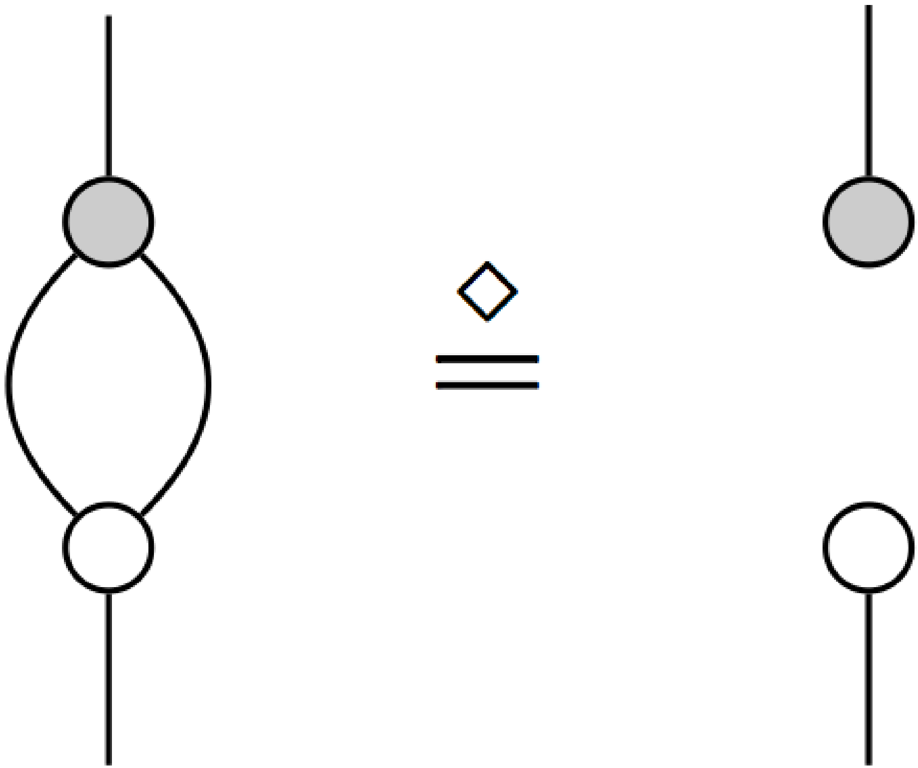
– complementary spiders –

— Ch. 7 – Picturing phases & complementarity —

– complementary spiders –

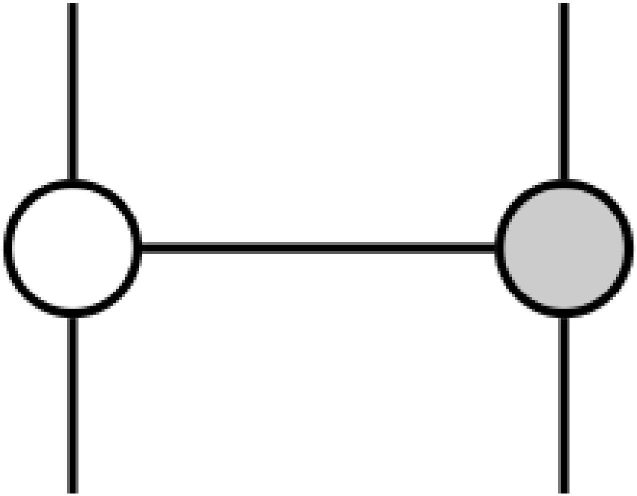


– complementary spiders –



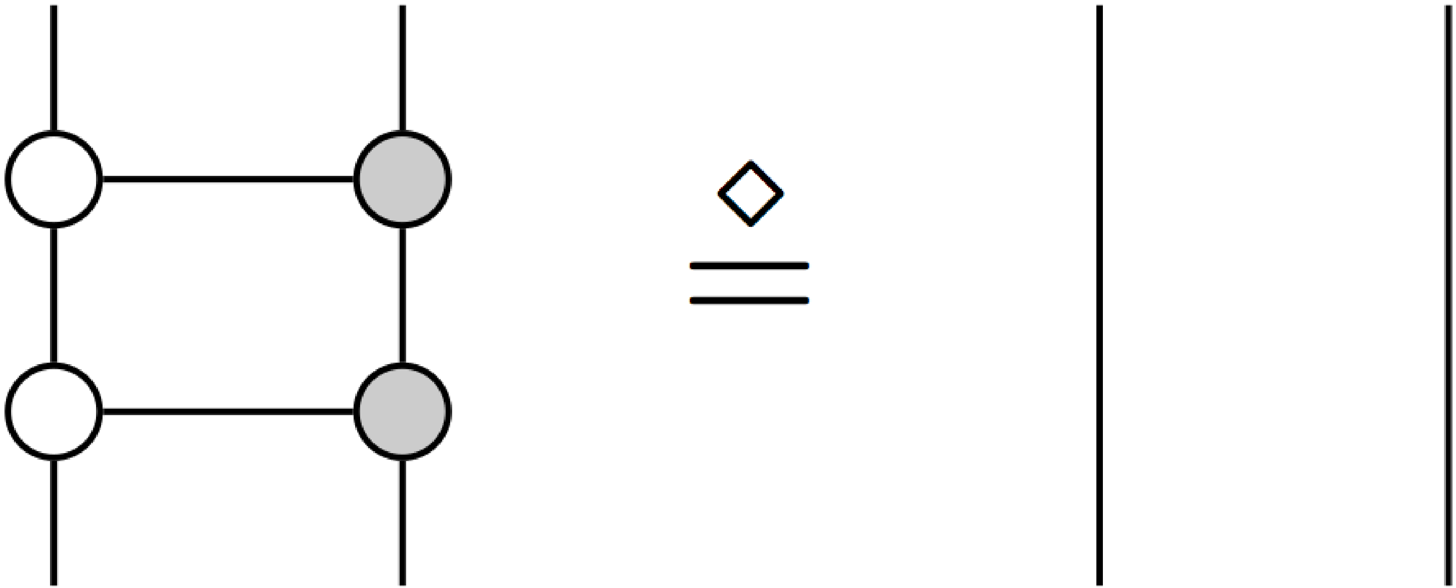
– complementary spiders –

CNOT :=



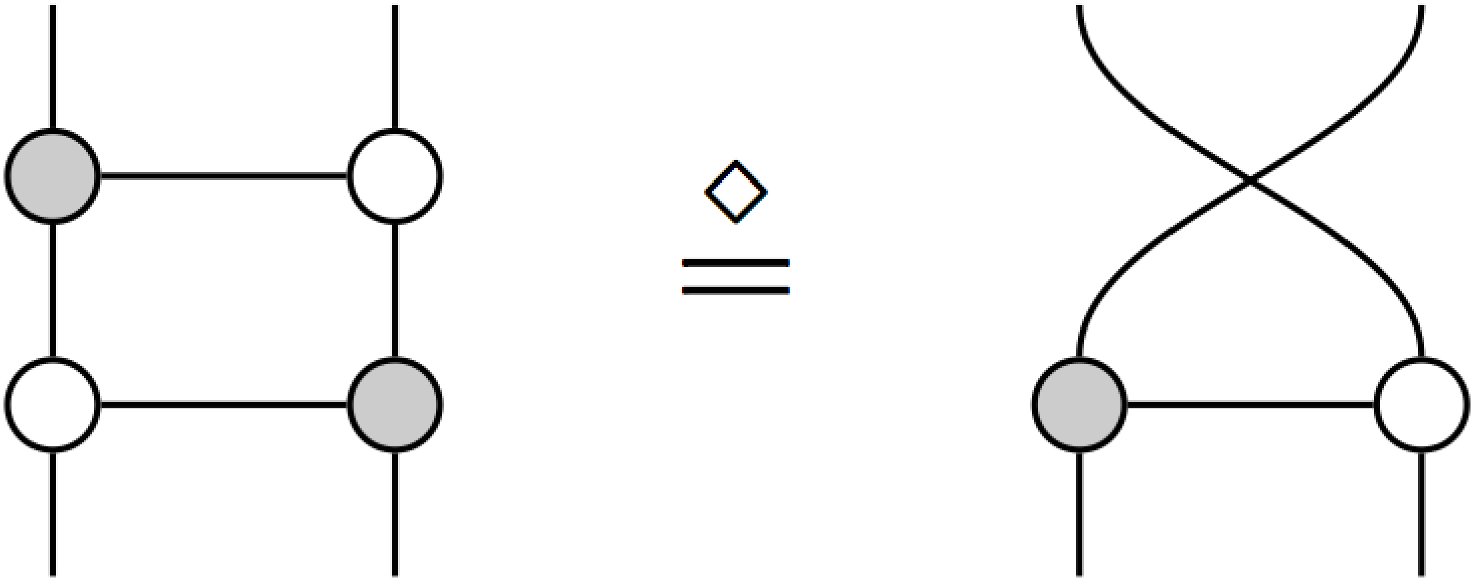
– complementary spiders –

Cor.



– complementary spiders –

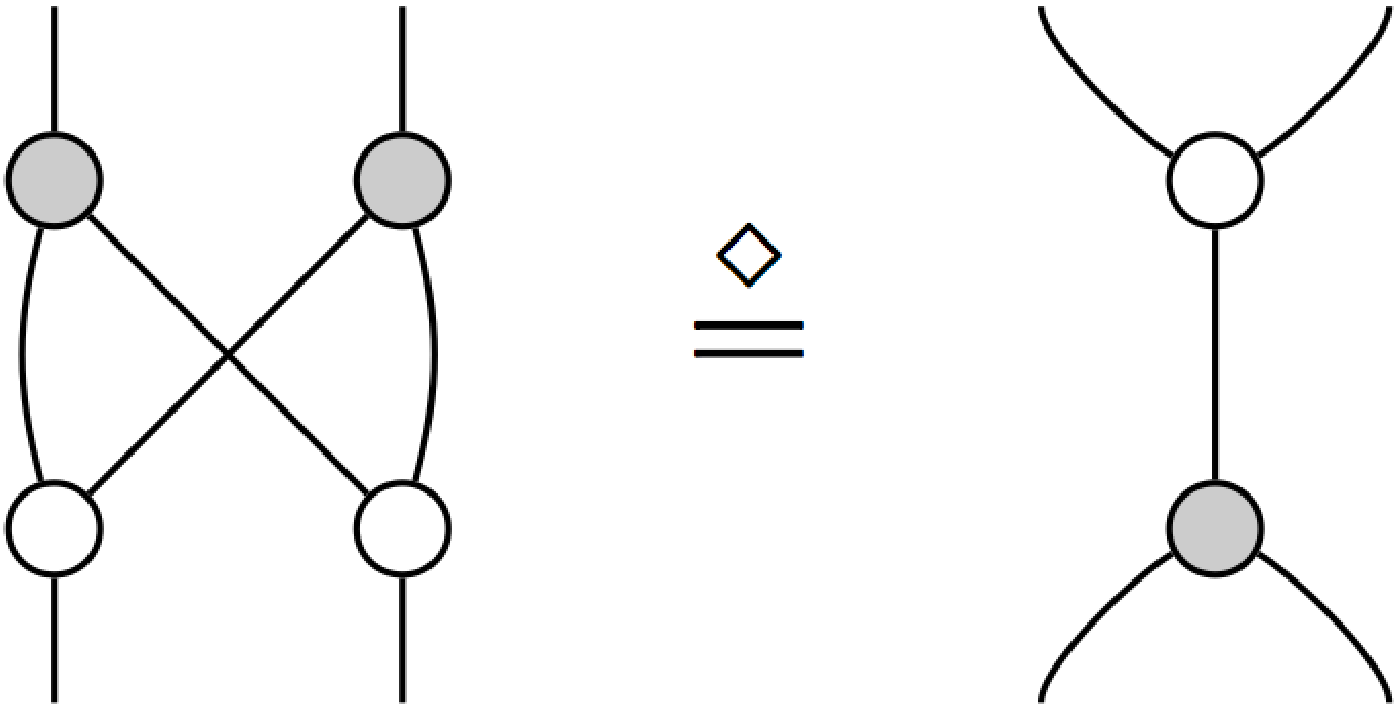
Desire.



— Ch. 7 – Picturing phases & complementarity —

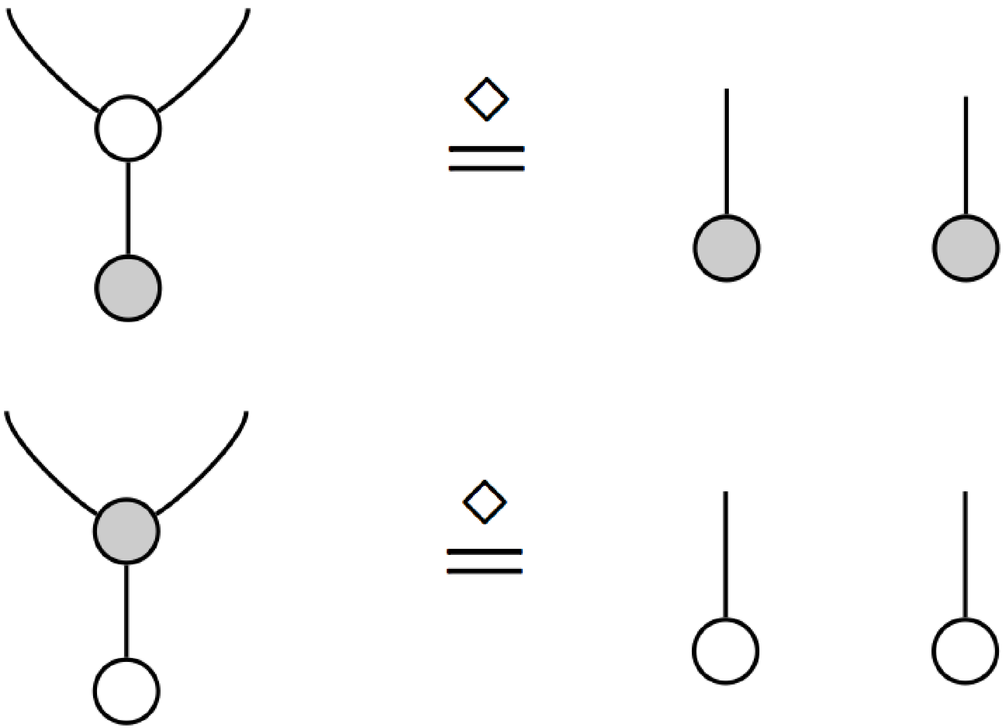
– strongly complementary spiders –

... :=



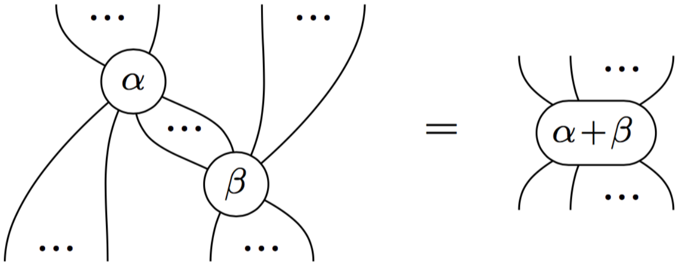
– strongly complementary spiders –

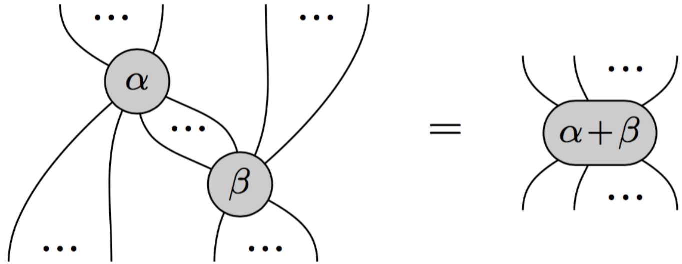
... :=

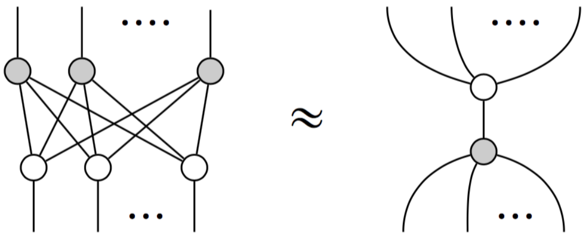


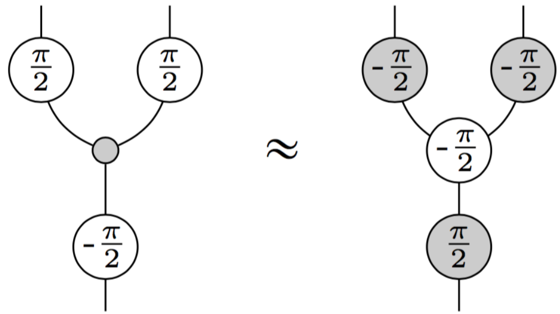
— Ch. 7 – Picturing phases & complementarity —

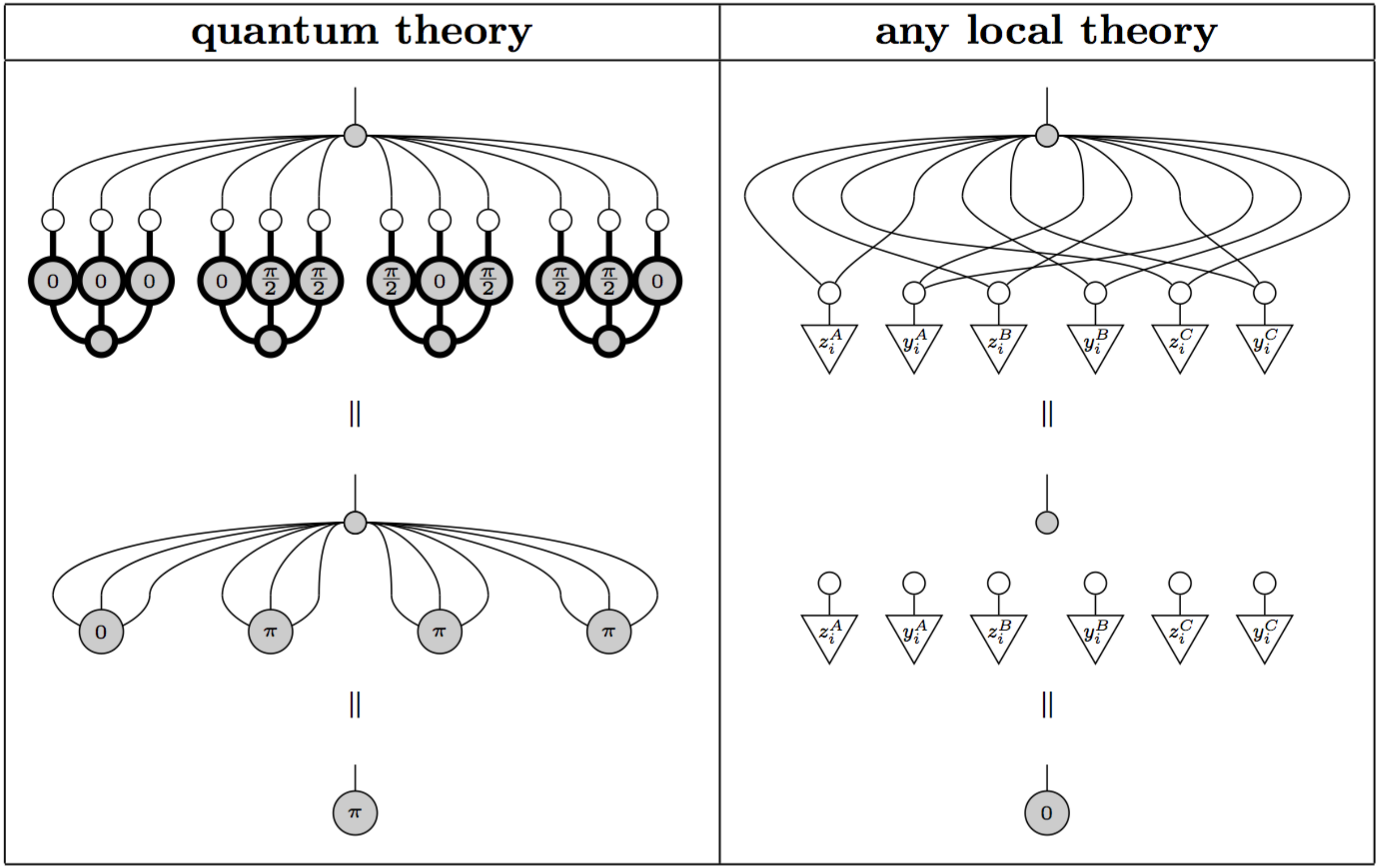
– ZX-calculus –










quantum theory
any local theory
0 0 0
0 $\frac{\pi}{2}$ $\frac{\pi}{2}$
$\frac{\pi}{2}$ 0 $\frac{\pi}{2}$
$\frac{\pi}{2}$ $\frac{\pi}{2}$ 0
z_i^A y_i^A z_i^B y_i^B z_i^C y_i^C
=
0 π π π
z_i^A y_i^A z_i^B y_i^B z_i^C y_i^C
=
π
0

— Ch. 7 – Picturing phases & complementarity —

– completeness –

— Ch. 7 – Picturing phases & complementarity —

– completeness –

M. Backens (2012) Any equational statement is provable in the stabiliser restriction of **ZX-calculus** if and only if it is provable for Hilbert spaces and linear maps.

— Ch. 7 – Picturing phases & complementarity —

– completeness –

M. Backens (2012) Any equational statement is provable in the stabiliser restriction of **ZX-calculus** if and only if it is provable for Hilbert spaces and linear maps.

A. Hadzihasanovic (2015) ... Z/W (with some restriction)...

— Ch. 7 – Picturing phases & complementarity —

– completeness –

M. Backens (2012) Any equational statement is provable in the stabiliser restriction of **ZX-calculus** if and only if it is provable for Hilbert spaces and linear maps.

A. Hadzihasanovic (2015) ... Z/W (with some restriction)...

E. Jeandel, S. Perdrix & R. Vilmart (2017) ... Clifford +T...

— Ch. 7 – Picturing phases & complementarity —

– completeness –

M. Backens (2012) Any equational statement is provable in the stabiliser restriction of **ZX-calculus** if and only if it is provable for Hilbert spaces and linear maps.

A. Hadzihasanovic (2015) ... Z/W (with some restriction)...

E. Jeandel, S. Perdrix & R. Vilmart (2017) ... Clifford +T...

A. Hadzihasanovic (2017) ... Z/W (no restriction)...

— Ch. 7 – Picturing phases & complementarity —

– completeness –

M. Backens (2012) Any equational statement is provable in the stabiliser restriction of **ZX-calculus** if and only if it is provable for Hilbert spaces and linear maps.

A. Hadzihasanovic (2015) ... Z/W (with some restriction)...

E. Jeandel, S. Perdrix & R. Vilmart (2017) ... Clifford +T...

A. Hadzihasanovic (2017) ... Z/W (no restriction)...

Kang Feng Ng and Quanlong Wang (2017) ... everything...

— Ch. 7 – Picturing phases & complementarity —

– completeness –

M. Backens (2012) Any equational statement is provable in the stabiliser restriction of **ZX-calculus** if and only if it is provable for Hilbert spaces and linear maps.

A. Hadzihasanovic (2015) ... Z/W (with some restriction)...

E. Jeandel, S. Perdrix & R. Vilmart (2017) ... Clifford +T...

A. Hadzihasanovic (2017) ... Z/W (no restriction)...

Kang Feng Ng and Quanlong Wang (2017) ... everything...

E. Jeandel, S. Perdrix & R. Vilmart (2017) ... everything, better...

— Ch. 7 – Picturing phases & complementarity —

– completeness –

M. Backens (2012) Any equational statement is provable in the stabiliser restriction of **ZX-calculus** if and only if it is provable for Hilbert spaces and linear maps.

A. Hadzihasanovic (2015) ... Z/W (with some restriction)...

E. Jeandel, S. Perdrix & R. Vilmart (2017) ... Clifford +T...

A. Hadzihasanovic (2017) ... Z/W (no restriction)...

Kang Feng Ng and Quanlong Wang (2017) ... everything...

E. Jeandel, S. Perdrix & R. Vilmart (2017) ... everything, better...

Kang Feng Ng and Quanlong Wang (2017) ... everything^{+}, even better...

— Ch. 7 – Picturing phases & complementarity —

– completeness –

M. Backens (2012) Any equational statement is provable in the stabiliser restriction of **ZX-calculus** if and only if it is provable for Hilbert spaces and linear maps.

A. Hadzihasanovic (2015) ... Z/W (with some restriction)...

E. Jeandel, S. Perdrix & R. Vilmart (2017) ... Clifford +T...

A. Hadzihasanovic (2017) ... Z/W (no restriction)...

Kang Feng Ng and Quanlong Wang (2017) ... everything...

E. Jeandel, S. Perdrix & R. Vilmart (2017) ... everything, better...

Kang Feng Ng and Quanlong Wang (2017) ... everything$^+$, even better...

E. Jeandel, S. Perdrix & R. Vilmart (51 minutes ago) ... everything, even^2 better...

Kang Feng Ng and Quanlong Wang (37 minutes ago) ... everything$^+$, even^3 better...

E. Jeandel, S. Perdrix & R. Vilmart (13.7 minutes ago) ... everything, even^4 better...

Kang Feng Ng and Quanlong Wang (3.4 seconds ago) ... everything$^+$, even^5 better...

Ongoing collaboration with:

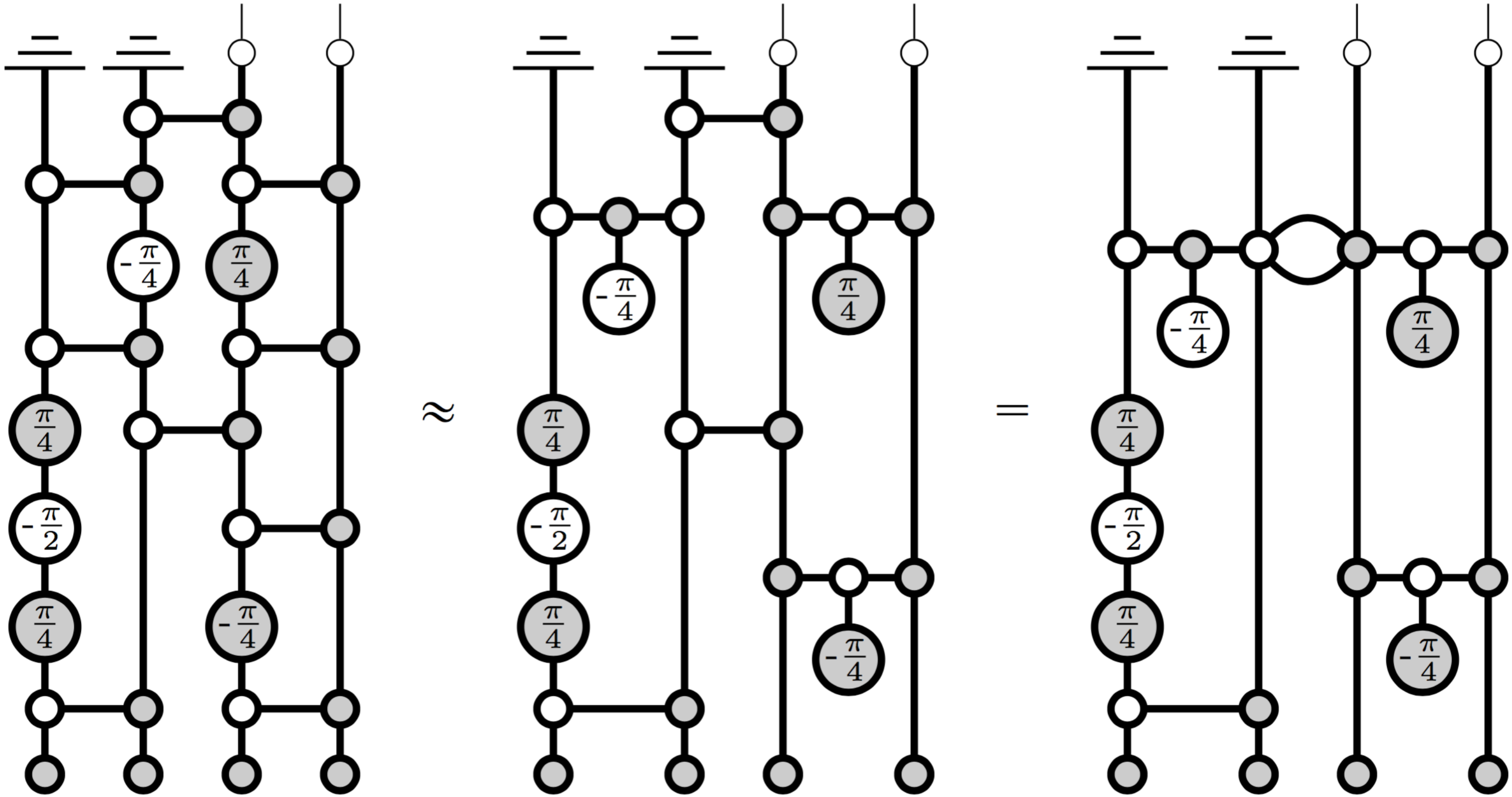
- **Cambridge Quantum Computing Inc.**

towards:

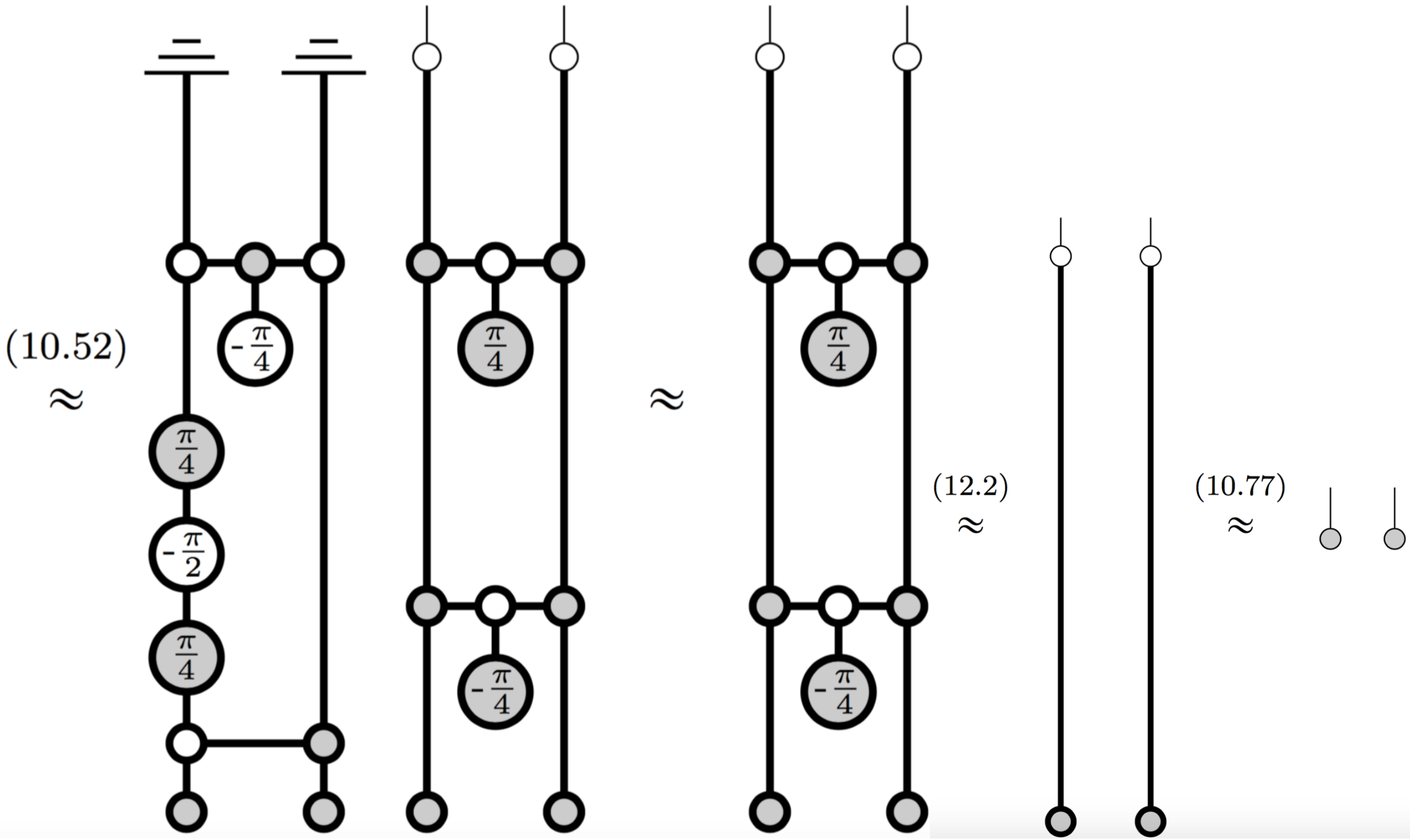
- **architecture-independent**
- **exact-efficient**

quantum compiler.

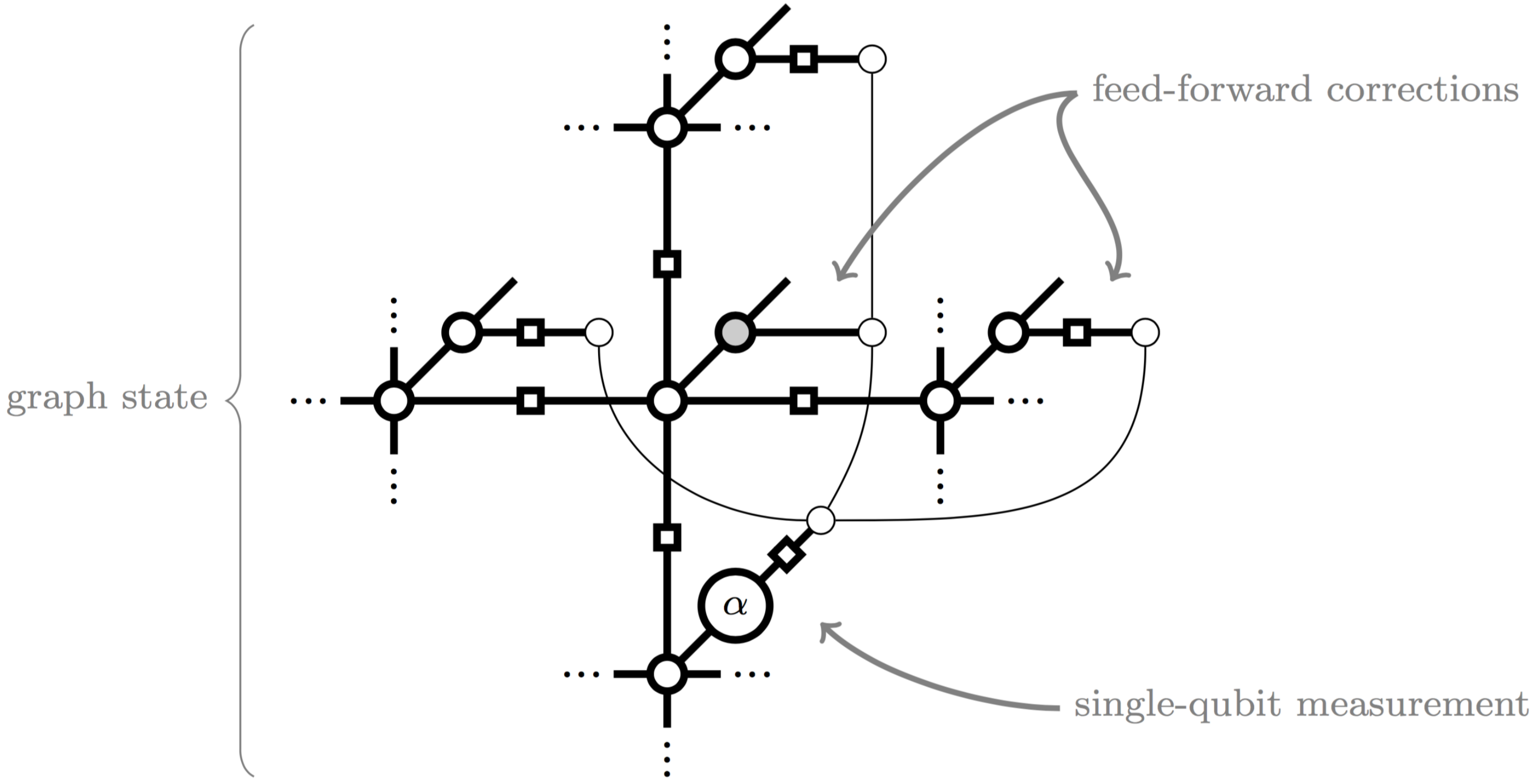
circuit rewriting :=



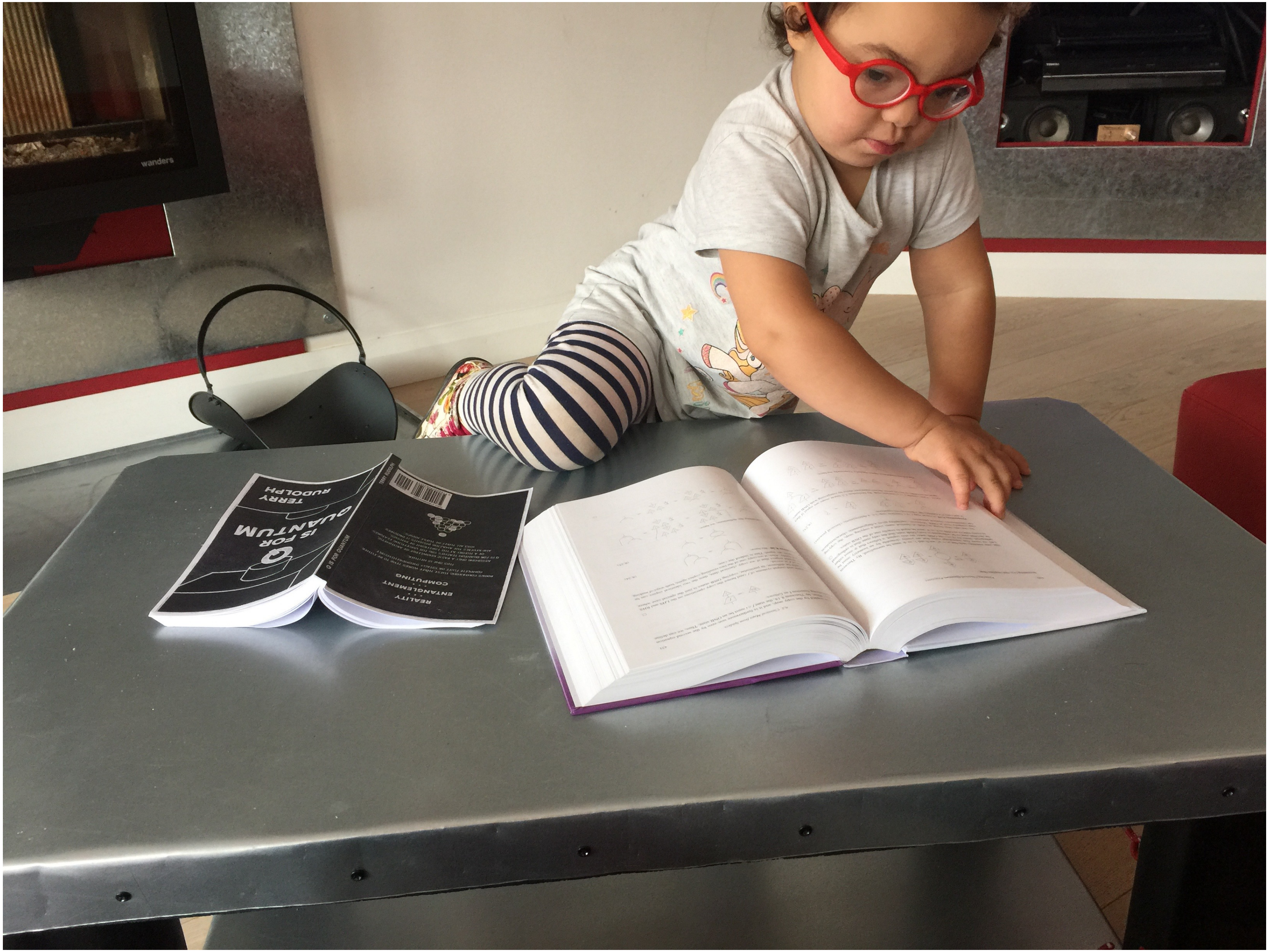
circuit rewriting :=

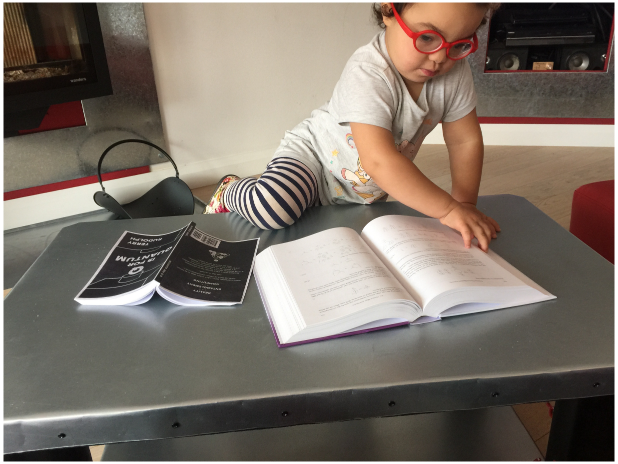


measurement based quantum computing :=



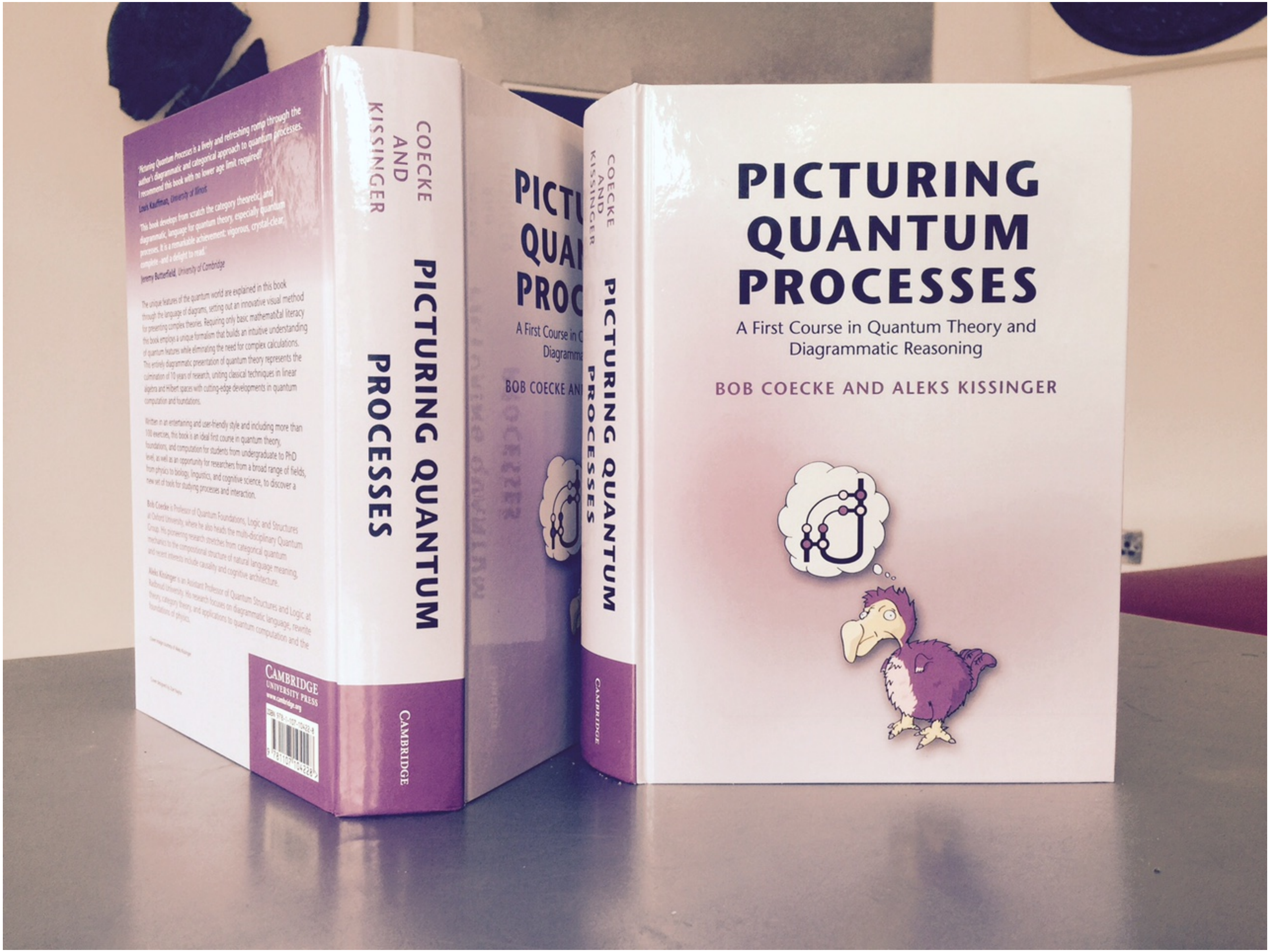
How young can one start this business?





KIDS OUTPERFORM THEIR TEACHERS AND DISCOVER
QUANTUM FEATURES THAT TOOK TOP SCIENTISTS 60y

EXPERIMENTS THIS SPRING !


PICTURING QUANTUM PROCESSES
A First Course in Quantum Theory and Diagrammatic Reasoning
BOB COECKE AND ALEKS KISSINGER
COECKE AND KISSINGER
PICTURING QUANTUM PROCESSES
CAMBRIDGE
CAMBRIDGE UNIVERSITY PRESS