Center for
Quantum&
Topological
Systems

Quantum Data Types via Linear HoTT

presentation at:

Workshop on Quantum Software @ QTML 2022

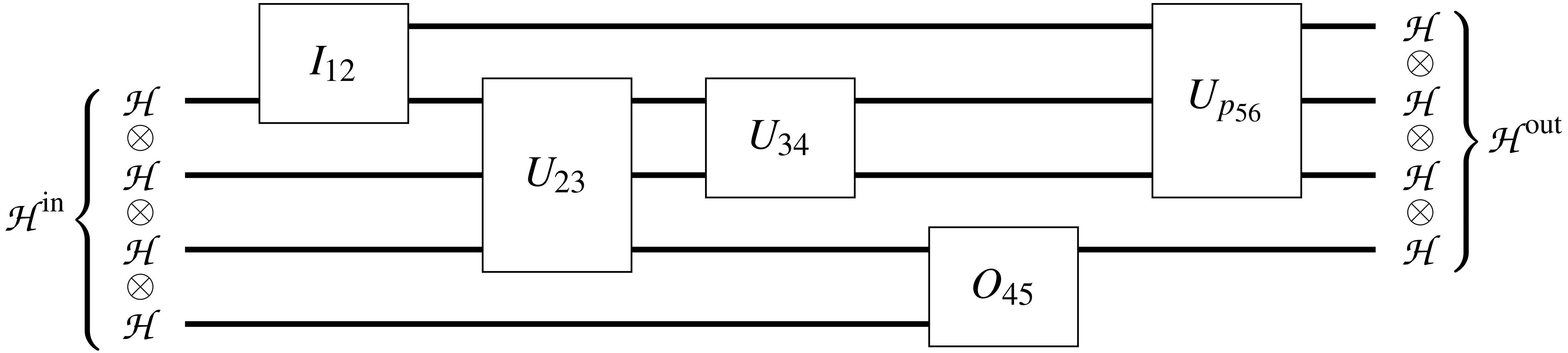
Urs Schreiber (NYU Abu Dhabi)
on joint work at CQTS with
D. J. Myers, M. Riley,
and Hisham Sati

slides and further pointers at: ncatlab.org/schreiber/show/QDataInLHoTT#QTML2022

The Problem

Pure quantum circuits are easy...

Linear operator composed & tensored from given *quantum logic gates*



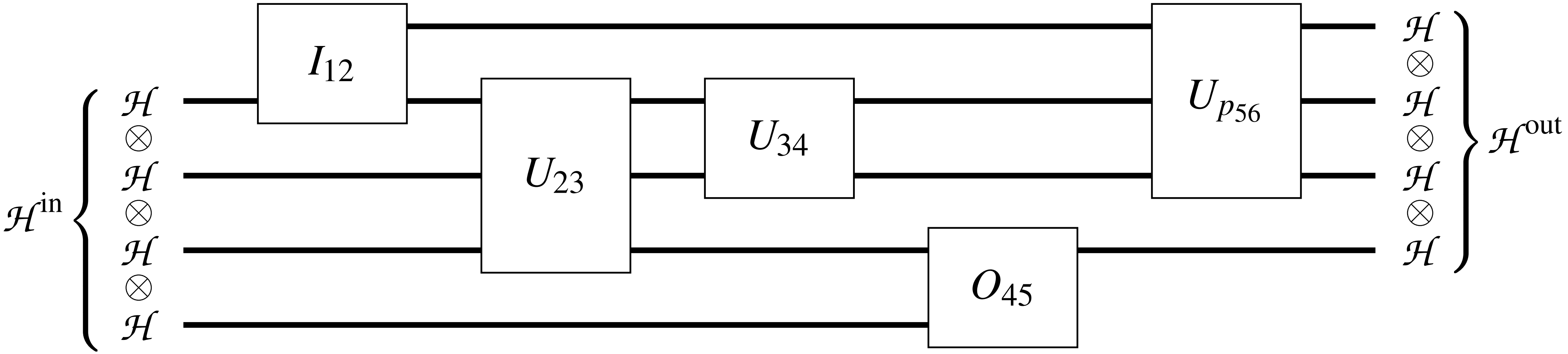
Hilbert space of possible input quantum states

linear transformation
upon execution

Hilbert space of possible output quantum states

Pure quantum circuits are easy...

Linear operator composed & tensored from given *quantum logic gates*



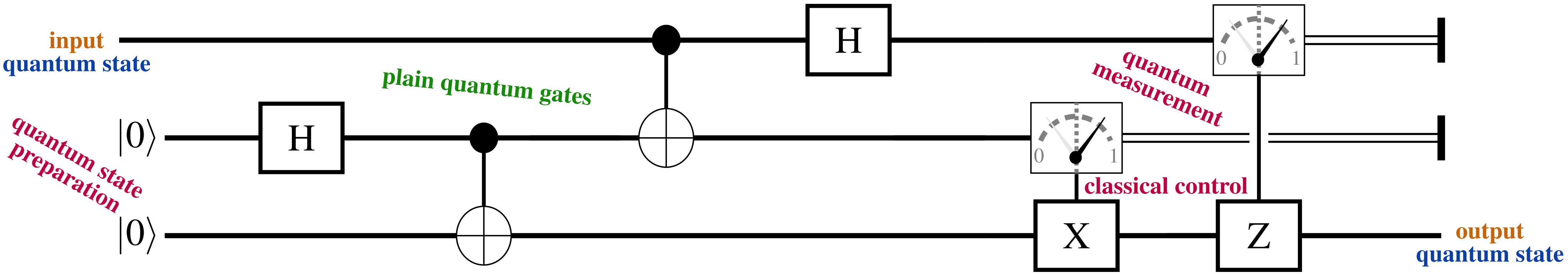
Hilbert space of possible input quantum states

linear transformation upon execution $\longrightarrow$

Hilbert space of possible output quantum states

but real quantum circuits have classical control & effects

(Example: QBit Teleportation protocol)



input quantum state

quantum state preparation

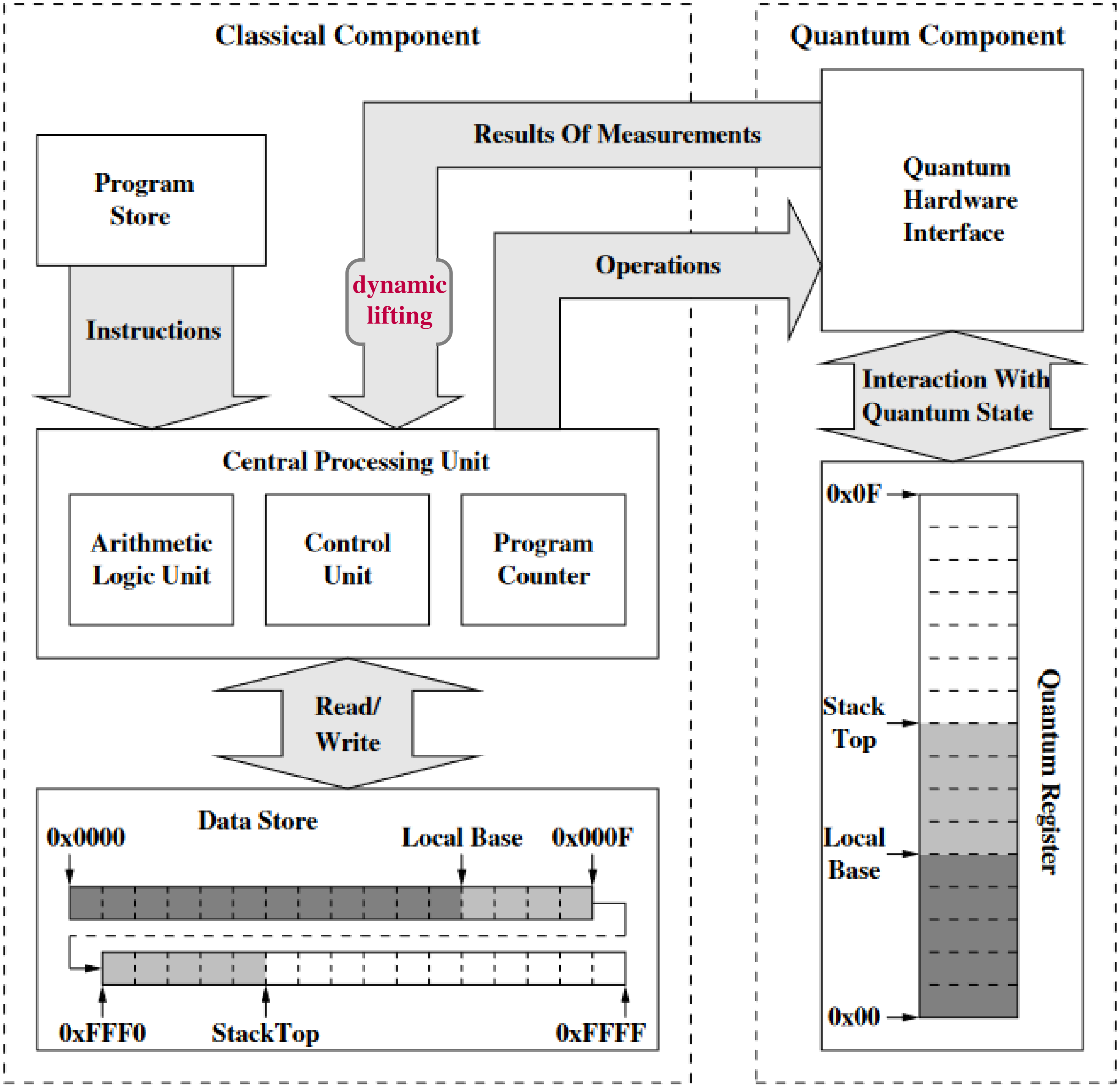
plain quantum gates

quantum measurement

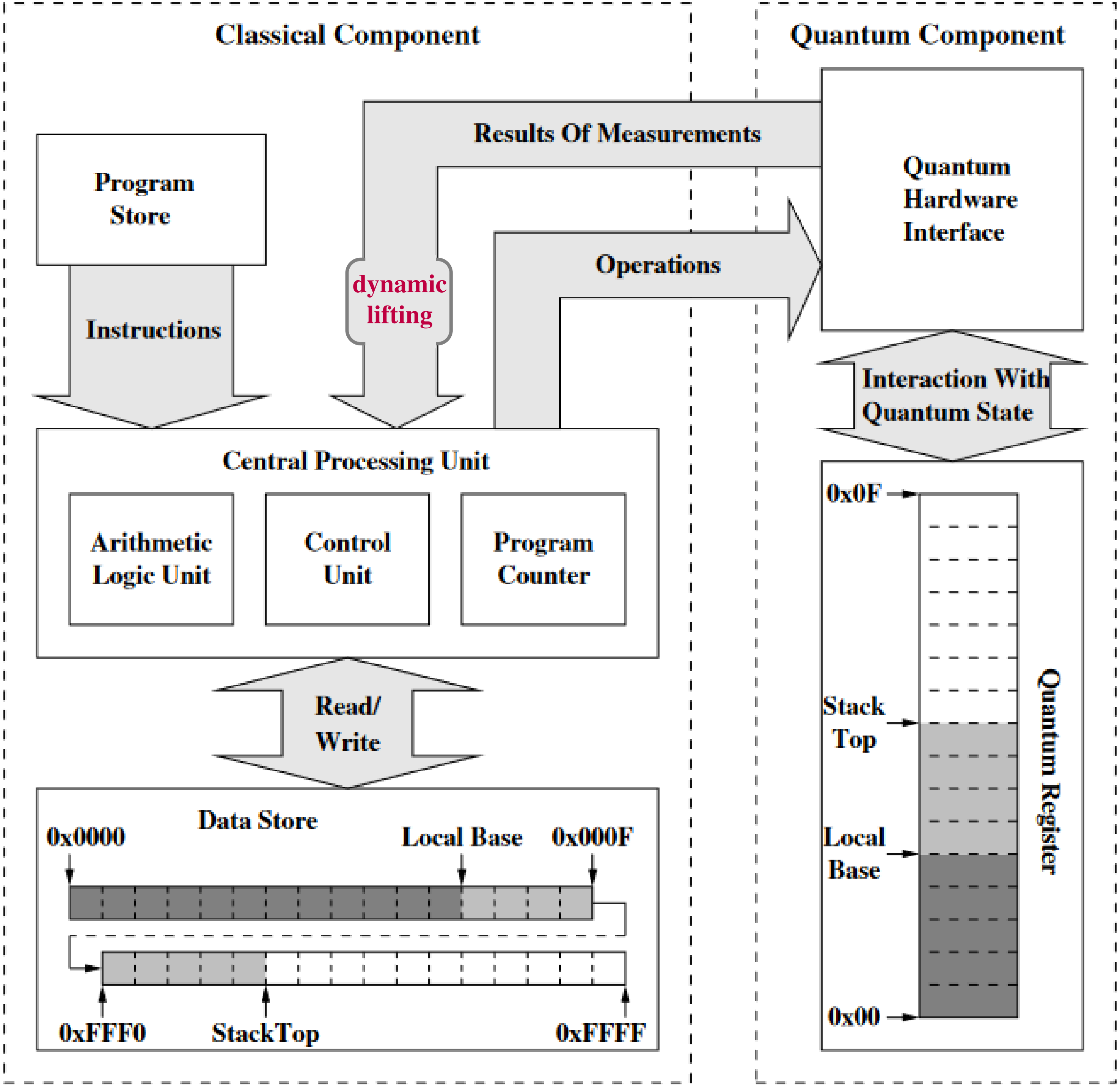
classical control

output quantum state

full reality is a loop: Classical $\underset{\text{prepare}}{\overset{\text{measure}}{\leftrightarrows}}$ Quantum

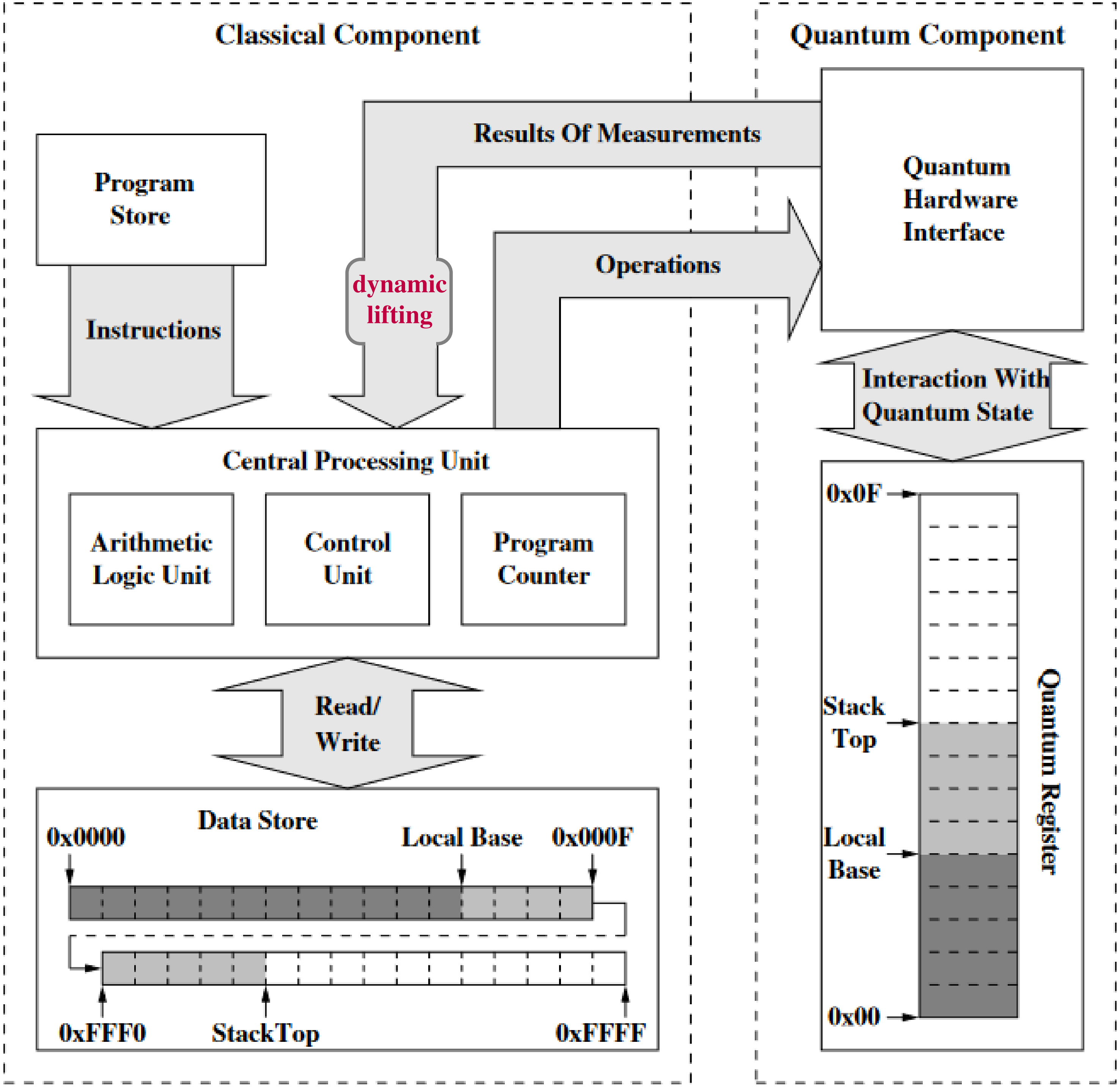


full reality is a loop: Classical $\underset{\text{prepare}}{\overset{\text{measure}}{\leftrightarrows}}$ Quantum



full loop needed e.g. for quantum error correction

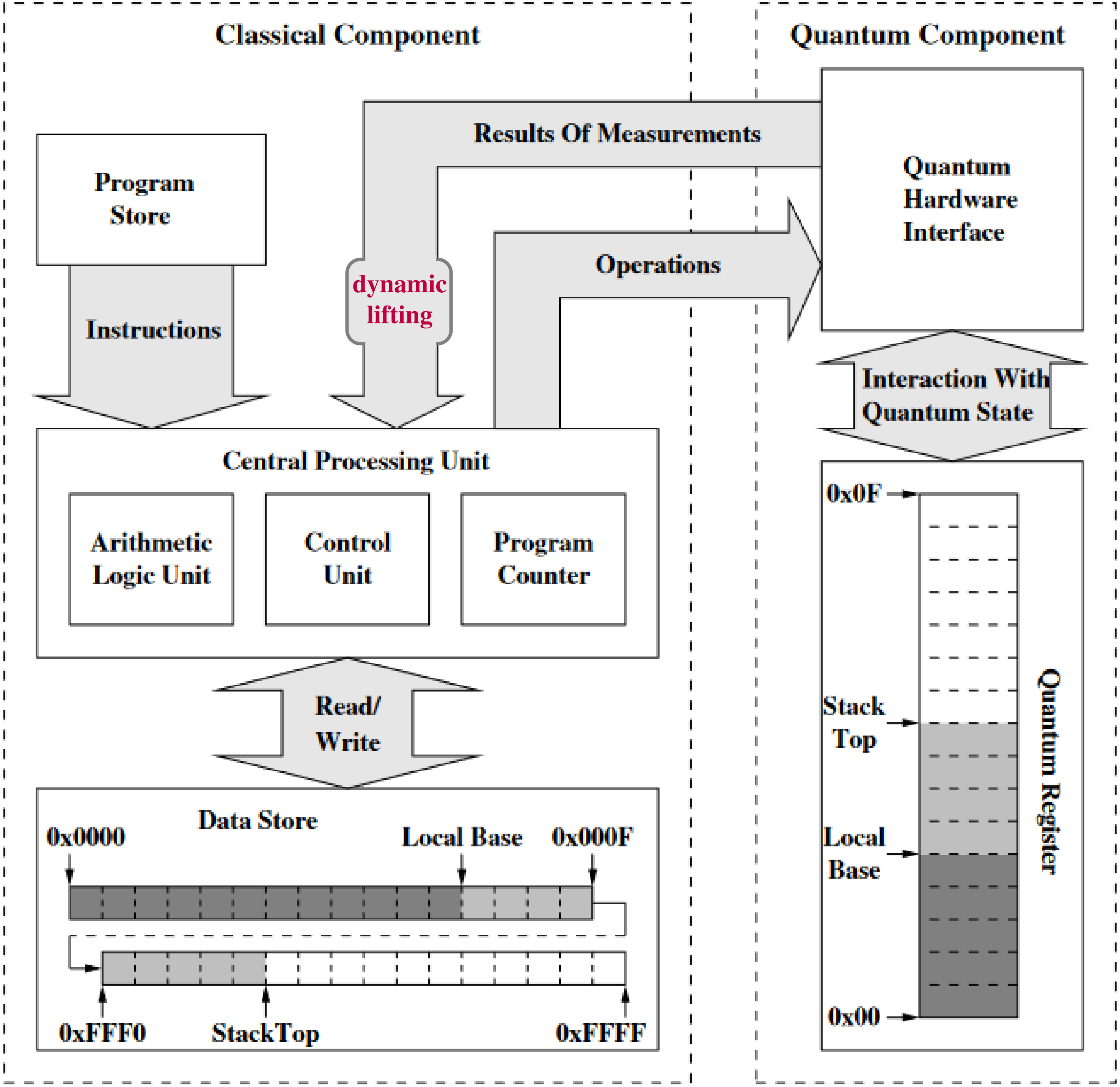
full reality is a loop: Classical $\underset{\text{prepare}}{\overset{\text{measure}}{\leftrightarrows}}$ Quantum



full loop needed e.g. for quantum error correction

but its formal language theory had remained thin

full reality is a loop: Classical $\underset{\text{prepare}}{\overset{\text{measure}}{\leftrightarrows}}$ Quantum



full loop needed e.g. for quantum error correction

but its formal language theory had remained thin

existing models for dynamic lifting are ad hoc & unverified

Existing quantum typed circuit languages

are embedded inside *classical* type theories:

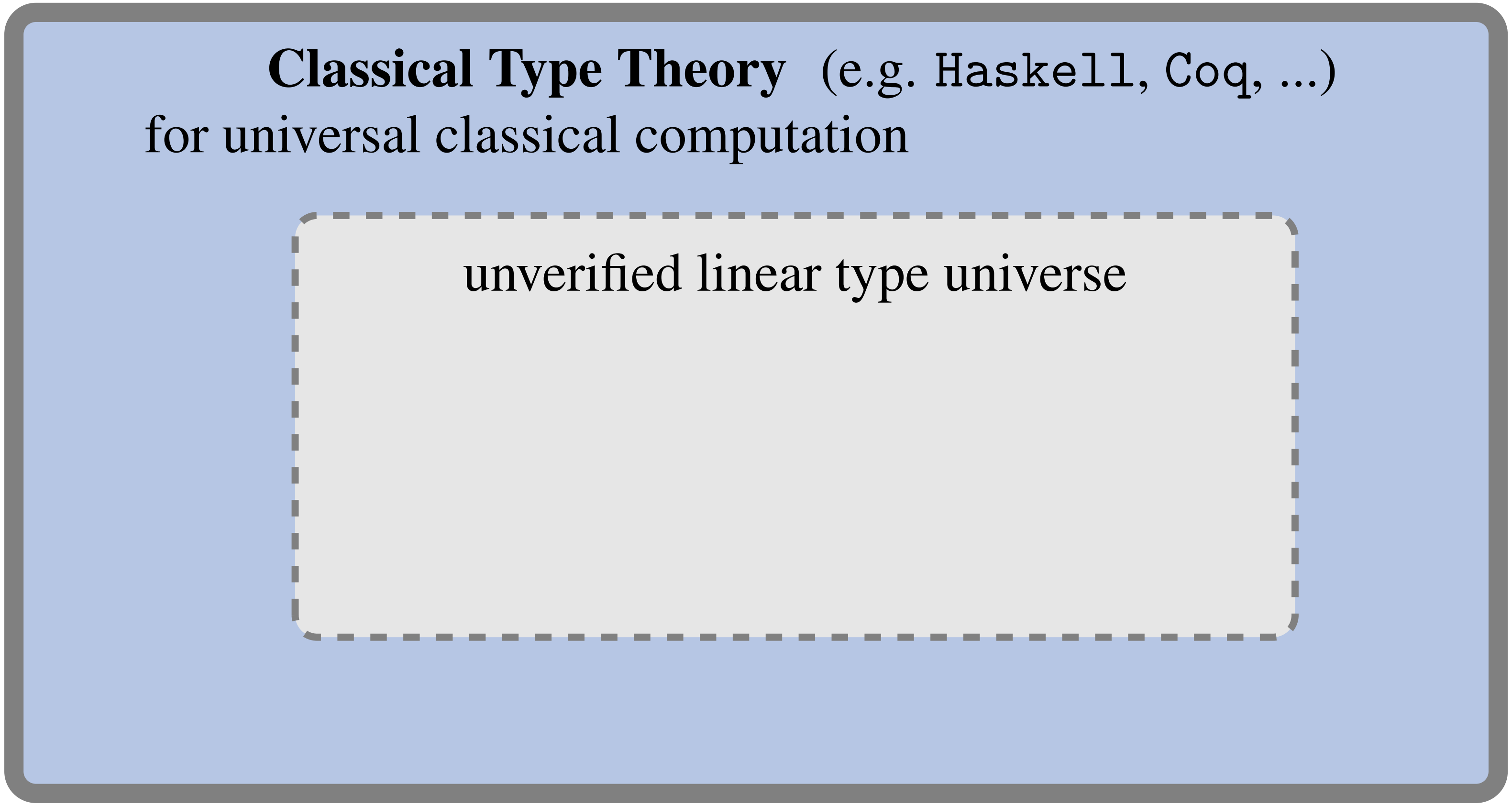
Existing quantum typed circuit languages

are embedded inside *classical* type theories:

Classical Type Theory (e.g. `Haskell`, `Coq`, ...)
for universal classical computation

Existing quantum typed circuit languages

are embedded inside *classical* type theories:



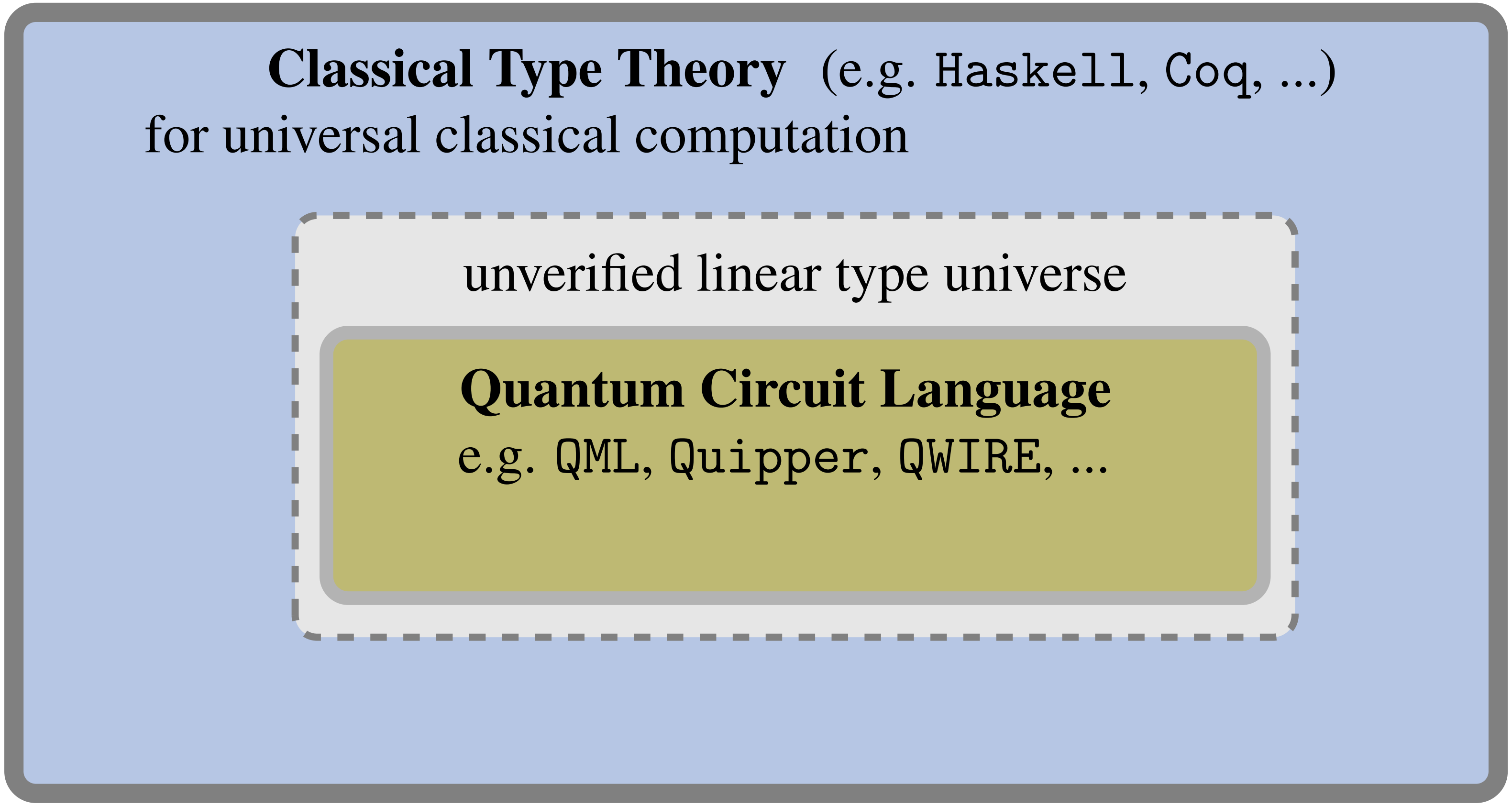
Classical Type Theory (e.g. `Haskell`, `Coq`, ...)

for universal classical computation

unverified linear type universe

Existing quantum typed circuit languages

are embedded inside *classical* type theories:



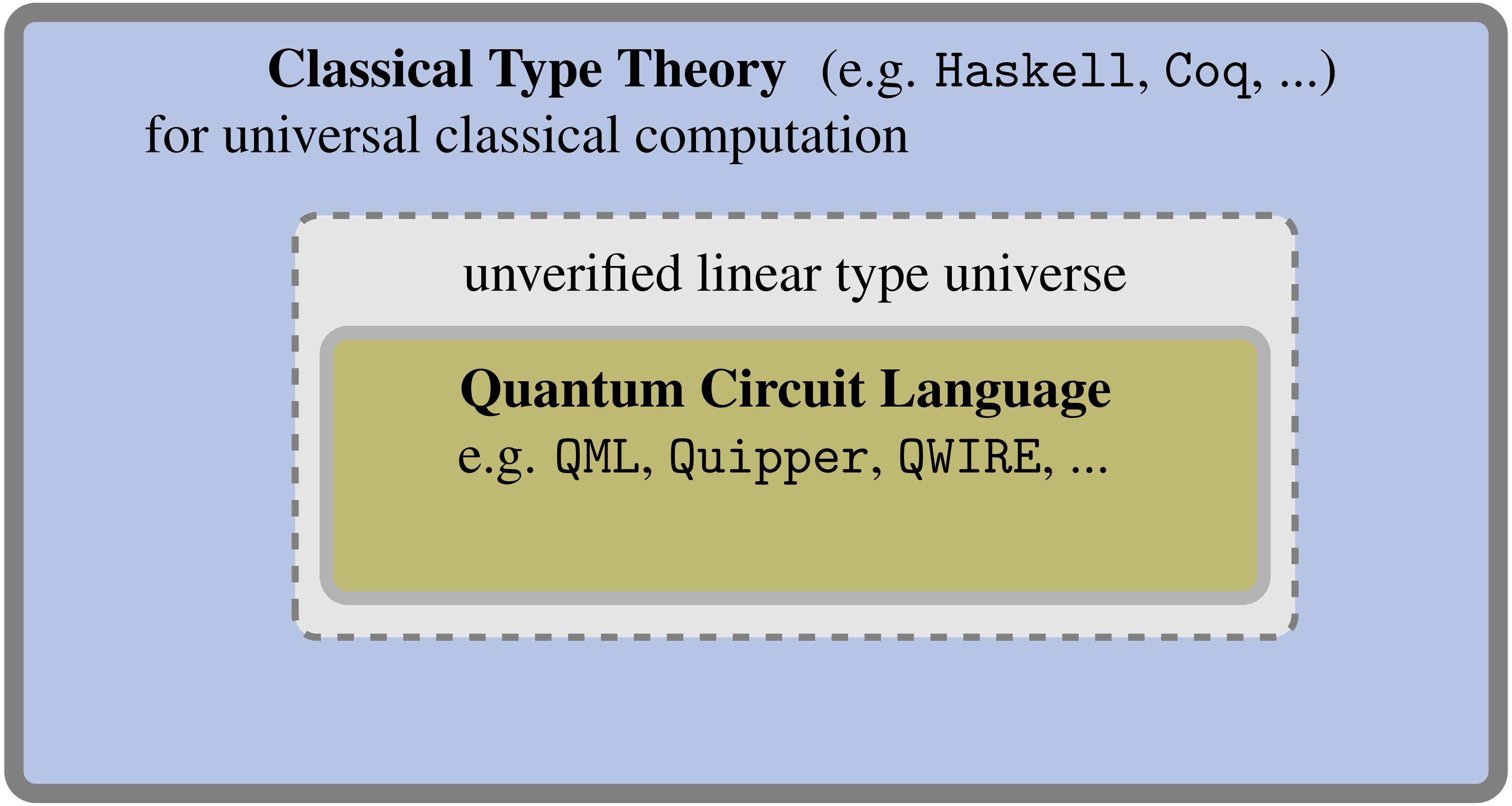
Classical Type Theory (e.g. `Haskell`, `Coq`, ...)
for universal classical computation

unverified linear type universe

Quantum Circuit Language
e.g. `QML`, `Quipper`, `QWIRE`, ...

Existing quantum typed circuit languages

are embedded inside *classical* type theories:



Classical Type Theory (e.g. `Haskell`, `Coq`, ...)
for universal classical computation

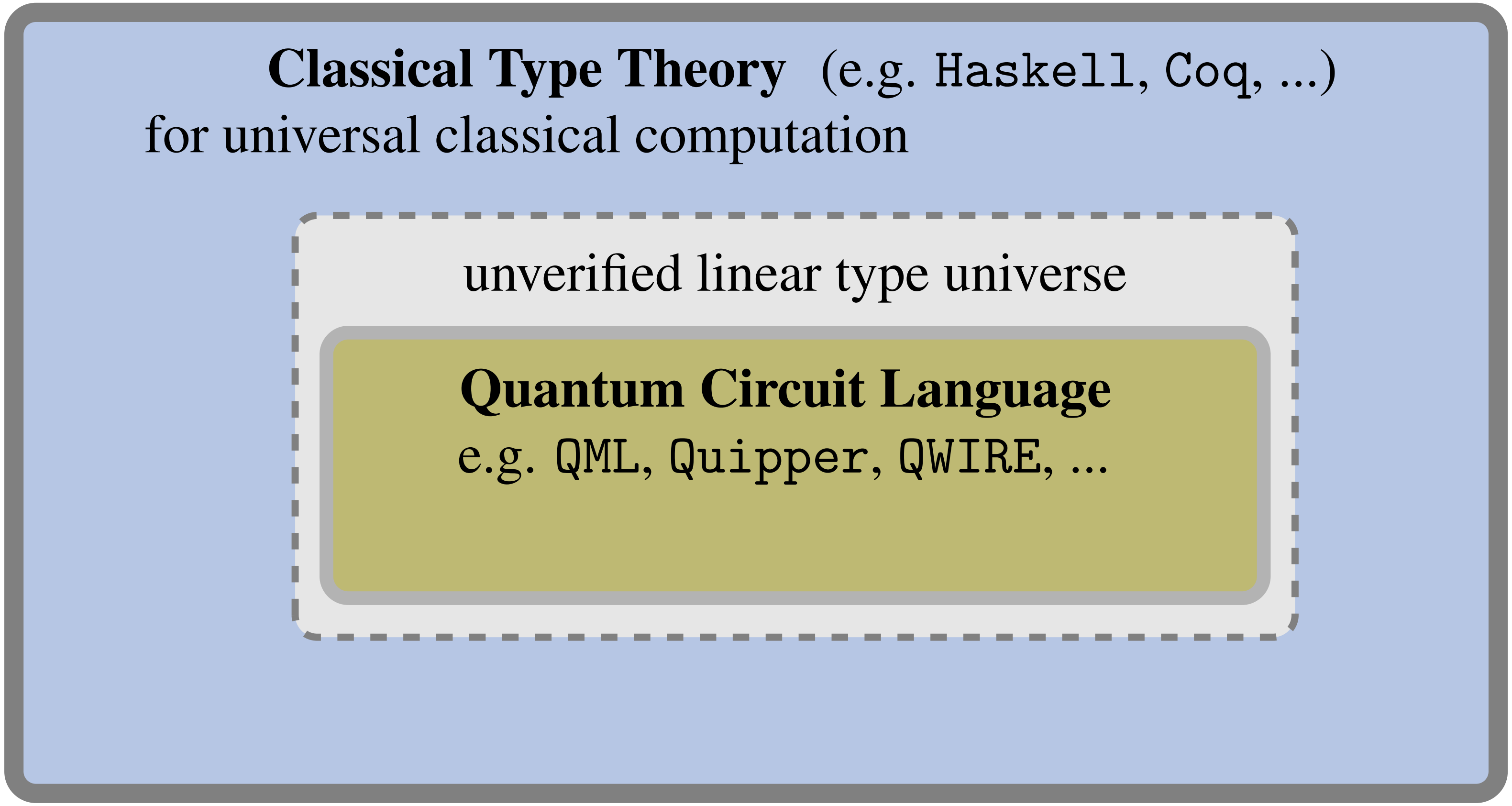
unverified linear type universe

Quantum Circuit Language
e.g. `QML`, `Quipper`, `QWIRE`, ...

for lack of a universal linear type theory.

Existing quantum typed circuit languages

are embedded inside *classical* type theories:



Classical Type Theory (e.g. `Haskell`, `Coq`, ...)
for universal classical computation

unverified linear type universe

Quantum Circuit Language
e.g. `QML`, `Quipper`, `QWIRE`, ...

for lack of a universal linear type theory.

Until now...

Our Solution

Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:

Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:

$\exists$ *universal* quantum+classical specification language

conservative over classical *Homotopy Type Theory* (`HoTT`)

Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:

∃ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]

(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)

Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:

$\exists$ *universal* quantum+classical specification language

conservative over classical *Homotopy Type Theory* (`HoTT`)

and

verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]

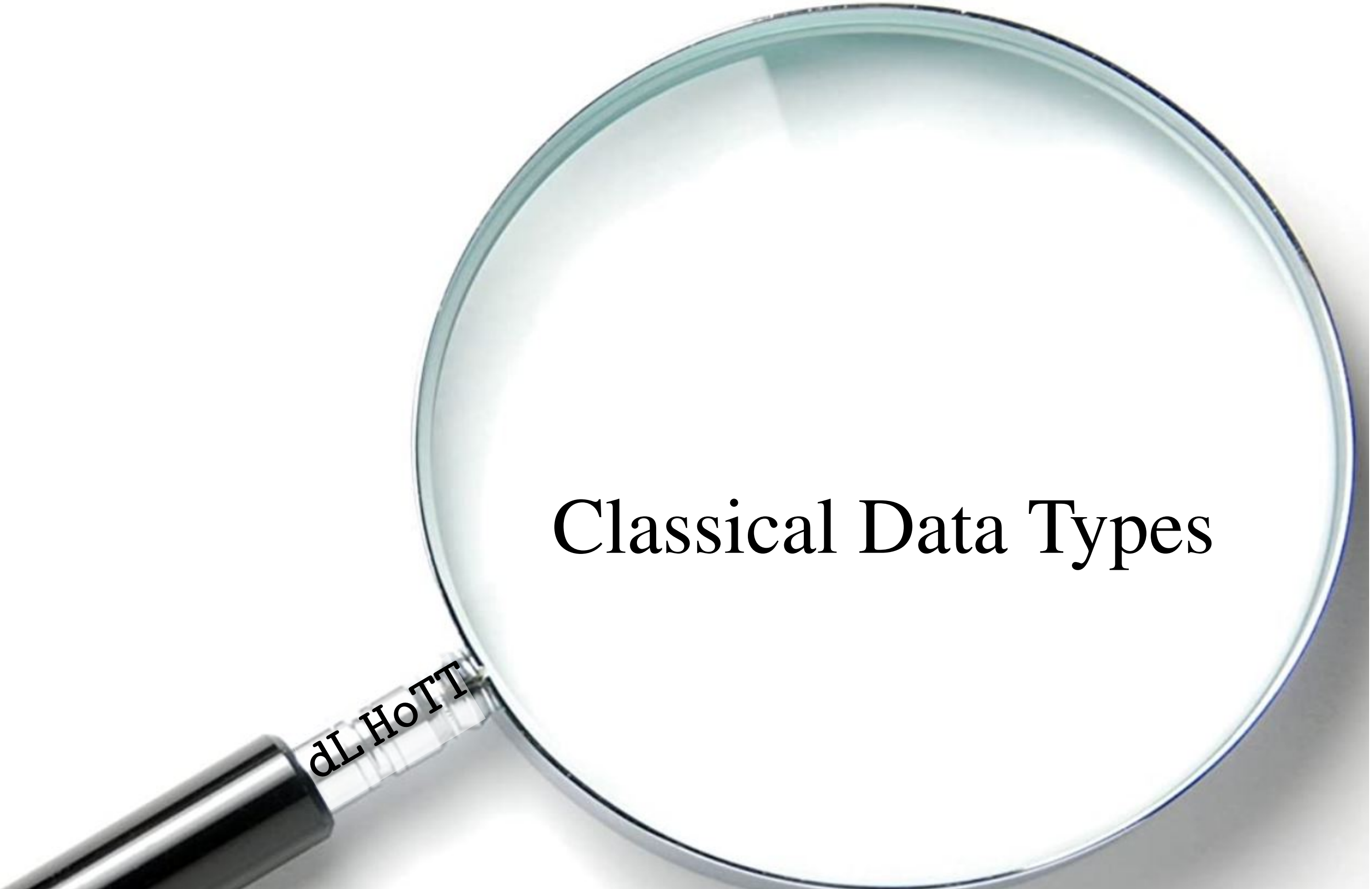
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)

Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
∃ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)

`dLHoTT` is like a quantum microscope for Classical Data Types

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
$\exists$ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)



Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:

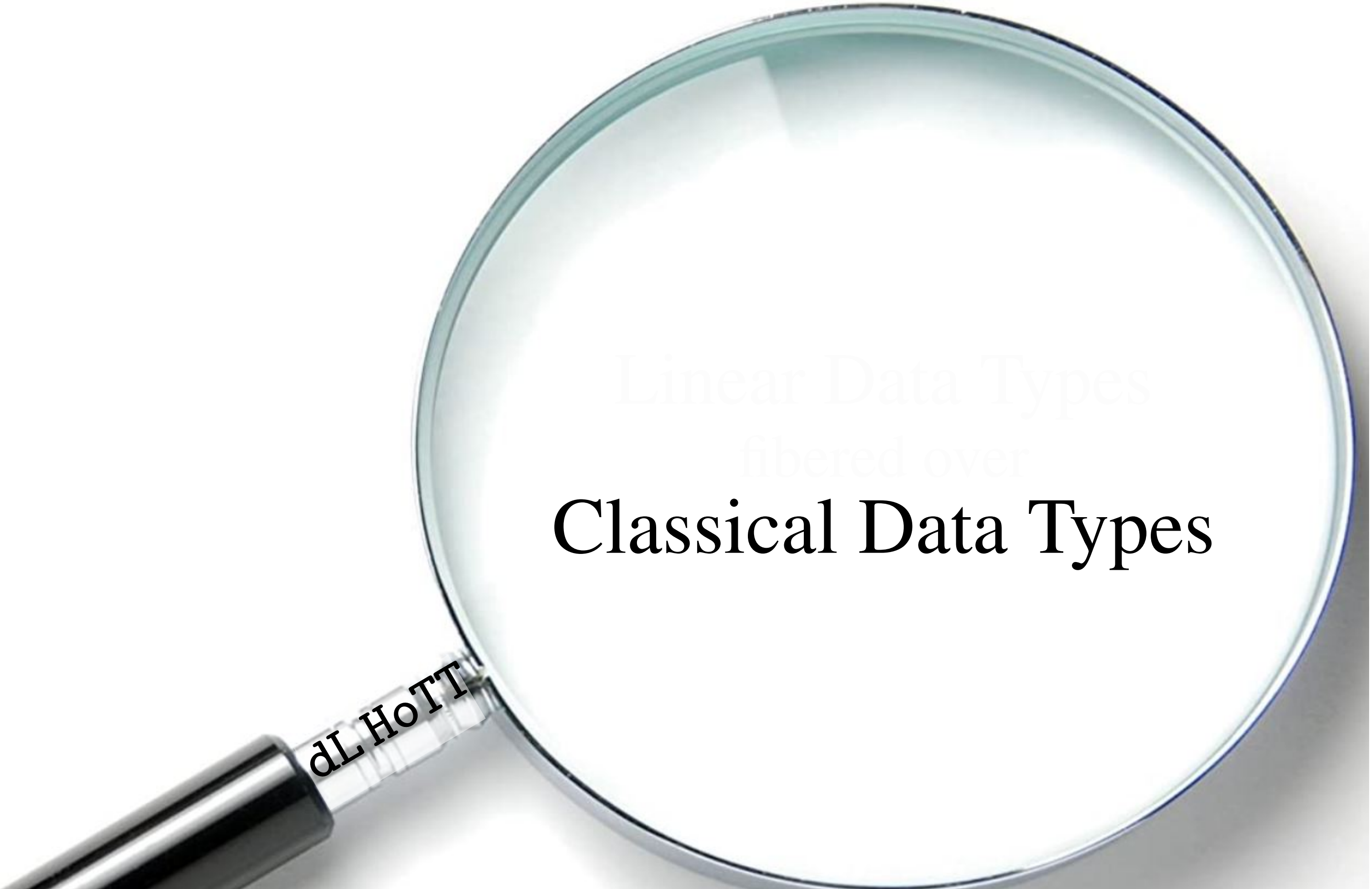
∃ *universal* quantum+classical specification language

conservative over classical *Homotopy Type Theory* (`HoTT`)

and

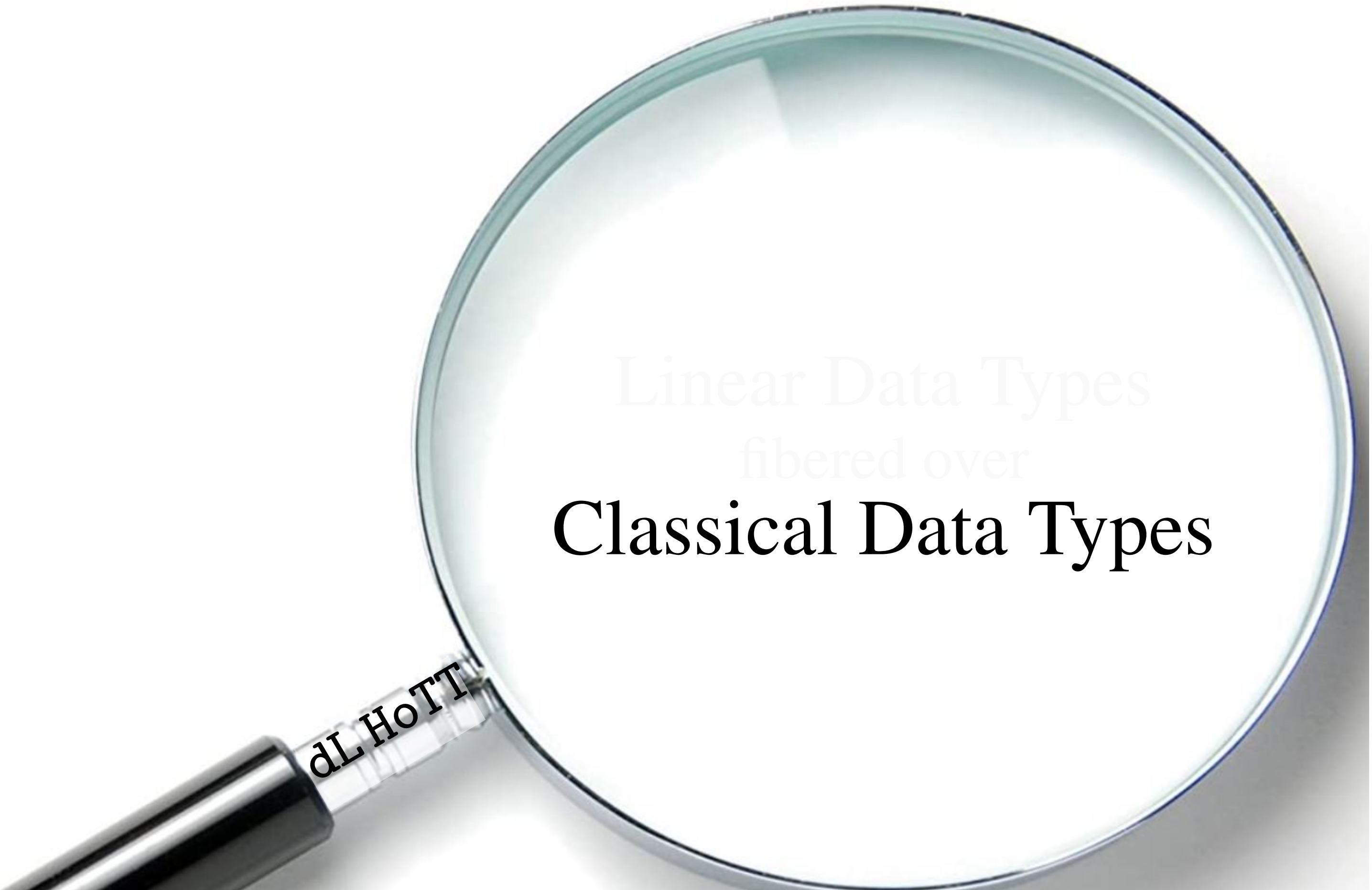
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]

(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)



Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
∃ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)



Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:

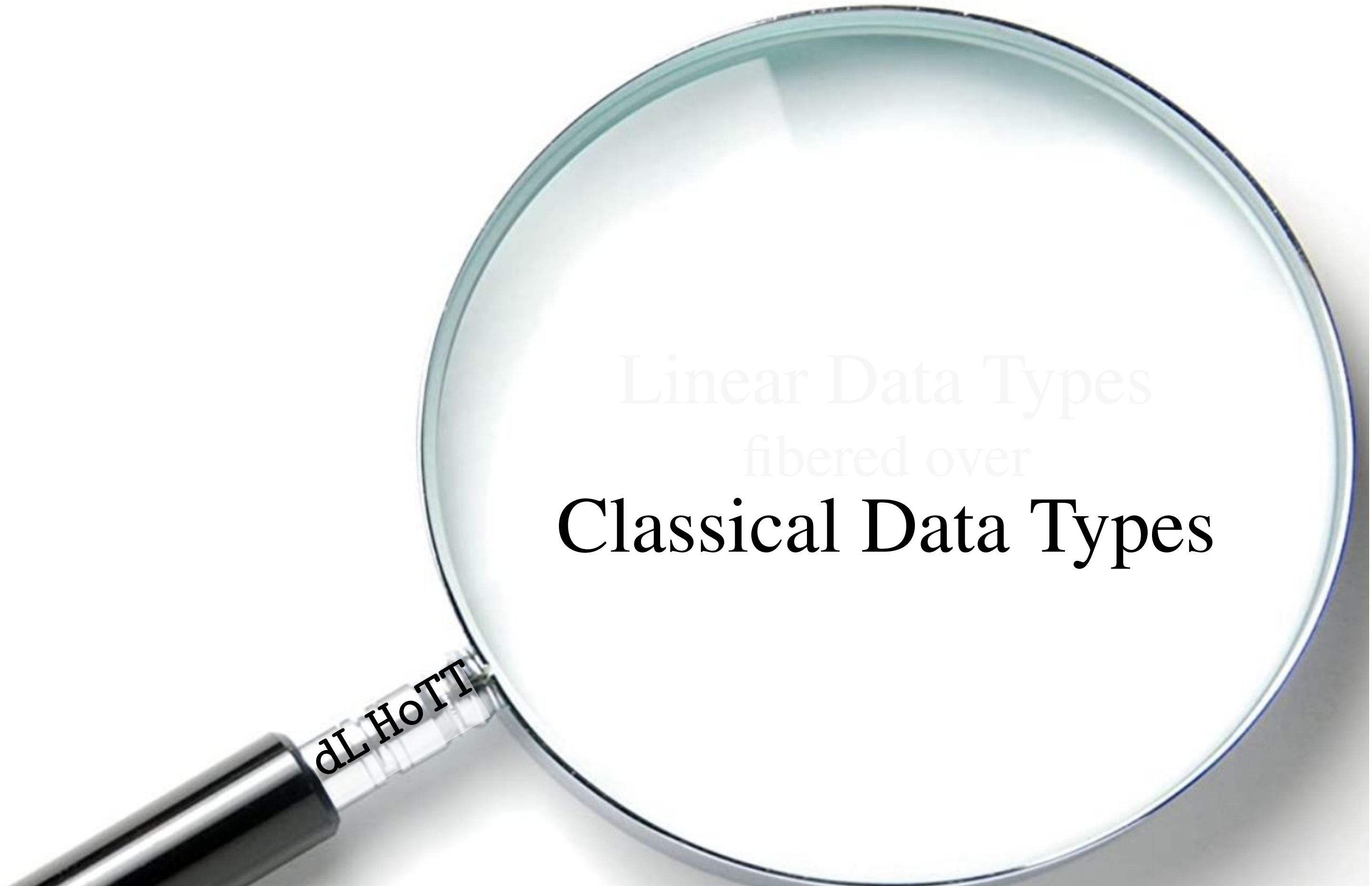
$\exists$ *universal* quantum+classical specification language

conservative over classical *Homotopy Type Theory* (`HoTT`)

and

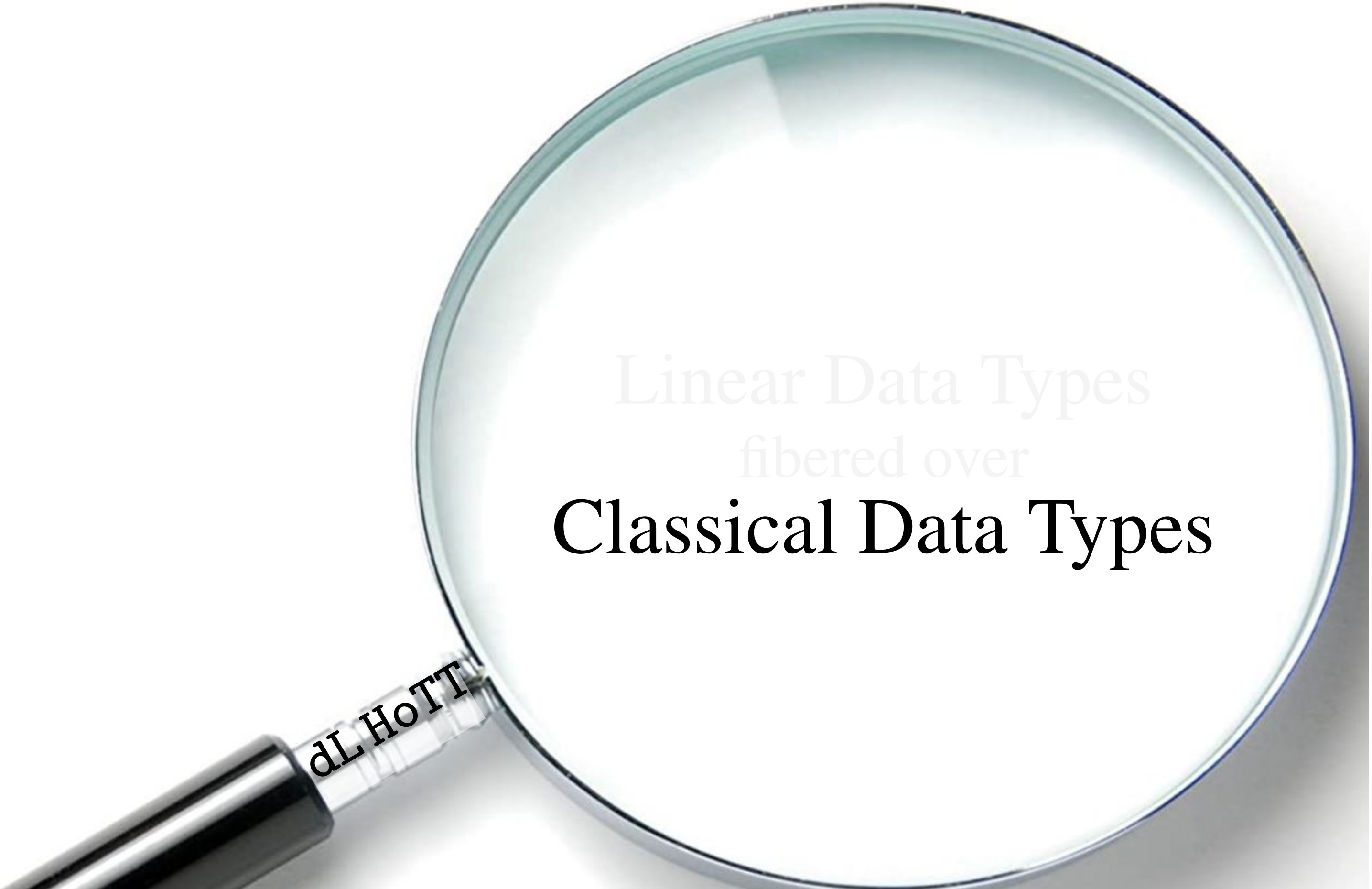
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]

(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)



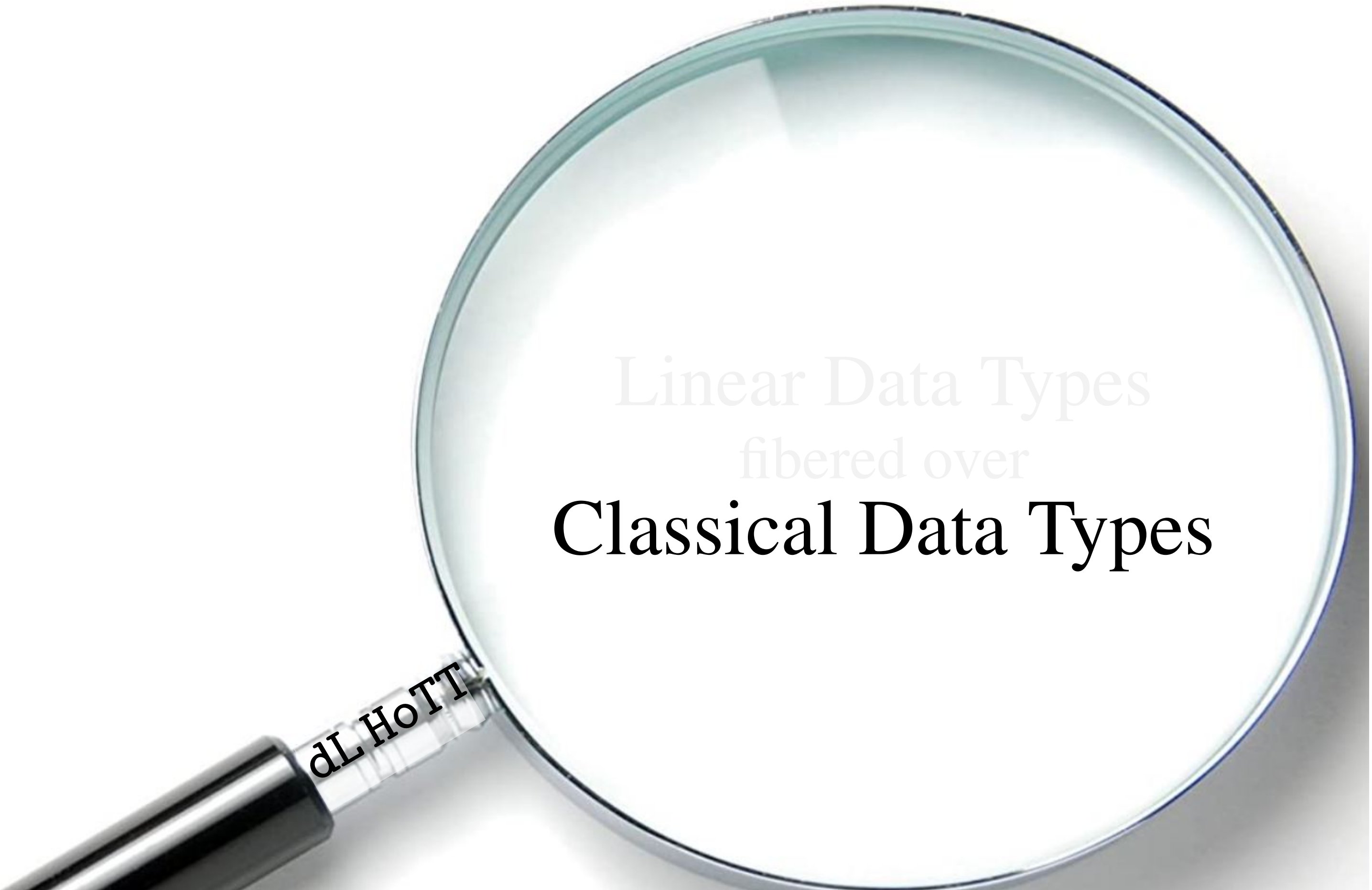
Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
∃ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)



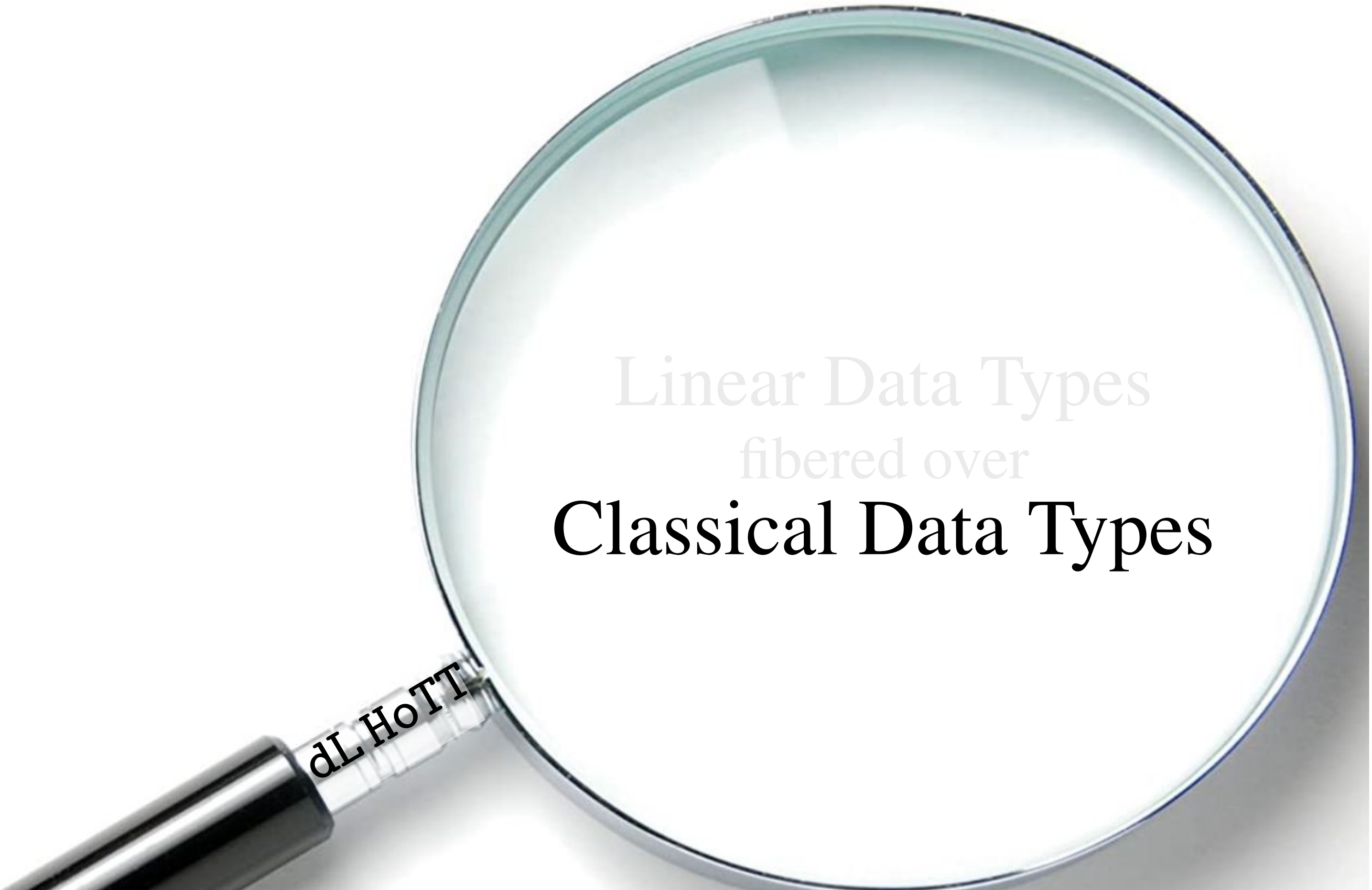
Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
∃ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)



Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
∃ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme “Motivic Yoga” [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck’s six operations *à la* Wirthmüller — more on all this below)

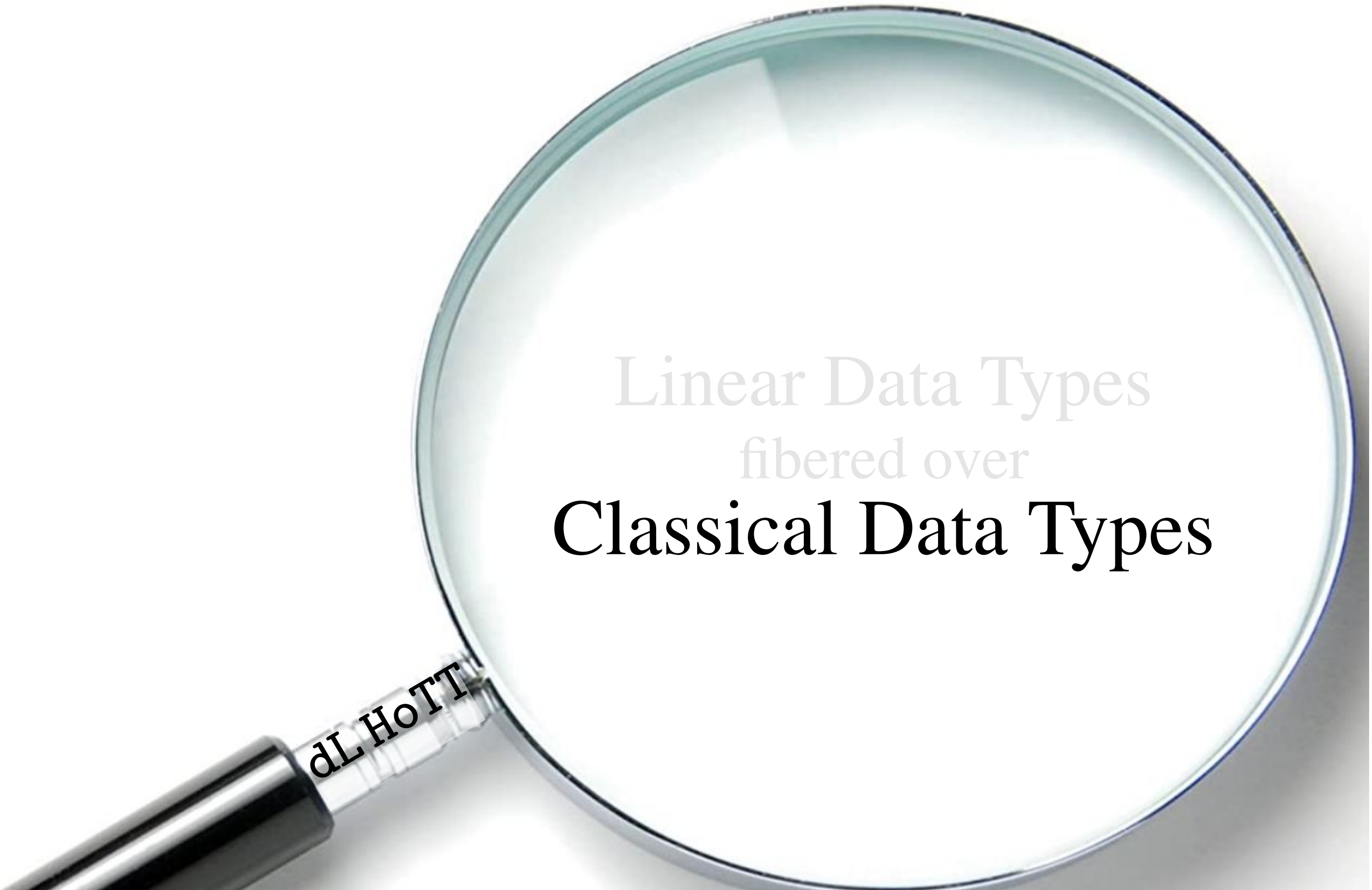


Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:

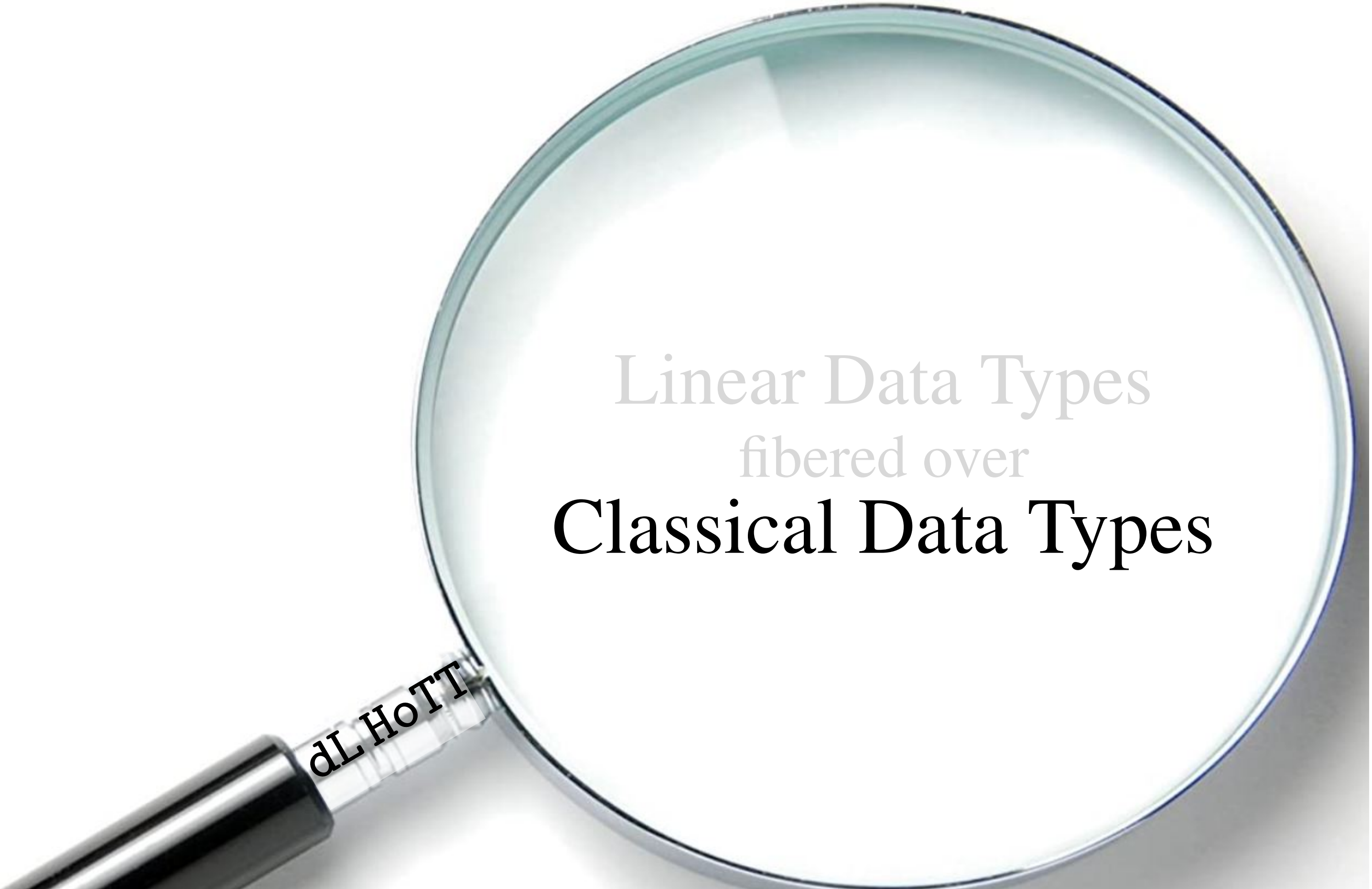
$\exists$ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]

(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)



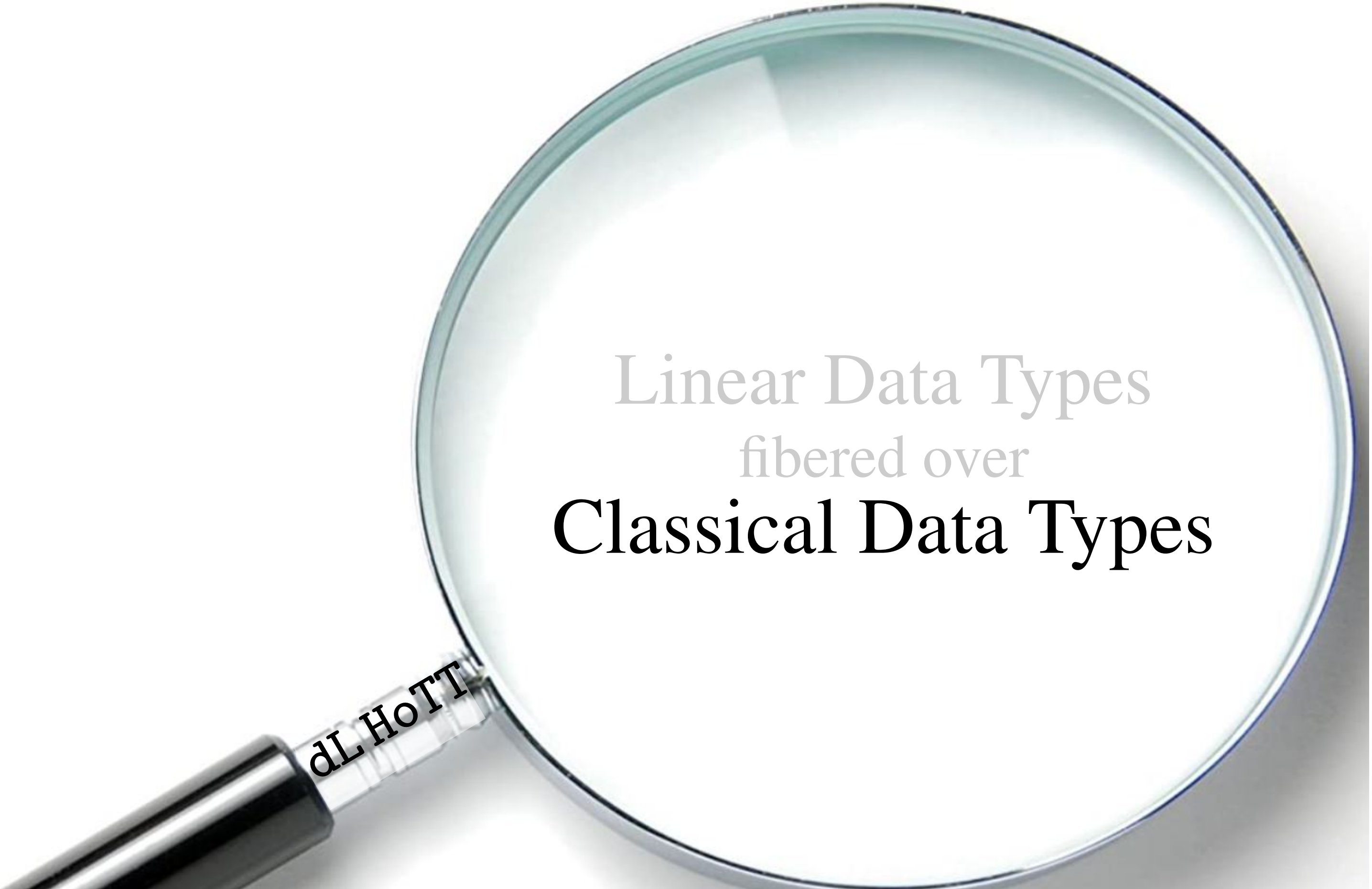
Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
$\exists$ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)



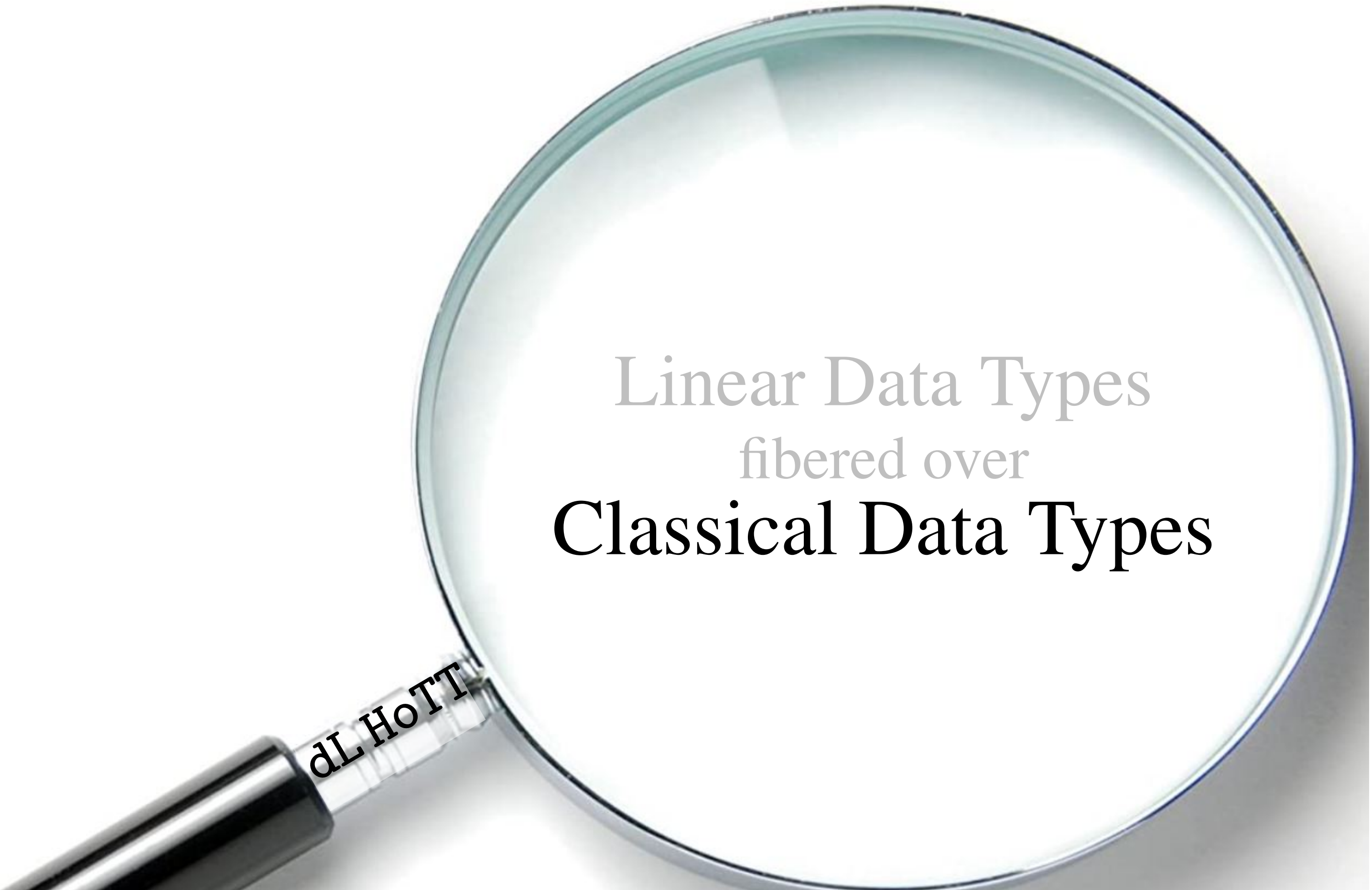
Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
$\exists$ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)



Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
$\exists$ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)



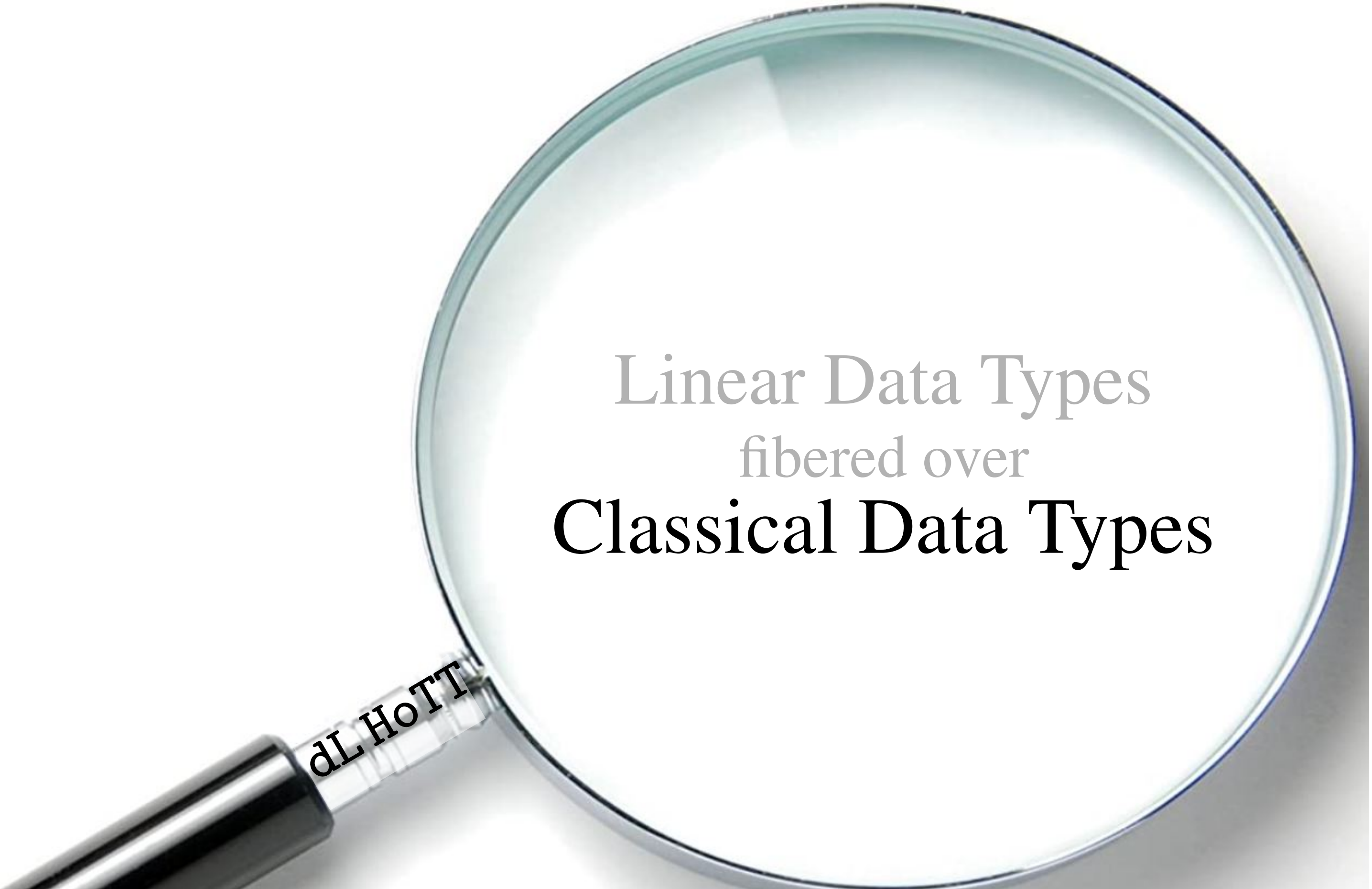
Linear Data Types
fibered over
Classical Data Types

Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:

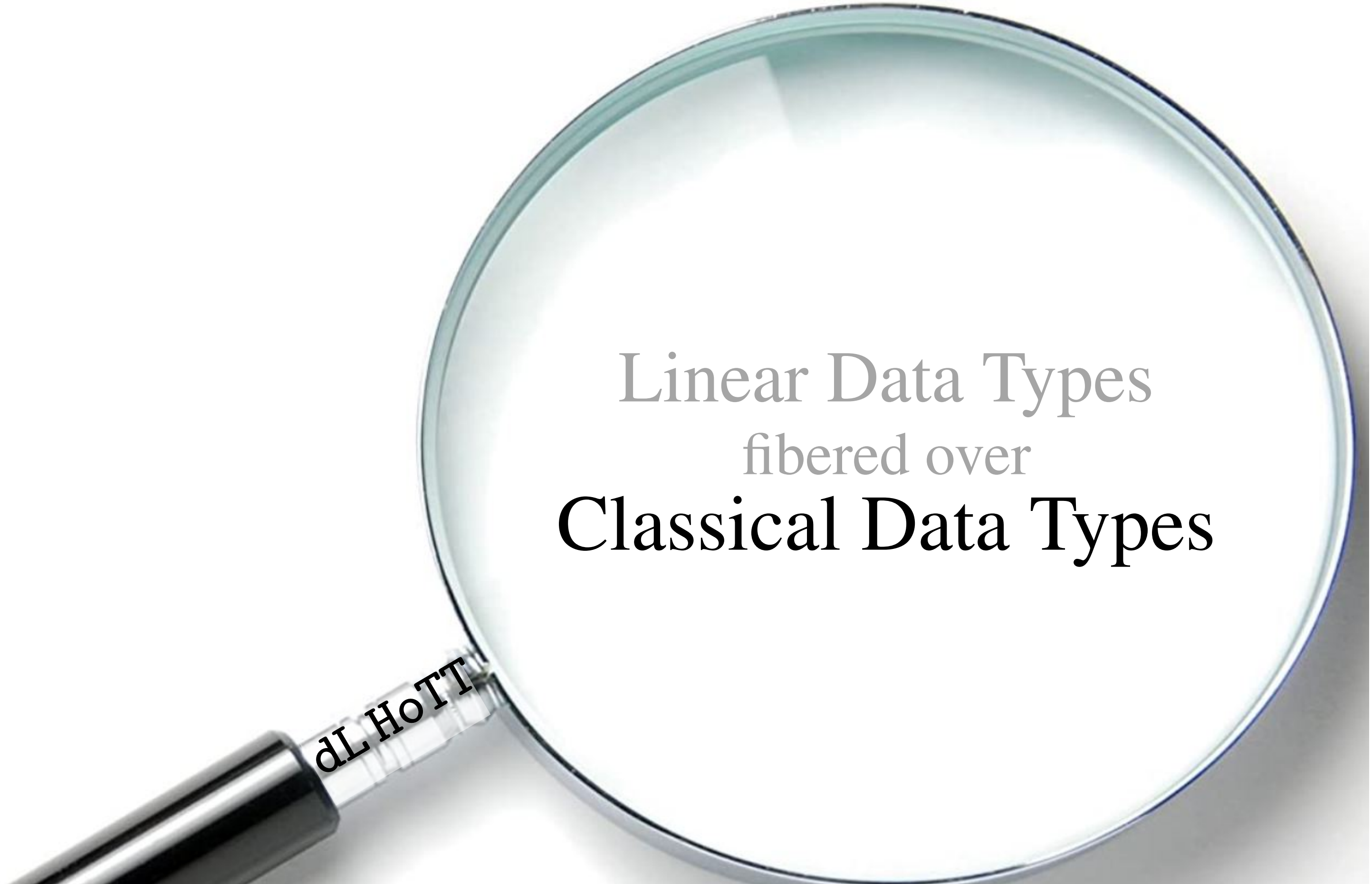
$\exists$ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]

(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)



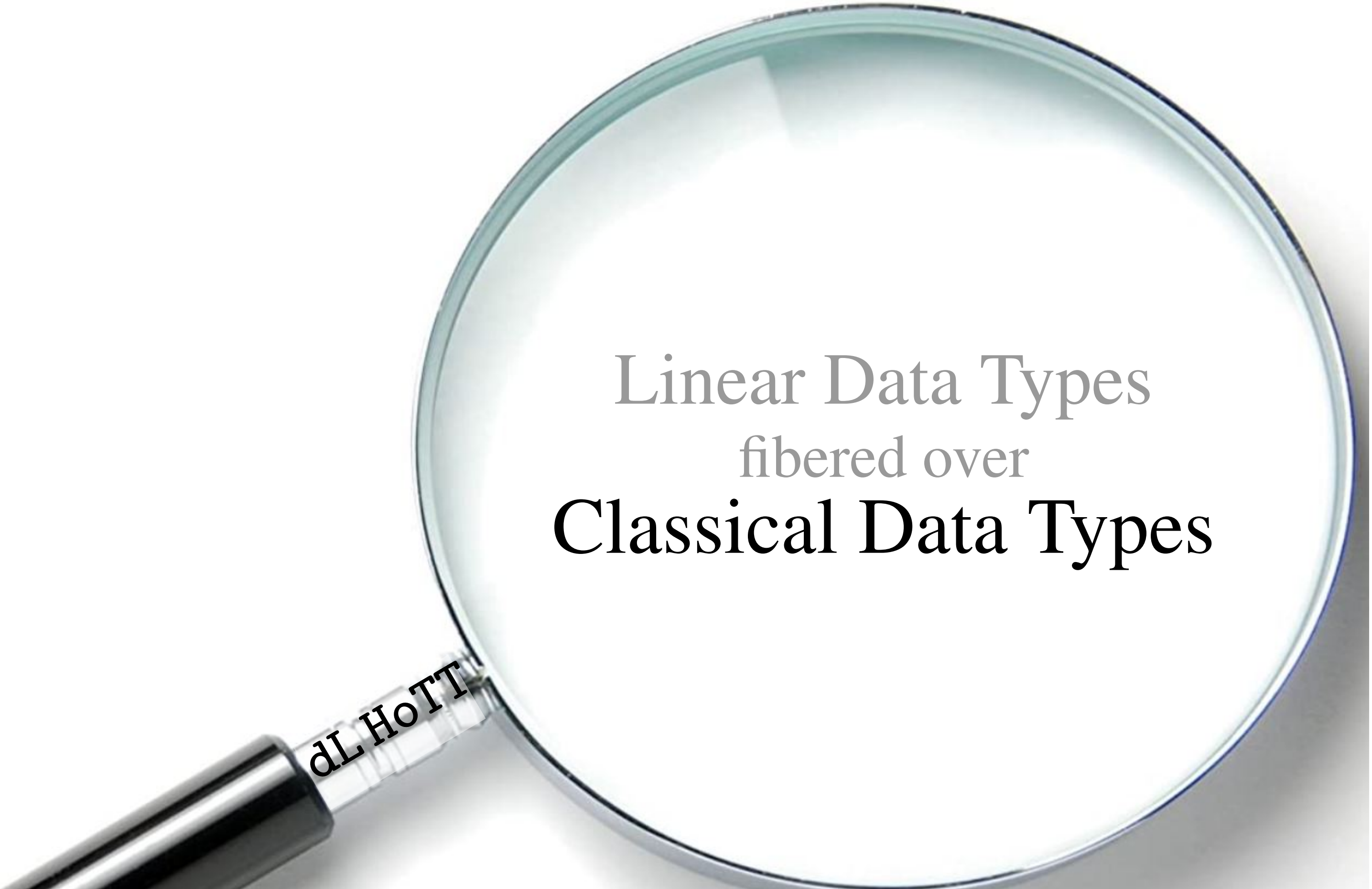
Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
∃ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)



Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
∃ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)

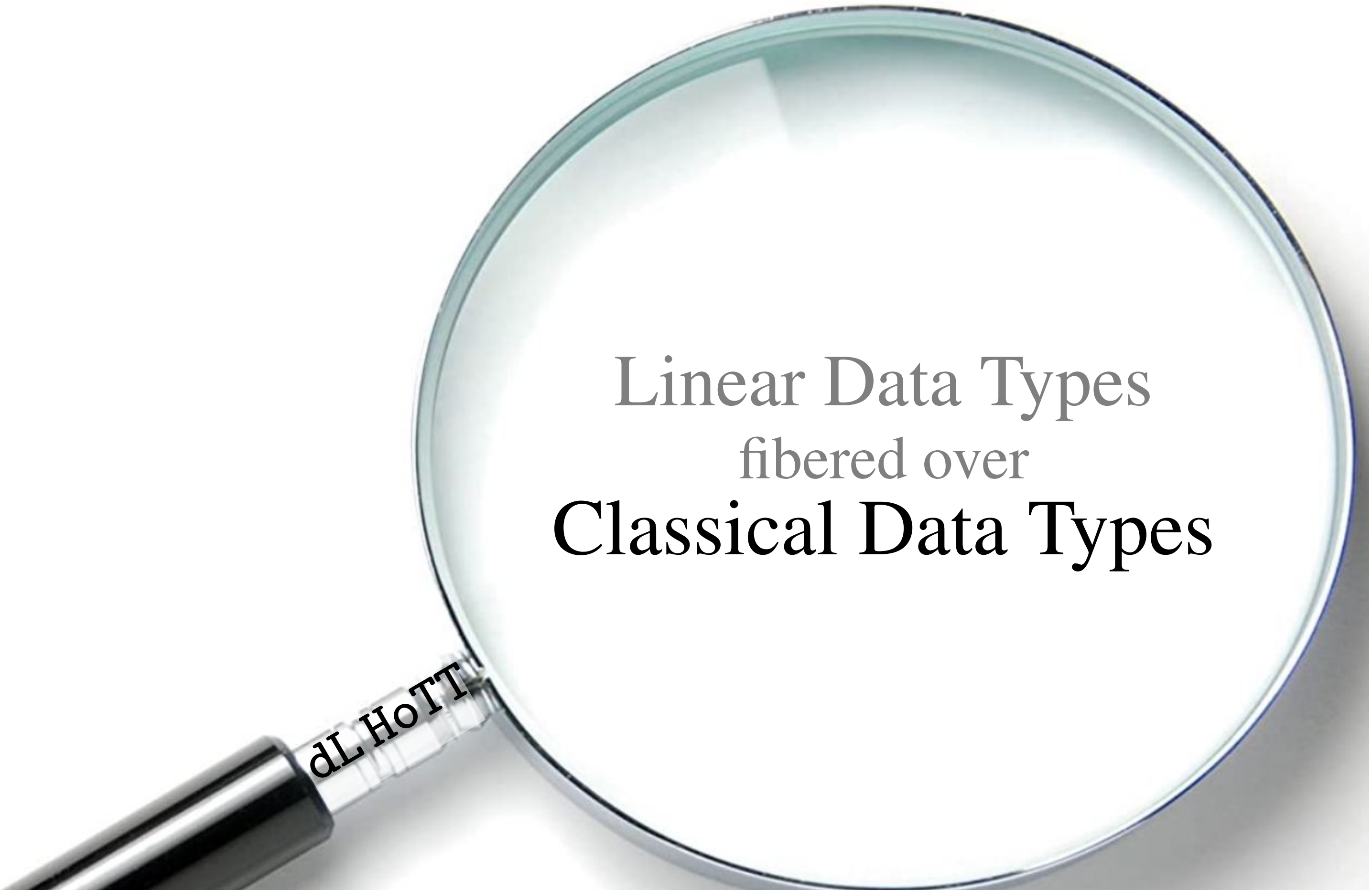


Linear Data Types
fibered over
Classical Data Types

Dependent Linear Homotopy Type Theory (`dLHoTT`)

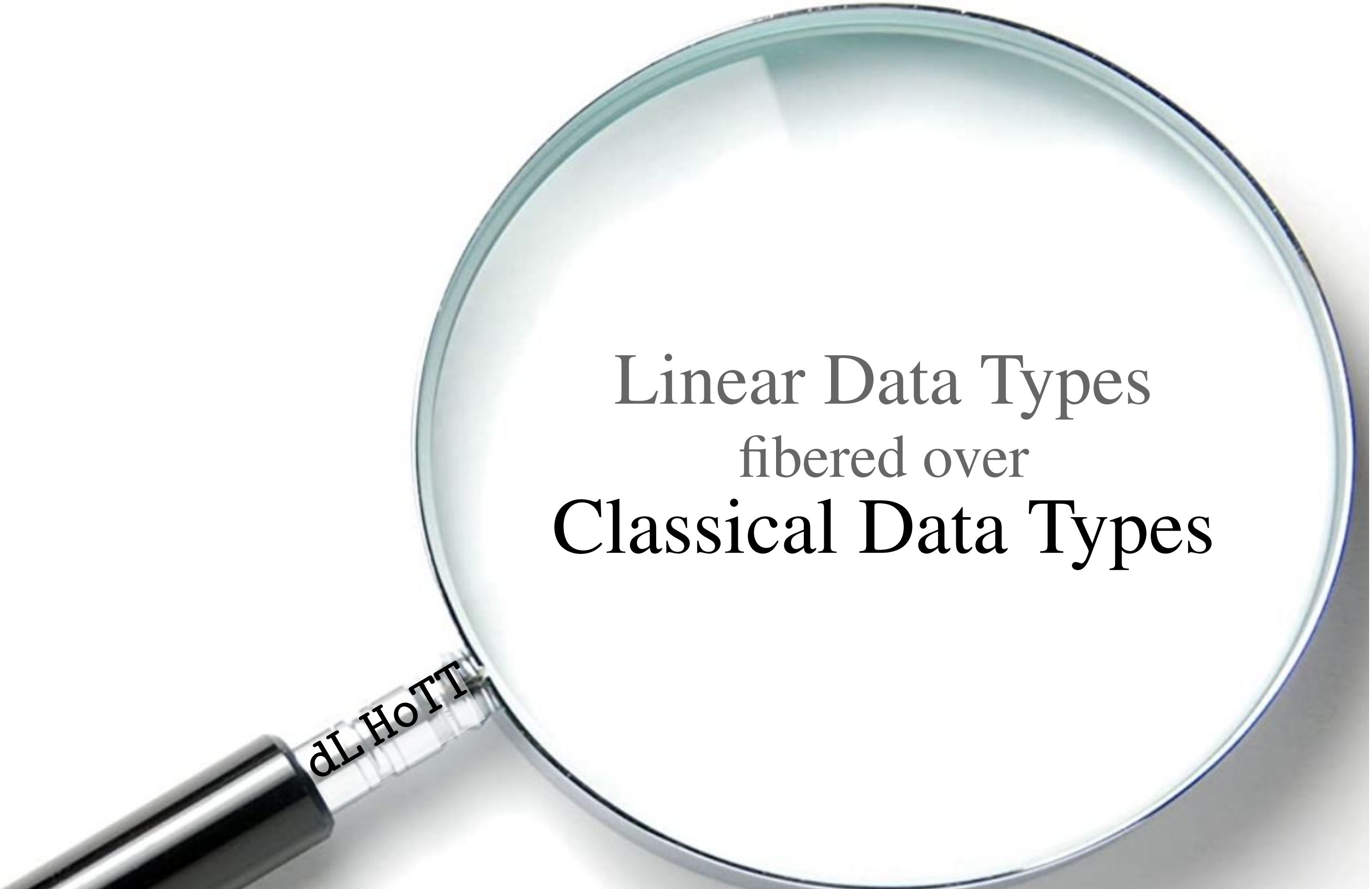
Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:

$\exists$ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)



Dependent Linear Homotopy Type Theory (`dLHoTT`)

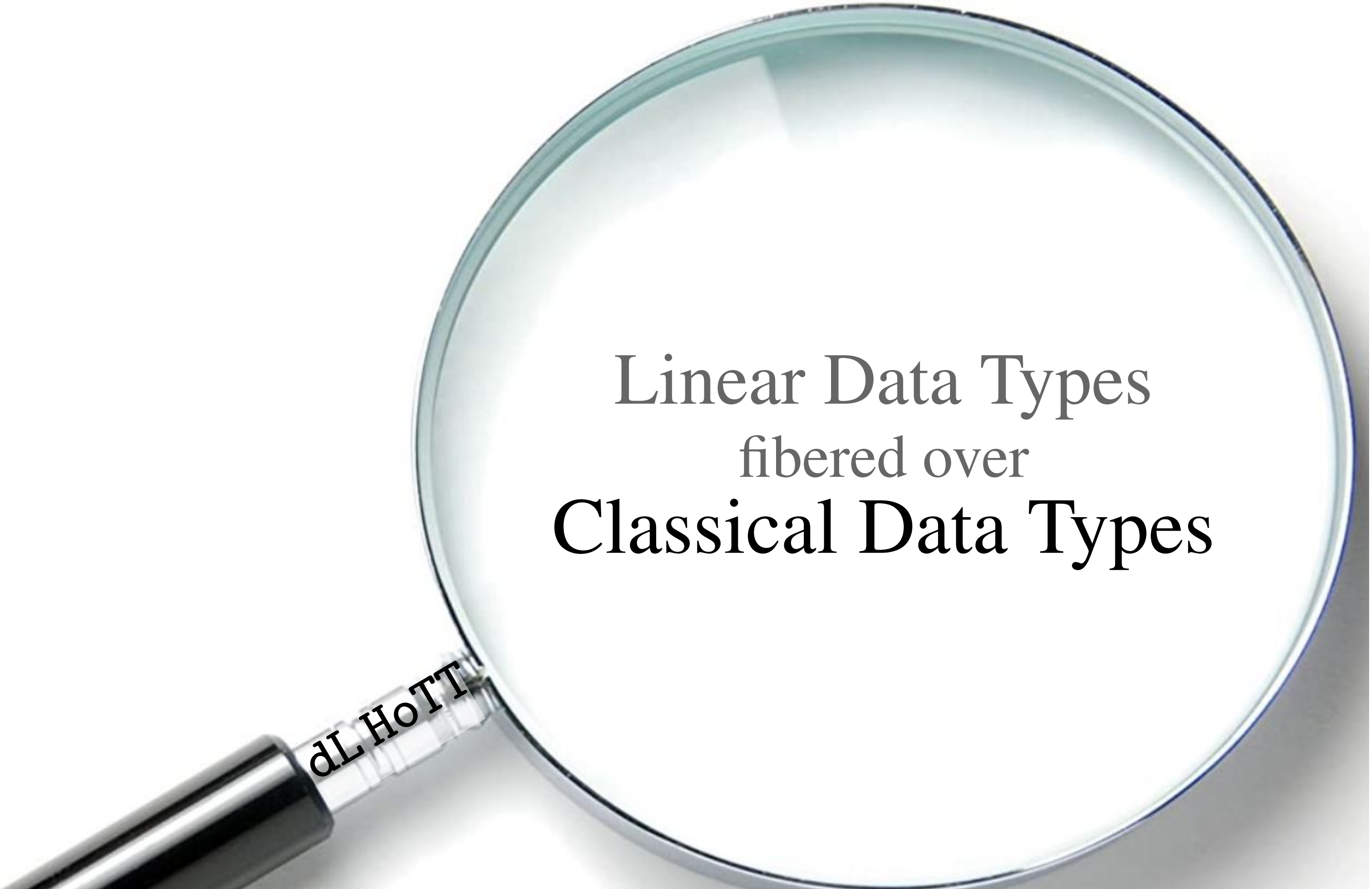
Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
$\exists$ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)



Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
∃ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)

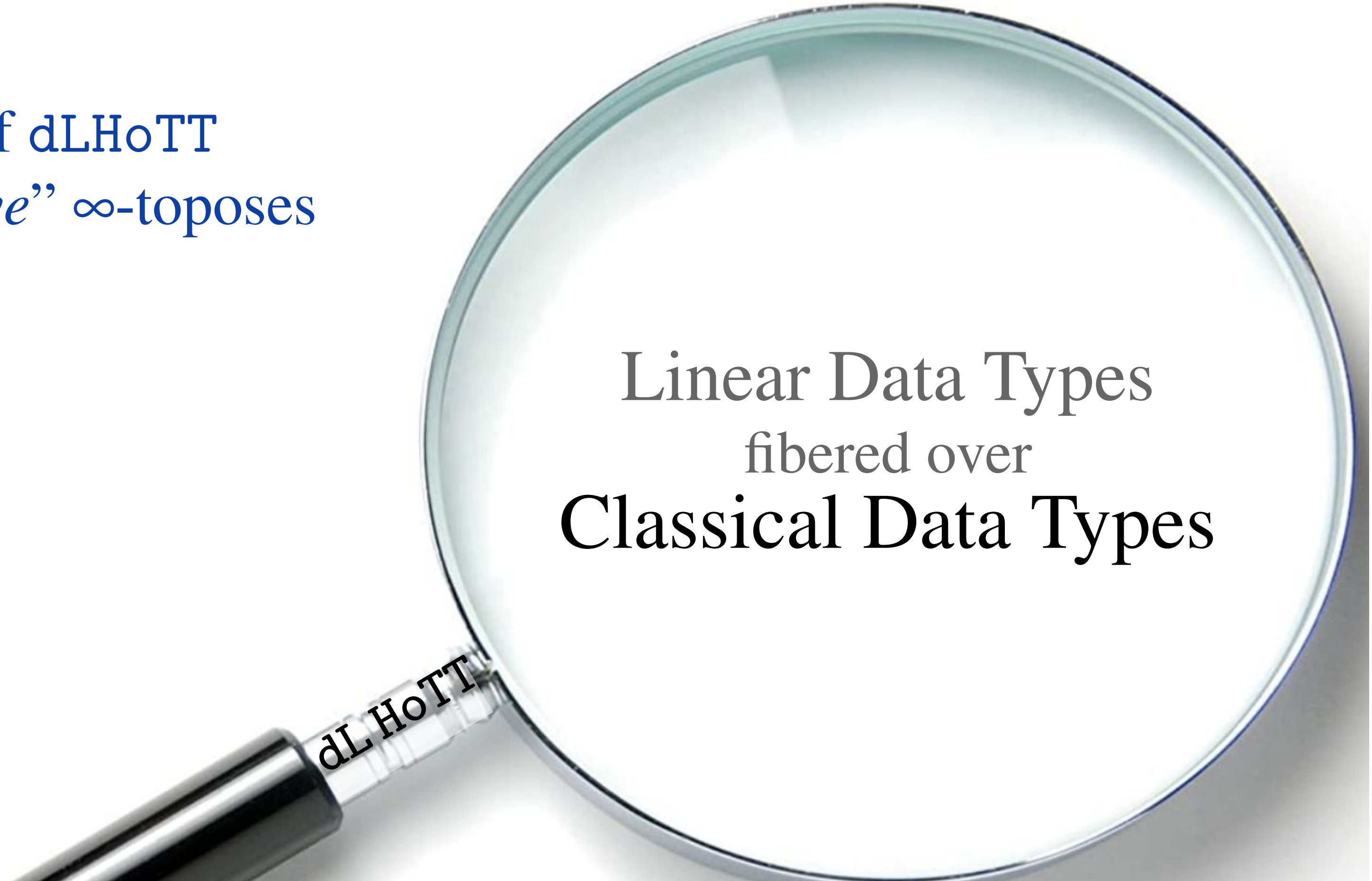
More technically:



Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
$\exists$ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)

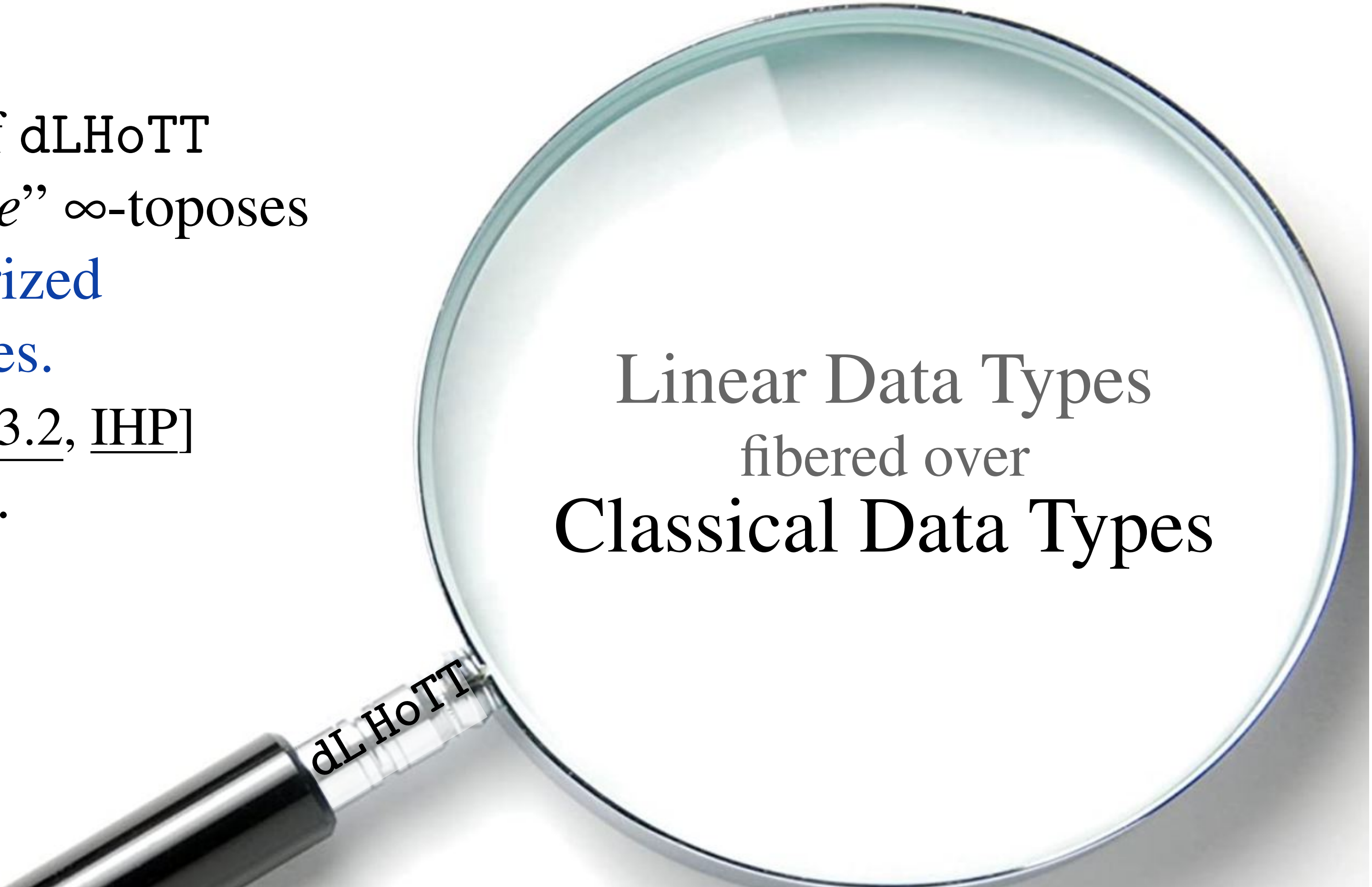
More technically:
The categorical semantics of `dLHoTT`
is in "*infinitesimally cohesive*" ∞-toposes



Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
$\exists$ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)

More technically:
The categorical semantics of `dLHoTT`
is in "*infinitesimally cohesive*" ∞-toposes
of module spectra parameterized
over classical homotopy types.
[S. (2013), §4.1.2] [S. (2014), §3.2, IHP]
[Riley, Finster & Licata (2021)].



Dependent Linear Homotopy Type Theory (`dLHoTT`)

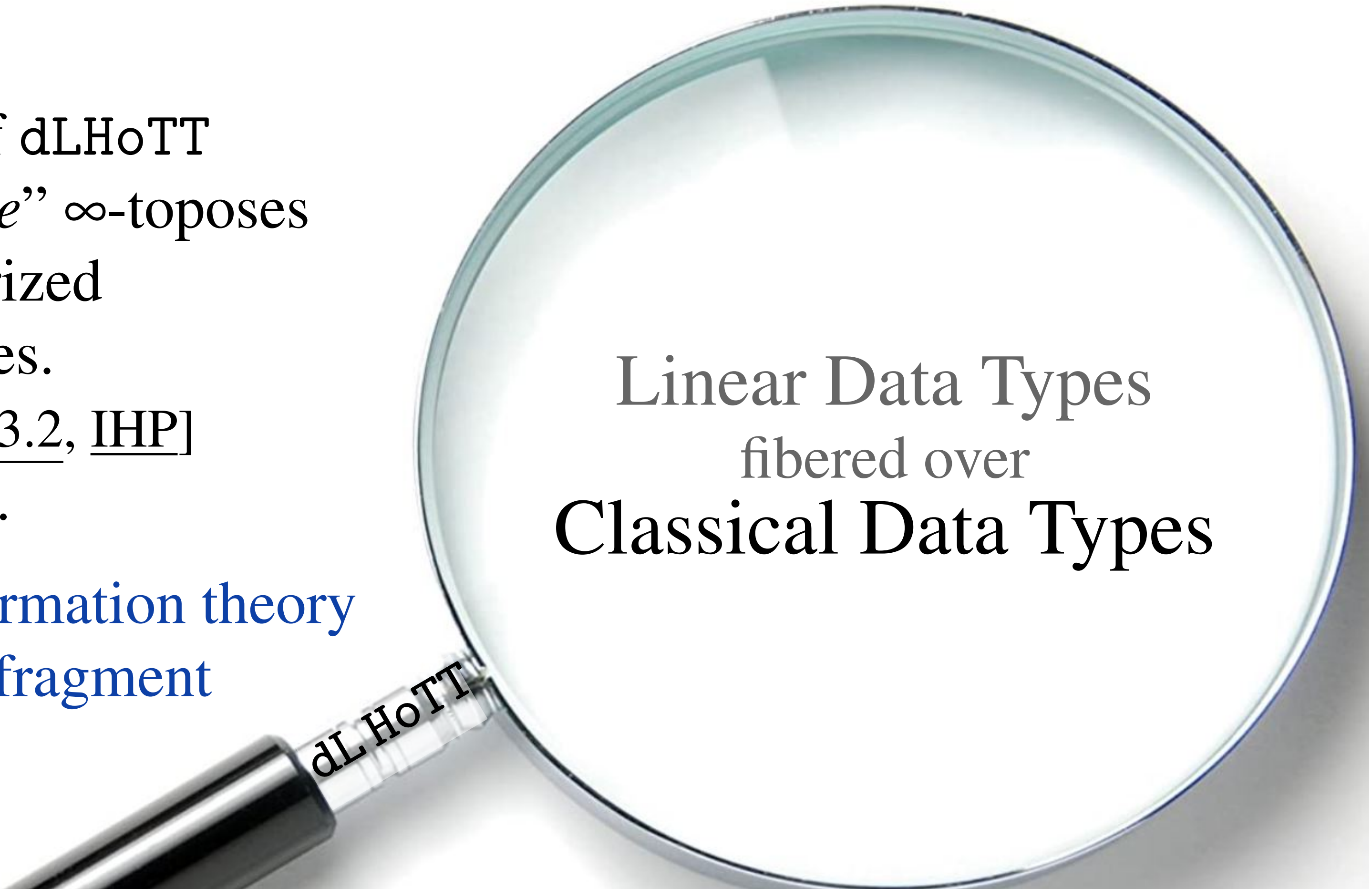
Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:

$\exists$ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)

More technically:

The categorical semantics of `dLHoTT`
is in "*infinitesimally cohesive*" ∞-toposes
of module spectra parameterized
over classical homotopy types.
[S. (2013), §4.1.2] [S. (2014), §3.2, IHP]
[Riley, Finster & Licata (2021)].

For traditional quantum information theory
this faithfully subsumes the fragment



Dependent Linear Homotopy Type Theory (`dLHoTT`)

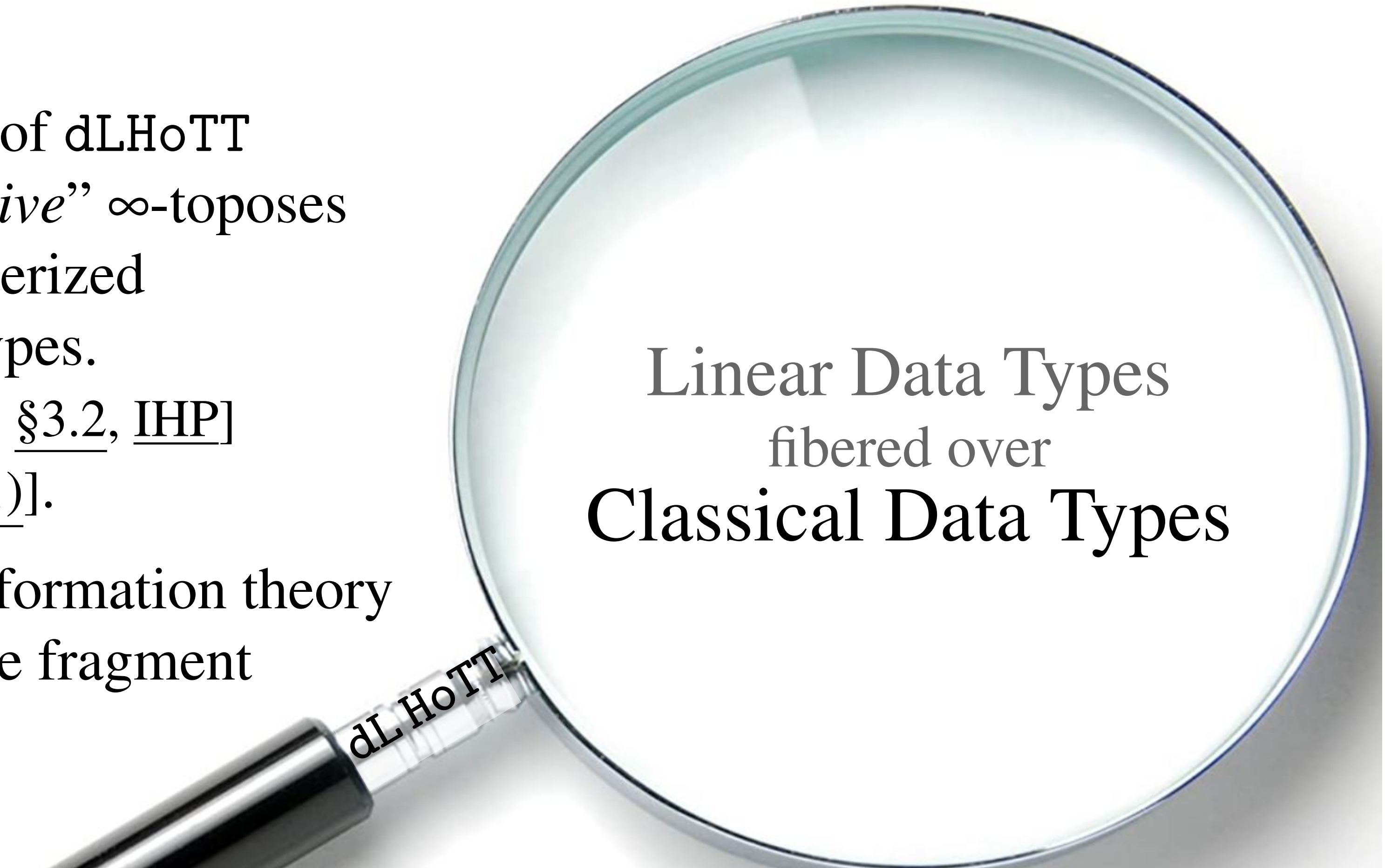
Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:

∃ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)

More technically:

The categorical semantics of `dLHoTT`
is in "*infinitesimally cohesive*" ∞-toposes
of module spectra parameterized
over classical homotopy types.
[S. (2013), §4.1.2] [S. (2014), §3.2, IHP]
[Riley, Finster & Licata (2021)].

For traditional quantum information theory
this faithfully subsumes the fragment
of complex vector bundles
over finite sets.



Dependent Linear Homotopy Type Theory (`dLHoTT`)

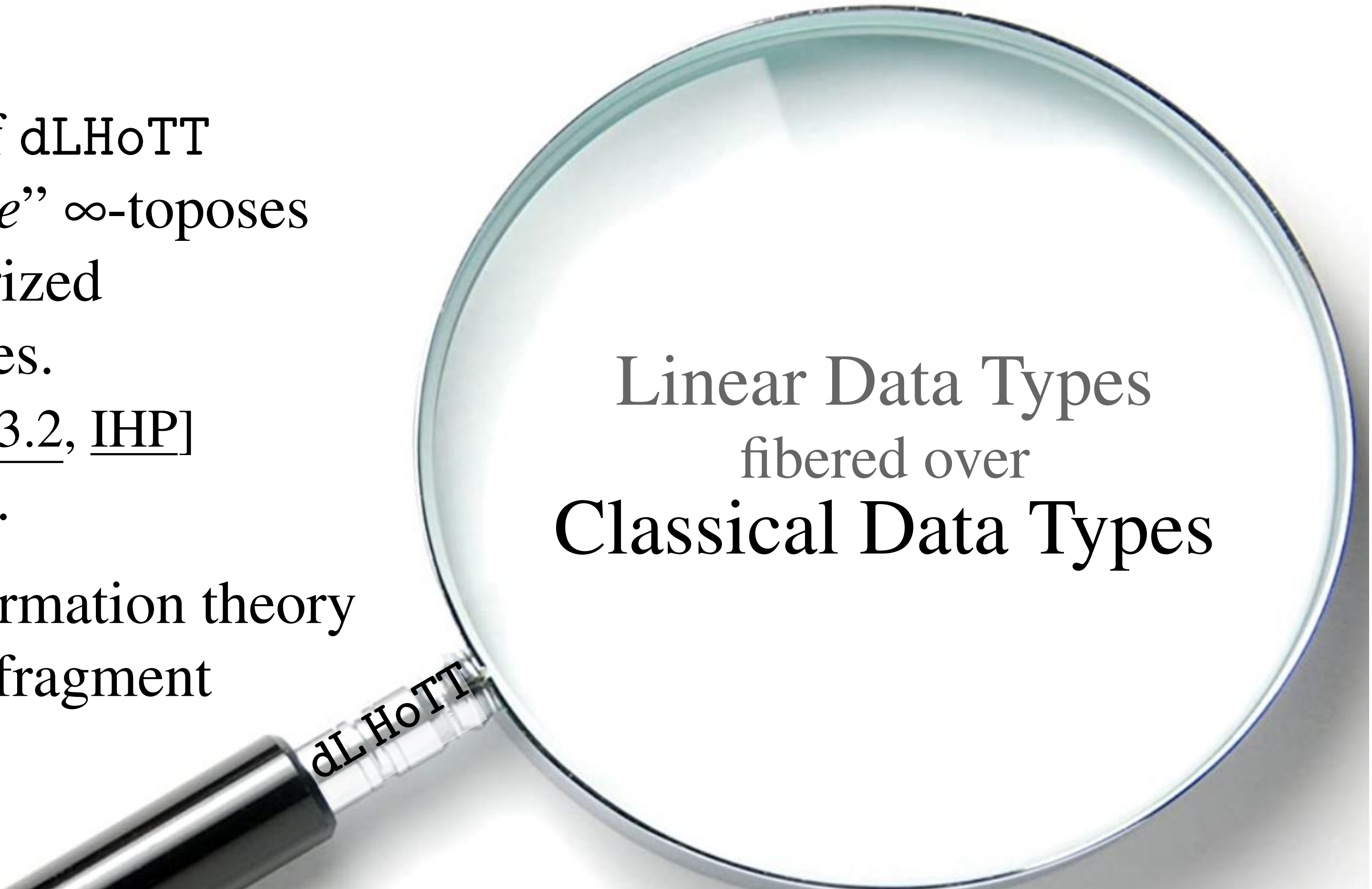
Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:

∃ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)

More technically:

The categorical semantics of `dLHoTT`
is in "*infinitesimally cohesive*" ∞-toposes
of module spectra parameterized
over classical homotopy types.
[S. (2013), §4.1.2] [S. (2014), §3.2, IHP]
[Riley, Finster & Licata (2021)].

For traditional quantum information theory
this faithfully subsumes the fragment
of complex vector bundles
over finite sets.



Linear Data Types
fibered over
Classical Data Types

Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
$\exists$ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)

Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
$\exists$ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)

Theorem [CQTS (2022)]:

Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
∃ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)

Theorem [CQTS (2022)]:
Motivic Yoga induces a system of monadic computational effects
constituting
linear modalities of actuality and potentiality

Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
$\exists$ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)

Theorem [CQTS (2022)]:
Motivic Yoga induces a system of monadic computational effects
constituting
linear modalities of actuality and potentiality
which happen to
know all about quantum information theory:

Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
∃ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)

Theorem [CQTS (2022)]:
Motivic Yoga induces a system of monadic computational effects
constituting
linear modalities of actuality and potentiality
which happen to
know all about quantum information theory:

quantum measurement is handling of linear indefiniteness effects

Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:

∃ *universal* quantum+classical specification language

conservative over classical *Homotopy Type Theory* (`HoTT`)

and

verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]

(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)

Theorem [CQTS (2022)]:

Motivic Yoga induces a system of monadic computational effects

constituting

linear modalities of actuality and potentiality

which happen to

know all about quantum information theory:

quantum measurement	is	handling of linear indefiniteness effects
quantum state preparation	is	handling of linear randomness co-effects

Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
∃ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)

Theorem [CQTS (2022)]:
Motivic Yoga induces a system of monadic computational effects
constituting
linear modalities of actuality and potentiality
which happen to
know all about quantum information theory:

quantum measurement	is	handling of linear indefiniteness effects
quantum state preparation	is	handling of linear randomness co-effects
quantum+classical circuits	are	the effectful string diagrams

Dependent Linear Homotopy Type Theory (`dLHoTT`)

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:
∃ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)

Theorem [CQTS (2022)]:
Motivic Yoga induces a system of monadic computational effects
constituting
linear modalities of actuality and potentiality
which happen to
know all about quantum information theory:

quantum measurement	is	handling of linear indefiniteness effects
quantum state preparation	is	handling of linear randomness co-effects
quantum+classical circuits	are	the effectful string diagrams
quantum dynamic lifting	is	comparison functor of monadicity theorem

Theorem [M. Riley (2022), `doi:10.14418/wes01.3.139`]:

$\exists$ *universal* quantum+classical specification language
conservative over classical *Homotopy Type Theory* (`HoTT`)
and
verifying axiom scheme "Motivic Yoga" [Riley, §2.4, anticipated in S. (2014), §3.2]
(i.e. Grothendieck's six operations *à la* Wirthmüller — more on all this below)

Theorem [CQTS (2022)]:

Motivic Yoga induces a system of monadic computational effects
constituting
linear modalities of actuality and potentiality
which happen to
know all about quantum information theory:

quantum measurement	is	handling of linear indefiniteness effects
quantum state preparation	is	handling of linear randomness co-effects
quantum+classical circuits	are	the effectful string diagrams
quantum dynamic lifting	is	comparison functor of monadicity theorem

`QS` – Quantum Systems language @ CQTS

$\rightsquigarrow$ full-blown Quantum Systems language emerges embedded in `dLHoTT`

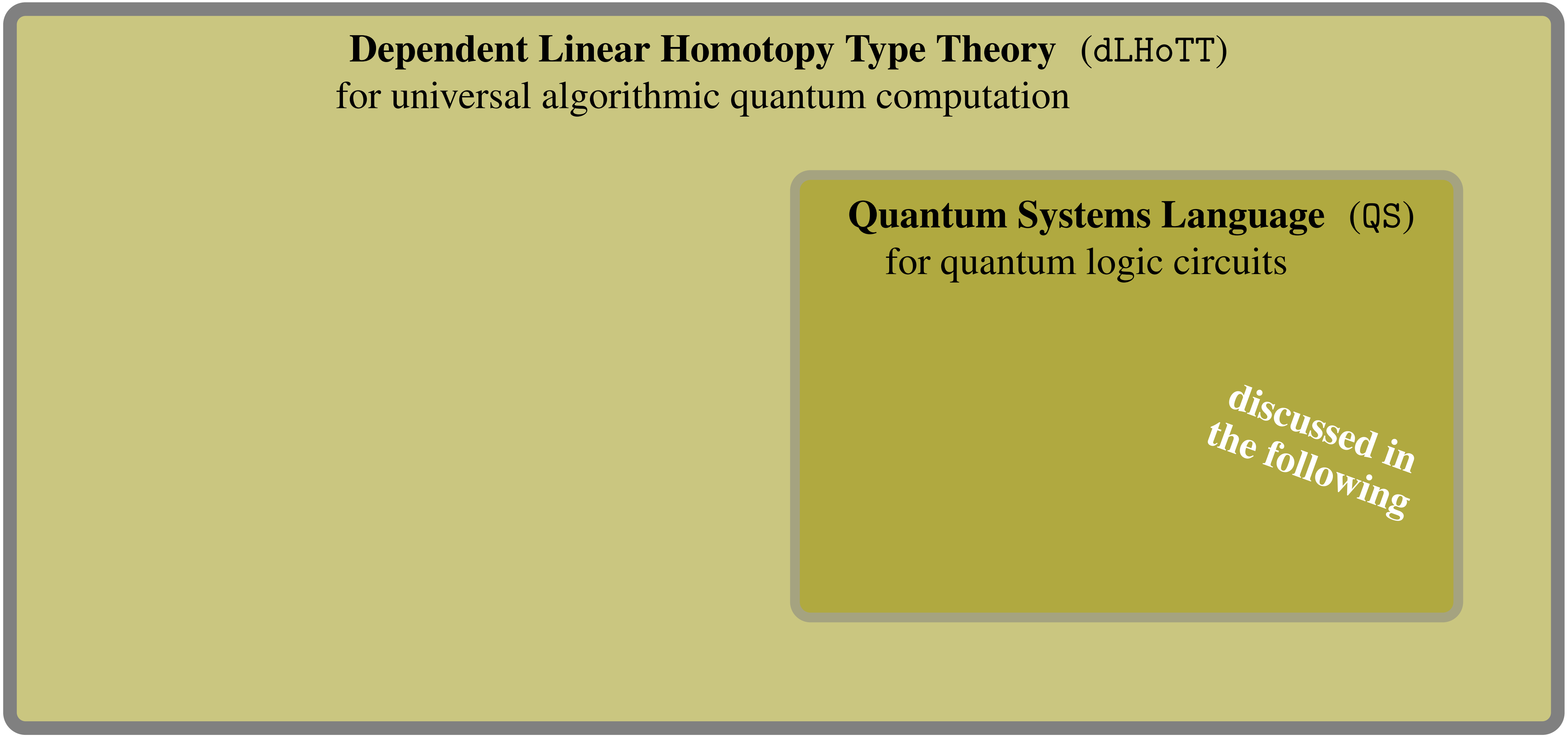
`QS` – Quantum Systems language @ CQTS

$\rightsquigarrow$ full-blown Quantum Systems language emerges embedded in `dLHoTT`

Dependent Linear Homotopy Type Theory (`dLHoTT`)
for universal algorithmic quantum computation

QS – Quantum Systems language @ CQTS

⇝ full-blown Quantum Systems language emerges embedded in dLHoTT



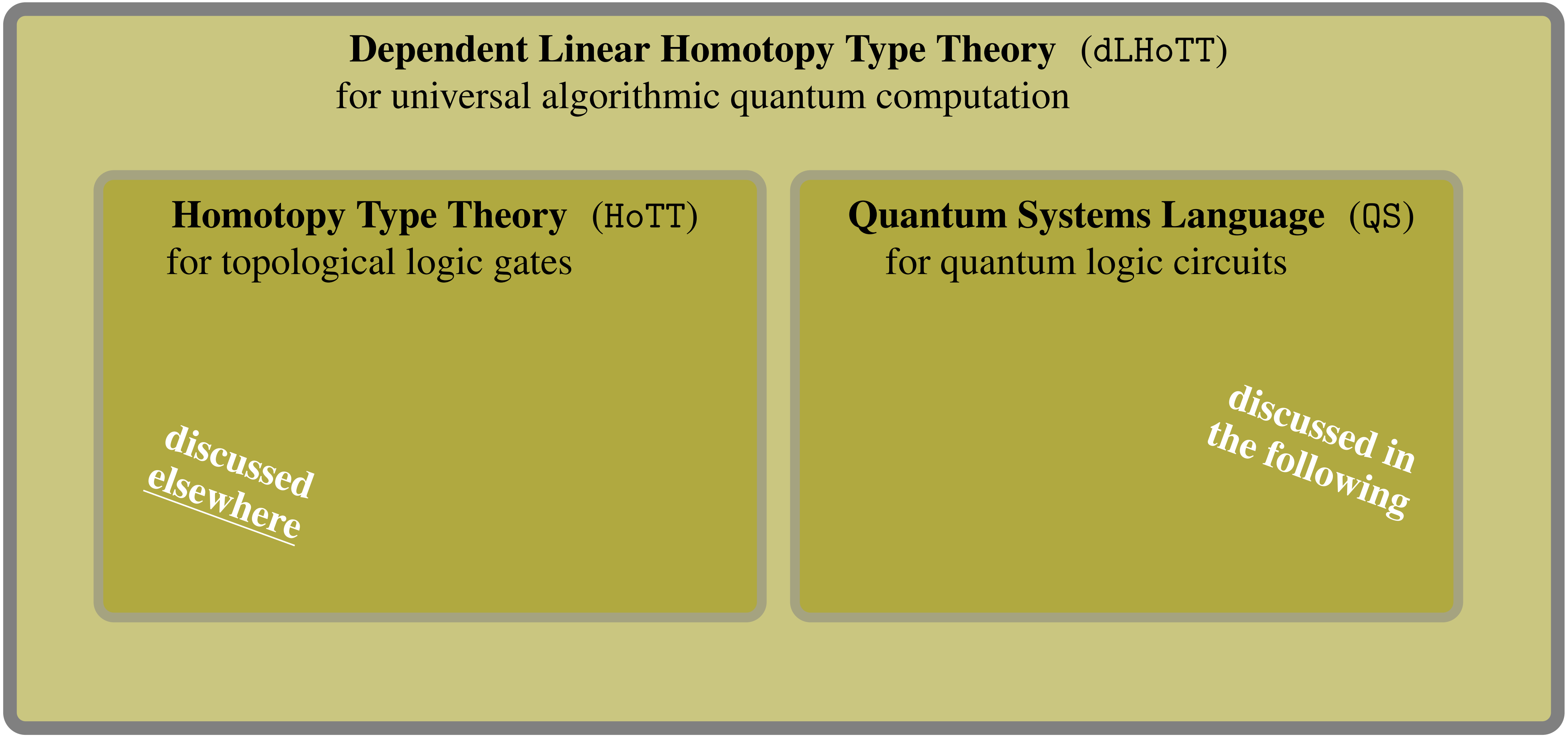
Dependent Linear Homotopy Type Theory (dLHoTT)
for universal algorithmic quantum computation

Quantum Systems Language (QS)
for quantum logic circuits

discussed in the following

`QS` – Quantum Systems language @ CQTS

⇝ full-blown Quantum Systems language emerges embedded in `dLHoTT`



Dependent Linear Homotopy Type Theory (`dLHoTT`)
for universal algorithmic quantum computation

Homotopy Type Theory (`HoTT`)
for topological logic gates

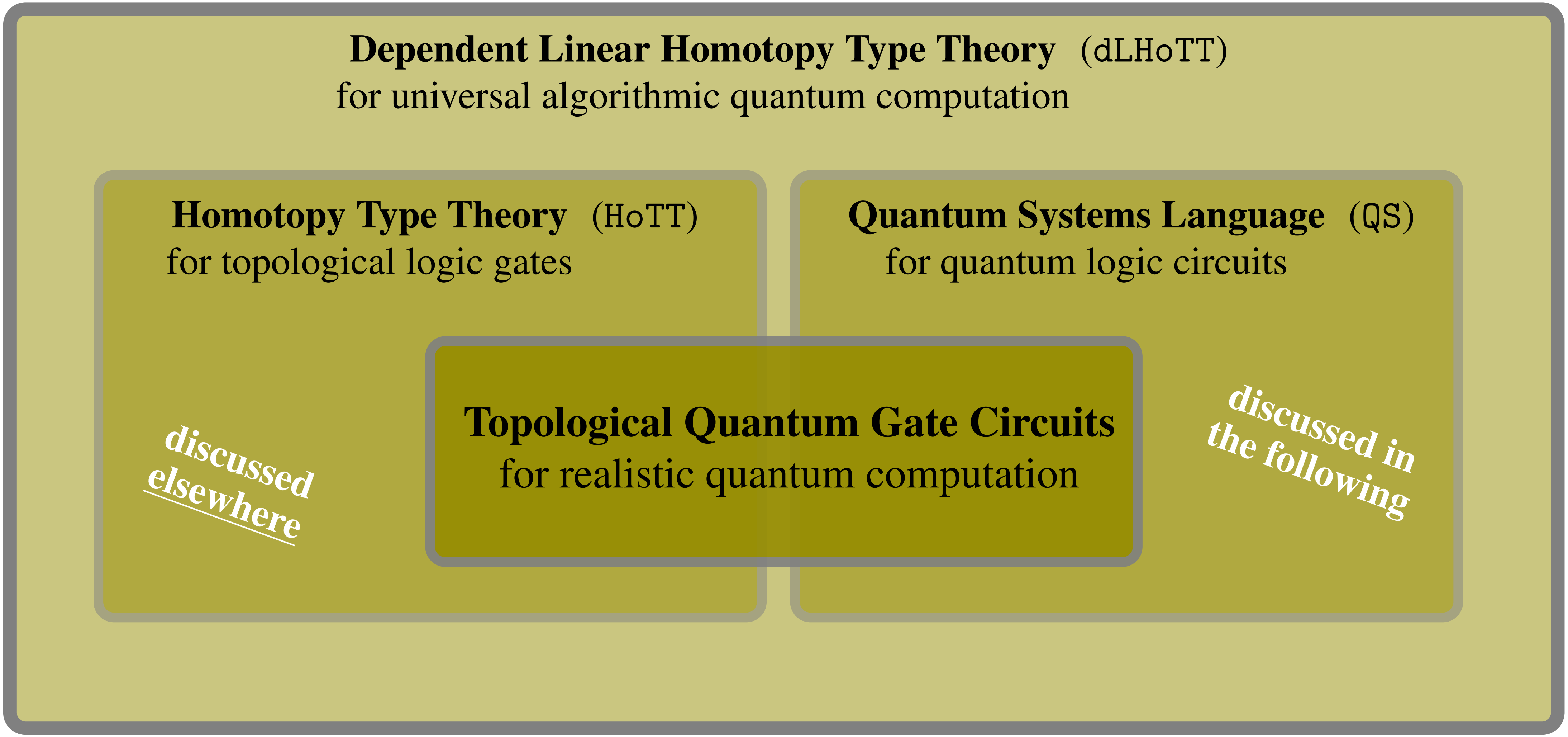
discussed elsewhere

Quantum Systems Language (`QS`)
for quantum logic circuits

discussed in the following

`QS` – Quantum Systems language @ CQTS

⇝ full-blown Quantum Systems language emerges embedded in `dLHoTT`



Dependent Linear Homotopy Type Theory (`dLHoTT`)
for universal algorithmic quantum computation

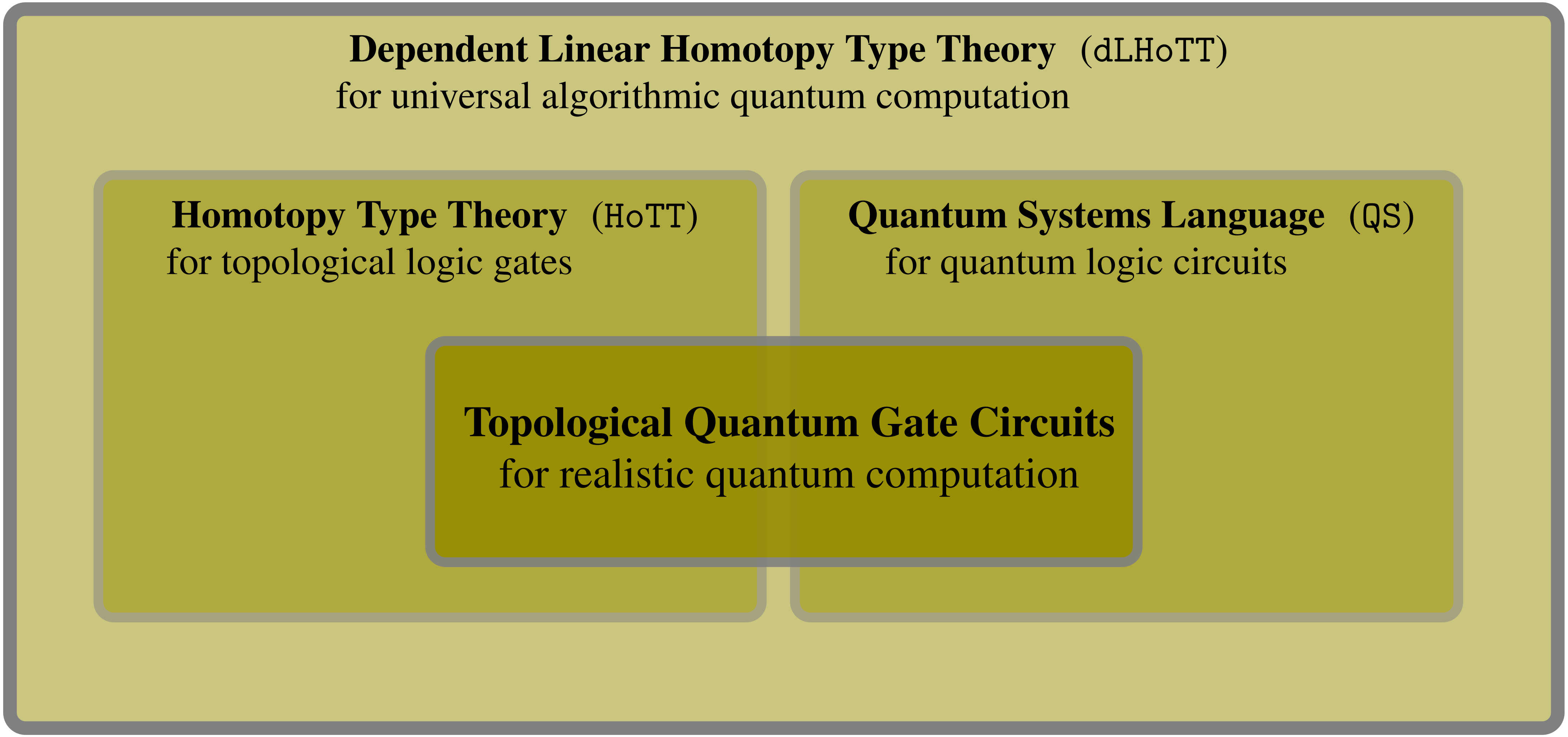
Homotopy Type Theory (`HoTT`)
for topological logic gates

Quantum Systems Language (`QS`)
for quantum logic circuits

Topological Quantum Gate Circuits
for realistic quantum computation

QS – Quantum Systems language @ CQTS

⇝ full-blown Quantum Systems language emerges embedded in dLHoTT



Dependent Linear Homotopy Type Theory (dLHoTT)
for universal algorithmic quantum computation

Homotopy Type Theory (HoTT)
for topological logic gates

Quantum Systems Language (QS)
for quantum logic circuits

Topological Quantum Gate Circuits
for realistic quantum computation

ambient dLHoTT	verifies	classically dependent quantum linear types
ambient HoTT	provides	specification of topological quantum gates
ambient dTT	provides	full verified classical control

Quantum Data Types

Linear/Quantum Data Types

Characteristic Property			
Symbol			
Formula (for B : FinType)			
AlgTop Jargon			
Linear Logic			
Physics Meaning			

Linear/Quantum Data Types

<table>
<tr><td>Characteristic Property</td><td>1. their cartesian product blends into the co-product:</td><td></td><td></td></tr>
<tr><td>Symbol</td><td></td><td></td><td></td></tr>
<tr><td>Formula
(for B : FinType)</td><td></td><td></td><td></td></tr>
<tr><td rowspan="2">AlgTop Jargon</td><td rowspan="2"></td><td></td><td></td></tr>
<tr><td colspan="2"></td></tr>
<tr><td>Linear Logic</td><td></td><td></td><td></td></tr>
<tr><td>Physics Meaning</td><td></td><td></td><td></td></tr>
</table>

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:		
Symbol	$\oplus$ direct sum		
Formula (for B : FinType)			
AlgTop Jargon			
Linear Logic			
Physics Meaning			

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	
Symbol	$\oplus$ direct sum		
Formula (for B : FinType)			
AlgTop Jargon			
Linear Logic			
Physics Meaning			

Linear/Quantum Data Types

<table>
<tr><td>Characteristic Property</td><td>1. their cartesian product blends into the co-product:</td><td>2. a tensor product appears & distributes over direct sum</td><td></td></tr>
<tr><td>Symbol</td><td>$\oplus$ direct sum</td><td>$\otimes$ tensor product</td><td></td></tr>
<tr><td>Formula
(for B : FinType)</td><td></td><td></td><td></td></tr>
<tr><td rowspan="2">AlgTop Jargon</td><td rowspan="2"></td><td></td><td></td></tr>
<tr><td colspan="2"></td></tr>
<tr><td>Linear Logic</td><td></td><td></td><td></td></tr>
<tr><td>Physics Meaning</td><td></td><td></td><td></td></tr>
</table>

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\oplus$ direct sum	$\otimes$ tensor product	
Formula (for B : FinType)			
AlgTop Jargon			
Linear Logic			
Physics Meaning			

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\oplus$ direct sum	$\otimes$ tensor product	$\multimap$ linear function type
Formula (for B : FinType)			
AlgTop Jargon			
Linear Logic			
Physics Meaning			

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\oplus$ direct sum	$\otimes$ tensor product	$\multimap$ linear function type
Formula (for B : FinType)	$\overset{\textbf{cart. product}}{\prod_B \mathcal{H}_b} \simeq \underset{\textbf{direct sum}}{\bigoplus_B \mathcal{H}_b} \simeq \overset{\textbf{co-product}}{\coprod_B \mathcal{H}_b}$		
AlgTop Jargon			
Linear Logic			
Physics Meaning			

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\oplus$ direct sum	$\otimes$ tensor product	$\multimap$ linear function type
Formula (for B : FinType)	$\overset{\textbf{cart. product}}{\prod_B \mathcal{H}_b} \simeq \underset{\textbf{direct sum}}{\bigoplus_B \mathcal{H}_b} \simeq \overset{\textbf{co-product}}{\coprod_B \mathcal{H}_b}$	$\mathcal{V} \otimes \Big(\bigoplus_{b:B} \mathcal{H}_b \Big) \simeq \bigoplus_{b:B} (\mathcal{V} \otimes \mathcal{H}_b)$	
AlgTop Jargon			
Linear Logic			
Physics Meaning			

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\oplus$ direct sum	$\otimes$ tensor product	$\multimap$ linear function type
Formula (for B : FinType)	$\overset{\textbf{cart. product}}{\prod_B \mathcal{H}_b} \simeq \underset{\textbf{direct sum}}{\bigoplus_B \mathcal{H}_b} \simeq \overset{\textbf{co-product}}{\coprod_B \mathcal{H}_b}$	$\mathcal{V} \otimes \Big(\bigoplus_{b:B} \mathcal{H}_b \Big) \simeq \bigoplus_{b:B} (\mathcal{V} \otimes \mathcal{H}_b)$	$(\mathcal{V} \otimes \mathcal{H}) \multimap \mathcal{K}$ $\simeq \mathcal{V} \multimap (\mathcal{H} \multimap \mathcal{K})$
AlgTop Jargon			
Linear Logic			
Physics Meaning			

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\oplus$ direct sum	$\otimes$ tensor product	$\multimap$ linear function type
Formula (for B : FinType)	$\overset{\textbf{cart. product}}{\prod_B \mathcal{H}_b} \simeq \underset{\textbf{direct sum}}{\bigoplus_B \mathcal{H}_b} \simeq \overset{\textbf{co-product}}{\coprod_B \mathcal{H}_b}$	$\mathcal{V} \otimes \Big(\bigoplus\limits_{b:B} \mathcal{H}_b \Big) \simeq \bigoplus\limits_{b:B} (\mathcal{V} \otimes \mathcal{H}_b)$	$(\mathcal{V} \otimes \mathcal{H}) \multimap \mathcal{K}$ $\simeq \mathcal{V} \multimap (\mathcal{H} \multimap \mathcal{K})$
AlgTop Jargon	biproduct, stability, ambidexterity	Frobenius reciprocity	mapping spectrum
Linear Logic			
Physics Meaning			

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\oplus$ direct sum	$\otimes$ tensor product	$\multimap$ linear function type
Formula (for B : FinType)	$\overset{\textbf{cart. product}}{\prod_B \mathcal{H}_b} \simeq \underset{\textbf{direct sum}}{\bigoplus_B \mathcal{H}_b} \simeq \overset{\textbf{co-product}}{\coprod_B \mathcal{H}_b}$	$\mathcal{V} \otimes \Big(\bigoplus_{b:B} \mathcal{H}_b \Big) \simeq \bigoplus_{b:B} (\mathcal{V} \otimes \mathcal{H}_b)$	$(\mathcal{V} \otimes \mathcal{H}) \multimap \mathcal{K}$ $\simeq \mathcal{V} \multimap (\mathcal{H} \multimap \mathcal{K})$
AlgTop Jargon	biproduct, stability, ambidexterity	Frobenius reciprocity	mapping spectrum
		Grothendieck's Motivic Yoga of 6 oper. (Wirthmüller form)	
Linear Logic			
Physics Meaning			

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\oplus$ direct sum	$\otimes$ tensor product	$\multimap$ linear function type
Formula (for B : FinType)	$\overset{\textbf{cart. product}}{\prod_B \mathcal{H}_b} \simeq \underset{\textbf{direct sum}}{\bigoplus_B \mathcal{H}_b} \simeq \overset{\textbf{co-product}}{\coprod_B \mathcal{H}_b}$	$\mathcal{V} \otimes \Big(\bigoplus_{b:B} \mathcal{H}_b \Big) \simeq \bigoplus_{b:B} (\mathcal{V} \otimes \mathcal{H}_b)$	$(\mathcal{V} \otimes \mathcal{H}) \multimap \mathcal{K}$ $\simeq \mathcal{V} \multimap (\mathcal{H} \multimap \mathcal{K})$
AlgTop Jargon	biproduct, stability, ambidexterity	Frobenius reciprocity	mapping spectrum
		Grothendieck's Motivic Yoga of 6 oper. (Wirthmüller form)	
Linear Logic	additive disjunction	multiplicative conjunction	linear implication
Physics Meaning			

Linear/Quantum Data Types

<table>
<tr><td>Characteristic Property</td><td>1. their cartesian product blends into the co-product:</td><td>2. a tensor product appears & distributes over direct sum</td><td>3. a linear function type appears adjoint to tensor</td></tr>
<tr><td>Symbol</td><td>$\bigoplus$ direct sum</td><td>$\bigotimes$ tensor product</td><td>$\multimap$ linear function type</td></tr>
<tr><td>Formula
(for B : FinType)</td><td>$\overset{\text{cart. product}}{\prod_B \mathcal{H}_b} \simeq \underset{\text{direct sum}}{\bigoplus_B \mathcal{H}_b} \simeq \overset{\text{co-product}}{\coprod_B \mathcal{H}_b}$</td><td>$\mathcal{V} \otimes \Big(\bigoplus_{b:B} \mathcal{H}_b \Big) \simeq \bigoplus_{b:B} (\mathcal{V} \otimes \mathcal{H}_b)$</td><td>$(\mathcal{V} \otimes \mathcal{H}) \multimap \mathcal{K}$
$\simeq \mathcal{V} \multimap (\mathcal{H} \multimap \mathcal{K})$</td></tr>
<tr><td rowspan="2">AlgTop Jargon</td><td rowspan="2">biproduct,
stability, ambidexterity</td><td>Frobenius reciprocity</td><td>mapping spectrum</td></tr>
<tr><td colspan="2">Grothendieck’s Motivic Yoga of 6 oper. (Wirthmüller form)</td></tr>
<tr><td>Linear Logic</td><td>additive disjunction</td><td>multiplicative conjunction</td><td>linear implication</td></tr>
<tr><td>Physics Meaning</td><td>superselection sectors
/ quantum parallelism</td><td>compound quantum systems
/ quantum entanglement</td><td>QRAM systems</td></tr>
</table>

Linear/Quantum Data Types

<table>
<tr><td>Characteristic Property</td><td>1. their cartesian product blends into the co-product:</td><td>2. a tensor product appears & distributes over direct sum</td><td>3. a linear function type appears adjoint to tensor</td></tr>
<tr><td>Symbol</td><td>$\oplus$ direct sum</td><td>$\otimes$ tensor product</td><td>$\multimap$ linear function type</td></tr>
<tr><td>Formula
(for B : FinType)</td><td>$\overset{\text{cart. product}}{\prod_B \mathcal{H}_b} \simeq \underset{\text{direct sum}}{\bigoplus_B \mathcal{H}_b} \simeq \overset{\text{co-product}}{\coprod_B \mathcal{H}_b}$</td><td>$\mathcal{V} \otimes \Big(\bigoplus_{b:B} \mathcal{H}_b \Big) \simeq \bigoplus_{b:B} (\mathcal{V} \otimes \mathcal{H}_b)$</td><td>$(\mathcal{V} \otimes \mathcal{H}) \multimap \mathcal{K}$
$\simeq \mathcal{V} \multimap (\mathcal{H} \multimap \mathcal{K})$</td></tr>
<tr><td rowspan="2">AlgTop Jargon</td><td rowspan="2">biproduct,
stability, ambidexterity</td><td>Frobenius reciprocity</td><td>mapping spectrum</td></tr>
<tr><td colspan="2">Grothendieck's Motivic Yoga of 6 oper. (Wirthmüller form)</td></tr>
<tr><td>Linear Logic</td><td>additive disjunction</td><td>multiplicative conjunction</td><td>linear implication</td></tr>
<tr><td>Physics Meaning</td><td>superselection sectors
/ quantum parallelism</td><td>compound quantum systems
/ quantum entanglement</td><td>QRAM systems</td></tr>
</table>

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\oplus$ direct sum	$\otimes$ tensor product	$\multimap$ linear function type
Formula (for B : FinType)	$\overset{\text{cart. product}}{\prod_B \mathcal{H}_b} \simeq \underset{\text{direct sum}}{\bigoplus_B \mathcal{H}_b} \simeq \overset{\text{co-product}}{\coprod_B \mathcal{H}_b}$	$\mathcal{V} \otimes \Big(\bigoplus_{b:B} \mathcal{H}_b \Big) \simeq \bigoplus_{b:B} (\mathcal{V} \otimes \mathcal{H}_b)$	$(\mathcal{V} \otimes \mathcal{H}) \multimap \mathcal{K}$ $\simeq \mathcal{V} \multimap (\mathcal{H} \multimap \mathcal{K})$

Dependent linear Type Formers

finite classical context
(variables, parameters, ...)

reference context

$$\begin{tikzcd} B \arrow[r, "p_B"] & * \end{tikzcd}$$

classical type system
dependent on context

Type_B

Type

classical
type system

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\bigoplus$ direct sum	$\bigotimes$ tensor product	$\multimap$ linear function type
Formula (for B : FinType)	$\overset{\text{cart. product}}{\prod_B \mathcal{H}_b} \simeq \underset{\text{direct sum}}{\bigoplus_B \mathcal{H}_b} \simeq \overset{\text{co-product}}{\coprod_B \mathcal{H}_b}$	$\mathcal{V} \otimes \Big(\bigoplus_{b:B} \mathcal{H}_b \Big) \simeq \bigoplus_{b:B} (\mathcal{V} \otimes \mathcal{H}_b)$	$(\mathcal{V} \otimes \mathcal{H}) \multimap \mathcal{K}$ $\simeq \mathcal{V} \multimap (\mathcal{H} \multimap \mathcal{K})$

Dependent linear Type Formers

finite classical context (variables, parameters, ...)

reference context

$$\begin{tikzcd}
B \arrow[r, "p_B"] & *
\end{tikzcd}$$

classical context extension

classical type system dependent on context

$$\begin{tikzcd}
\mathrm{Type}_B & \mathrm{Type} \arrow[l, "*_B \times" description]
\end{tikzcd}$$

classical type system

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\oplus$ direct sum	$\otimes$ tensor product	$\multimap$ linear function type
Formula (for B : FinType)	**cart. product** **co-product** $\prod_B \mathcal{H}_b \simeq \bigoplus_B \mathcal{H}_b \simeq \coprod_B \mathcal{H}_b$ **direct sum**	$\mathcal{V} \otimes \Big(\bigoplus\limits_{b:B} \mathcal{H}_b \Big) \simeq \bigoplus\limits_{b:B} (\mathcal{V} \otimes \mathcal{H}_b)$	$(\mathcal{V} \otimes \mathcal{H}) \multimap \mathcal{K}$ $\simeq \mathcal{V} \multimap (\mathcal{H} \multimap \mathcal{K})$

Dependent linear Type Formers

finite classical context (variables, parameters, ...)

reference context

$$\begin{tikzcd}
B \arrow[rr, "p_B"] & & *
\end{tikzcd}$$

co-product

classical type system dependent on context

$$\begin{tikzcd}
\mathrm{Type}_B \arrow[rr, bend left, "\coprod_{b:B}"] \arrow[rr, phantom, "\perp"] & & \mathrm{Type} \arrow[ll, "*_B \times"]
\end{tikzcd}$$

classical type system

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\bigoplus$ direct sum	$\otimes$ tensor product	$\multimap$ linear function type
Formula (for B : FinType)	$\overset{\text{cart. product}}{\prod_B \mathcal{H}_b} \simeq \underset{\text{direct sum}}{\bigoplus_B \mathcal{H}_b} \simeq \overset{\text{co-product}}{\coprod_B \mathcal{H}_b}$	$\mathcal{V} \otimes \Big(\bigoplus_{b:B} \mathcal{H}_b \Big) \simeq \bigoplus_{b:B} (\mathcal{V} \otimes \mathcal{H}_b)$	$(\mathcal{V} \otimes \mathcal{H}) \multimap \mathcal{K}$ $\simeq \mathcal{V} \multimap (\mathcal{H} \multimap \mathcal{K})$

Dependent linear Type Formers

finite classical context (variables, parameters, ...)

reference context

$$\begin{tikzcd} B \arrow[r, "p_B"] & * \end{tikzcd}$$

classical type system dependent on context

$$\begin{tikzcd}[column sep=huge] \mathrm{Type}_B \arrow[r, bend left=60, "\overset{\text{co-product}}{\coprod_{b:B}}"{name=U}] \arrow[r, bend right=60, "\underset{\text{product}}{\prod_{b:B}}"'{name=D}] & \mathrm{Type} \arrow[l, "*_B \times"{name=M, description}] \arrow[from=U, to=M, phantom, "\perp"] \arrow[from=M, to=D, phantom, "\perp"] \end{tikzcd}$$

classical type system

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\bigoplus$ direct sum	$\otimes$ tensor product	$\multimap$ linear function type
Formula (for B : FinType)	$\underset{\text{cart. product}}{\prod_B \mathcal{H}_b} \simeq \underset{\text{direct sum}}{\bigoplus_B \mathcal{H}_b} \simeq \underset{\text{co-product}}{\coprod_B \mathcal{H}_b}$	$\mathcal{V} \otimes \Big(\bigoplus_{b:B} \mathcal{H}_b \Big) \simeq \bigoplus_{b:B} (\mathcal{V} \otimes \mathcal{H}_b)$	$(\mathcal{V} \otimes \mathcal{H}) \multimap \mathcal{K}$ $\simeq \mathcal{V} \multimap (\mathcal{H} \multimap \mathcal{K})$

Dependent linear Type Formers

finite classical context (variables, parameters, ...)

reference context

$$\begin{tikzcd}
B \arrow[r, "p_B"] & *
\end{tikzcd}$$

classical type system dependent on context

co-product

$$\begin{tikzcd}
\mathrm{Type}_B \arrow[r, shift left=6, "\coprod_{b:B}"] \arrow[r, phantom, shift left=3, "\perp"] \arrow[r, phantom, shift right=3, "\perp"] \arrow[r, shift right=6, "\prod_{b:B}"'] & \mathrm{Type} \arrow[l, "*_B \times"']
\end{tikzcd}$$

product

classical type system

classical base change / classical quantification

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\oplus$ direct sum	$\otimes$ tensor product	$\multimap$ linear function type
Formula (for B : FinType)	**cart. product** **co-product** $\prod_B \mathcal{H}_b \simeq \bigoplus_B \mathcal{H}_b \simeq \coprod_B \mathcal{H}_b$ **direct sum**	$\mathcal{V} \otimes \Big(\bigoplus_{b:B} \mathcal{H}_b \Big) \simeq \bigoplus_{b:B} (\mathcal{V} \otimes \mathcal{H}_b)$	$(\mathcal{V} \otimes \mathcal{H}) \multimap \mathcal{K}$ $\simeq \mathcal{V} \multimap (\mathcal{H} \multimap \mathcal{K})$

Dependent linear Type Formers

finite classical context (variables, parameters, ...)

reference context

$$\begin{tikzcd}
B \arrow[r, "p_B"] & *
\end{tikzcd}$$

classical type system dependent on context

$$\begin{tikzcd}[column sep=huge, row sep=large]
\mathrm{Type}_B \arrow[r, shift left=8, "\overset{\text{co-product}}{\coprod_{b:B}}"] \arrow[r, shift right=8, "\underset{\text{product}}{\prod_{b:B}}"'] \arrow[r, phantom, shift left=4, "\perp"] \arrow[r, phantom, shift right=4, "\perp"] & \mathrm{Type} \arrow[l, "*_B \times"']
\end{tikzcd}$$

classical type system

classical base change /
classical quantification

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\bigoplus$ direct sum	$\otimes$ tensor product	$\multimap$ linear function type
Formula (for B : FinType)	$\overset{\text{cart. product}}{\prod_B \mathcal{H}_b} \simeq \underset{\text{direct sum}}{\bigoplus_B \mathcal{H}_b} \simeq \overset{\text{co-product}}{\coprod_B \mathcal{H}_b}$	$\mathcal{V} \otimes \Big(\bigoplus_{b:B} \mathcal{H}_b \Big) \simeq \bigoplus_{b:B} (\mathcal{V} \otimes \mathcal{H}_b)$	$(\mathcal{V} \otimes \mathcal{H}) \multimap \mathcal{K}$ $\simeq \mathcal{V} \multimap (\mathcal{H} \multimap \mathcal{K})$

Dependent linear Type Formers

finite classical context (variables, parameters, ...)

reference context

$$\begin{tikzcd} B \arrow[rr, "p_B"] & & * \end{tikzcd}$$

classical type system dependent on context

$$\begin{tikzcd} \mathrm{Type}_B \arrow[rr, shift left=8, "\overset{\text{co-product}}{\coprod_{b:B}}" description] \arrow[rr, phantom, shift left=4, "\perp"] \arrow[rr, phantom, shift right=4, "\perp"] \arrow[rr, shift right=8, "\underset{\text{product}}{\prod_{b:B}}" description] & & \mathrm{Type} \arrow[ll, "*_B \times" description] \end{tikzcd}$$

classical type system

classical base change / classical quantification

linear type system in classical context

$$\Big(\mathrm{LType}_B,\ \overset{\text{tensor}}{\otimes_B}\Big) \qquad \Big(\mathrm{LType},\ \overset{\text{tensor}}{\otimes}\Big)$$

linear type system

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\bigoplus$ direct sum	$\bigotimes$ tensor product	$\multimap$ linear function type
Formula (for B : FinType)	$\overset{\textbf{cart. product}}{\prod_B \mathcal{H}_b} \simeq \underset{\textbf{direct sum}}{\bigoplus_B \mathcal{H}_b} \simeq \overset{\textbf{co-product}}{\coprod_B \mathcal{H}_b}$	$\mathcal{V} \otimes \Big(\bigoplus_{b:B} \mathcal{H}_b \Big) \simeq \bigoplus_{b:B} \big(\mathcal{V} \otimes \mathcal{H}_b \big)$	$(\mathcal{V} \otimes \mathcal{H}) \multimap \mathcal{K}$ $\simeq \mathcal{V} \multimap (\mathcal{H} \multimap \mathcal{K})$

Dependent linear Type Formers

finite classical context (variables, parameters, ...)

reference context

$$\begin{tikzcd}[column sep=huge]
B \arrow[r, "p_B"] & *
\end{tikzcd}$$

classical type system dependent on context

$$\begin{tikzcd}[column sep=huge]
\mathrm{Type}_B \arrow[r, shift left=14pt, "\overset{\text{co-product}}{\coprod_{b:B}}"] \arrow[r, phantom, shift left=7pt, "\perp"] \arrow[r, phantom, shift right=7pt, "\perp"] \arrow[r, shift right=14pt, "\underset{\text{product}}{\prod_{b:B}}"'] & \mathrm{Type} \arrow[l, "*_B \times" description]
\end{tikzcd}$$

classical type system

linear type system in classical context

$$\begin{tikzcd}[column sep=huge]
\Big(\mathrm{LType}_B, \overset{\text{tensor}}{\otimes_B} \Big) & \Big(\mathrm{LType}, \overset{\text{tensor}}{\otimes} \Big) \arrow[l, "\textbf{linear context extension}"', "\mathbb{1}_B \otimes" description]
\end{tikzcd}$$

linear type system

classical base change / classical quantification

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\oplus$ direct sum	$\otimes$ tensor product	$\multimap$ linear function type
Formula (for B : FinType)	$\overset{\text{cart. product}}{\prod_B \mathcal{H}_b} \simeq \underset{\text{direct sum}}{\bigoplus_B \mathcal{H}_b} \simeq \overset{\text{co-product}}{\coprod_B \mathcal{H}_b}$	$\mathcal{V} \otimes \Big(\bigoplus_{b:B} \mathcal{H}_b \Big) \simeq \bigoplus_{b:B} (\mathcal{V} \otimes \mathcal{H}_b)$	$(\mathcal{V} \otimes \mathcal{H}) \multimap \mathcal{K}$ $\simeq \mathcal{V} \multimap (\mathcal{H} \multimap \mathcal{K})$

Dependent linear Type Formers

finite classical context (variables, parameters, ...)

reference context

$$\begin{tikzcd}[column sep=huge]
B \arrow[r, "p_B"] & *
\end{tikzcd}$$

classical type system dependent on context

$$\begin{tikzcd}[column sep=huge]
\mathrm{Type}_B \arrow[r, shift left=4, "\overset{\text{co-product}}{\coprod_{b:B}}"] \arrow[r, phantom, shift left=2, "\perp"] \arrow[r, phantom, shift right=2, "\perp"] \arrow[r, shift right=4, "\underset{\text{product}}{\prod_{b:B}}"'] & \mathrm{Type} \arrow[l, "{*_B} \times" description]
\end{tikzcd}$$

classical type system

linear type system in classical context

$$\begin{tikzcd}[column sep=huge]
\big(\mathrm{LType}_B, \overset{\text{tensor}}{\otimes_B}\big) \arrow[r, shift left=4, "\overset{\text{direct sum}}{\bigoplus_{b:B}}"] \arrow[r, phantom, shift left=2, "\perp"] \arrow[r, phantom, shift right=2, "\perp"] \arrow[r, shift right=4, "\bigoplus_{b:B}"'] & \big(\mathrm{LType}, \overset{\text{tensor}}{\otimes}\big) \arrow[l, "\mathbb{1}_B \otimes" description]
\end{tikzcd}$$

linear type system

classical base change /
classical quantification

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\oplus$ direct sum	$\otimes$ tensor product	$\multimap$ linear function type
Formula (for B : FinType)	$\overset{\textbf{cart. product}}{\prod_B \mathcal{H}_b} \simeq \underset{\textbf{direct sum}}{\bigoplus_B \mathcal{H}_b} \simeq \overset{\textbf{co-product}}{\coprod_B \mathcal{H}_b}$	$\mathcal{V} \otimes \Big(\bigoplus_{b:B} \mathcal{H}_b \Big) \simeq \bigoplus_{b:B} (\mathcal{V} \otimes \mathcal{H}_b)$	$(\mathcal{V} \otimes \mathcal{H}) \multimap \mathcal{K}$ $\simeq \mathcal{V} \multimap (\mathcal{H} \multimap \mathcal{K})$

Dependent linear Type Formers

finite classical context (variables, parameters, ...)

reference context

$$\begin{tikzcd}
B \arrow[rr, "p_B"] & & *
\end{tikzcd}$$

classical type system dependent on context

$$\begin{tikzcd}[column sep=huge]
\mathrm{Type}_B \arrow[r, shift left=14pt, "\overset{\text{co-product}}{\coprod_{b:B}}" description] \arrow[r, phantom, shift left=7pt, "\perp"] \arrow[r, phantom, shift right=7pt, "\perp"] \arrow[r, shift right=14pt, "\underset{\text{product}}{\prod_{b:B}}" description] & \mathrm{Type} \arrow[l, "*_B \times" description]
\end{tikzcd}$$

classical type system

classical base change / classical quantification

linear type system in classical context

$$\begin{tikzcd}[column sep=huge]
\big(\mathrm{LType}_B, \overset{\text{tensor}}{\otimes_B}\big) \arrow[r, shift left=14pt, "\overset{\text{direct sum}}{\bigoplus_{b:B}}" description] \arrow[r, phantom, shift left=7pt, "\perp"] \arrow[r, phantom, shift right=7pt, "\perp"] \arrow[r, shift right=14pt, "\bigoplus_{b:B}" description] & \big(\mathrm{LType}, \overset{\text{tensor}}{\otimes}\big) \arrow[l, "\mathbb{1}_B \otimes" description]
\end{tikzcd}$$

linear type system

quantum base change / Motivic Yoga

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\bigoplus$ direct sum	$\bigotimes$ tensor product	$\multimap$ linear function type
Formula (for B : FinType)	$\overset{\text{cart. product}}{\prod_B \mathcal{H}_b} \simeq \underset{\text{direct sum}}{\bigoplus_B \mathcal{H}_b} \simeq \overset{\text{co-product}}{\coprod_B \mathcal{H}_b}$	$\mathcal{V} \otimes \Big(\bigoplus_{b:B} \mathcal{H}_b \Big) \simeq \bigoplus_{b:B} (\mathcal{V} \otimes \mathcal{H}_b)$	$(\mathcal{V} \otimes \mathcal{H}) \multimap \mathcal{K}$ $\simeq \mathcal{V} \multimap (\mathcal{H} \multimap \mathcal{K})$

Dependent linear Type Formers

finite classical context (variables, parameters, ...)

reference context

$$\begin{tikzcd}[column sep=huge]
B \arrow[r, "p_B"] & *
\end{tikzcd}$$

classical type system dependent on context

$$\begin{tikzcd}[column sep=huge, row sep=large]
\mathrm{Type}_B \arrow[r, shift left=4, "\overset{\text{co-product}}{\coprod_{b:B}}"] \arrow[r, shift right=4, "\underset{\text{product}}{\prod_{b:B}}"'] \arrow[r, phantom, shift left=2, "\perp"] \arrow[r, phantom, shift right=2, "\perp"] & \mathrm{Type} \arrow[l, "*_B \times" description]
\end{tikzcd}$$

classical type system

classical base change / classical quantification

linear type system in classical context

$$\begin{tikzcd}[column sep=huge, row sep=large]
\big(\mathrm{LType}_B, \overset{\text{tensor}}{\otimes_B}\big) \arrow[r, shift left=4, "\overset{\text{direct sum}}{\bigoplus_{b:B}}"] \arrow[r, shift right=4, "\bigoplus_{b:B}"'] \arrow[r, phantom, shift left=2, "\perp"] \arrow[r, phantom, shift right=2, "\perp"] & \big(\mathrm{LType}, \overset{\text{tensor}}{\otimes}\big) \arrow[l, "\mathbb{1}_B \otimes" description]
\end{tikzcd}$$

linear type system

quantum base change / Motivic Yoga

Linear/Quantum Data Types

Characteristic Property	**1.** their cartesian product blends into the co-product:	**2.** a tensor product appears & distributes over direct sum	**3.** a linear function type appears adjoint to tensor
Symbol	$\bigoplus$ direct sum	$\bigotimes$ tensor product	$\multimap$ linear function type
Formula (for B : FinType)	$\overset{\text{cart. product}}{\prod_B \mathcal{H}_b} \simeq \underset{\text{direct sum}}{\bigoplus_B \mathcal{H}_b} \simeq \overset{\text{co-product}}{\coprod_B \mathcal{H}_b}$	$\mathcal{V} \otimes \Big(\bigoplus_{b:B} \mathcal{H}_b \Big) \simeq \bigoplus_{b:B} (\mathcal{V} \otimes \mathcal{H}_b)$	$(\mathcal{V} \otimes \mathcal{H}) \multimap \mathcal{K}$ $\simeq \mathcal{V} \multimap (\mathcal{H} \multimap \mathcal{K})$

Dependent linear Type Formers

all available in dLHoTT

finite classical context (variables, parameters, ...)

reference context

$$\begin{tikzcd}
B \arrow[r, "p_B"] & *
\end{tikzcd}$$

classical type system dependent on context

$$\begin{tikzcd}[column sep=huge, row sep=large]
\mathrm{Type}_B \arrow[r, shift left=6, "\overset{\text{co-product}}{\coprod_{b:B}}"] \arrow[r, phantom, shift left=3, "\perp"] \arrow[r, phantom, shift right=3, "\perp"] \arrow[r, shift right=6, "\underset{\text{product}}{\prod_{b:B}}"'] & \mathrm{Type} \arrow[l, "*_B \times"']
\end{tikzcd}$$

classical type system

classical base change / classical quantification

linear type system in classical context

$$\begin{tikzcd}[column sep=huge, row sep=large]
\big(\mathrm{LType}_B,\, \overset{\text{tensor}}{\otimes_B}\big) \arrow[r, shift left=6, "\overset{\text{direct sum}}{\bigoplus_{b:B}}"] \arrow[r, phantom, shift left=3, "\perp"] \arrow[r, phantom, shift right=3, "\perp"] \arrow[r, shift right=6, "\bigoplus_{b:B}"'] & \big(\mathrm{LType},\, \overset{\text{tensor}}{\otimes}\big) \arrow[l, "\mathbb{1}_B \otimes"']
\end{tikzcd}$$

linear type system

quantum base change / Motivic Yoga

Quantum Effects

Recall: **Monadic computational effects.**

A *monad* $\mathscr{E}(-)$ on a data type system encodes *computational effects*:

effectful program

$$D_1 \xrightarrow{\mathrm{prog}_{12}} \mathscr{E}(D_2)$$

output data of nominal type D_2
causing effects of type $\mathscr{E}(-)$

Recall: **Monadic computational effects.**

A *monad* $\mathscr{E}(-)$ on a data type system encodes *computational effects*:

first program

$$\begin{tikzcd} D_1 \arrow[r, "\mathrm{prog}_{12}"] & \mathscr{E}(D_2) \end{tikzcd}$$

output data of nominal type D_2
causing effects of type $\mathscr{E}(-)$

second program

$$\begin{tikzcd} D_2 \arrow[r, "\mathrm{prog}_{23}"] & \mathscr{E}(D_3) \end{tikzcd}$$

input data of type D_2
causing effects of type $\mathscr{E}(-)$

Recall: **Monadic computational effects.**

A *monad* $\mathscr{E}(-)$ on a data type system encodes *computational effects*:

first program

second program

$$\begin{tikzcd} D_1 \arrow[r, "\mathrm{prog}_{12}"] & \mathscr{E}(D_2) & D_2 \arrow[r, "\mathrm{prog}_{23}"] & \mathscr{E}(D_3) \arrow[d, mapsto, bend left, "\text{bind previous effects into second program}"] \\ D_1 \arrow[r, "\mathrm{prog}_{12}"] & \mathscr{E}(D_2) & \mathscr{E}(D_2) \arrow[r, "\mathrm{bind}^{\mathscr{E}}\mathrm{prog}_{23}"] & \mathscr{E}(D_3) \end{tikzcd}$$

output data of nominal type D_2
causing effects of type $\mathscr{E}(-)$

input data of type D_2
causing effects of type $\mathscr{E}(-)$

bind previous effects
into second program

carry any previous $\mathscr{E}(-)$-effects along

Recall: **Monadic computational effects.**

A *monad* $\mathscr{E}(-)$ on a data type system encodes *computational effects*:

first program

second program

$\mathscr{E}$-composite program

$$\begin{tikzcd}
D_1 \arrow[r, "\mathrm{prog}_{12}", "\text{output data of nominal type } D_2 \text{ causing effects of type } \mathscr{E}(-)"'] & \mathscr{E}(D_2) & D_2 \arrow[r, "\mathrm{prog}_{23}", "\text{input data of type } D_2 \text{ causing effects of type } \mathscr{E}(-)"'] & \mathscr{E}(D_3) \arrow[d, mapsto, bend left, "\text{bind previous effects into second program}"] \\
D_1 \arrow[r, "\mathrm{prog}_{12}"] & \mathscr{E}(D_2) & \mathscr{E}(D_2) \arrow[r, "\mathrm{bind}^{\mathscr{E}}\,\mathrm{prog}_{23}", "\text{carry any previous } \mathscr{E}(-)\text{-effects along}"'] & \mathscr{E}(D_3) \arrow[d, mapsto, bend left, "\text{compose}"] \\
D_1 \arrow[rrr, "\mathrm{bind}^{\mathscr{E}}\,\mathrm{prog}_{23} \circ \mathrm{prog}_{12}", "\text{causing cumulative } \mathscr{E}(-)\text{-effects}"'] & & & \mathscr{E}(D_3)
\end{tikzcd}$$

Recall: **Monadic effect handlers.**

$$\begin{tikzcd} D_1 \arrow[rr, "\mathrm{prog}_{12}", "\textbf{in-effectful program}"'] & & D_2 \end{tikzcd}$$ **data type to absorb $\mathscr{E}$-effects**

Recall: **Monadic effect handlers.**

$$\begin{tikzcd}
D_1 \arrow[r, "\mathrm{prog}_{12}", "\text{in-effectful program}"'] & D_2 & {} \arrow[d, mapsto, bend left, "\text{incorporate handling of } \mathscr{E}(-)\text{-effects}"] \\
\mathscr{E}(D_1) \arrow[r, "\mathrm{hndl}^{\mathscr{E}}_{D_2} \mathrm{prog}_{12}", "\text{in-effectful program handling effects of type } \mathscr{E}(-)"'] & D_2 & {}
\end{tikzcd}$$

Recall: **Monadic effect handlers.**

$$\begin{tikzcd}
D_1 \arrow[r, "\mathrm{prog}_{12}", "\text{in-effectful program}"'] & D_2 & {} \arrow[d, mapsto, bend left, "\text{incorporate handling of } \mathscr{E}(-)\text{-effects}"] \\
\mathscr{E}(D_1) \arrow[r, "\mathrm{hndl}^{\mathscr{E}}_{D_2}\mathrm{prog}_{12}", "\text{in-effectful program handling effects of type } \mathscr{E}(-)"'] & D_2 & {}
\end{tikzcd}$$

$$\begin{tikzcd}
D_1 \arrow[r, "\mathrm{ret}^{\mathscr{E}}_{D_1}", "\text{produce trivial effect}"'] \arrow[rr, bend right, "\mathrm{prog}_{12}"', "\text{no effect}"'] & \mathscr{E}(D_1) \arrow[r, "\mathrm{hndl}^{\mathscr{E}}_{D_2}\mathrm{prog}_{12}", "\text{handle effects running program}"'] & D_2
\end{tikzcd}$$

consistency conditions

Recall: **Monadic effect handlers.**

$$\begin{tikzcd}
D_1 \arrow[r, "\mathrm{prog}_{12}", "\text{in-effectful program}"'] & D_2 & {} \arrow[d, mapsto, bend left, "\begin{array}{c}\text{incorporate handling}\\ \text{of } \mathscr{E}(-)\text{-effects}\end{array}"] \\
\mathscr{E}(D_1) \arrow[r, "\mathrm{hndl}^{\mathscr{E}}_{D_2}\,\mathrm{prog}_{12}", "\begin{array}{c}\text{in-effectful program}\\ \text{handling effects of type } \mathscr{E}(-)\end{array}"'] & D_2 & {}
\end{tikzcd}$$

$$\begin{tikzcd}
D_1 \arrow[r, "\mathrm{ret}^{\mathscr{E}}_{D_1}", "\begin{array}{c}\text{produce}\\ \text{trivial effect}\end{array}"'] \arrow[rr, bend right, "\begin{array}{c}\mathrm{prog}_{12}\\ \text{no effect}\end{array}"'] & \mathscr{E}(D_1) \arrow[r, "\mathrm{hndl}^{\mathscr{E}}_{D_2}\,\mathrm{prog}_{12}", "\begin{array}{c}\text{handle effects}\\ \text{running program}\end{array}"'] & D_2
\end{tikzcd}$$

consistency conditions

$$\begin{tikzcd}
\mathscr{E}(D_0) \arrow[r, "\mathrm{bind}^{\mathscr{E}}\,\mathrm{prog}_{01}", "\begin{array}{c}\text{carry effects}\\ \text{along}\end{array}"'] \arrow[rr, bend right, "\mathrm{hndl}^{\mathscr{E}}_{D_2}\left(D_0 \xrightarrow{\mathrm{prog}_{01}} \mathscr{E}(D_1) \xrightarrow{\mathrm{hndl}^{\mathscr{E}}_{D_2}\mathrm{prog}_{12}} D_2\right)"'] & \mathscr{E}(D_1) \arrow[r, "\mathrm{hndl}^{\mathscr{E}}_{D_2}\,\mathrm{prog}_{12}", "\begin{array}{c}\text{handle}\\ \text{cumulative effects}\end{array}"'] & D_2
\end{tikzcd}$$

handle effects... consecutively

Recall: **Data type system of Monadic effect handlers.**

$$\begin{tikzcd}
D_1 \arrow[r, "\mathrm{prog}_{12}", "\textbf{in-effectful program}"'] & D_2 \arrow[d, mapsto, bend left, "\begin{array}{c}\textbf{incorporate handling}\\ \textbf{of } \mathscr{E}(-)\textbf{-effects}\end{array}"] \\
\mathscr{E}(D_1) \arrow[r, "\mathrm{hndl}^{\mathscr{E}}_{D_2}\mathrm{prog}_{12}", "\begin{array}{c}\textbf{in-effectful program}\\ \textbf{handling effects of type } \mathscr{E}(-)\end{array}"'] & D_2
\end{tikzcd}$$

"$\mathscr{E}$-modal data type"

Monadicity:

$\mathscr{E}$-modales in Type
("EM-category")

$\mathrm{Type}^{\mathscr{E}}$

Recall: **Data type system of Monadic effect handlers.**

$$\begin{tikzcd}
D_1 \arrow[r, "\mathrm{prog}_{12}", "\textbf{in-effectful program}"'] & D_2 \arrow[d, mapsto, bend left=60, "\substack{\textbf{incorporate handling}\\ \textbf{of } \mathscr{E}(-)\textbf{-effects}}"] \\
\mathscr{E}(D_1) \arrow[r, "\mathrm{hndl}^{\mathscr{E}}_{D_2}\mathrm{prog}_{12}", "\substack{\textbf{in-effectful program}\\ \textbf{handling effects of type } \mathscr{E}(-)}"'] & D_2
\end{tikzcd}$$

“$\mathscr{E}$-modal data type”

Monadicity:

$$\begin{tikzcd}
\mathrm{Type} \arrow[loop right, "\mathscr{E}"]
\end{tikzcd}$$

monad

$\mathscr{E}$-modales in Type
(“EM-category”)

$\mathrm{Type}^{\mathscr{E}}$

Recall: **Data type system of Monadic effect handlers.**

$$\begin{tikzcd}
D_1 \arrow[r, "\mathrm{prog}_{12}", "\text{in-effectful program}"'] & D_2 \arrow[d, mapsto, bend left, "\text{incorporate handling of } \mathscr{E}(-)\text{-effects}"] \\
\mathscr{E}(D_1) \arrow[r, "\mathrm{hndl}^{\mathscr{E}}_{D_2}\mathrm{prog}_{12}", "\text{in-effectful program handling effects of type } \mathscr{E}(-)"'] & D_2 \quad \text{“}\mathscr{E}\text{-modal data type”}
\end{tikzcd}$$

Monadicity:

$$\begin{tikzcd}
 & \mathrm{Type} \arrow[loop right, "\mathscr{E}"] \arrow[dl, shift right=2, "F^{\mathscr{E}}"'] & \text{induced monad} \\
\mathrm{Type}^{\mathscr{E}} \arrow[ur, shift right=2, "U^{\mathscr{E}}"', "\text{monadic adjunction}"] \arrow[ur, phantom, "\perp"] & &
\end{tikzcd}$$

$\mathscr{E}$**-modales in** Type

(“EM-category”)

Recall: **Data type system of Monadic effect handlers.**

$$\begin{tikzcd}
D_1 \arrow[r, "\mathrm{prog}_{12}", "\text{in-effectful program}"'] & D_2 \arrow[d, maps to, bend left, "\text{incorporate handling of } \mathscr{E}(-)\text{-effects}"] \\
\mathscr{E}(D_1) \arrow[r, "\mathrm{hndl}^{\mathscr{E}}_{D_2}\mathrm{prog}_{12}", "\text{in-effectful program handling effects of type } \mathscr{E}(-)"'] & D_2
\end{tikzcd}$$

“$\mathscr{E}$-modal data type”

Monadicity:

any adjunction
on type system

$$\begin{tikzcd}
\mathrm{Type}' \arrow[r, shift right, "U"'] \arrow[r, phantom, "\perp"] \arrow[d, "K^{UF}", "\text{terminal comparison functor}"'] & \mathrm{Type} \arrow[l, shift right, "F"'] \arrow[loop right, "\mathscr{E}"] \arrow[dl, shift right, "F^{\mathscr{E}}"'] \\
\mathrm{Type}^{\mathscr{E}} \arrow[ur, shift right, "U^{\mathscr{E}}"'] \arrow[ur, phantom, "\perp"] &
\end{tikzcd}$$

induced monad

monadic adjunction

$\mathscr{E}$-modales in Type
(“EM-category”)

Recall: **Data type system of Monadic effect handlers.**

$$\begin{tikzcd}
D_1 \arrow[r, "\mathrm{prog}_{12}", "\text{in-effectful program}"'] & D_2 \arrow[d, mapsto, bend left, "\text{incorporate handling of } \mathscr{E}(-)\text{-effects}"] \\
\mathscr{E}(D_1) \arrow[r, "\mathrm{hndl}^{\mathscr{E}}_{D_2}\mathrm{prog}_{12}", "\text{in-effectful program handling effects of type } \mathscr{E}(-)"'] & D_2 \; \text{“}\mathscr{E}\text{-modal data type”}
\end{tikzcd}$$

in-effectful program

incorporate handling of $\mathscr{E}(-)$-effects

“$\mathscr{E}$-modal data type”

in-effectful program handling effects of type $\mathscr{E}(-)$

Monadicity:

$$\begin{tikzcd}
\mathrm{Type}_{\mathscr{E}} \arrow[d, hook, "K_{UF}"] \arrow[dr, shift right, "U_{\mathscr{E}}"'] & \\
\mathrm{Type}' \arrow[r, shift right, "U"'] \arrow[d, "K^{UF}"] & \mathrm{Type} \arrow[loop right, "\mathscr{E}"] \arrow[l, shift right, "F"'] \arrow[l, phantom, "\perp"] \arrow[ul, two heads, shift right, "F_{\mathscr{E}}"'] \arrow[ul, phantom, "\perp"] \arrow[dl, shift right, "F^{\mathscr{E}}"'] \arrow[dl, phantom, "\perp"] \\
\mathrm{Type}^{\mathscr{E}} \arrow[ur, shift right, "U^{\mathscr{E}}"'] &
\end{tikzcd}$$

free $\mathscr{E}$-modales in Type
(“Kleisli category”)

initial comparison functor

any adjunction
on type system

induced monad

terminal comparison functor

monadic adjunction

$\mathscr{E}$-modales in Type
(“EM-category”)

Recall: **Data type system of Monadic effect handlers.**

$$\begin{tikzcd}
D_1 \arrow[r, "\mathrm{prog}_{12}"] & D_2 \arrow[d, mapsto, bend left, "\text{incorporate handling of } \mathscr{E}(-)\text{-effects}"] \\
\mathscr{E}(D_1) \arrow[r, "\mathrm{hndl}^{\mathscr{E}}_{D_2}\mathrm{prog}_{12}"] & D_2
\end{tikzcd}$$

in-effectful program

incorporate handling of $\mathscr{E}(-)$-effects

“$\mathscr{E}$-modal data type”

in-effectful program handling effects of type $\mathscr{E}(-)$

Monadicity:

$$\begin{tikzcd}
\mathrm{Type}_{\mathscr{E}} \arrow[d, hook, "K_{UF}"] \arrow[drr, shift left, "U_{\mathscr{E}}"] \arrow[drr, phantom, "\perp"] & & \\
\mathrm{Type}' \arrow[d, "K^{UF}"] \arrow[rr, shift right, "U"'] \arrow[rr, phantom, "\perp"] & & \mathrm{Type} \arrow[ll, shift right, "F"'] \arrow[ull, two heads, shift left, "F_{\mathscr{E}}"] \arrow[dll, shift right, "F^{\mathscr{E}}"'] \arrow["\mathscr{E}"', loop right] \\
\mathrm{Type}^{\mathscr{E}} \arrow[urr, shift right, "U^{\mathscr{E}}"'] \arrow[urr, phantom, "\perp"] & &
\end{tikzcd}$$

free $\mathscr{E}$-modales in Type
(“Kleisli category”)

initial comparison functor

any adjunction
on type system

terminal comparison functor

$\mathscr{E}$-modales in Type
(“EM-category”)

monadic adjunction

induced monad

Now just to work this out
for the **effects induced by**
dependent data type formers
in dLHoTT ⟶

Given B : Type of possible measurement outcomes (“possible worlds”) **the monadic effects of B-dependent data type formers** constitute modalities of **actual and potential B-measurements**:

classical data types

$$\begin{tikzcd}
 & & \textbf{possibly} & & \textbf{randomly} & & \\
\textbf{actual data} & \perp & \mathrm{Type}_B \arrow[loop above, "\Diamond_B"] \arrow[loop below, "\Box_B"'] \arrow[rr, shift left=4, "\coprod_B"] \arrow[rr, phantom, shift left=2, "\perp"] \arrow[rr, phantom, shift right=2, "\perp"] \arrow[rr, shift right=4, "\prod_B"'] & & \mathrm{Type} \arrow[ll, "\times B"'] \arrow[loop above, "\star_B"] \arrow[loop below, "\bigcirc_B"'] & \perp & \textbf{potential data} \\
 & & \textbf{necessarily} & & \textbf{indefinitely} & &
\end{tikzcd}$$

Given B : Type of possible measurement outcomes ("possible worlds") **the monadic effects of B-dependent data type formers** constitute modalities of **actual and potential B-measurements**:

classical data types

possibly **randomly**

actual data $\perp$ $\perp$ **potential data**

$$\begin{tikzcd}
\mathrm{Type}_B \arrow[loop above, "\Diamond_B"] \arrow[loop below, "\Box_B"] \arrow[r, shift left=4, "\coprod_B"] \arrow[r, phantom, shift left=2, "\perp"] \arrow[r, phantom, shift right=2, "\perp"] \arrow[r, shift right=4, "\prod_B"'] & \mathrm{Type} \arrow[l, "\times B" description] \arrow[loop above, "\star_B"] \arrow[loop below, "\bigcirc_B"]
\end{tikzcd}$$

necessarily **indefinitely**

necessarily $P_\bullet$

$$\Box_B P_\bullet$$

$$b : B \;\vdash\; \prod_{b' : B} P_{b'}$$

Given B : Type of possible measurement outcomes (“possible worlds”) **the monadic effects of B-dependent data type formers** constitute modalities of **actual and potential B-measurements**:

classical data types

possibly **randomly**

actual data $\perp$ $\perp$ **potential data**

$$\begin{tikzcd}
\mathrm{Type}_B \arrow[loop above, "\Diamond_B"] \arrow[loop below, "\Box_B"'] \arrow[r, shift left=4, "\coprod_B"] \arrow[r, phantom, shift left=2, "\perp"] \arrow[r, phantom, shift right=2, "\perp"] \arrow[r, shift right=4, "\prod_B"'] & \mathrm{Type} \arrow[l, "\times B" description] \arrow[loop above, "\star_B"] \arrow[loop below, "\bigcirc_B"']
\end{tikzcd}$$

necessarily **indefinitely**

---

necessarily $P_\bullet$ **entails** **actually $P_\bullet$**

$$\Box_B P_\bullet \xrightarrow{\;\;\epsilon^{\Box_B}_{P_\bullet}\;\;} P_\bullet$$

$$b : B \;\vdash\; \prod_{b' : B} P_{b'} \xrightarrow{(p_{b'})_{b':B} \;\mapsto\; p_b} P_b$$

Given B : Type of possible measurement outcomes (“possible worlds”) **the monadic effects of B-dependent data type formers** constitute modalities of **actual and potential B-measurements**:

classical data types

possibly **randomly**

actual data $\perp$ **potential data** $\perp$

necessarily **indefinitely**

$$\begin{tikzcd}
\mathrm{Type}_B \arrow[loop above, "\Diamond_B"] \arrow[loop below, "\Box_B"'] \arrow[rr, shift left=6, "\coprod_B"] \arrow[rr, phantom, shift left=3, "\perp"] \arrow[rr, shift right=6, "\prod_B"'] \arrow[rr, phantom, shift right=3, "\perp"] & & \mathrm{Type} \arrow[ll, "\times B"'] \arrow[loop above, "\star_B"] \arrow[loop below, "\bigcirc_B"']
\end{tikzcd}$$

---

necessarily $P_\bullet$ **entails** **actually $P_\bullet$** **entails** **possibly $P_\bullet$**

$$\begin{tikzcd}
\Box_B P_\bullet \arrow[r, "\epsilon^{\Box_B}_{P_\bullet}"] & P_\bullet \arrow[r, "\eta^{\Diamond_B}_{P_\bullet}"] & \Diamond_B P_\bullet
\end{tikzcd}$$

$$b : B \;\vdash\; \begin{tikzcd}
\prod\limits_{b':B} P_{b'} \arrow[r, "(p_{b'})_{b':B} \mapsto p_b"] & P_b \arrow[r, "p_b \mapsto (p_b)_b"] & \coprod\limits_{b':B} P_{b'}
\end{tikzcd}$$

Given B : Type of possible measurement outcomes (“possible worlds”) **the monadic effects of B-dependent data type formers** constitute modalities of **actual and potential B-measurements**:

classical
data types

possibly **randomly**

actual data $\perp$ $\perp$ **potential data**

$$\begin{tikzcd}
\mathrm{Type}_B \arrow[loop above, "\Diamond_B"] \arrow[loop below, "\Box_B"] \arrow[r, shift left=4, "\coprod_B"] \arrow[r, shift right=4, "\prod_B"'] & \mathrm{Type} \arrow[l, "\times B" description] \arrow[loop above, "\star_B"] \arrow[loop below, "\bigcirc_B"]
\end{tikzcd}$$

$$\perp \quad \perp$$

necessarily **indefinitely**

necessarily $P_\bullet$ **entails** **actually $P_\bullet$** **entails** **possibly $P_\bullet$**

$$\Box_B P_\bullet \xrightarrow{\;\epsilon^{\Box_B}_{P_\bullet}\;} P_\bullet \xrightarrow{\;\eta^{\Diamond_B}_{P_\bullet}\;} \Diamond_B P_\bullet$$

$$b : B \;\vdash\; \prod_{b':B} P_{b'} \xrightarrow{(p_{b'})_{b':B} \mapsto p_b} P_b \xrightarrow{p_b \mapsto (p_b)_b} \coprod_{b':B} P_{b'}$$

randomly P

$$\star_B P$$

$$\coprod_{b:B} P$$

Given B : Type of possible measurement outcomes (“possible worlds”) **the monadic effects of B-dependent data type formers** constitute modalities of **actual and potential B-measurements**:

classical data types

possibly **randomly**

actual data $\perp$ **potential data** $\perp$

necessarily **indefinitely**

$$\begin{tikzcd}
\mathrm{Type}_B \arrow[loop above, "\lozenge_B"] \arrow[loop below, "\square_B"'] \arrow[r, shift left=3, "\coprod_B"] \arrow[r, shift right=3, "\prod_B"'] & \mathrm{Type} \arrow[l, "\times B" description] \arrow[loop above, "\star_B"'] \arrow[loop below, "\bigcirc_B"]
\end{tikzcd}$$

$$\coprod_B \dashv \times B \dashv \prod_B$$

necessarily $P_\bullet$ **entails** **actually $P_\bullet$** **entails** **possibly $P_\bullet$**

$$\square_B P_\bullet \xrightarrow{\epsilon^{\square_B}_{P_\bullet}} P_\bullet \xrightarrow{\eta^{\lozenge_B}_{P_\bullet}} \lozenge_B P_\bullet$$

$$b : B \;\vdash\; \prod_{b':B} P_{b'} \xrightarrow{(p_{b'})_{b':B} \mapsto p_b} P_b \xrightarrow{p_b \mapsto (p_b)_b} \coprod_{b':B} P_{b'}$$

randomly P **entails** **potentially P**

$$\star_B P \xrightarrow{\epsilon^{\star_B}_{P}} P$$

$$\coprod_{b:B} P \xrightarrow{(p)_b \mapsto p} P$$

Given B : Type of possible measurement outcomes (“possible worlds”) **the monadic effects of B-dependent data type formers** constitute modalities of **actual and potential B-measurements**:

classical data types

possibly **randomly**

$$\begin{tikzcd}
\textbf{actual data} \arrow[r, phantom, "\perp"] & \mathrm{Type}_B \arrow[loop above, "\Diamond_B"] \arrow[loop below, "\Box_B"'] \arrow[rr, shift left=6, "\coprod_B"] \arrow[rr, phantom, shift left=3, "\perp"] \arrow[rr, phantom, shift right=3, "\perp"] \arrow[rr, shift right=6, "\prod_B"'] & & \mathrm{Type} \arrow[ll, "\times B"'] \arrow[loop above, "\star_B"] \arrow[loop below, "\bigcirc_B"'] \arrow[r, phantom, "\perp"] & \textbf{potential data}
\end{tikzcd}$$

necessarily **indefinitely**

necessarily $P_\bullet$ **entails** **actually $P_\bullet$** **entails** **possibly $P_\bullet$**

$$\Box_B P_\bullet \xrightarrow{\;\epsilon^{\Box_B}_{P_\bullet}\;} P_\bullet \xrightarrow{\;\eta^{\Diamond_B}_{P_\bullet}\;} \Diamond_B P_\bullet$$

$$b : B \;\vdash\; \prod_{b':B} P_{b'} \xrightarrow{(p_{b'})_{b':B} \mapsto p_b} P_b \xrightarrow{p_b \mapsto (p_b)_b} \coprod_{b':B} P_{b'}$$

randomly P **entails** **potentially P** **entails** **indefinitely P**

$$\star_B P \xrightarrow{\;\epsilon^{\star_B}_{P}\;} P \xrightarrow{\;\eta^{\bigcirc_B}_{P}\;} \bigcirc_B P$$

$$\coprod_{b:B} P \xrightarrow{(p)_b \mapsto p} P \xrightarrow{p \mapsto (p)_{b:B}} \prod_{b:B} P$$

Given B : Type of possible measurement outcomes (“possible worlds”)
the monadic effects of B-dependent linear data type formers constitute modalities of **actual and potential quantum B-measurements**.

quantum
data types

possibly

randomly

$$\begin{tikzcd}
\textbf{actual data} \perp & \mathrm{LType}_B \simeq \mathrm{LType}^{\bigcirc_B} \arrow[loop above, "\Diamond_B"] \arrow[loop below, "\Box_B"'] \arrow[r, phantom] \arrow[dr, phantom] & (\mathrm{LType}_B)_{\Diamond_B} \arrow[r, hook, "\sim"] & (\mathrm{LType}_B)^{\Diamond_B} \arrow[dr, phantom, "\simeq" sloped] & & \\
 & & \mathrm{LType}_{\bigcirc_B} \arrow[ull, phantom] \arrow[ul, hook'] \arrow[u, phantom, "\perp"] \arrow[d, phantom, "\perp"] & & \mathrm{LType} \arrow[ll, two heads, "\mathbb{1}_B \otimes"'] \arrow[loop above, "\star_B"] \arrow[loop below, "\bigcirc_B"] & \perp \textbf{ potential data} \\
 & & (\mathrm{LType}_B)_{\Box_B} \arrow[r, hook, "\sim"'] & (\mathrm{LType}_B)^{\Box_B} \arrow[ur, phantom, "\simeq" sloped] & &
\end{tikzcd}$$

$$\begin{tikzcd}
\mathrm{LType}_B \arrow[r, two heads, "\oplus_B"] & (\mathrm{LType}_B)_{\Diamond_B} \\
\mathrm{LType}_B \arrow[r, two heads, "\oplus_B"'] & (\mathrm{LType}_B)_{\Box_B}
\end{tikzcd}$$

necessarily

indefinitely

Given B : Type of possible measurement outcomes (“possible worlds”) **the monadic effects of B-dependent linear data type formers** constitute modalities of **actual and potential quantum B-measurements**.

quantum data types

possibly **randomly**

$$\begin{tikzcd}
 & & (\mathrm{LType}_B)_{\Diamond_B} \arrow[r, hook, "\sim"] \arrow[d, phantom, "\perp"] & (\mathrm{LType}_B)^{\Diamond_B} \arrow[dr, phantom, "\sim" sloped] & \\
\text{actual data } \perp \;\; \mathrm{LType}_B \arrow[loop above, "\Diamond_B"] \arrow[loop below, "\Box_B"'] \arrow[rru, two heads, "\oplus_B"] \arrow[rrd, two heads, "\oplus_B"'] \arrow[r, phantom, "\simeq"] & \mathrm{LType}^{\bigcirc_B} & \mathrm{LType}_{\bigcirc_B} \arrow[l, hook'] \arrow[d, phantom, "\perp"] & & \mathrm{LType} \;\; \perp \text{ potential data} \arrow[ll, two heads, "\mathbb{1}_B \otimes"'] \arrow[loop above, "\star_B"] \arrow[loop below, "\bigcirc_B"'] \\
 & & (\mathrm{LType}_B)_{\Box_B} \arrow[r, hook, "\sim"'] & (\mathrm{LType}_B)^{\Box_B} \arrow[ur, phantom, "\sim" sloped] &
\end{tikzcd}$$

necessarily **indefinitely**

---

necessarily $\mathcal{H}_\bullet$

$\Box_B \mathcal{H}_\bullet$

Given... **obtain...**

$b : B \;\vdash\; \mathcal{H}$

measurement result

where $\mathcal{H} := \bigoplus_{b':B} \mathcal{H}_{b'}$

Given B : Type of possible measurement outcomes (“possible worlds”)
the monadic effects of B-dependent linear data type formers constitute modalities of **actual and potential quantum B-measurements**.

quantum data types

possibly **randomly**

actual data $\perp$ $\perp$ **potential data**

$$\begin{tikzcd}
 & & (\mathrm{LType}_B)_{\Diamond_B} \arrow[r, hook, "\sim"] & (\mathrm{LType}_B)^{\Diamond_B} & \\
\mathrm{LType}_B \arrow[loop above, "\Diamond_B"] \arrow[loop below, "\Box_B"'] \arrow[r, phantom, "\simeq"] \arrow[urr, two heads, "\oplus_B"] \arrow[drr, two heads, "\oplus_B"'] & \mathrm{LType}^{\bigcirc_B} & \mathrm{LType}_{\bigcirc_B} \arrow[l, hook'] \arrow[u, phantom, "\perp"] \arrow[d, phantom, "\perp"] & & \mathrm{LType} \arrow[ll, two heads, "\mathbb{1}_B \otimes"'] \arrow[ul, phantom, "\sim"] \arrow[dl, phantom, "\sim"] \arrow[loop above, "\star_B"] \arrow[loop below, "\bigcirc_B"] \\
 & & (\mathrm{LType}_B)_{\Box_B} \arrow[r, hook, "\sim"'] & (\mathrm{LType}_B)^{\Box_B} &
\end{tikzcd}$$

necessarily **indefinitely**

---

necessarily $\mathcal{H}_\bullet$ **entails** **actually $\mathcal{H}_\bullet$**

$$\Box_B \mathcal{H}_\bullet \xrightarrow{\;\epsilon^{\Box_B}_{\mathcal{H}_\bullet}\;} \mathcal{H}_\bullet$$

Given... **obtain...**

$$b : B \;\vdash\; \mathcal{H} \xrightarrow[\text{measurement collapse}]{\sum_{b'} |\psi_{b'}\rangle \;\mapsto\; |\psi_b\rangle} \mathcal{H}_b \qquad \text{where } \mathcal{H} := \bigoplus_{b':B} \mathcal{H}_{b'}$$

measurement result

Given B : Type of possible measurement outcomes (“possible worlds”)
the monadic effects of B-dependent linear data type formers constitute modalities of **actual and potential quantum B-measurements**.

quantum data types

possibly **randomly**

actual data $\perp$ **potential data** $\perp$

$$\begin{tikzcd}
 & (\mathrm{LType}_B)_{\lozenge_B} \arrow[r, hook, "\sim"] \arrow[d, phantom, "\perp"] & (\mathrm{LType}_B)^{\lozenge_B} \arrow[dr, phantom, "\sim" sloped] & \\
\mathrm{LType}_B \simeq \mathrm{LType}^{\bigcirc_B} \arrow[ur, two heads, "\oplus_B"] \arrow[dr, two heads, "\oplus_B"'] \arrow[loop above, "\lozenge_B"] \arrow[loop below, "\Box_B"] & \mathrm{LType}_{\bigcirc_B} \arrow[l, hook'] \arrow[d, phantom, "\perp"] & & \mathrm{LType} \arrow[ll, two heads, "\mathbb{1}_B \otimes"'] \arrow[loop above, "\star_B"] \arrow[loop below, "\bigcirc_B"] \\
 & (\mathrm{LType}_B)_{\Box_B} \arrow[r, hook, "\sim"'] & (\mathrm{LType}_B)^{\Box_B} \arrow[ur, phantom, "\sim" sloped] &
\end{tikzcd}$$

necessarily **indefinitely**

---

necessarily $\mathcal{H}_\bullet$ **entails** **actually $\mathcal{H}_\bullet$** **entails** **possibly $\mathcal{H}_\bullet$**

$$\begin{tikzcd}
\Box_B \mathcal{H}_\bullet \arrow[r, "\varepsilon^{\Box_B}_{\mathcal{H}_\bullet}"] & \mathcal{H}_\bullet \arrow[r, "\eta^{\lozenge_B}_{\mathcal{H}_\bullet}"] & \lozenge_B \mathcal{H}_\bullet
\end{tikzcd}$$

Given... **obtain...**

$$\begin{tikzcd}
b : B \quad \vdash & \mathcal{H} \arrow[r, two heads, "\sum_{b'} |\psi_{b'}\rangle \;\mapsto\; |\psi_b\rangle", "\text{measurement collapse}"'] \arrow[rr, bend right, "\text{linear projector onto sub-Hilbert space } \mathcal{H}_b"'] & \mathcal{H}_b \arrow[r, hook, "|\psi_b\rangle \mapsto \oplus_{b'} \begin{cases} |\psi_b\rangle & \text{if } b'=b \\ 0 & \text{else} \end{cases}", "\text{state preparation}"'] & \mathcal{H},
\end{tikzcd}$$

measurement result

$$\text{where } \mathcal{H} := \bigoplus_{b':B} \mathcal{H}_{b'}$$

Given B : Type of possible measurement outcomes ("possible worlds")
the monadic effects of B-dependent linear data type formers constitute
modalities of **actual and potential quantum B-measurements**.

quantum data types

$$\begin{tikzcd}
 & & (\mathrm{LType}_B)_{\Diamond_B} \arrow[d, phantom, "\perp"] \arrow[r, hook, "\sim"] & (\mathrm{LType}_B)^{\Diamond_B} \arrow[dr, phantom, "\simeq"] & & \\
\textbf{actual data} \; \perp & \mathrm{LType}_B \simeq \mathrm{LType}^{\bigcirc_B} \arrow[ur, ->>, "\oplus_B"] \arrow[dr, ->>, "\oplus_B"'] \arrow[loop above, "\Diamond_B \; \text{possibly}"] \arrow[loop below, "\Box_B \; \text{necessarily}"'] & \mathrm{LType}_{\bigcirc_B} \arrow[l, hook'] \arrow[d, phantom, "\perp"] & & \mathrm{LType} \arrow[ll, ->>, "\mathbb{1}_B \otimes"'] \arrow[loop above, "\text{randomly} \; \star_B"] \arrow[loop below, "\bigcirc_B \; \text{indefinitely}"'] & \perp \; \textbf{potential data} \\
 & & (\mathrm{LType}_B)_{\Box_B} \arrow[r, hook, "\sim"'] & (\mathrm{LType}_B)^{\Box_B} \arrow[ur, phantom, "\simeq"] & &
\end{tikzcd}$$

principle of quantum compulsion:

necessarily $\mathcal{H}_\bullet$ **entails** **actually $\mathcal{H}_\bullet$** **entails** **possibly $\mathcal{H}_\bullet$** **is** **necessarily $\mathcal{H}_\bullet$**

$$\Box_B \mathcal{H}_\bullet \xrightarrow{\;\epsilon^{\Box_B}_{\mathcal{H}_\bullet}\;} \mathcal{H}_\bullet \xrightarrow{\;\eta^{\Diamond_B}_{\mathcal{H}_\bullet}\;} \Diamond_B \mathcal{H}_\bullet \underset{\text{ambidexterity}}{\simeq} \Box_B \mathcal{H}_\bullet$$

Given... **obtain...**

$$\underset{\textbf{measurement result}}{b : B} \;\vdash\; \begin{tikzcd}
\mathcal{H} \arrow[r, ->>, "\sum_{b'} |\psi_{b'}\rangle \;\mapsto\; |\psi_b\rangle", "\text{measurement collapse}"'] \arrow[rr, bend right, "\text{linear projector onto sub-Hilbert space } \mathcal{H}_b"'] & \mathcal{H}_b \arrow[r, hook, "|\psi_b\rangle \mapsto \oplus_{b'} \begin{cases} |\psi_b\rangle & \text{if } b'{=}b \\ 0 & \text{else} \end{cases}", "\text{state preparation}"'] & \mathcal{H},
\end{tikzcd} \qquad \text{where } \mathcal{H} := \bigoplus_{b':B} \mathcal{H}_{b'}$$

Given B : Type of possible measurement outcomes (“possible worlds”)
the monadic effects of B-dependent linear data type formers constitute modalities of **actual and potential quantum B-measurements**.

quantum data types

possibly **randomly**

$$\begin{tikzcd}
 & & (\mathrm{LType}_B)_{\lozenge_B} \arrow[r, hook, "\sim"] \arrow[d, phantom, "\perp"] & (\mathrm{LType}_B)^{\lozenge_B} \arrow[dr, phantom, sloped, "\simeq"] & \\
\mathrm{LType}_B \arrow[loop above, "\lozenge_B"] \arrow[loop below, "\square_B"'] \arrow[r, phantom, "\simeq"] \arrow[urr, two heads, "\oplus_B"] \arrow[drr, two heads, "\oplus_B"'] & \mathrm{LType}^{\bigcirc_B} & \mathrm{LType}_{\bigcirc_B} \arrow[l, hook'] \arrow[d, phantom, "\perp"] & & \mathrm{LType} \arrow[ll, two heads, "\mathbb{1}_B \otimes"'] \arrow[loop above, "\star_B"] \arrow[loop below, "\bigcirc_B"'] \\
 & & (\mathrm{LType}_B)_{\square_B} \arrow[r, hook, "\sim"'] & (\mathrm{LType}_B)^{\square_B} \arrow[ur, phantom, sloped, "\simeq"] &
\end{tikzcd}$$

actual data $\perp$ LType_B … LType $\perp$ **potential data**

necessarily **indefinitely**

---

principle of quantum compulsion:

necessarily $\mathcal{H}_\bullet$ **entails** **actually $\mathcal{H}_\bullet$** **entails** **possibly $\mathcal{H}_\bullet$** **is** **necessarily $\mathcal{H}_\bullet$**

$$\begin{tikzcd}
\square_B \mathcal{H}_\bullet \arrow[r, "\epsilon^{\square_B}_{\mathcal{H}_\bullet}"] & \mathcal{H}_\bullet \arrow[r, "\eta^{\lozenge_B}_{\mathcal{H}_\bullet}"] & \lozenge_B \mathcal{H}_\bullet \arrow[r, phantom, "\simeq"] & \square_B \mathcal{H}_\bullet
\end{tikzcd}$$

ambidexterity

Given... **obtain...**

$$\begin{tikzcd}
b : B \arrow[r, phantom, "\vdash"] & \mathcal{H} \arrow[r, two heads, "\sum_{b'} \lvert \psi_{b'} \rangle \;\mapsto\; \lvert \psi_b \rangle", "\text{measurement collapse}"'] \arrow[rr, bend right, "\text{linear projector onto sub-Hilbert space } \mathcal{H}_b"'] & \mathcal{H}_b \arrow[r, hook, "\lvert \psi_b \rangle \mapsto \oplus_{b'} \begin{cases} \lvert \psi_b \rangle & \text{if } b' = b \\ 0 & \text{else} \end{cases}", "\text{state preparation}"'] & \mathcal{H},
\end{tikzcd}$$

where $\mathcal{H} := \bigoplus\limits_{b':B} \mathcal{H}_{b'}$

measurement result

linear projector onto sub-Hilbert space $\mathcal{H}_b$

---

randomly $\mathcal{H}$

$\star_B \mathcal{H}$

$\bigoplus\limits_{b:B} \mathcal{H}$

Given B : Type of possible measurement outcomes ("possible worlds")
the monadic effects of B-dependent linear data type formers constitute modalities of **actual and potential quantum B-measurements**.

quantum
data types

possibly **randomly**

$$\begin{tikzcd}
 & (\mathrm{LType}_B)_{\lozenge_B} \arrow[r, hook, "\sim"] \arrow[d, phantom, "\perp"] & (\mathrm{LType}_B)^{\lozenge_B} & \\
\mathrm{LType}_B \simeq \mathrm{LType}^{\bigcirc_B} \arrow[loop above, "\lozenge_B"] \arrow[loop below, "\square_B"'] \arrow[ur, two heads, "\bigoplus_B"] \arrow[dr, two heads, "\bigoplus_B"'] & \mathrm{LType}_{\bigcirc_B} \arrow[l, hook'] \arrow[d, phantom, "\perp"] & & \mathrm{LType} \arrow[ll, two heads, "\mathbb{1}_B \otimes"'] \arrow[loop above, "\star_B"] \arrow[loop below, "\bigcirc_B"'] \arrow[ul, "\sim" sloped] \arrow[dl, "\sim" sloped] \\
 & (\mathrm{LType}_B)_{\square_B} \arrow[r, hook, "\sim"'] & (\mathrm{LType}_B)^{\square_B} &
\end{tikzcd}$$

actual data $\perp$ $\perp$ **potential data**

necessarily **indefinitely**

principle of quantum compulsion:

necessarily $\mathcal{H}_\bullet$ **entails** **actually $\mathcal{H}_\bullet$** **entails** **possibly $\mathcal{H}_\bullet$** **is** **necessarily $\mathcal{H}_\bullet$**

$$\square_B \mathcal{H}_\bullet \xrightarrow{\epsilon^{\square_B}_{\mathcal{H}_\bullet}} \mathcal{H}_\bullet \xrightarrow{\eta^{\lozenge_B}_{\mathcal{H}_\bullet}} \lozenge_B \mathcal{H}_\bullet \underset{\text{ambidexterity}}{\simeq} \square_B \mathcal{H}_\bullet$$

Given... **obtain...**

$$\begin{tikzcd}
b : B \vdash & \mathcal{H} \arrow[r, two heads, "\Sigma_{b'} \vert \psi_{b'} \rangle \;\mapsto\; \vert \psi_b \rangle", "\text{measurement collapse}"'] \arrow[rr, bend right, "\text{linear projector onto sub-Hilbert space } \mathcal{H}_b"'] & \mathcal{H}_b \arrow[r, hook, "\vert \psi_b \rangle \mapsto \oplus_{b'} \begin{cases} \vert \psi_b \rangle & \text{if } b' = b \\ 0 & \text{else} \end{cases}", "\text{state preparation}"'] & \mathcal{H},
\end{tikzcd}$$

measurement result

$$\text{where } \mathcal{H} := \bigoplus_{b' : B} \mathcal{H}_{b'}$$

randomly $\mathcal{H}$ **entails** **potentially $\mathcal{H}$**

$$\star_B \mathcal{H} \xrightarrow{\epsilon^{\star_B}_{\mathcal{H}}} \mathcal{H}$$

$$\bigoplus_{b:B} \mathcal{H} \xrightarrow[\text{quantum superposition}]{\oplus_b \vert \psi_b \rangle \mapsto \Sigma_b \vert \psi_b \rangle} \mathcal{H}$$

Given B : Type of possible measurement outcomes ("possible worlds")
the monadic effects of B-dependent linear data type formers constitute
modalities of **actual and potential quantum B-measurements**.

quantum
data types

possibly **randomly**

actual data $\perp$ $\perp$ **potential data**

necessarily **indefinitely**

$$\begin{tikzcd}
 & (\mathrm{LType}_B)_{\lozenge_B} \arrow[r, hook, "\sim"] \arrow[d, phantom, "\perp"] & (\mathrm{LType}_B)^{\lozenge_B} \\
\mathrm{LType}_B \simeq \mathrm{LType}^{\bigcirc_B} \arrow[ur, two heads, "\bigoplus_B"] \arrow[dr, two heads, "\bigoplus_B"'] \arrow[loop above, "\lozenge_B"] \arrow[loop below, "\Box_B"'] & \mathrm{LType}_{\bigcirc_B} \arrow[l, hook'] \arrow[d, phantom, "\perp"] & \mathrm{LType} \arrow[l, two heads, "\mathbb{1}_B \otimes"'] \arrow[u, phantom, sloped, "\simeq"] \arrow[d, phantom, sloped, "\simeq"] \arrow[loop above, "\star_B"] \arrow[loop below, "\bigcirc_B"'] \\
 & (\mathrm{LType}_B)_{\Box_B} \arrow[r, hook, "\sim"'] & (\mathrm{LType}_B)^{\Box_B}
\end{tikzcd}$$

---

principle of quantum compulsion:

necessarily $\mathcal{H}_\bullet$ **entails** **actually $\mathcal{H}_\bullet$** **entails** **possibly $\mathcal{H}_\bullet$** **is** **necessarily $\mathcal{H}_\bullet$**

$$\begin{tikzcd}
\Box_B \mathcal{H}_\bullet \arrow[r, "\epsilon^{\Box_B}_{\mathcal{H}_\bullet}"] & \mathcal{H}_\bullet \arrow[r, "\eta^{\lozenge_B}_{\mathcal{H}_\bullet}"] & \lozenge_B \mathcal{H}_\bullet \arrow[r, phantom, "\simeq"] & \Box_B \mathcal{H}_\bullet
\end{tikzcd}$$

ambidexterity

Given... **obtain...**

$$\begin{tikzcd}
b : B \arrow[r, phantom, "\vdash"] & \mathcal{H} \arrow[r, two heads, "\Sigma_{b'}\,|\psi_{b'}\rangle \;\mapsto\; |\psi_b\rangle", "\text{measurement collapse}"'] \arrow[rr, bend right, "\text{linear projector onto sub-Hilbert space } \mathcal{H}_b"'] & \mathcal{H}_b \arrow[r, hook, "|\psi_b\rangle \mapsto \oplus_{b'} \begin{cases} |\psi_b\rangle & \text{if } b'{=}b \\ 0 & \text{else} \end{cases}", "\text{state preparation}"'] & \mathcal{H},
\end{tikzcd}$$

measurement result

where $\mathcal{H} := \bigoplus\limits_{b':B} \mathcal{H}_{b'}$

---

randomly $\mathcal{H}$ **entails** **potentially $\mathcal{H}$** **entails** **indefinitely $\mathcal{H}$**

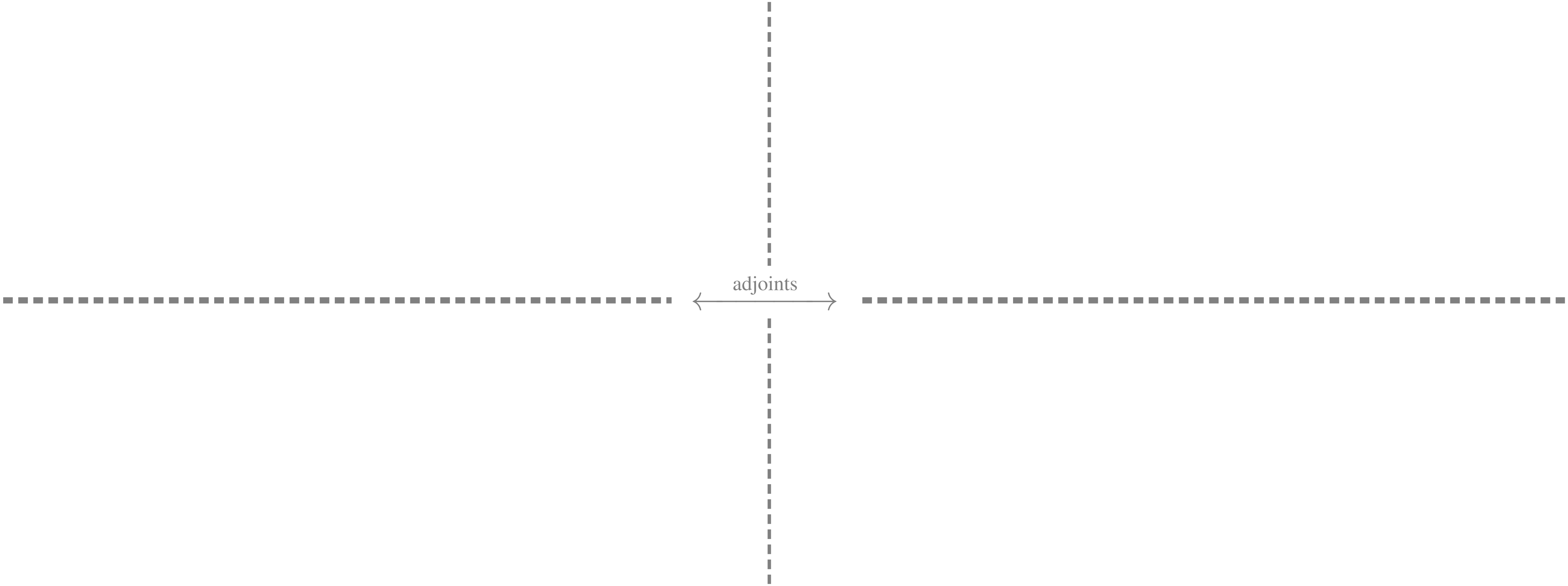
$$\begin{tikzcd}
\star_B \mathcal{H} \arrow[r, "\epsilon^{\star_B}_{\mathcal{H}}"] & \mathcal{H} \arrow[r, "\eta^{\bigcirc_B}_{\mathcal{H}}"] & \bigcirc_B \mathcal{H}
\end{tikzcd}$$

$$\begin{tikzcd}
\bigoplus\limits_{b:B} \mathcal{H} \arrow[r, "\oplus_b\,|\psi_b\rangle \mapsto \Sigma_b\,|\psi_b\rangle", "\text{quantum superposition}"'] & \mathcal{H} \arrow[r, "|\psi\rangle \mapsto \oplus_b |\psi\rangle_b", "\text{quantum parallelization}"'] & \bigoplus\limits_{b:B} \mathcal{H}
\end{tikzcd}$$

The pure effects of these modalities of dependent linear data type formation

are remarkable in their sheer quantum information-theoretic content.

To repeat:



The pure effects of these modalities of dependent linear data type formation

are remarkable in their sheer quantum information-theoretic content.

To repeat:

$$\overbrace{(p_B)^*(p_B)_*}^{\Box_B}\mathcal{H}_\bullet \xrightarrow[\textbf{necessity counit}]{\varepsilon^{\Box_B}_{\mathcal{H}_\bullet}} \mathcal{H}_\bullet$$

$$b : B \vdash \bigoplus_{b':B} \mathcal{H}_{b'} \xrightarrow[\text{quantum } \textbf{measurement}]{\oplus_{b'}|\psi_{b'}\rangle \;\mapsto\; |\psi_b\rangle} \mathcal{H}_b$$

“ the necessary becomes actual ”

adjoints

$\longleftrightarrow$

The pure effects of these modalities of dependent linear data type formation

are remarkable in their sheer quantum information-theoretic content.

To repeat:

$$\overbrace{(p_B)^*(p_B)_*}^{\Box_B}\mathcal{H}_\bullet \xrightarrow[\textbf{necessity counit}]{\epsilon^{\Box_B}_{\mathcal{H}_\bullet}} \mathcal{H}_\bullet$$

$$b : B \vdash \bigoplus_{b':B}\mathcal{H}_{b'} \xrightarrow[\text{quantum }\textbf{measurement}]{\oplus_{b'}|\psi_{b'}\rangle \;\mapsto\; |\psi_b\rangle} \mathcal{H}_b$$

“ the necessary becomes actual ”

$$\xleftrightarrow{\text{adjoints}}$$

$$\mathcal{H}_\bullet \xrightarrow[\textbf{possibility unit}]{\eta^{\Diamond_B}_{\mathcal{H}_\bullet}} \overbrace{(p_B)^*(p_B)_!}^{\Diamond_B}\mathcal{H}_\bullet$$

$$b : B \vdash \mathcal{H}_b \xrightarrow[\text{quantum }\textbf{state preparation}]{|\psi_b\rangle \;\mapsto\; \oplus_{b'}\begin{cases}|\psi_b\rangle & \text{if } b'{=}b\\ 0 & \text{else}\end{cases}} \bigoplus_{b':B}\mathcal{H}_{b'}$$

“ the actual is possible ”

The pure effects of these modalities of dependent linear data type formation

---

are remarkable in their sheer quantum information-theoretic content.

To repeat:

$$\overbrace{(p_B)^*(p_B)_*}^{\Box_B}\mathcal{H}_\bullet \xrightarrow[\textbf{necessity counit}]{\varepsilon^{\Box_B}_{\mathcal{H}_\bullet}} \mathcal{H}_\bullet$$

$$b : B \vdash \bigoplus_{b':B}\mathcal{H}_{b'} \xrightarrow[\text{quantum }\textbf{measurement}]{\oplus_{b'}|\psi_{b'}\rangle \;\mapsto\; |\psi_b\rangle} \mathcal{H}_b$$

“ the necessary becomes actual ”

$$\mathcal{H}_\bullet \xrightarrow[\textbf{possibility unit}]{\eta^{\Diamond_B}_{\mathcal{H}_\bullet}} \overbrace{(p_B)^*(p_B)_!}^{\Diamond_B}\mathcal{H}_\bullet$$

$$b : B \vdash \mathcal{H}_b \xrightarrow[\text{quantum }\textbf{state preparation}]{|\psi_b\rangle \;\mapsto\; \oplus_{b'}\begin{cases}|\psi_b\rangle & \text{if } b'{=}b\\ 0 & \text{else}\end{cases}} \bigoplus_{b':B}\mathcal{H}_{b'}$$

“ the actual is possible ”

adjoints

$\longleftrightarrow$

“ the random becomes potential ”

$$\overbrace{(p_B)_!(p_B)^*}^{\star_B}\mathcal{H} \xhookrightarrow[\textbf{randomness counit}]{\varepsilon^{\star_B}_{\mathcal{H}}} \mathcal{H}$$

$$\bigoplus_{b:B}\mathcal{H} \xrightarrow[\text{quantum }\textbf{superposition}]{\oplus_b|\psi_b\rangle \;\mapsto\; \Sigma_b|\psi_b\rangle} \mathcal{H}$$

The pure effects of these modalities of dependent linear data type formation

are remarkable in their sheer quantum information-theoretic content.

To repeat:

$$\begin{tikzcd} \overbrace{(p_B)^*(p_B)_*}^{\Box_B}\mathcal{H}_\bullet \arrow[rr, "\varepsilon^{\Box_B}_{\mathcal{H}_\bullet}", "\textbf{necessity counit}"'] && \mathcal{H}_\bullet \end{tikzcd}$$

$$b : B \vdash \begin{tikzcd} \bigoplus\limits_{b':B}\mathcal{H}_{b'} \arrow[rr, "\oplus_{b'}|\psi_{b'}\rangle \;\mapsto\; |\psi_b\rangle", "\text{quantum } \textbf{measurement}"'] && \mathcal{H}_b \end{tikzcd}$$

“ the necessary becomes actual ”

$$\begin{tikzcd} \mathcal{H}_\bullet \arrow[rr, "\eta^{\Diamond_B}_{\mathcal{H}_\bullet}", "\textbf{possibility unit}"'] && \overbrace{(p_B)^*(p_B)_!}^{\Diamond_B}\mathcal{H}_\bullet \end{tikzcd}$$

$$b : B \vdash \begin{tikzcd} \mathcal{H}_b \arrow[rr, "|\psi_b\rangle \;\mapsto\; \oplus_{b'}\begin{cases} |\psi_b\rangle & \text{if } b'{=}b \\ 0 & \text{else} \end{cases}", "\text{quantum } \textbf{state preparation}"'] && \bigoplus\limits_{b':B}\mathcal{H}_{b'} \end{tikzcd}$$

“ the actual is possible ”

adjoints ⟷

“ the random becomes potential ”

$$\begin{tikzcd} \overbrace{(p_B)_!(p_B)^*}^{\star_B}\mathcal{H} \arrow[rr, hook, "\varepsilon^{\star_B}_{\mathcal{H}}", "\textbf{randomness counit}"'] && \mathcal{H} \end{tikzcd}$$

$$\begin{tikzcd} \bigoplus\limits_{b:B}\mathcal{H} \arrow[rr, "\oplus_b|\psi_b\rangle \;\mapsto\; \Sigma_b|\psi_b\rangle", "\text{quantum } \textbf{superposition}"'] && \mathcal{H} \end{tikzcd}$$

“ the potential is inderminate ”

$$\begin{tikzcd} \mathcal{H} \arrow[rr, "\eta^{\bigcirc_B}_{\mathcal{H}}", "\textbf{indeterminacy unit}"'] && \overbrace{(p_B)_*(p_B)^*}^{\bigcirc_B}\mathcal{H} \end{tikzcd}$$

$$\begin{tikzcd} \mathcal{H} \arrow[rr, "|\psi\rangle \;\mapsto\; \oplus_b|\psi\rangle_b", "\text{quantum } \textbf{parallelism}"'] && \bigoplus\limits_{b:B}\mathcal{H} \end{tikzcd}$$

Q-bits are the free linear indeterminacy-effect handlers over $\mathrm{Bool} = \{0, 1\}$

Coherent q-bits:

$$\begin{tikzcd}
\mathrm{QBit} \arrow[d, equal, "\cdot\cdot"] & : & \mathrm{LType} \arrow[r, hook, "\mathbb{1}_{\mathrm{Bool}}^{\otimes}"] & \mathrm{LType}_{\mathrm{Bool}} \arrow[r, "\bigoplus_{\mathrm{Bool}}", "\sim"'] & \mathrm{LType}^{\bigcirc_B} \\
\bigcirc_{\mathrm{Bool}}\, \mathbb{1}
\end{tikzcd}$$

De-cohered (measured) q-bits:

Quantum gate with q-bit output:

Q-bits are the free linear indeterminacy-effect handlers over $\mathrm{Bool} = \{0,1\}$

Coherent q-bits:

$$\mathrm{QBit} \;:\; \mathrm{LType} \xhookrightarrow{\mathbb{1}_{\mathrm{Bool}} \otimes} \mathrm{LType}_{\mathrm{Bool}} \xrightarrow[\sim]{\bigoplus_{\mathrm{Bool}}} \mathrm{LType}^{\bigcirc_B}$$

$$\overset{\cdot\cdot}{\Vert}$$

$$\bigcirc_{\mathrm{Bool}} \mathbb{1} \;=\; \bigoplus_{\{0,1\}} \mathbb{C} \;=\; \mathbb{C} \cdot |0\rangle \oplus \mathbb{C} \cdot |1\rangle$$

Quantum gate with q-bit output:

De-cohered (measured) q-bits:

Q-bits are the free linear indeterminacy-effect handlers over $\mathrm{Bool} = \{0, 1\}$

Coherent q-bits:

$$\begin{tikzcd} \mathrm{QBit} \arrow[d, phantom, "\rotatebox{90}{$=:$}"] & : \mathrm{LType} \arrow[r, hook, "\mathbb{1}_{\mathrm{Bool}}{}^{\otimes}"] & \mathrm{LType}_{\mathrm{Bool}} \arrow[r, "\bigoplus_{\mathrm{Bool}}", "\sim"'] & \mathrm{LType}^{\bigcirc_B} \\ \bigcirc_{\mathrm{Bool}} \mathbb{1} & = \bigoplus_{\{0,1\}} \mathbb{C} = \mathbb{C} \cdot \lvert 0 \rangle \oplus \mathbb{C} \cdot \lvert 1 \rangle & & \end{tikzcd}$$

Quantum gate with q-bit output:

De-cohered (measured) q-bits:

Q-bits are the free linear indeterminacy-effect handlers over $\mathrm{Bool} = \{0,1\}$

Coherent q-bits:

$$\mathrm{QBit} \;:\; \mathrm{LType} \xhookrightarrow{\mathbb{1}_{\mathrm{Bool}}\otimes} \mathrm{LType}_{\mathrm{Bool}} \xrightarrow[\sim]{\bigoplus_{\mathrm{Bool}}} \mathrm{LType}^{\bigcirc_B}$$

$$\overset{\cdot\cdot}{\|}$$

$$\bigcirc_{\mathrm{Bool}}\mathbb{1} \;=\; \bigoplus_{\{0,1\}}\mathbb{C} \;=\; \mathbb{C}\cdot|0\rangle \oplus \mathbb{C}\cdot|1\rangle$$

$$\mathrm{QBit}$$

$$\otimes$$

$$\mathcal{H}$$

$$\|$$

$$\bigcirc_{\mathrm{Bool}}\mathcal{H} \;=\; \bigoplus_{\{0,1\}}\mathcal{H} \;=\; \mathcal{H}\otimes|0\rangle \;\oplus\; \mathcal{H}\otimes|0\rangle$$

De-cohered (measured) q-bits:

Quantum gate with q-bit output:

Q-bits are the free linear indeterminacy-effect handlers over $\mathrm{Bool} = \{0,1\}$

Coherent q-bits:

$$\mathrm{QBit} \;:\; \mathrm{LType} \xhookrightarrow{\mathbb{1}_{\mathrm{Bool}}\otimes} \mathrm{LType}_{\mathrm{Bool}} \xrightarrow[\sim]{\bigoplus_{\mathrm{Bool}}} \mathrm{LType}^{\bigcirc_B}$$

$$\overset{\cdot\cdot}{\|}$$

$$\bigcirc_{\mathrm{Bool}}\mathbb{1} \;=\; \bigoplus_{\{0,1\}}\mathbb{C} \;=\; \mathbb{C}\cdot|0\rangle \oplus \mathbb{C}\cdot|1\rangle$$

$$\mathrm{QBit}$$

$$\otimes$$

$$\mathcal{H}$$

$$\|$$

$$\bigcirc_{\mathrm{Bool}}\mathcal{H} \;=\; \bigoplus_{\{0,1\}}\mathcal{H} \;=\; \mathcal{H}\otimes|0\rangle \;\oplus\; \mathcal{H}\otimes|0\rangle$$

De-cohered (measured) q-bits:

$$\mathbb{1}_{\mathrm{Bool}} \;:\; \mathrm{LType}_{\mathrm{Bool}} \xrightarrow[\sim]{\bigoplus_{\mathrm{Bool}}} \mathrm{LType}^{\bigcirc_{\mathrm{Bool}}}$$

$$b : \mathrm{Bool} \quad \vdash \quad \mathbb{C}\cdot|b\rangle \;:\; \mathrm{LType}$$

Quantum gate with q-bit output:

Q-bits are the free linear indeterminacy-effect handlers over $\mathrm{Bool} = \{0,1\}$

Coherent q-bits:

$$\mathrm{QBit} \;:\; \mathrm{LType} \xhookrightarrow{\mathbb{1}_{\mathrm{Bool}}\otimes} \mathrm{LType}_{\mathrm{Bool}} \underset{\sim}{\xrightarrow{\bigoplus_{\mathrm{Bool}}}} \mathrm{LType}^{\bigcirc_B}$$

$$\overset{\cdot\cdot}{\|}$$

$$\bigcirc_{\mathrm{Bool}}\mathbb{1} \;=\; \bigoplus_{\{0,1\}}\mathbb{C} \;=\; \mathbb{C}\cdot|0\rangle \oplus \mathbb{C}\cdot|1\rangle$$

$$\mathrm{QBit}$$

$$\otimes$$

$$\mathcal{H}$$

$$\|$$

$$\bigcirc_{\mathrm{Bool}}\mathcal{H} \;=\; \bigoplus_{\{0,1\}}\mathcal{H} \;=\; \mathcal{H}\otimes|0\rangle \;\oplus\; \mathcal{H}\otimes|0\rangle$$

De-cohered (measured) q-bits:

$$\mathbb{1}_{\mathrm{Bool}} \;:\; \mathrm{LType}_{\mathrm{Bool}} \underset{\sim}{\xrightarrow{\bigoplus_{\mathrm{Bool}}}} \mathrm{LType}^{\bigcirc_{\mathrm{Bool}}}$$

$$b : \mathrm{Bool} \quad \vdash \quad \mathbb{C}\cdot|b\rangle \;:\; \mathrm{LType}$$

$$\mathbb{1}_{\mathrm{Bool}}$$

$$\otimes \qquad b : \mathrm{Bool} \quad \vdash \quad \mathcal{H}\otimes|b\rangle \;:\; \mathrm{LType}$$

$$\mathcal{H}$$

Quantum gate with q-bit output:

Q-bits are the free linear indeterminacy-effect handlers over $\mathrm{Bool} = \{0,1\}$

Coherent q-bits:

$$\mathrm{QBit} \;:\; \mathrm{LType} \xhookrightarrow{\mathbb{1}_{\mathrm{Bool}}\otimes} \mathrm{LType}_{\mathrm{Bool}} \xrightarrow[\sim]{\bigoplus_{\mathrm{Bool}}} \mathrm{LType}^{\bigcirc_B}$$

$$\overset{\cdot\cdot}{\|}$$

$$\bigcirc_{\mathrm{Bool}}\mathbb{1} \;=\; \bigoplus_{\{0,1\}}\mathbb{C} \;=\; \mathbb{C}\cdot|0\rangle \oplus \mathbb{C}\cdot|1\rangle$$

$$\mathrm{QBit}$$

$$\otimes$$

$$\mathcal{H}$$

$$\|$$

$$\bigcirc_{\mathrm{Bool}}\mathcal{H} \;=\; \bigoplus_{\{0,1\}}\mathcal{H} \;=\; \mathcal{H}\otimes|0\rangle \;\oplus\; \mathcal{H}\otimes|0\rangle$$

De-cohered (measured) q-bits:

$$\mathbb{1}_{\mathrm{Bool}} \;:\; \mathrm{LType}_{\mathrm{Bool}} \xrightarrow[\sim]{\bigoplus_{\mathrm{Bool}}} \mathrm{LType}^{\bigcirc_{\mathrm{Bool}}}$$

$$b : \mathrm{Bool} \quad\vdash\quad \mathbb{C}\cdot|b\rangle \;:\; \mathrm{LType}$$

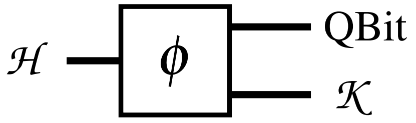
$$\mathbb{1}_{\mathrm{Bool}}$$

$$\otimes \qquad b : \mathrm{Bool} \quad\vdash\quad \mathcal{H}\otimes|b\rangle \;:\; \mathrm{LType}$$

$$\mathcal{H}$$

Quantum gate with q-bit output:

A quantum gate which may handle $\bigcirc_{\mathrm{Bool}}$-effects is one with a QBit-output:



$$\mathcal{H} \xrightarrow{\phi} \mathrm{QBit}\otimes\mathcal{K} \simeq \bigcirc_{\mathrm{Bool}}\mathcal{K}$$

Q-bits are the free linear indeterminacy-effect handlers over $\mathrm{Bool} = \{0,1\}$

Coherent q-bits:

$$\mathrm{QBit} \;:\; \mathrm{LType} \xhookrightarrow{\;\mathbb{1}_{\mathrm{Bool}}\otimes\;} \mathrm{LType}_{\mathrm{Bool}} \xrightarrow[\sim]{\;\bigoplus_{\mathrm{Bool}}\;} \mathrm{LType}^{\bigcirc_B}$$

$$\rotatebox{90}{$=:$}$$

$$\bigcirc_{\mathrm{Bool}}\mathbb{1} \;=\; \bigoplus_{\{0,1\}}\mathbb{C} \;=\; \mathbb{C}\cdot|0\rangle \oplus \mathbb{C}\cdot|1\rangle$$

$$\mathrm{QBit}$$

$$\otimes$$

$$\mathcal{H}$$

$$\shortparallel$$

$$\bigcirc_{\mathrm{Bool}}\mathcal{H} \;=\; \bigoplus_{\{0,1\}}\mathcal{H} \;=\; \mathcal{H}\otimes|0\rangle \;\oplus\; \mathcal{H}\otimes|0\rangle$$

De-cohered (measured) q-bits:

$$\mathbb{1}_{\mathrm{Bool}} \;:\; \mathrm{LType}_{\mathrm{Bool}} \xrightarrow[\sim]{\;\bigoplus_{\mathrm{Bool}}\;} \mathrm{LType}^{\bigcirc_{\mathrm{Bool}}}$$

$$b : \mathrm{Bool} \quad\vdash\quad \mathbb{C}\cdot|b\rangle \;:\; \mathrm{LType}$$

$$\mathbb{1}_{\mathrm{Bool}}$$

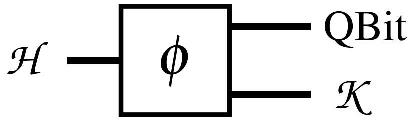
$$\otimes$$

$$\mathcal{H}$$

$$b : \mathrm{Bool} \quad\vdash\quad \mathcal{H}\otimes|b\rangle \;:\; \mathrm{LType}$$

Quantum gate with q-bit output:

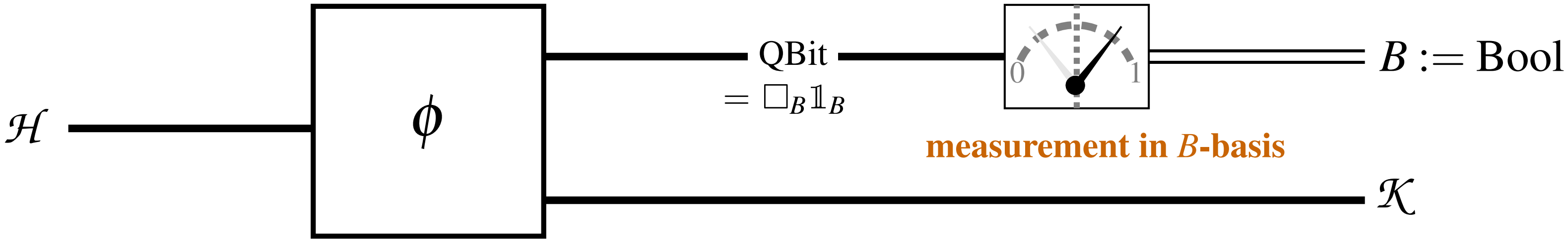
A quantum gate which
may handle $\bigcirc_{\mathrm{Bool}}$-effects
is one with a QBit-output:



$$\mathcal{H} \xrightarrow{\;\phi\;} \mathrm{QBit}\otimes\mathcal{K} \simeq \bigcirc_{\mathrm{Bool}}\mathcal{K}$$

Quantum measurement is Linear indefiniteness-effect handling.

quantum circuit



quantum gate

$$\mathcal{H} \xrightarrow{\phi} \mathrm{QBit} \otimes \mathcal{K} \simeq \bigcirc_B \mathcal{K}$$

$\bigcirc_B$-effect handling

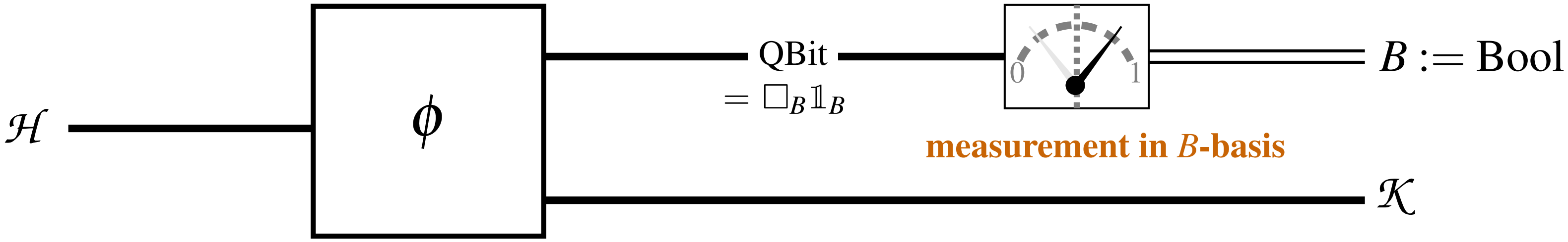
Quantum measurement is Linear indefiniteness-effect handling.

quantum circuit

$\rightsquigarrow$ formalization

$\bigcirc_B$-modal linear types

$\mathrm{LType}_{\bigcirc_B}$



quantum gate

$$\mathcal{H} \xrightarrow{\phi} \mathrm{QBit} \otimes \mathcal{K} \simeq \bigcirc_B \mathcal{K}$$

$\bigcirc_B$-effect handling

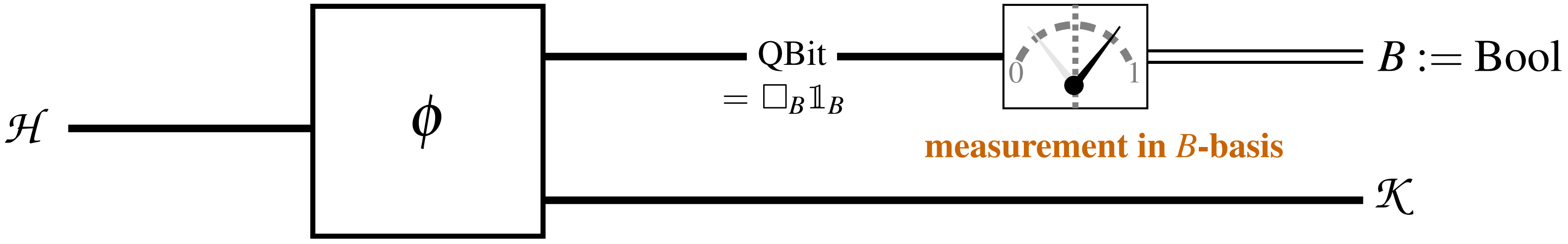
Quantum measurement is Linear indefiniteness-effect handling.

quantum circuit

$$\begin{tikzcd} \text{quantum circuit} \arrow[d, squiggly, "\text{formalization}"] \\ \bigcirc_B\text{-modal linear types} \\ \mathrm{LType}_{\bigcirc_B} \arrow[d, hook, "K_{(p_B)_*(p_B)^*}"', "\text{comparison functor}"] \\ \mathrm{LType}_B \\ B\text{-dependent linear types} \end{tikzcd}$$

$\bigcirc_B$-modal linear types

B-dependent linear types



measurement in B-basis

subjection to $\bigcirc_B$-effects

quantum gate

$$\begin{tikzcd} \bigcirc_B \mathcal{H} \arrow[loop above, "\bigcirc_B"] \arrow[r, "\bigcirc_B\left(\mathcal{H} \xrightarrow{\phi} \mathrm{QBit} \otimes \mathcal{K} \simeq \bigcirc_B \mathcal{K}\right)"] & \bigcirc_B \bigcirc_B \mathcal{K} \arrow[loop above, "\bigcirc_B"] \arrow[r, "\mathrm{hndl}^{\bigcirc_B}_{\bigcirc_B \mathcal{K}}", "\bigcirc_B\text{-effect handling}"'] & \bigcirc_B \mathcal{K} \arrow[loop above, "\bigcirc_B"] \end{tikzcd}$$

Quantum measurement is Linear indefiniteness-effect handling.

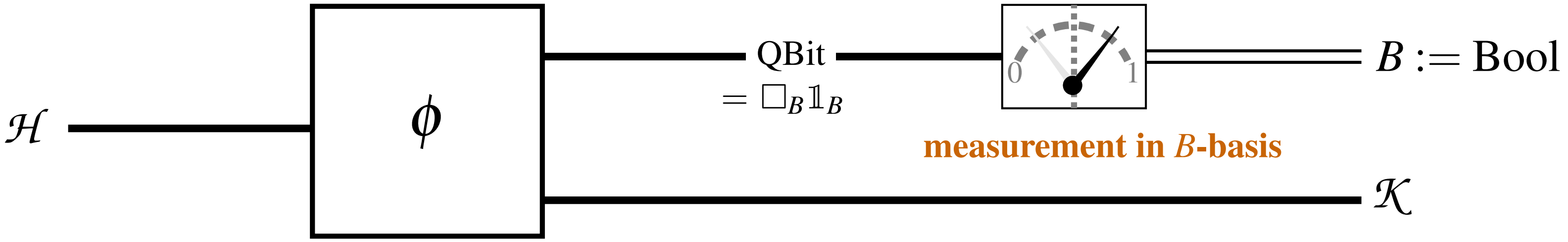
quantum circuit

$$\begin{tikzcd} \text{quantum circuit} \arrow[d, squiggly, "\text{formalization}"] \\ \bigcirc_B\text{-modal linear types} \end{tikzcd}$$

$\bigcirc_B$-modal linear types

$$\begin{tikzcd} \mathrm{LType}_{\bigcirc_B} \arrow[d, "K_{(p_B)^*(p_B)_*}"', "\text{comparison functor}"] \\ \mathrm{LType}_B \end{tikzcd}$$

B-dependent linear types



measurement in B-basis

$$\begin{tikzcd}[column sep=huge] \bigcirc_B \mathcal{H} \arrow[loop above, "\bigcirc_B"] \arrow[r, "\bigcirc_B\left(\mathcal{H} \xrightarrow{\phi} \mathrm{QBit} \otimes \mathcal{K} \simeq \bigcirc_B \mathcal{K}\right)", "\text{quantum gate}"'] & \bigcirc_B \bigcirc_B \mathcal{K} \arrow[loop above, "\bigcirc_B"] \arrow[r, "\mathrm{hndl}^{\bigcirc_B}_{\bigcirc_B \mathcal{K}}", "\bigcirc_B\text{-effect handling}"'] & \bigcirc_B \mathcal{K} \arrow[loop above, "\bigcirc_B"] \end{tikzcd}$$

subjection to $\bigcirc_B$-effects

quantum gate

$\bigcirc_B$-effect handling

$$\begin{tikzcd}[column sep=huge] \mathbb{1}_B \otimes \mathcal{H} \arrow[r, "\mathbb{1}_B \otimes \phi"] & \square_B \mathbb{1}_B \otimes \mathcal{K} \arrow[r, "\epsilon^{\square_B}_{\mathbb{1}_B} \otimes \mathcal{K}", "\square_B\text{-counit}"'] & \mathbb{1}_B \otimes \mathcal{K} \end{tikzcd}$$

$\square_B$-counit

$$\begin{tikzcd}[column sep=huge] b : B \vdash \mathcal{H} \arrow[r, "\phi"] & \mathrm{QBit} \otimes \mathcal{K} \arrow[r, "\text{quantum state collapse}"] & \mathcal{K} \end{tikzcd}$$

measurement result

quantum state collapse

$$\begin{array}{c} |0\rangle \otimes |\psi_0\rangle \\ + |1\rangle \otimes |\psi_1\rangle \end{array} \mapsto |\psi_b\rangle$$

Quantum measurement is Linear indefiniteness-effect handling.

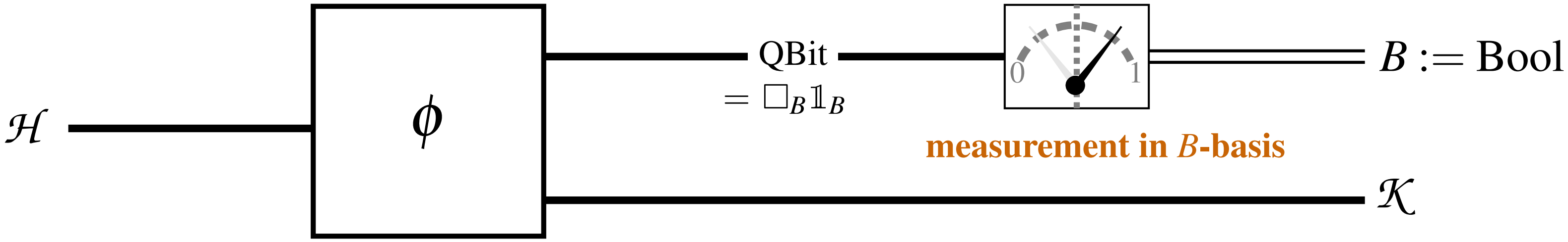
quantum circuit

$$\begin{tikzcd} \text{quantum circuit} \arrow[d, squiggly, "\text{formalization}"] \\ \bigcirc_B\text{-modal linear types} \end{tikzcd}$$

$\bigcirc_B$-modal linear types

$$\begin{tikzcd} \mathrm{LType}_{\bigcirc_B} \arrow[d, "K_{(p_B)^*(p_B)_*}"', "\text{comparison functor}"] \\ \mathrm{LType}_B \end{tikzcd}$$

B-dependent linear types



measurement in B-basis

$$\begin{tikzcd}[column sep=huge] \bigcirc_B \mathcal{H} \arrow[loop above, "\bigcirc_B"] \arrow[r, "\bigcirc_B\left(\mathcal{H} \xrightarrow{\phi} \mathrm{QBit} \otimes \mathcal{K} \simeq \bigcirc_B \mathcal{K}\right)"] & \bigcirc_B \bigcirc_B \mathcal{K} \arrow[loop above, "\bigcirc_B"] \arrow[r, "\mathrm{hndl}^{\bigcirc_B}_{\bigcirc_B \mathcal{K}}", "\bigcirc_B\text{-effect handling}"'] & \bigcirc_B \mathcal{K} \arrow[loop above, "\bigcirc_B"] \end{tikzcd}$$

subjection to $\bigcirc_B$-effects

quantum gate

$\bigcirc_B$-effect handling

$$\begin{tikzcd}[column sep=huge] \mathbb{1}_B \otimes \mathcal{H} \arrow[r, "\mathbb{1}_B \otimes \phi"] & \Box_B \mathbb{1}_B \otimes \mathcal{K} \arrow[r, "\epsilon^{\Box_B}_{\mathbb{1}_B} \otimes \mathcal{K}", "\Box_B\text{-counit}"'] & \mathbb{1}_B \otimes \mathcal{K} \end{tikzcd}$$

$\Box_B$-counit

$$b : B \;\vdash\; \begin{tikzcd}[column sep=huge] \mathcal{H} \arrow[r, "\phi"] & \mathrm{QBit} \otimes \mathcal{K} \arrow[r, "\text{quantum state collapse}"] & \mathcal{K} \end{tikzcd}$$

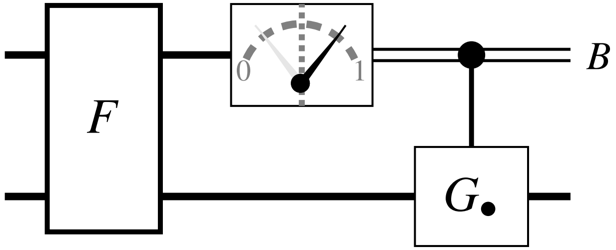
measurement result

quantum state collapse

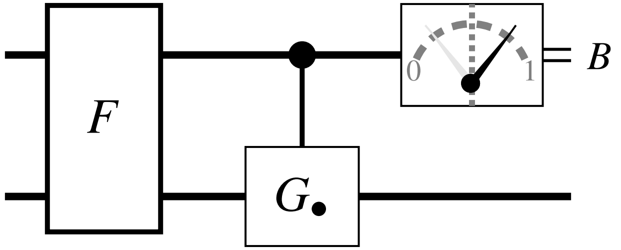
$$\begin{array}{l} |0\rangle \otimes |\psi_0\rangle \\ + |1\rangle \otimes |\psi_1\rangle \end{array} \mapsto |\psi_b\rangle$$

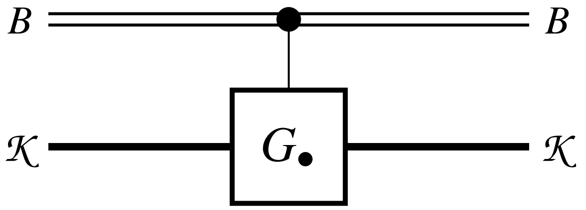
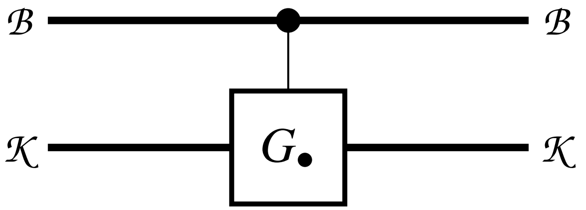
full linearly-typed detail of
quantum measurement logic
is emergent effect in dLHoTT

E.g.: Deferred measurement principle – Proven by monadic effect logic.



$$\xleftrightarrow{\text{Deferred Measurement Principle}}$$

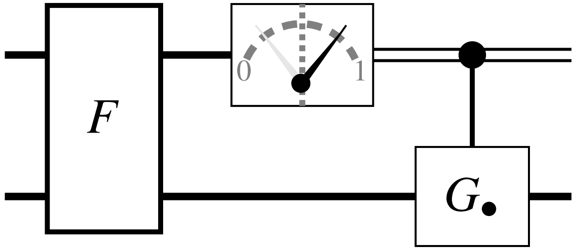


classically controlled gate	**quantumly controlled gate**
 $\mathcal{B}_\bullet \boxtimes \mathcal{K} \xrightarrow{G_\bullet} \mathcal{B}_\bullet \boxtimes \mathcal{K}$ $b : B \ \vdash \quad \mathcal{K} \xrightarrow{G_b} \mathcal{K}$	 $\Box_B \mathcal{B}_\bullet \boxtimes \mathcal{K} \xrightarrow{\Box_B G_\bullet} \Box_B \mathcal{B}_\bullet \boxtimes \mathcal{K}$ $b : B \ \vdash \quad \bigoplus_{b' B} \mathcal{K} \xrightarrow{\bigoplus_{b' : B} G_{b'}} \bigoplus_{b' B} \mathcal{K}$

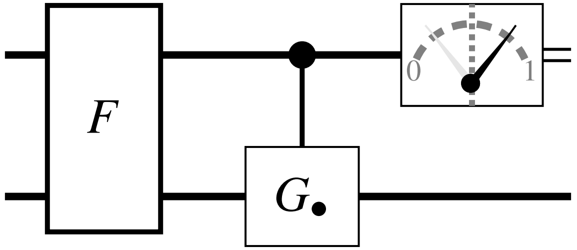
E.g.: Deferred measurement principle – Proven by monadic effect logic.

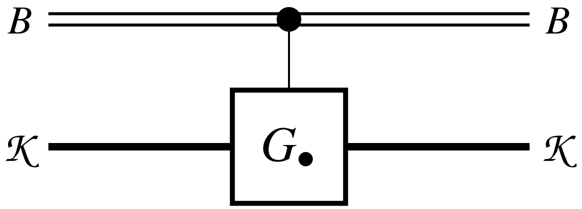
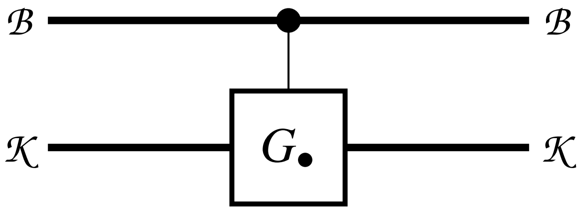
$$\Box_B \mathcal{H}_\bullet \xrightarrow{F} \Box_B \mathcal{H}_\bullet \xrightarrow{\varepsilon^{\Box_B}_{\mathcal{H}_\bullet}} \mathcal{H}_\bullet \xrightarrow{G_\bullet} \mathcal{H}_\bullet \quad\mapsto\quad \Box_B \mathcal{H}_\bullet \xrightarrow{\mathrm{diag}_B(G_\bullet)\circ F} \Box_B \mathcal{H}_\bullet \quad\mapsto\quad \Box_B \mathcal{H}_\bullet \xrightarrow{\mathrm{diag}_B(G_\bullet)\circ F} \Box_B \mathcal{H}_\bullet \xrightarrow{\varepsilon^{\Box_B}_{\mathcal{H}_\bullet}} \mathcal{H}_\bullet$$

measurement-controlled quantum gate **quantum-controlled quantum gate...** **...followed by measurement**



$\longleftrightarrow$ **Deferred Measurement Principle**



classically controlled gate	**quantumly controlled gate**
 $\mathcal{B}_\bullet \boxtimes \mathcal{K} \xrightarrow{G_\bullet} \mathcal{B}_\bullet \boxtimes \mathcal{K}$ $b : B \;\vdash\; \mathcal{K} \xrightarrow{G_b} \mathcal{K}$	 $\Box_B \mathcal{B}_\bullet \boxtimes \mathcal{K} \xrightarrow{\Box_B G_\bullet} \Box_B \mathcal{B}_\bullet \boxtimes \mathcal{K}$ $b : B \;\vdash\; \bigoplus_{b' B} \mathcal{K} \xrightarrow{\bigoplus_{b':B} G_{b'}} \bigoplus_{b' B} \mathcal{K}$

E.g.: Deferred measurement principle – Proven by monadic effect logic.

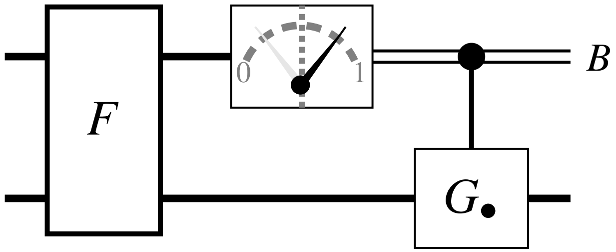
$$\begin{tikzcd}
\mathrm{Kl}(\Box_B) \arrow[rr, bend left, "\mathrm{id}"] \arrow[r, "\sim", "\delta^B \circ \Box_B(-)"'] & \mathrm{LType}_{B\Box_B} \arrow[r, "\sim", "\varepsilon^{\Box_B} \circ (-)"'] & \mathrm{Kl}(\Box_B)
\end{tikzcd}$$

Kleisli equivalence

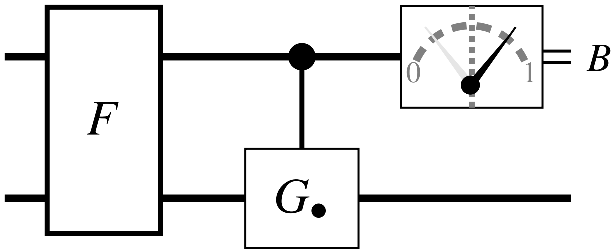
$\Box_B$-Kleisli morphisms **$\Box_B$-coalgebra homomorphisms** **$\Box_B$-Kleisli morphisms**

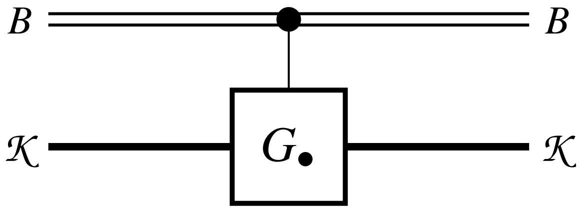
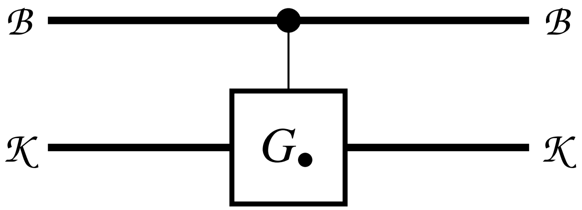
$$\Box_B\mathcal{H}_\bullet \xrightarrow{F} \Box_B\mathcal{H}_\bullet \xrightarrow{\varepsilon^{\Box_B}_{\mathcal{H}_\bullet}} \mathcal{H}_\bullet \xrightarrow{G_\bullet} \mathcal{H}_\bullet \quad \mapsto \quad \Box_B\mathcal{H}_\bullet \xrightarrow{\mathrm{diag}_B(G_\bullet)\circ F} \Box_B\mathcal{H}_\bullet \quad \mapsto \quad \Box_B\mathcal{H}_\bullet \xrightarrow{\mathrm{diag}_B(G_\bullet)\circ F} \Box_B\mathcal{H}_\bullet \xrightarrow{\varepsilon^{\Box_B}_{\mathcal{H}_\bullet}} \mathcal{H}_\bullet$$

measurement-controlled quantum gate **quantum-controlled quantum gate...** **...followed by measurement**



Deferred Measurement Principle ⟷



classically controlled gate	**quantumly controlled gate**
 $\mathcal{B}_\bullet \boxtimes \mathcal{K} \xrightarrow{G_\bullet} \mathcal{B}_\bullet \boxtimes \mathcal{K}$ $b : B \ \vdash \quad \mathcal{K} \xrightarrow{G_b} \mathcal{K}$	 $\Box_B\mathcal{B}_\bullet \boxtimes \mathcal{K} \xrightarrow{\Box_B G_\bullet} \Box_B\mathcal{B}_\bullet \boxtimes \mathcal{K}$ $b : B \ \vdash \quad \bigoplus_{b'B} \mathcal{K} \xrightarrow{\bigoplus_{b':B} G_{b'}} \bigoplus_{b'B} \mathcal{K}$

Also the Exponential modality

traditionally postulated in linear logic
is an emergent effect in `dLHoTT`

linear randomization

aka: stabilization/motivization

$$\begin{tikzcd}
\mathrm{Type} \arrow[r, shift left=2, "\Sigma^\infty_+ \,:\, B \,\mapsto\, \star_B \mathbb{1}"] \arrow[r, phantom, "\perp"] & \mathrm{LType} \arrow[l, shift left=2, "\Omega^\infty"] \arrow[loop right, "!"]
\end{tikzcd}$$

exponential modality

classical types

linear types

Also the Exponential modality

traditionally postulated in linear logic
is an emergent effect in `dLHoTT`

linear randomization

aka: stabilization/motivization

$$\begin{tikzcd} \mathrm{Type} \arrow[rr, shift left=2, "\Sigma^\infty_+ \,:\, B \,\mapsto\, \star_B \mathbb{1}"] \arrow[rr, phantom, "\perp"] & & \mathrm{LType} \arrow[ll, shift left=2, "\Omega^\infty"] \arrow[loop right, "!"] \end{tikzcd}$$

exponential modality

classical types

linear types

In summary, we see that:

The *Motive* or *Linear Randomization* of B : FinType is
the quantum data type spanned by eigenstates $|b\rangle, b : B$
equipped with the structure of a free effect handler
for quantum measurement logic in the B-basis.

$$\star_{\mathrm{Bool}} \mathbb{1} \;\simeq\; \bigcirc_{\mathrm{Bool}} \mathbb{1} \;\simeq\; \mathrm{QBit}$$

Quantum Circuits

Quantum effects are compatible with tensor product.

Linear Randomness and Indefiniteness are “very strong” effects, in that:

$$\bigcirc_B\big(D \otimes D'\big) \;\simeq\; \big(\bigcirc_B D\big) \otimes D' \,, \qquad \star_B\big(D \otimes D'\big) \;\simeq\; \big(\star_B D\big) \otimes D'$$

There is a whole system of them:

$$\bigcirc_B \bigcirc_{B'} \;\simeq\; \bigcirc_{B \times B'} , \qquad \text{NB: } \bigcirc_B \bigcirc'_B \;\simeq\; \bigcirc_B \mathbb{1} \otimes \bigcirc'_B$$

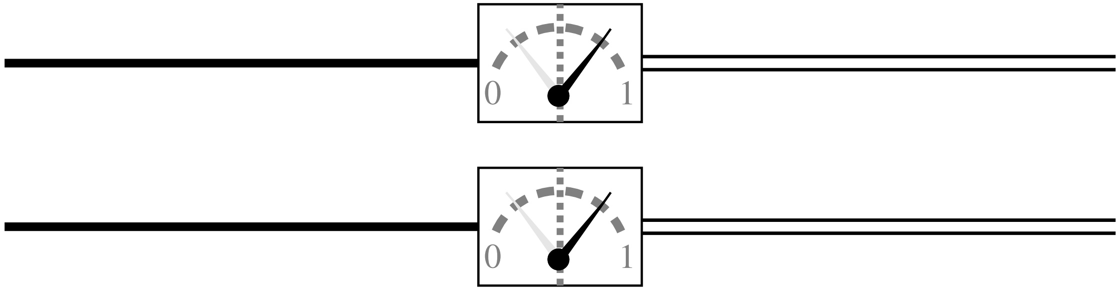
which under dynamic lifting (monadicity comparison functor)
gives the external tensor product of dependent linear types:

$$\begin{tikzcd}
\textbf{free } \bigcirc_B\textbf{-effect handlers in linear data types} & \mathrm{LType}_{\bigcirc_B} \arrow[d, hook, "K_{\oplus_B \mathbb{1}_B \otimes}"', "\textbf{comparison functor}"] & \bigcirc_{B \times B'} \bigcirc_{B \times B'} \mathcal{H} \arrow[loop above, "\bigcirc_{B \times B'}"] \arrow[d, mapsto] \\
\textbf{B-dependent linear data types} & \mathrm{LType}_B & \big(\Box_B \mathbb{1}_B\big) \underset{\textbf{external tensor product}}{\boxtimes} \big(\Box_{B'} \mathbb{1}_{B'}\big) \otimes \mathcal{H}
\end{tikzcd}$$

Quantum circuits with classical control & effects

are the *effectful* string diagrams in the linear type system

E.g.
The dependent linear type of a measurement on a pair of qbits:



type of a pair of coherent qbits

pair of measurements

type of collapsed qbits dependent on measured bits b, b'

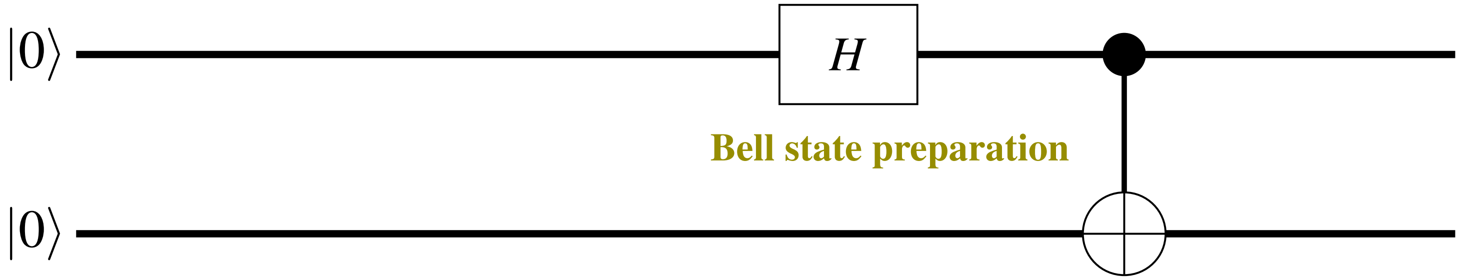
$$\square_{\mathrm{Bool}^2}\big(\mathrm{QBit}_\bullet \boxtimes \mathrm{QBit}_\bullet\big) \xrightarrow{\;\varepsilon_{\mathrm{Bool}^2}(\mathrm{QBit}_\bullet \boxtimes \mathrm{QBit}_\bullet)\;} \mathrm{QBit}_\bullet \boxtimes \mathrm{QBit}_\bullet$$

measured bits

$$(b, b') : \mathrm{Bool}^2 \;\;\vdash\;\; \square_{\mathrm{Bool}^2}\big(\mathrm{QBit}_\bullet \boxtimes \mathrm{QBit}_\bullet\big)_{(b,b')} \simeq \mathbb{C}^2 \otimes \mathbb{C}^2 \xrightarrow{\;\sum_{d,d'} q_{dd'}\, |d\rangle \otimes |d'\rangle \;\mapsto\; q_{bb'}\, |b\rangle \otimes |b'\rangle\;} \mathbb{C}\,.$$

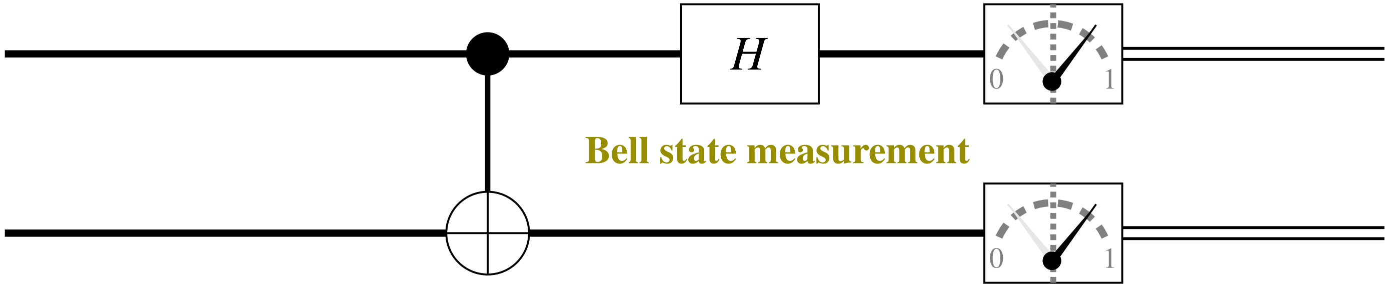
collapse of the quantum state

Example: Bell states of q-bits are typed as follows (regarded in $\mathrm{LType}_{\mathrm{Bool}\times\mathrm{Bool}}$):



$$\mathrm{QBit}_\bullet \boxtimes \mathrm{QBit}_\bullet \longrightarrow \big(\Diamond_{\mathrm{Bool}}\mathrm{QBit}_\bullet\big) \boxtimes \big(\Diamond_{\mathrm{Bool}}\mathrm{QBit}_\bullet\big) \simeq \Box_{\mathrm{Bool}^2}\big(\mathrm{QBit}_\bullet \boxtimes \mathrm{QBit}_\bullet\big) \longrightarrow \Box_{\mathrm{Bool}^2}\big(\mathrm{QBit}_\bullet \boxtimes \mathrm{QBit}_\bullet\big)$$

$$b, b' : \mathrm{Bool} \vdash \mathbb{C} \xrightarrow{\;1 \;\mapsto\; |0\rangle\otimes|0\rangle \;\mapsto\; \frac{1}{\sqrt{2}}\big(|0\rangle+|1\rangle\big)\otimes|0\rangle \;\mapsto\; \frac{1}{\sqrt{2}}\big(|0\rangle\otimes|0\rangle+|1\rangle\otimes|1\rangle\big)\;} \mathbb{C}^2\otimes\mathbb{C}^2$$

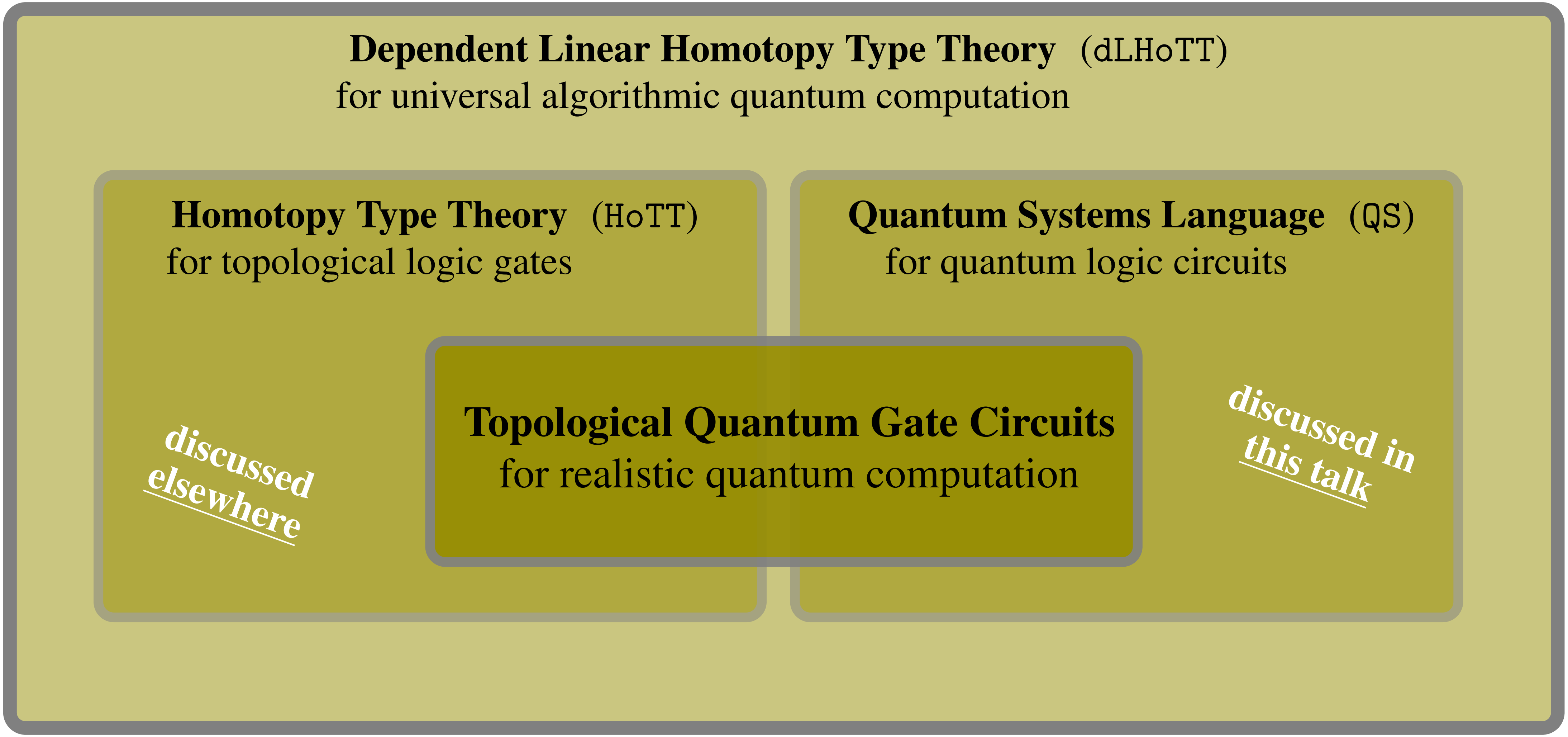


$$\Box_{\mathrm{Bool}^2}\big(\mathrm{QBit}_\bullet \boxtimes \mathrm{QBit}_\bullet\big) \longrightarrow \mathrm{QBit}_\bullet \boxtimes \mathrm{QBit}_\bullet$$

$$b_1, b_2 : \mathrm{Bool} \vdash \mathbb{C}^2\otimes\mathbb{C}^2 \xrightarrow{\;\sum_{b'_1 b'_2} q_{b'_1 b'_2}\cdot|b'_1\rangle\otimes|b'_2\rangle \;\mapsto\; \big(q_{0,b_2}+(-1)^{b_1}\cdot q_{1,(1-b_2)}\big)\cdot|b_1\rangle\otimes|b_2\rangle\;} \mathbb{C}$$

`QS` – Quantum Systems language @ CQTS

⇝ full-blown Quantum Systems language emerges embedded in `dLHoTT`



Dependent Linear Homotopy Type Theory (`dLHoTT`)
for universal algorithmic quantum computation

Homotopy Type Theory (`HoTT`)
for topological logic gates

Quantum Systems Language (`QS`)
for quantum logic circuits

Topological Quantum Gate Circuits
for realistic quantum computation

discussed elsewhere

discussed in this talk

Center for
Quantum &
Topological
Systems

Thanks!

Quantum Data Types via Linear HoTT

presentation at:

Workshop on Quantum Software @ QTML 2022

Urs Schreiber (NYU Abu Dhabi)
on joint work at CQTS with
D. J. Myers, M. Riley,
and Hisham Sati

slides and further pointers at: ncatlab.org/schreiber/show/QDataInLHoTT#QTML2022