A 2-categorical approach to logical relations

FioreFest, Cambridge, June 2026

These slides available at philipsaville.co.uk

Pedro H. Azevedo de Amorim (University of Bath, UK)
Satoshi Kura (Waseda University, Japan)
Philip Saville (University of Sussex, UK)
Linus Schönfelder (University of Sussex, UK)

UNIVERSITY OF SUSSEX BRIGHTON

UNIVERSITY OF BATH

WASEDA University

Things I learned from Marcelo

Things I learned from Marcelo

1. Look for the right abstractions.

Things I learned from Marcelo

1. Look for the right **abstractions**.
2. Let the algebra tell you **how things should be**.

"correct-by-construction"

"principled"

"synthesise"

Logical relations, at varying levels of abstraction

Level 1: logical relations as families of relations Milner, Plotkin, ….

A logical relation R consists of:

a predicate $R_A \subseteq [\![A]\!]$ for each type A…

…determined inductively at higher types by a logical relations condition

Level 1: logical relations as families of relations

Milner, Plotkin, ….

A logical relation R consists of:

a predicate $R_A \subseteq [\![A]\!]$ for each type A…

…determined inductively at higher types by a logical relations condition

E.g. logical relations condition for $\rightarrow$ in STLC:

$$f \in R_{A \rightarrow B} \subseteq [\![A]\!] \Rightarrow [\![B]\!] \quad \text{iff} \quad f(x) \in R_B \text{ whenever } x \in R_A$$

Level 1: logical relations as families of relations Milner, Plotkin, ...

A logical relation R consists of:

a predicate $R_A \subseteq [\![A]\!]$ for each type A...

...determined inductively at higher types by a logical relations condition

Basic Lemma:

every program is in the relation: $[\![P]\!] \in R_A$ for any closed $P : A$

so long as this holds for the basic constants

Level 2: logical relations as relations models

Statman,
Reynolds & Ma, ….

Level 2: logical relations as relations models

Statman,
Reynolds & Ma, ….

semantic model

Set

Level 2: logical relations as relations models

Statman,
Reynolds & Ma, ….

objects: $(X \in \mathbf{Set}, R \subseteq X)$
maps: functions preserving the predicate

$$\begin{array}{c}\mathbf{Pred}\\ \downarrow p\\ \mathbf{Set}\end{array}$$

semantic model

Level 2: logical relations as relations models

Statman,
Reynolds & Ma, ….

objects: $(X \in \mathbf{Set}, R \subseteq X)$
maps: functions preserving the predicate

$$\mathbf{Pred}$$
$$\downarrow p$$
$$\mathbf{Set}$$

semantic model

$\mathbf{Pred}$ is cartesian closed:

$$(X, R) \Rightarrow (Y, S) := (X \Rightarrow Y, R \supset S)$$

$f \in R \supset S$ iff $f(x) \in S$ whenever $x \in R$

$f \in [\![A \to B]\!]^{\mathbf{Pred}}$ iff f preserves the predicate on $[\![A]\!]^{\mathbf{Set}}$

Level 2: logical relations as relations models

Statman,
Reynolds & Ma, ….

objects: $(X \in \mathbf{Set}, R \subseteq X)$
maps: functions preserving the predicate

$$\begin{array}{c}\mathbf{Pred} \\ \big\downarrow p \\ \mathbf{Set}\end{array}$$

semantic model

Pred is cartesian closed:

$$(X, R) \Rightarrow (Y, S) := (X \Rightarrow Y, R \supset S)$$

$f \in R \supset S$ iff $f(x) \in S$ whenever $x \in R$

$f \in [\![A \to B]\!]^{\mathbf{Pred}}$ iff f preserves the predicate on $[\![A]\!]^{\mathbf{Set}}$

CC-structure encodes the logical relations condition!

Level 2: logical relations as relations models

Statman,
Reynolds & Ma, ….

objects: $(X \in \mathbf{Set}, R \subseteq X)$
maps: functions preserving the predicate

$$\mathbf{Pred}$$

$$\downarrow p$$

$$\mathbf{Set}$$

forgetful functor p
strictly preserves CC-structure

semantic model

Pred is cartesian closed:

$$(X, R) \Rightarrow (Y, S) := (X \Rightarrow Y, R \supset S)$$

$f \in R \supset S$ iff $f(x) \in S$ whenever $x \in R$

$f \in [\![A \to B]\!]^{\mathbf{Pred}}$ iff f preserves the predicate on $[\![A]\!]^{\mathbf{Set}}$

CC-structure encodes the logical relations condition!

Level 2: logical relations as relations models

Statman,
Reynolds & Ma, …

objects: $(X \in \mathbf{Set}, R \subseteq X)$
maps: functions preserving the predicate

$\mathbf{Pred}$

forgetful functor p
strictly preserves CC-structure

$\downarrow p$

semantic model

$\mathbf{Set}$

$\mathbf{Pred}$ is cartesian closed:

$$(X, R) \Rightarrow (Y, S) := (X \Rightarrow Y, R \supset S)$$

$f \in R \supset S$ iff $f(x) \in S$ whenever $x \in R$

$f \in [\![A \to B]\!]^{\mathbf{Pred}}$ iff f preserves the predicate on $[\![A]\!]^{\mathbf{Set}}$

CC-structure encodes the logical relations condition!

Basic Lemma holds automatically:

$$p\left([\![P]\!]^{\mathbf{Pred}}\right) = [\![P]\!]^{\mathbf{Set}}$$

Level 3: logical relations as certain fibrations

Hermida, Jacobs, Katsumata, …

Level 3: logical relations as certain fibrations

Hermida, Jacobs, Katsumata, ….

semantic model $\mathscr{M}$

Level 3: logical relations as certain fibrations

Hermida, Jacobs, Katsumata, ….

‘relations model’	$\widehat{\mathscr{M}}$
forgetful functor p strictly preserves model-structure	$\downarrow p$
semantic model	$\mathscr{M}$

Level 3: logical relations as certain fibrations

Hermida, Jacobs, Katsumata, ….

‘relations model’ = fibration

forgetful functor p
strictly preserves model-structure

semantic model

$$\begin{array}{c} \widehat{\mathscr{M}} \\ \downarrow p \\ \mathscr{M} \end{array}$$

Level 3: logical relations as certain fibrations

Hermida, Jacobs, Katsumata, ….

‘relations model’ = fibration

forgetful functor p
strictly preserves model-structure

semantic model

$$\begin{array}{c} \widehat{\mathscr{M}} \\ \downarrow p \\ \mathscr{M} \end{array}$$

model-structure encodes the logical relations condition

Level 3: logical relations as certain fibrations

Hermida, Jacobs, Katsumata, ….

‘relations model’ = fibration $\qquad \widehat{\mathscr{M}}$

forgetful functor p $\qquad \big\downarrow p$
strictly preserves model-structure

semantic model $\qquad \mathscr{M}$

model-structure encodes the
logical relations condition

Basic Lemma holds automatically:

$$p\left(\llbracket P \rrbracket^{\widehat{\mathscr{M}}}\right) = \llbracket P \rrbracket^{\mathscr{M}}$$

Level 3: logical relations as certain fibrations

Hermida, Jacobs, Katsumata, ….

‘relations model’ = fibration

forgetful functor p
strictly preserves model-structure

semantic model

$$\begin{array}{c} \widehat{\mathscr{M}} \\ \big\downarrow p \\ \mathscr{M} \end{array}$$

model-structure encodes the
logical relations condition

Basic Lemma holds automatically:

$$p\left(\llbracket P \rrbracket^{\widehat{\mathscr{M}}}\right) = \llbracket P \rrbracket^{\mathscr{M}}$$

fibration for logical relations:
fibration
strictly preserving
model structure

Logical relations can be seen as…

Level 1: type-indexed families of relations

Level 2: relations models

Level 3: fibrations for logical relations

fibration for logical relations:

fibration

strictly preserving

model structure

Logical relations can be seen as…

Level 1: type-indexed families of relations

Level 2: relations models

Level 3: fibrations for logical relations

also get:
a good theory for building
new logical relations

fibration for logical relations:
fibration
strictly preserving
model structure

Building fibrations for logical relations

e.g. effect simulation, ⊤⊤-lifting, Kripke relations of varying arity, …

fibration for logical relations:

fibration
strictly preserving
model structure

Building fibrations for logical relations

e.g. effect simulation, ⊤⊤-lifting, Kripke relations of varying arity, …

fibration for logical relations:

fibration
strictly preserving
model structure

semantic model $\mathscr{M}$

Building fibrations for logical relations

e.g. effect simulation, ⊤⊤-lifting, Kripke relations of varying arity, …

fibration for logical relations:

fibration
strictly preserving
model structure

semantic model $\mathscr{M}$

$$\begin{array}{c}\mathscr{E}\\ \downarrow p\\ \mathscr{B}\end{array}$$

abstract notion of 'relation'

ie. p a fibration for logical relations

Building fibrations for logical relations

e.g. effect simulation, ⊤⊤-lifting, Kripke relations of varying arity, …

fibration for logical relations:

fibration
strictly preserving
model structure

$$\begin{array}{ccc} & & \mathscr{E} \\ & & \downarrow{\scriptstyle p} \\ \mathscr{M} & \xrightarrow[F]{} & \mathscr{B} \end{array}$$

semantic model

abstract notion of ‘relation’

ie. p a fibration for logical relations

weak map of models

Building fibrations for logical relations

e.g. effect simulation, $\top\top$-lifting, Kripke relations of varying arity, …

fibration for logical relations:

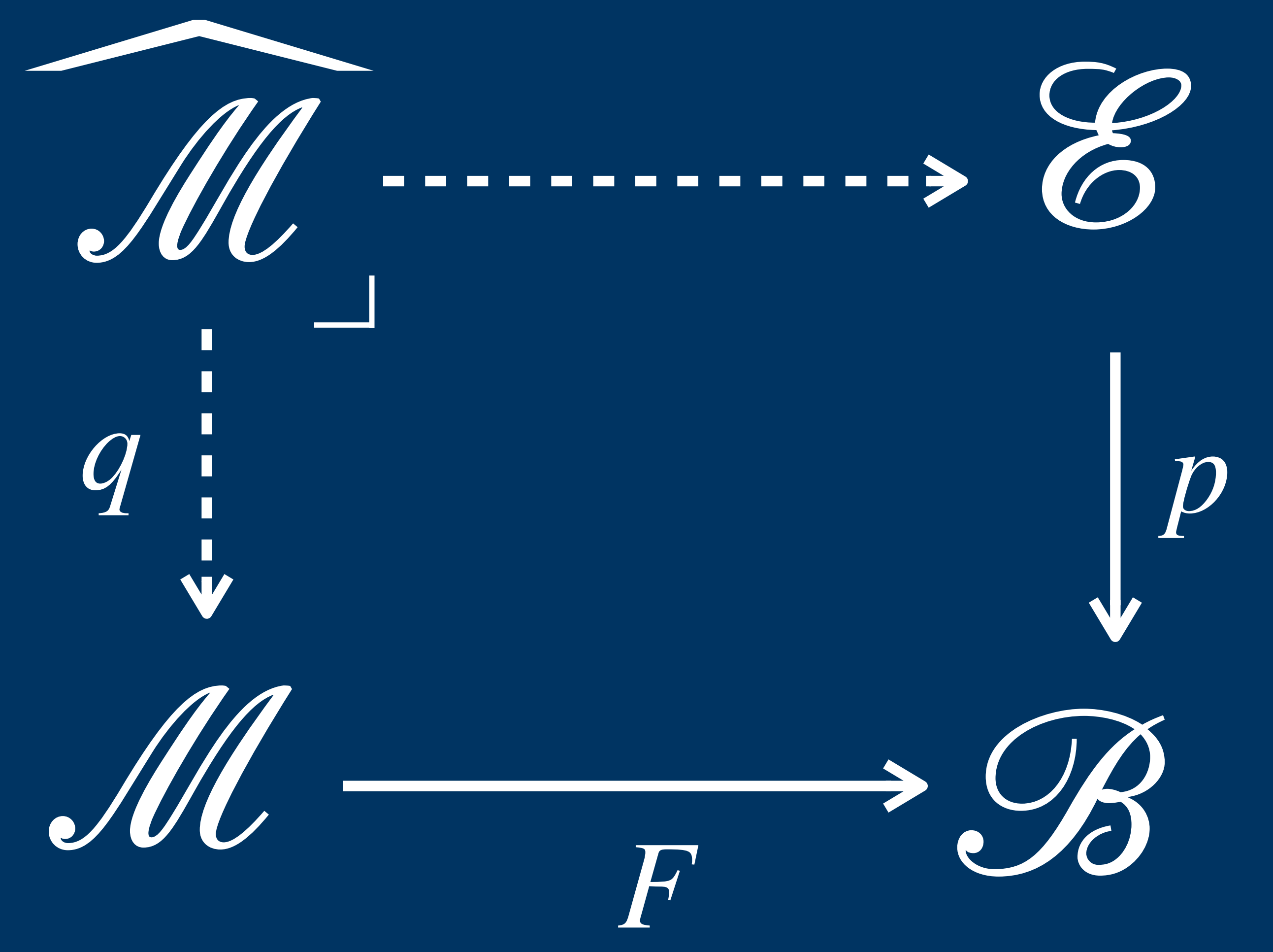
fibration
strictly preserving
model structure

lifting theorem:

universal choice of model

'glueing' objects of $\mathscr{M}$ to 'relations' in $\mathscr{E}$

semantic model



abstract notion of 'relation'

ie. p a fibration for logical relations

weak map of models

Building fibrations for logical relations

e.g. effect simulation, ⊤⊤-lifting, Kripke relations of varying arity, …

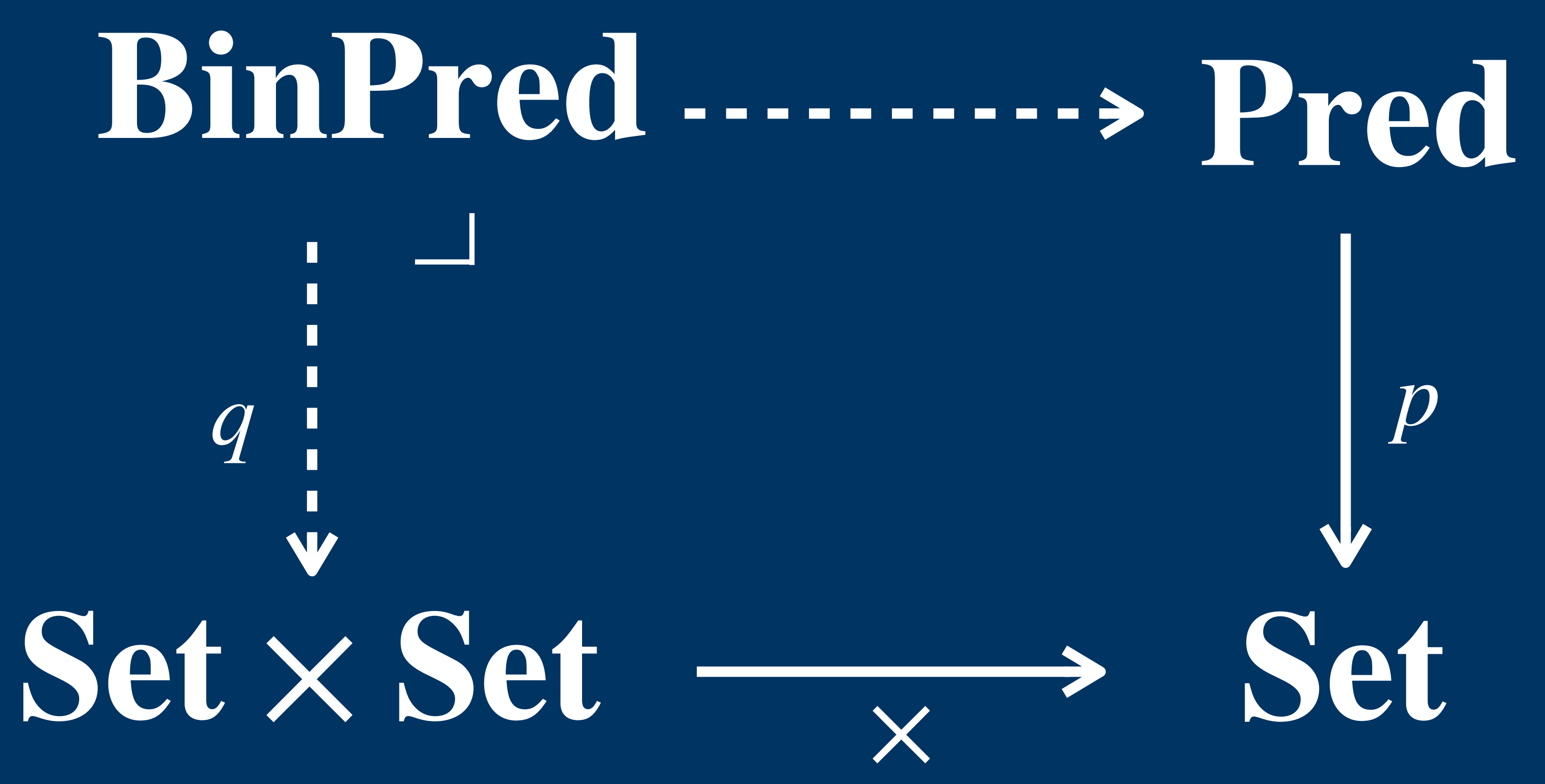
fibration for logical relations:

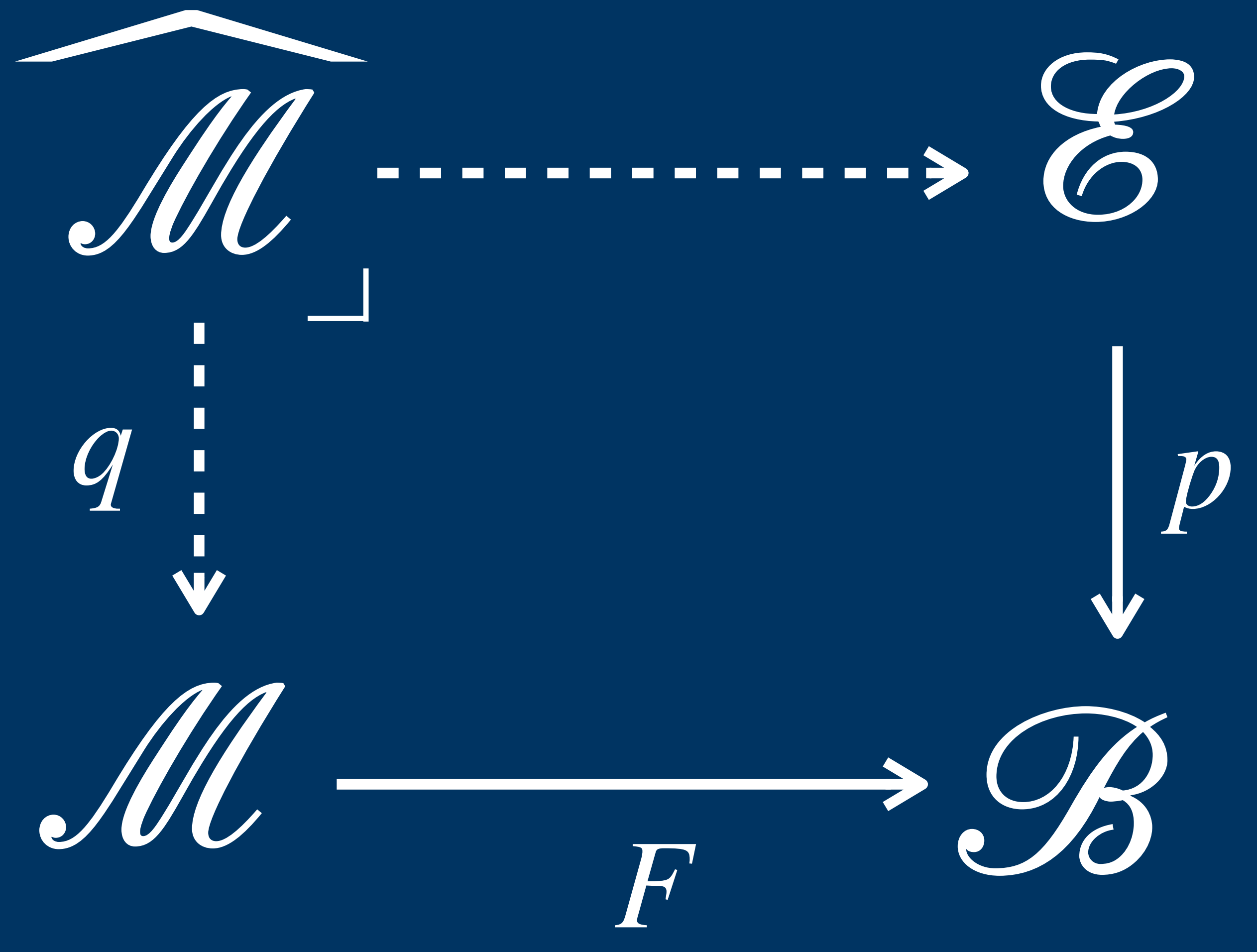
fibration
strictly preserving
model structure

lifting theorem:

universal choice of model
'glueing' objects of $\mathscr{M}$ to 'relations' in $\mathscr{E}$

objects: $(X \in \mathbf{Set}, R \subseteq X \times X)$
maps: pairs of maps preserving the relation





weak map of models

abstract notion of 'relation'

ie. p a fibration for logical relations

Logical relations can be seen as…

Level 1: type-indexed families of relations

Level 2: relations models

Level 3: fibrations for logical relations

+ can build new examples
from old ones by glueing

fibration for logical relations:

fibration
strictly preserving
model structure

Logical relations: a 2-categorical perspective

What's a fibration for logical relations for call-by-push-value?

A CBPV-model is a $\mathrm{Psh}(\mathscr{V})$-enriched adjunction

$$\mathrm{self}\,\mathscr{V} \underset{U}{\overset{F}{\rightleftarrows}} \mathscr{C} \quad (F \dashv U)$$

1. What is a CBPV fibration for logical relations?
2. Do they satisfy a lifting theorem?

What's a fibration for logical relations for call-by-push-value?

A CBPV-model is a $\mathrm{Psh}(\mathscr{V})$-enriched adjunction

$$\mathrm{self}\,\mathscr{V} \underset{U}{\overset{F}{\rightleftarrows}} \mathscr{C} \quad (F \dashv U)$$

1. What is a CBPV fibration for logical relations?
2. Do they satisfy a lifting theorem?

can no longer just say a fibration which preserves all the model structure

need a sensible notion of glueing

Observation:

“fibration which preserves model structure”	$\approx$	fibration internal to 2-category of models

A fibration internal to… is…

$\mathbf{CartCart}_{\text{strict}}$	a fibration strictly preserving products
$\mathbf{CCCart}_{\text{strict}}$	a fibration strictly preserving CC-structure
λ_{ml}-models	a fibration strictly preserving λ_{ml}-structure

Suggestion for level 4:

fibration for logical relations	:=	fibration internal to 2-category of models

*sometimes sufficient but not necessary condition, but a lot of the theory still applies

Suggestion for level 4:

fibration for logical relations	:=	fibration internal to 2-category of models

*sometimes sufficient but not necessary condition, but a lot of the theory still applies

Does this work correctly?

Internal fibrations are always fibrations which preserve model structure

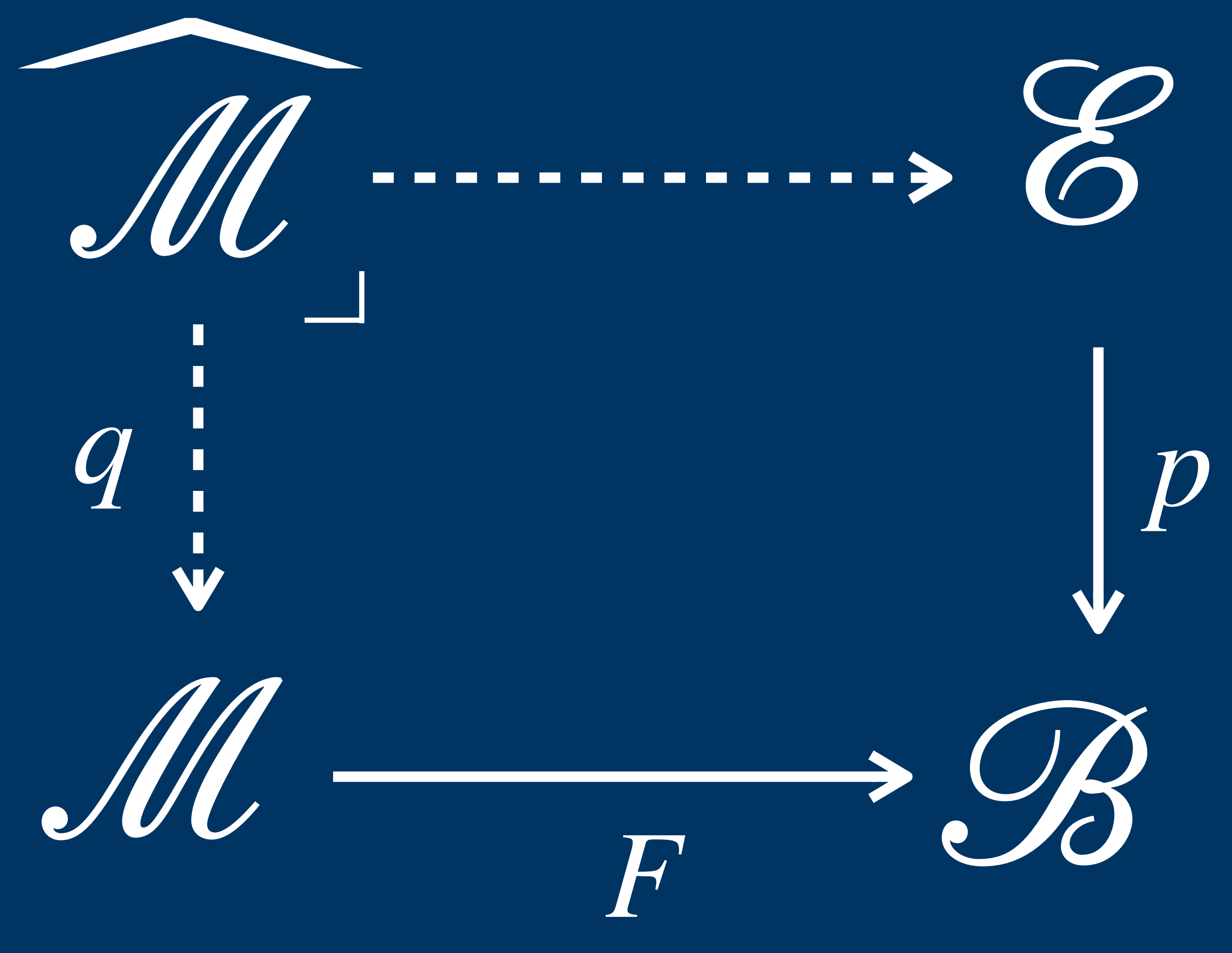
Theorem: let T be a relative pseudomonad along a pseudofunctor $J : \mathscr{A} \to \mathscr{C}$. Then a fibration in $\mathrm{Ps}\text{-}T\text{-}\mathrm{Alg}_{\mathrm{ps}}$ is exactly an algebra pseudomorphism $(p, \overline{p})$ such that p is a fibration in $\mathscr{C}$.

Glueing plays well with fibrations

lifting theorem:

universal choice of model
'glueing' objects of $\mathscr{M}$ to 'relations' in $\mathscr{E}$

semantic model



abstract notion of 'relation'

ie. p a fibration for logical relations

weak map of models

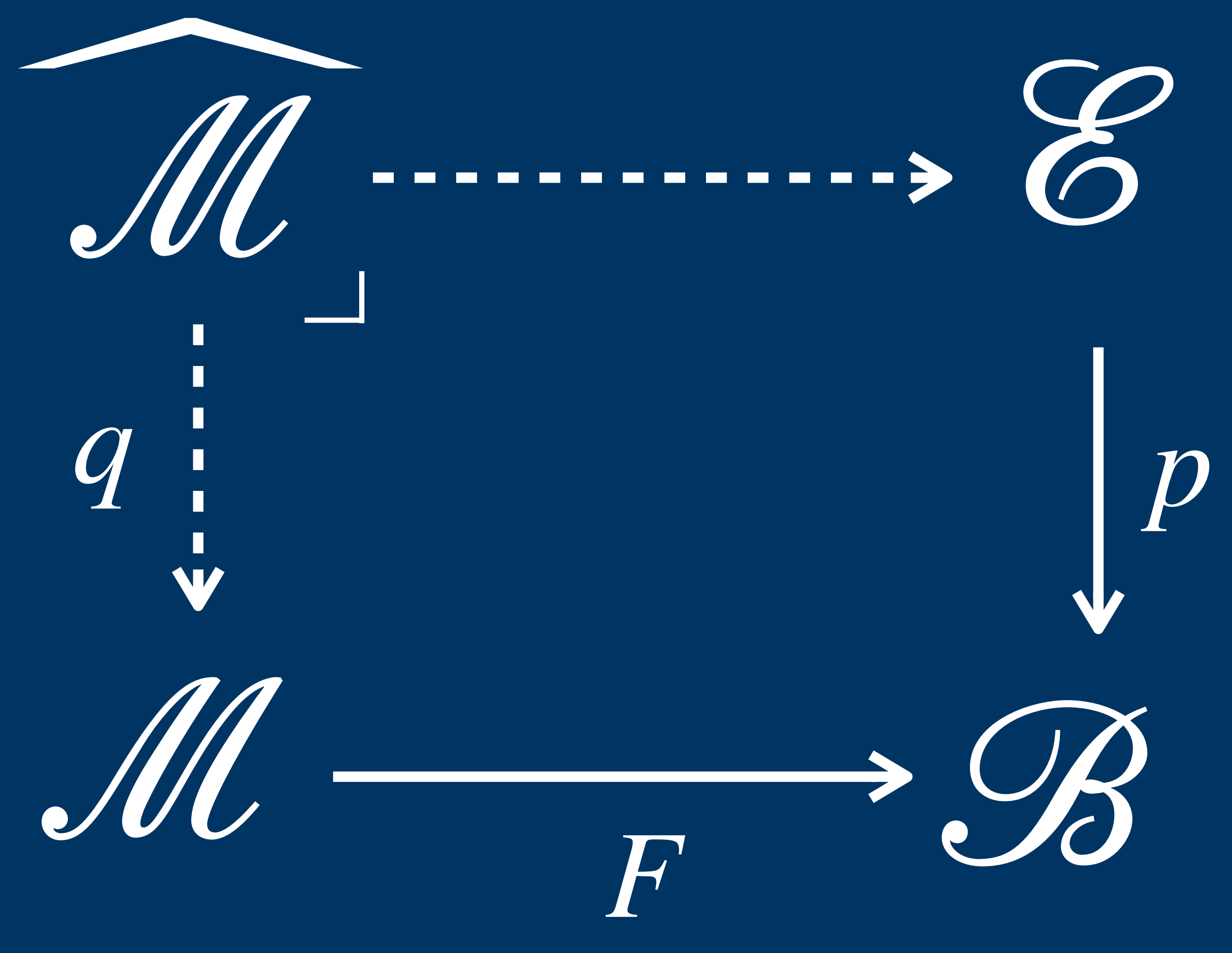
Proposition [classic]: fibrations are closed under pullback.

Glueing plays well with fibrations

lifting theorem:

universal choice of model
'glueing' objects of $\mathscr{M}$ to 'relations' in $\mathscr{E}$

semantic
model



abstract notion
of 'relation'

ie. p a fibration for
logical relations

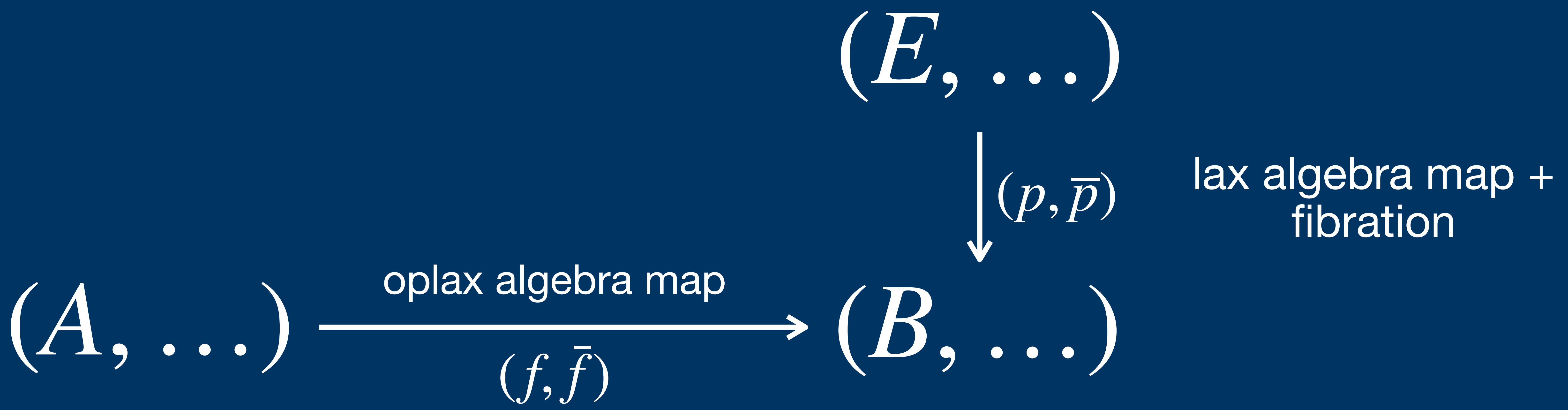
weak map of models

Proposition [classic]: fibrations are closed under pullback.

Glueing reduces to the existence of certain pullbacks in
2-categories of models.

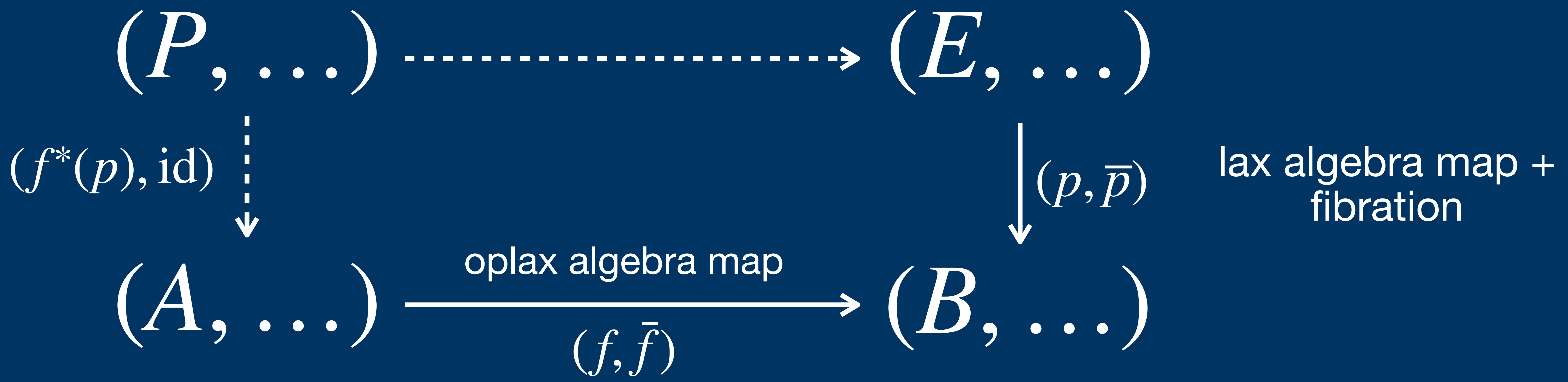
Glueing plays well with fibrations

Proposition: let T be a relative pseudomonad. Consider the following cospan of pseudoalgebras:



Glueing plays well with fibrations

Proposition: let T be a relative pseudomonad. Consider the following cospan of pseudoalgebras:



Then for the pullback $f^*(p) : P \to A$ in $\mathscr{C}$:

1. P acquires oplax algebra structure,
2. $f^*(p)$ is a strict algebra map, and
3. If $(p, \overline{p})$ is pseudo, this is a 2-pullback in $\mathrm{Oplax}\text{-}T\text{-}\mathrm{Alg}_{\mathrm{oplax}}$.

Glueing plays well with fibrations

…even when your models are diagrams

Glueing plays well with fibrations

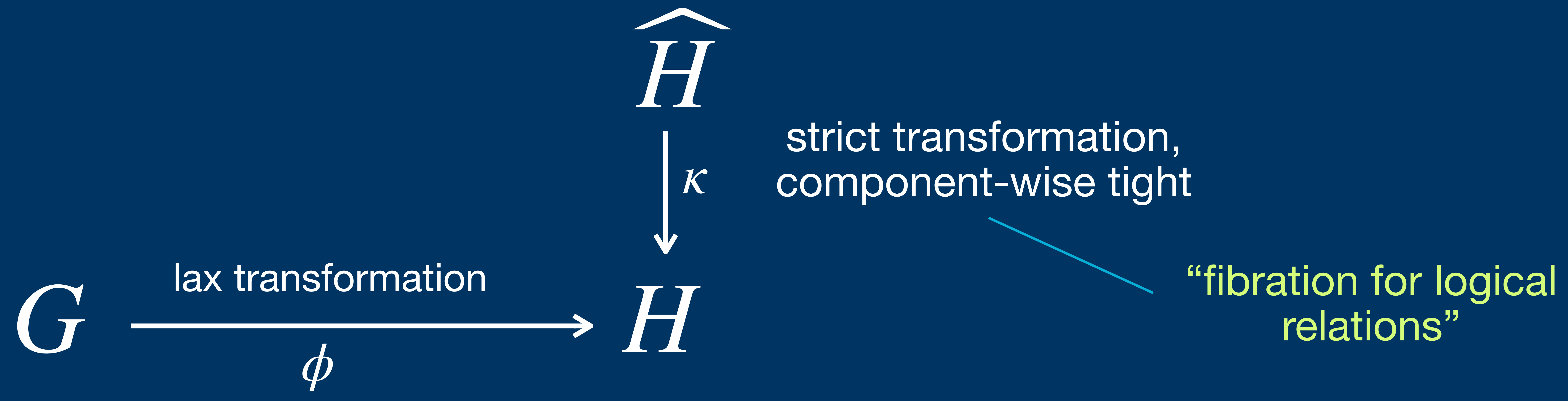
…even when your models are diagrams

If tight maps in $\mathscr{C}$ are closed under pullback (+ satisfy mild conditions), then in $[\mathscr{J}, \mathscr{C}]_{\mathrm{lax}}$ you can pull a tight map back along any map:

Glueing plays well with fibrations

…even when your models are diagrams

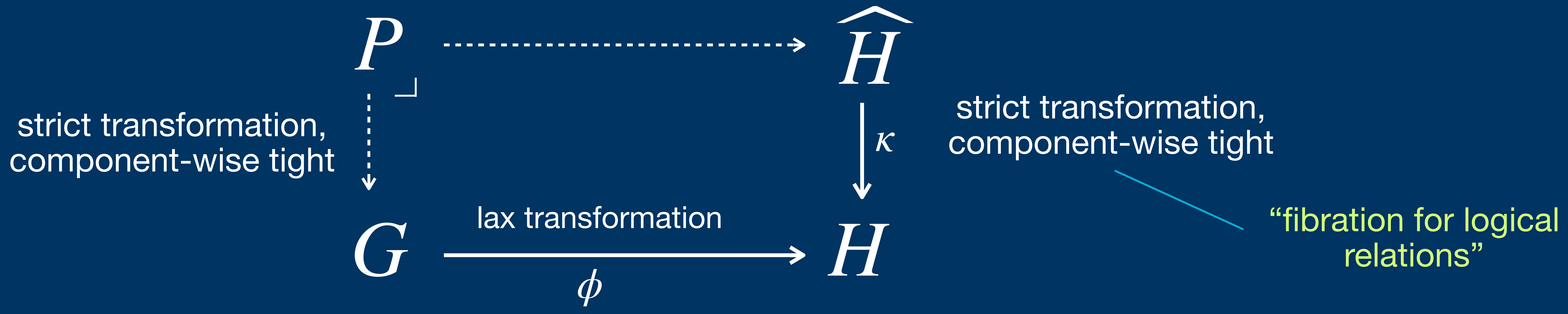
If tight maps in $\mathscr{C}$ are closed under pullback (+ satisfy mild conditions), then in $[\mathscr{J}, \mathscr{C}]_{\text{lax}}$ you can pull a tight map back along any map:



Glueing plays well with fibrations

…even when your models are diagrams

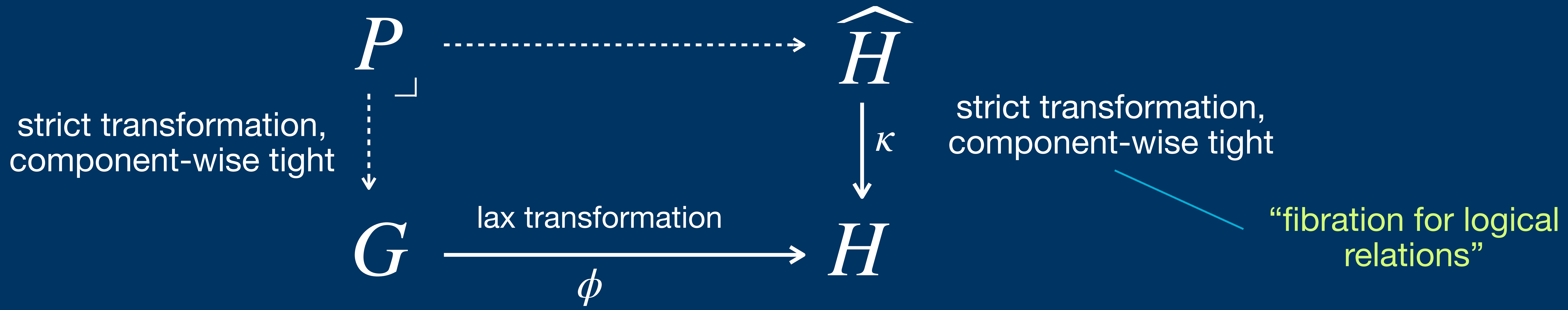
If tight maps in $\mathscr{C}$ are closed under pullback (+ satisfy mild conditions), then in $[\mathscr{J}, \mathscr{C}]_{\text{lax}}$ you can pull a tight map back along any map:



Glueing plays well with fibrations

…even when your models are diagrams

If tight maps in $\mathscr{C}$ are closed under pullback (+ satisfy mild conditions), then in $[\mathscr{J}, \mathscr{C}]_{\text{lax}}$ you can pull a tight map back along any map:



Moral: glueing works for monad models and adjunction models.

under slightly stronger conditions: can also take fibred products à la TT-lifting with multiple parameters

1. Fibrations internal to 2-categories tend to be the right thing*.
2. The theory of glueing works as you would hope, even in 2-categories of diagrams.

*sometimes sufficient but not necessary condition, but a lot of the theory still applies

Logical relations for CBPV models

(a recipe)

1. Define a 2-category LInd of locally-indexed categories,
2. Define CBPV as a sub-2-category of $[\mathrm{Adj}, \mathrm{LInd}]_{\mathrm{lax}}$,
3. Identify the fibrations internal to LInd,
4. Define a CBPV fibration to be a CBPV-morphism that's component-wise a LInd-fibration.

logical relations models + a glueing theorem follow from the algebra

a.k.a. Kripke relations of varying arity

Next steps: open logical relations

a.k.a. Kripke relations of varying arity

Next steps: open logical relations

characterise the definable maps in a model

[cf. Jung & Tiuryn]

semantic NbE à la *Semantic Analysis of Normalisation by Evaluation*

a.k.a. Kripke relations of varying arity

Next steps: open logical relations

characterise the definable maps in a model

[cf. Jung & Tiuryn]

semantic NbE à la *Semantic Analysis of Normalisation by Evaluation*

Missing ingredient: a presheaf construction for CBPV models

2-categorical statement of Lafont argument

Open logical relations

(a.k.a. Kripke relations of varying arity)

Many properties of interest require us to pay attention to contexts.

For example: definability (Jung & Tiuryn)

$$\mathrm{Def}_\sigma(\Gamma) := \{ [\![M]\!] \mid \Gamma \vdash M : \sigma \} \subseteq \mathscr{M}([\![\Gamma]\!], [\![\sigma]\!])$$

Open logical relations

(a.k.a. Kripke relations of varying arity)

Many properties of interest require us to pay attention to contexts.

For example: definability (Jung & Tiuryn)

$$\mathrm{Def}_{\sigma}(\Gamma) := \{[\![M]\!] \mid \Gamma \vdash M : \sigma\} \subseteq \mathscr{M}([\![\Gamma]\!], [\![\sigma]\!])$$

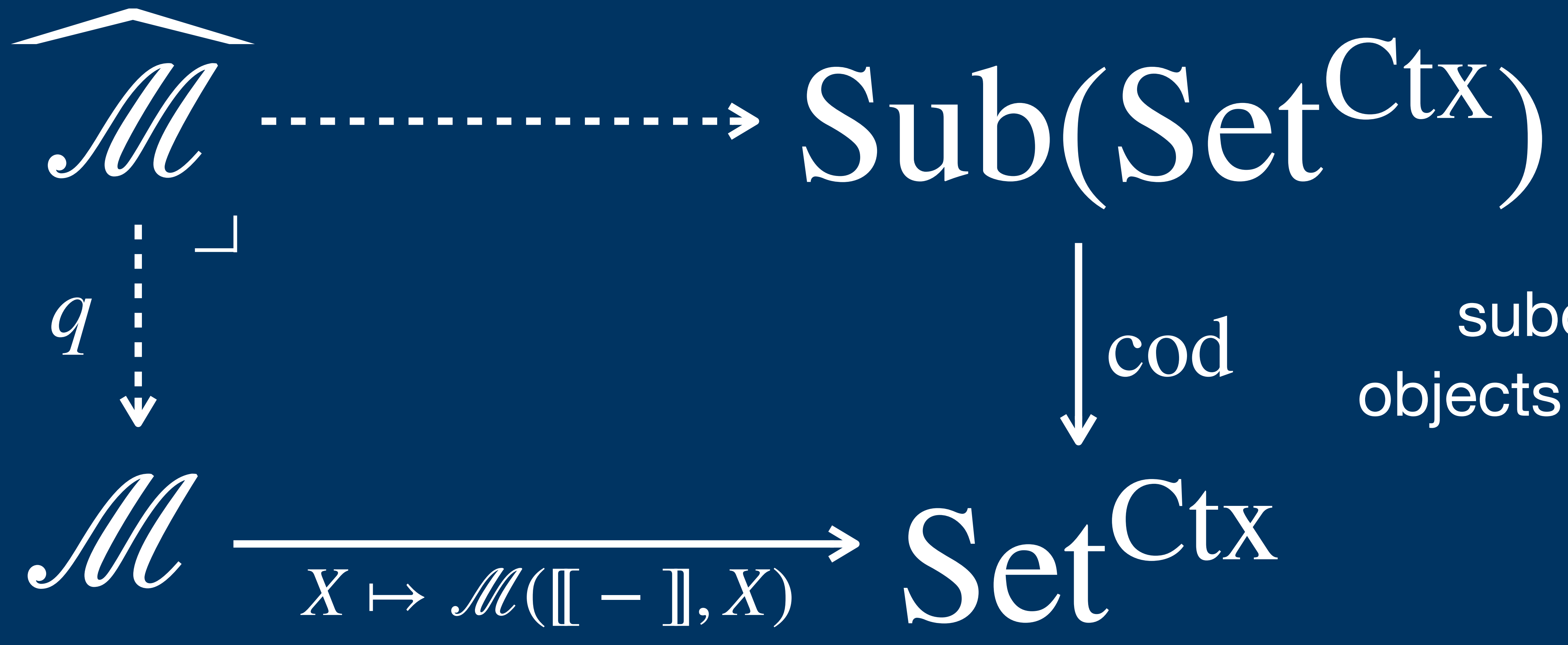
$$[\![\lambda x \,.\, M]\!] \in \mathrm{Def}_{\sigma \to \tau}(\underline{\Gamma}) \iff [\![M]\!] \in \mathrm{Def}_{\tau}(\underline{\Gamma, x : \sigma})$$

Open logical relations

(a.k.a. Kripke relations of varying arity)

objects:
$\big(X \in \mathscr{M}, R \hookrightarrow \mathscr{M}([\![-]\!], X)\big)$

i.e. families
$R(\Gamma) \subseteq \mathscr{M}([\![\Gamma]\!], X)$
indexed by contexts,
plus functoriality



semantic
model

subobject fibration:
objects are monos $\overline{P} \hookrightarrow P$

$\mathrm{Def}_\sigma(\Gamma) := \{[\![M]\!] \mid \Gamma \vdash M : \sigma\} \subseteq \mathscr{M}([\![\Gamma]\!], [\![\sigma]\!])$

open logical relations = glueing + presheaves

✔ ???

What about call-by-push-value?

= "locally indexed"

A CBPV-model is a $\mathrm{Psh}(\mathscr{V})$-enriched adjunction

$$\mathrm{self}\,\mathscr{V} \underset{U}{\overset{F}{\rightleftarrows}} \mathscr{C} \quad (F \dashv U)$$

What is a 'presheaf model' for CBPV?

What about call-by-push-value?

= “locally indexed”

A CBPV-model is a $\mathrm{Psh}(\mathscr{V})$-enriched adjunction

$$\mathrm{self}\,\mathscr{V} \underset{U}{\overset{F}{\rightleftarrows}} \mathscr{C} \quad (F \dashv U)$$

What is a ‘presheaf model’ for CBPV?

First approach: use the enriched presheaf construction.
But: size issues become troubling.

What about call-by-push-value?

= "locally indexed"

A CBPV-model is a $\mathrm{Psh}(\mathscr{V})$-enriched adjunction

$$\mathrm{self}\,\mathscr{V} \underset{U}{\overset{F}{\rightleftarrows}} \mathscr{C} \quad (F \dashv U)$$

What is a 'presheaf model' for CBPV?

First approach: use the enriched presheaf construction.
But: size issues become troubling.

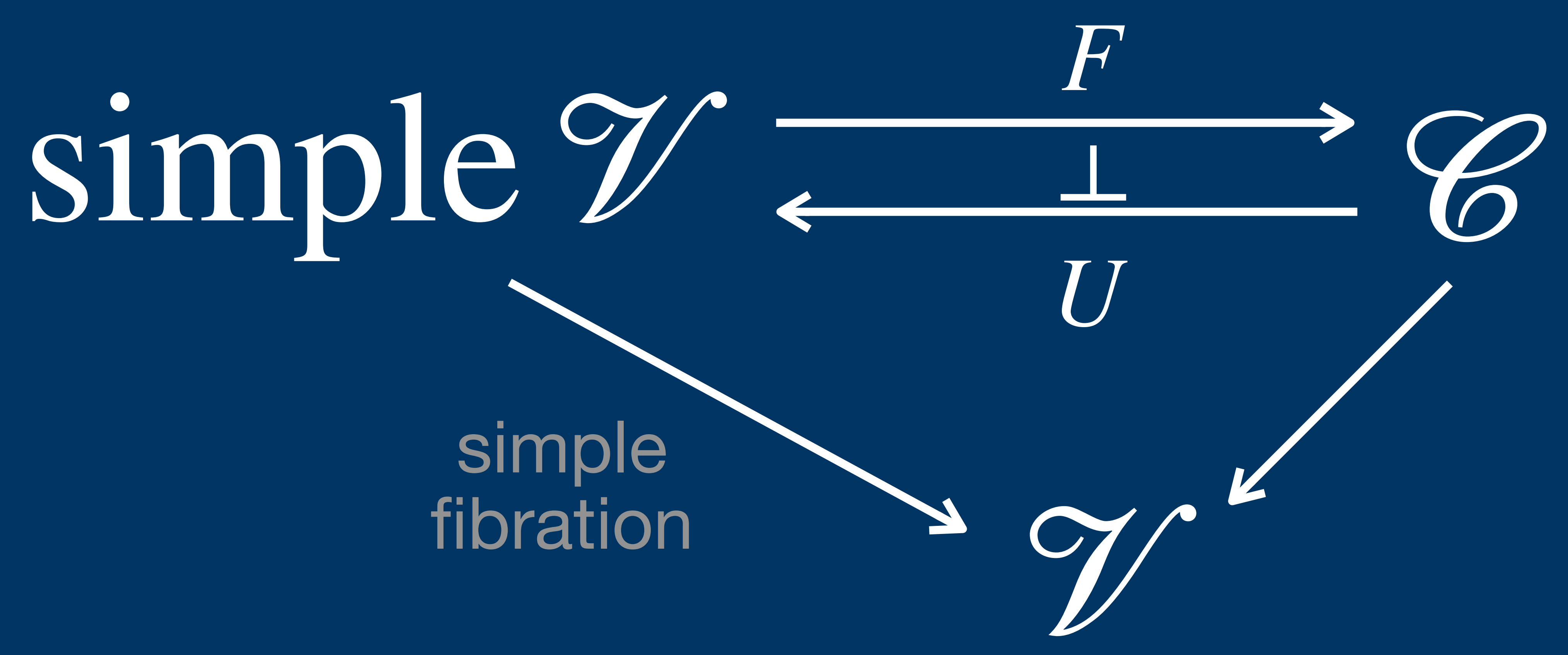
Alternative: recast CBPV models as certain fibrations.
(cf. Ahman, Vákár)

What about call-by-push-value?

each fibre has the same set of objects, and reindexing is ioo

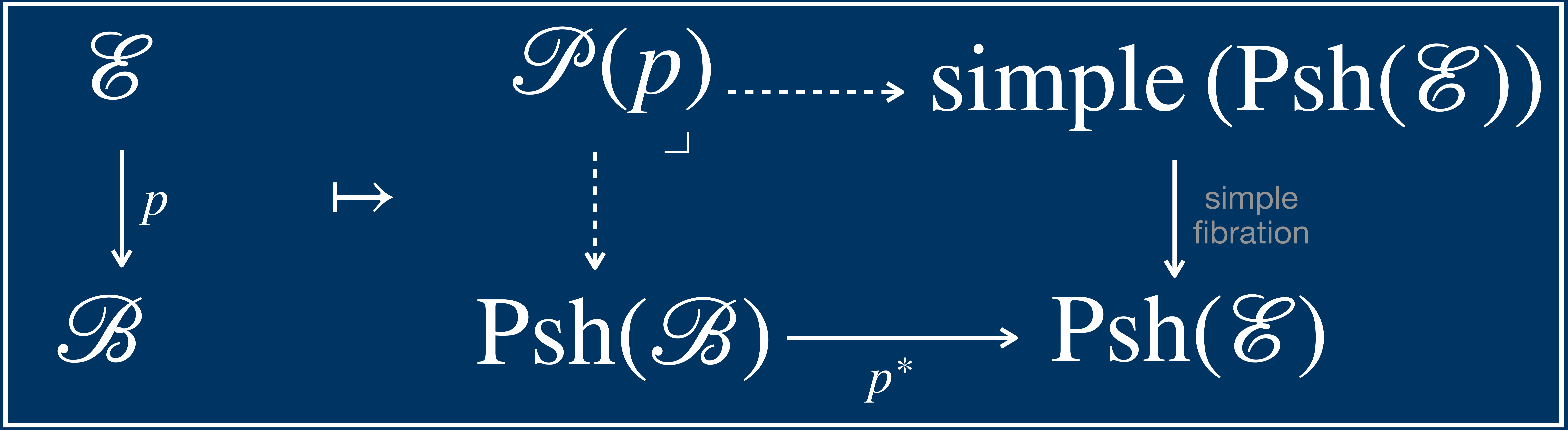
$\mathrm{Psh}(\mathscr{V})$-enriched category $\mathscr{C}$ $\overset{\text{folklore(?)}}{\longleftrightarrow}$ 'local' fibration $p : \mathscr{E} \to \mathscr{V}$

A CBPV-model is a (locally) fibred adjunction:



(cf. Ahman, Vákár)

The 2-category of local fibrations has a Yoneda structure



Induces a presheaf pseudofunctor $\rightsquigarrow$ sends adjunctions to adjunctions.

Determines a canonical presheaf operation on CBPV models.

Extracting the juice (WIP)

1. Open logical relations for CBPV: characterising definability, semantic NbE.
2. 2-categorical Lafont argument using this construction.
3. Seems to be quite a general way to build presheaf-like pseudomonads.

Summing up

1. Theory of logical relations includes fibrations for logical relations and glueing.
2. Fibrations internal to 2-categories of models seem to capture both aspects quite well.
3. Applies to examples where the models are more complex, e.g. CBPV.
4. Defining an appropriate presheaf construction should also lead to open logical relations, definability, NbE, conservativity, … .

the right abstraction (?)

follow what the algebra tells you

LICS paper link: arxiv.org/abs/2505.14482

Summing up

1. Theory of logical relations includes fibrations for logical relations and glueing.
2. Fibrations internal to 2-categories of models seem to capture both aspects quite well.
3. Applies to examples where the models are more complex, e.g. CBPV.
4. Defining an appropriate presheaf construction should also lead to open logical relations, definability, NbE, conservativity, … .

the right abstraction (?)

follow what the algebra tells you

Can we do the same for other languages?

Suggests a 'formal theory of logical relations' + applications.