Structured Frobenius for Fibrations Defined from a Generic Point

HoTT-UF 2024

Sina Hazratpour

j.w.w. Emily Riehl

April 2024

Frobenius condition

- The **Frobenius condition** is a property of weak factorization systems (WFS) that requires pullback along maps in the right class to preserve maps in the left class.

Frobenius condition

- The **Frobenius condition** is a property of weak factorization systems (WFS) that requires pullback along maps in the right class to preserve maps in the left class.
- In a locally cartesian closed category with a WFS:
  Frobenius condition $\iff$ R-maps closed under pushforwards along R-maps.

Frobenius condition

- The **Frobenius condition** is a property of weak factorization systems (WFS) that requires pullback along maps in the right class to preserve maps in the left class.
- In a locally cartesian closed category with a WFS:
  Frobenius condition $\iff$ R-maps closed under pushforwards along R-maps.
- In a full model category: Frobenius condition $\iff$ right properness.

Frobenius condition & HoTT

- Interpretation of Π-types; instrumental in obtaining models of HoTT in simplicial and cubical sets.

Frobenius condition & HoTT

- Interpretation of Π-types; instrumental in obtaining models of HoTT in simplicial and cubical sets.
- In Voevodsky's simplicial model of HoTT

Frobenius condition & HoTT

- Interpretation of Π-types; instrumental in obtaining models of HoTT in simplicial and cubical sets.
- In Voevodsky's simplicial model of HoTT
  - R-maps = Kan fibrations

Frobenius condition & HoTT

- Interpretation of Π-types; instrumental in obtaining models of HoTT in simplicial and cubical sets.
- In Voevodsky's simplicial model of HoTT
  - R-maps = Kan fibrations
  - The Frobenius condition is justified by the non-constructive use of minimal fibrations.

Frobenius condition & HoTT

- Interpretation of Π-types; instrumental in obtaining models of HoTT in simplicial and cubical sets.
- In Voevodsky's simplicial model of HoTT
  - R-maps = Kan fibrations
  - The Frobenius condition is justified by the non-constructive use of minimal fibrations.
- In Cubical Type Theory, Coquand gave a constructive proof of the Frobenius condition by reducing fibration structures to the more manageable composition structures.

Category-theoretic proofs of the Frobenius condition

- Previous work by Awodey and Gambino-Sattler on obtaining a categorical proof which is more conceptual and less syntactical.

Category-theoretic proofs of the Frobenius condition

- Previous work by Awodey and Gambino-Sattler on obtaining a categorical proof which is more conceptual and less syntactical.
- We used 2-categorical methods to give a proof of a functorial Frobenius condition.

Category-theoretic proofs of the Frobenius condition

- Previous work by Awodey and Gambino-Sattler on obtaining a categorical proof which is more conceptual and less syntactical.
- We used 2-categorical methods to give a proof of a functorial Frobenius condition.
    - Our proof does not require connection structures on the interval object since we work with the "unbiased" fibrations.

Category-theoretic proofs of the Frobenius condition

- Previous work by Awodey and Gambino-Sattler on obtaining a categorical proof which is more conceptual and less syntactical.
- We used 2-categorical methods to give a proof of a functorial Frobenius condition.
  - Our proof does not require connection structures on the interval object since we work with the "unbiased" fibrations.
  - We work in the more general setting of LCCCs.

Category-theoretic proofs of the Frobenius condition

- Previous work by Awodey and Gambino-Sattler on obtaining a categorical proof which is more conceptual and less syntactical.
- We used 2-categorical methods to give a proof of a functorial Frobenius condition.
  - Our proof does not require connection structures on the interval object since we work with the "unbiased" fibrations.
  - We work in the more general setting of LCCCs.
  - Equational approach based on mates from 2-category theory, instead of reasoning by universal properties.

Setup

- A pointed LCCC $(\mathcal{E}, \square \colon 1 \to \mathcal{E})$
- A category $\mathrm{TFib}_{\mathrm{cart}} \to \mathcal{E}^{2}_{\mathrm{cart}}$ of **stably structured trivial fibrations** satisfying the axioms STF1 - STF3 in below.

Axioms STF1-STF3

The **free retract** category is defined by the following pushout of categories:

$$\begin{array}{ccc} \mathbb{2} & \xrightarrow{!} & \mathbb{1} \\ {\scriptstyle d^1}\big\downarrow & \ulcorner & \big\downarrow{\scriptstyle r} \\ \mathbb{3} & \longrightarrow & \mathcal{R} \end{array}$$

And, the category of **maps with a specified section** is defined by the following pullback of categories:

$$\begin{array}{ccc} \mathcal{E}^{\mathcal{R}}_{\text{cart}} & \hookrightarrow & \mathcal{E}^{\mathcal{R}} \\ {\scriptstyle d^0}\big\downarrow & \lrcorner & \big\downarrow{\scriptstyle d^0} \\ \mathcal{E}^{\mathbb{2}}_{\text{cart}} & \hookrightarrow & \mathcal{E}^{\mathbb{2}} \end{array}$$

Axioms STF1-STF3

STF1 Trivial fibrations have a stable choice of section:

$$\begin{array}{ccc} \mathrm{TFib}_{\mathrm{cart}} & \overset{s}{\dashrightarrow} & \mathcal{E}^{\mathcal{R}}_{\mathrm{cart}} \\ & {\scriptstyle u}\searrow \quad \swarrow{\scriptstyle d^0} & \\ & \mathcal{E}^{\mathbb{2}}_{\mathrm{cart}} & \end{array}$$

STF2 Trivial fibrations are stable under pushforwards along any map:

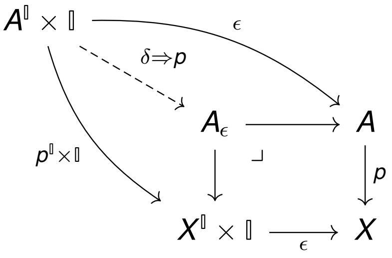
$$\begin{array}{ccc} \mathrm{TFib}_{\mathrm{cart}} \times_{\mathcal{E}} \mathcal{E}^{\mathbb{3}}_{\mathrm{cart}} & \dashrightarrow & \mathrm{TFib}_{\mathrm{cart}} \times_{\mathcal{E}} \mathcal{E}^{\mathbb{2}}_{\mathrm{cart}} \\ {\scriptstyle u\times\mathrm{id}}\downarrow & & \downarrow{\scriptstyle u\times\mathrm{id}} \\ \mathcal{E}^{\mathbb{4}}_{\mathrm{cart}} & \underset{\Pi}{\longrightarrow} & \mathcal{E}^{\mathbb{3}}_{\mathrm{cart}} \end{array}$$

STF3 Trivial fibrations are closed under retract:

$$\begin{array}{ccc} \mathrm{TFib}_{\mathrm{cart}} \times_{\mathcal{E}^{\mathbb{2}}_{\mathrm{cart}}} \mathcal{E}^{\mathbb{2}\times\mathcal{R}}_{\mathrm{cart}} & \overset{\mathrm{ev}_r}{\dashrightarrow} & \mathrm{TFib}_{\mathrm{cart}} \\ {\scriptstyle u\times\mathrm{id}}\downarrow & & \downarrow{\scriptstyle u} \\ \mathcal{E}^{\mathbb{2}\times\mathcal{R}}_{\mathrm{cart}} & \underset{\mathrm{ev}_r}{\longrightarrow} & \mathcal{E}^{\mathbb{2}}_{\mathrm{cart}} \end{array}$$

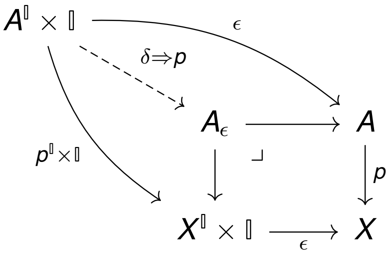
Fibrations from Trivial Fibrations

For the generic point $\delta\colon 1 \to \mathbb{I}$ and a map $p\colon A \to X$, the Leibniz exponential $\delta \Rightarrow p$ is the gap map to the pullback. This defines a cartesian functor $\delta \Rightarrow (-)\colon \mathcal{E}^{2} \to \mathcal{E}^{2}$.

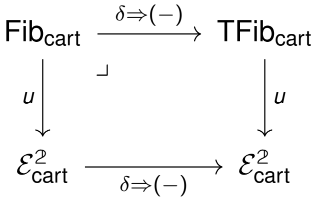


Fibrations from Trivial Fibrations

For the generic point $\delta\colon 1 \to \mathbb{I}$ and a map $p\colon A \to X$, the Leibniz exponential $\delta \Rightarrow p$ is the gap map to the pullback. This defines a cartesian functor $\delta \Rightarrow (-)\colon \mathcal{E}^{\mathbf{2}} \to \mathcal{E}^{\mathbf{2}}$.



We define a category of **stably structured fibrations** from the category of stably structured trivial fibrations:



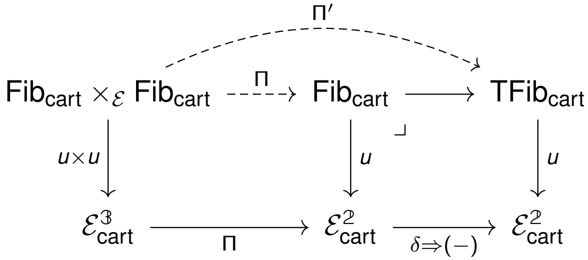
Functorial Structured Frobenius Theorem

$$\begin{array}{ccc}
\mathrm{Fib}_{\mathrm{cart}} \times_{\mathcal{E}} \mathrm{Fib}_{\mathrm{cart}} & \overset{\Pi}{\dashrightarrow} & \mathrm{Fib}_{\mathrm{cart}} \\
{\scriptstyle u \times u}\big\downarrow & & \big\downarrow{\scriptstyle u} \\
\mathcal{E}^{\mathbb{3}}_{\mathrm{cart}} & \underset{\Pi}{\longrightarrow} & \mathcal{E}^{\mathbb{2}}_{\mathrm{cart}}
\end{array}$$

Functorial Structured Frobenius Theorem

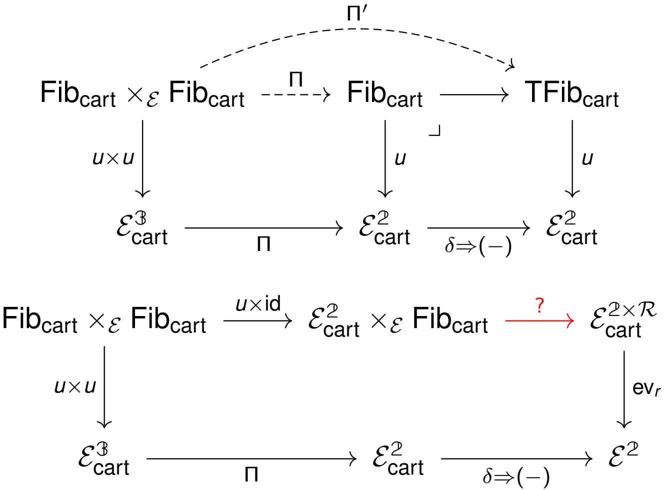
$$
\begin{array}{ccccc}
 & & \overset{\Pi'}{\curvearrowright} & & \\
\mathsf{Fib}_{\mathsf{cart}} \times_{\mathcal{E}} \mathsf{Fib}_{\mathsf{cart}} & \overset{\Pi}{\dashrightarrow} & \mathsf{Fib}_{\mathsf{cart}} & \longrightarrow & \mathsf{TFib}_{\mathsf{cart}} \\
\Big\downarrow{\scriptstyle u \times u} & & \Big\downarrow{\scriptstyle u} \quad \lrcorner & & \Big\downarrow{\scriptstyle u} \\
\mathcal{E}^{\mathbb{3}}_{\mathsf{cart}} & \underset{\Pi}{\longrightarrow} & \mathcal{E}^{\mathbb{2}}_{\mathsf{cart}} & \underset{\delta \Rightarrow (-)}{\longrightarrow} & \mathcal{E}^{\mathbb{2}}_{\mathsf{cart}}
\end{array}
$$

Functorial Structured Frobenius Theorem



To obtain Π' we post-compose with the map $\mathrm{ev}_r \colon \mathrm{TFib}_{\mathrm{cart}} \times_{\mathcal{E}^{2}_{\mathrm{cart}}} \mathcal{E}^{2\times\mathcal{R}}_{\mathrm{cart}} \to \mathit{TFib}_{\mathrm{cart}}$.

Functorial Structured Frobenius Theorem



Constructing the red arrow

$\delta \Rightarrow p_* q$ is a retract of a pushforward of $\delta \Rightarrow q$:

$$\begin{array}{ccccc}
(\Pi_A B)^{\square} \times \square & \xrightarrow{\kappa} & \Pi_{A^{\square} \times \square} B^{\square} \times \square & \xrightarrow{\rho} & (\Pi_A B)^{\square} \times \square \\
\Big\downarrow {\scriptstyle \delta \Rightarrow p_* q} & & \Big\downarrow {\scriptstyle (p^{\square} \times \square)_*(\delta \Rightarrow q)} & & \Big\downarrow {\scriptstyle \delta \Rightarrow p_* q} \\
(\Pi_A B)_\epsilon & \xrightarrow[\kappa_\epsilon]{} & \Pi_{A^{\square} \times \square} (B_\epsilon) & \xrightarrow[\rho_\epsilon]{} & (\Pi_A B)_\epsilon
\end{array}$$

Constructing the red arrow

$\delta \Rightarrow p_* q$ is a retract of a pushforward of $\delta \Rightarrow q$:

$$
\begin{array}{ccccc}
(\Pi_A B)^{\square} \times \square & \xrightarrow{\kappa} & \Pi_{A^{\square} \times \square} B^{\square} \times \square & \xrightarrow{\rho} & (\Pi_A B)^{\square} \times \square \\
\downarrow{\scriptstyle \delta \Rightarrow p_* q} & & \downarrow{\scriptstyle (p^{\square} \times \square)_*(\delta \Rightarrow q)} & & \downarrow{\scriptstyle \delta \Rightarrow p_* q} \\
(\Pi_A B)_\epsilon & \xrightarrow[\kappa_\epsilon]{} & \Pi_{A^{\square} \times \square}(B_\epsilon) & \xrightarrow[\rho_\epsilon]{} & (\Pi_A B)_\epsilon
\end{array}
$$

This we show later using the calculus of mates from 2-category theory.

Finishing the proof

- By **(BC)**, the vertical morphisms and the canonical transformations κ_ϵ and κ are stable under pullback.

Finishing the proof

- By **(BC)**, the vertical morphisms and the canonical transformations κ_ϵ and κ are stable under pullback.
- By **(BC) + (STF1)**, ρ_ϵ and ρ are pullback stable.

Finishing the proof

- By **(BC)**, the vertical morphisms and the canonical transformations κ_ϵ and κ are stable under pullback.
- By **(BC) + (STF1)**, ρ_ϵ and ρ are pullback stable.
- **(STF2)** provides a pullback stable trivial fibration structure on $(p^{\mathbb{I}} \times \mathbb{I})_*(\delta \Rightarrow q)$

Finishing the proof

- By **(BC)**, the vertical morphisms and the canonical transformations κ_ϵ and κ are stable under pullback.
- By **(BC) + (STF1)**, ρ_ϵ and ρ are pullback stable.
- **(STF2)** provides a pullback stable trivial fibration structure on $(p^{\mathbb{I}} \times \mathbb{I})_*(\delta \Rightarrow q)$
- $$\begin{array}{ccccc}
\mathsf{Fib}_{\mathsf{cart}} \times_{\mathcal{E}} \mathsf{Fib}_{\mathsf{cart}} & \dashrightarrow & \mathsf{TFib}_{\mathsf{cart}} \times_{\mathcal{E}^{2}_{\mathsf{cart}}} \mathcal{E}^{2\times\mathcal{R}}_{\mathsf{cart}} & \xrightarrow{\mathsf{ev}_r} & \mathsf{TFib}_{\mathsf{cart}} \\
\Big\downarrow{\scriptstyle \mathsf{id}\times u} & & \Big\downarrow & & \Big\downarrow{\scriptstyle u} \\
\mathcal{E}^{2}_{\mathsf{cart}} \times_{\mathcal{E}} \mathsf{Fib}_{\mathsf{cart}} & \longrightarrow & \mathcal{E}^{2\times\mathcal{R}}_{\mathsf{cart}} & \xrightarrow{\mathsf{ev}_r} & \mathcal{E}^{2}_{\mathsf{cart}}
\end{array}$$

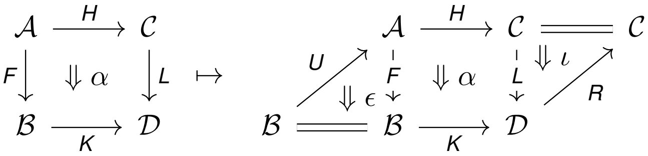
Finishing the proof

- By **(BC)**, the vertical morphisms and the canonical transformations κ_ϵ and κ are stable under pullback.
- By **(BC) + (STF1)**, ρ_ϵ and ρ are pullback stable.
- **(STF2)** provides a pullback stable trivial fibration structure on $(p^{\square} \times \square)_*(\delta \Rightarrow q)$
- $$\begin{array}{ccccc} \mathsf{Fib}_{\mathsf{cart}} \times_{\mathcal{E}} \mathsf{Fib}_{\mathsf{cart}} & \dashrightarrow & \mathsf{TFib}_{\mathsf{cart}} \times_{\mathcal{E}^{\mathbb{2}}_{\mathsf{cart}}} \mathcal{E}^{\mathbb{2}\times\mathcal{R}}_{\mathsf{cart}} & \xrightarrow{\mathsf{ev}_r} & \mathsf{TFib}_{\mathsf{cart}} \\ {\scriptstyle \mathsf{id}\times u}\downarrow & & \downarrow & & \downarrow{\scriptstyle u} \\ \mathcal{E}^{\mathbb{2}}_{\mathsf{cart}} \times_{\mathcal{E}} \mathsf{Fib}_{\mathsf{cart}} & \longrightarrow & \mathcal{E}^{\mathbb{2}\times\mathcal{R}}_{\mathsf{cart}} & \xrightarrow{\mathsf{ev}_r} & \mathcal{E}^{\mathbb{2}}_{\mathsf{cart}} \end{array}$$
- Composing with **(STF3)** construct the desired lift $\mathsf{Fib}_{\mathsf{cart}} \times_{\mathcal{E}} \mathsf{Fib}_{\mathsf{cart}} \to \mathsf{TFib}_{\mathsf{cart}}$.

Proof of the Retract Diagram Using Mates

The Mate Correspondence

The mates correspondence gives an extended, double-categorical, version of adjoint transposition: a suitably-oriented 2-cell in a square involving parallel left adjoints is mates with another 2-cell in the corresponding square formed by their right adjoints.



Theorem (Kelly-Street)

Consider the pair of double categories $\mathbb{L}\text{adj}$ *and* $\mathbb{R}\text{adj}$ *whose:*

- *objects are categories,*
- *horizontal arrows are functors,*
- *vertical arrows are fully-specified adjunctions pointing in the direction of the left adjoint, and*
- *squares of* $\mathbb{L}\text{adj}$ *(resp.* $\mathbb{R}\text{adj}$*) are natural transformations between the squares of functors formed by the left (resp. right) adjoints.*

Then

$$\mathbb{L}\text{adj} \cong \mathbb{R}\text{adj}$$

which acts on squares by taking mates.

The basic 2-cells

From the counit 2-cells

$$
\begin{array}{ccc}
/\square & \xrightarrow{\;\square_!\;} & /1 \\
{\scriptstyle \square^*}\uparrow & \Downarrow \pi & \| \\
/1 & = & /1
\end{array}
\qquad\qquad
\begin{array}{ccc}
/1 & \xrightarrow{\;\square^*\;} & /\square \\
{\scriptstyle \square_*}\uparrow & \Downarrow \nu & \| \\
/\square & = & /\square
\end{array}
$$

The basic 2-cells

From the counit 2-cells

$$\begin{array}{ccc} /\square & \xrightarrow{\square_!} & /1 \\ {\scriptstyle \square^*}\uparrow & \Downarrow \pi & \| \\ /1 & = & /1 \end{array} \qquad \begin{array}{ccc} /1 & \xrightarrow{\square^*} & /\square \\ {\scriptstyle \square_*}\uparrow & \Downarrow \nu & \| \\ /\square & = & /\square \end{array}$$

we obtain the spans

$$X^{\square} \xleftarrow{\pi} X^{\square} \times \square \xrightarrow{\epsilon} X$$

natural in X.

The basic 2-cells

From the counit 2-cells

$$\begin{array}{ccc} /\Box & \xrightarrow{\Box_!} & /1 \\ {\scriptstyle \Box^*}\uparrow & \Downarrow \pi & \| \\ /1 & = & /1 \end{array} \qquad \begin{array}{ccc} /1 & \xrightarrow{\Box^*} & /\Box \\ {\scriptstyle \Box_*}\uparrow & \Downarrow \nu & \| \\ /\Box & = & /\Box \end{array}$$

we obtain the spans

$$X^{\Box} \xleftarrow{\pi} X^{\Box} \times \Box \xrightarrow{\epsilon} X$$

natural in X.

$$\begin{array}{ccc} A^{\Box} \times \Box & \xrightarrow{\pi} & A^{\Box} \\ {\scriptstyle p^{\Box}\times\Box}\downarrow & \lrcorner & \downarrow{\scriptstyle p} \\ X^{\Box} \times \Box & \xrightarrow[\pi]{} & X^{\Box} \end{array}$$

Leibniz Exponential from the basic 2-cells

The component of the whiskered counit

$$
\begin{array}{ccccc}
 & & /X^{\square} & \xrightarrow{\pi^*} & /X^{\square} \times \square \\
 & & \pi_* \uparrow & \Downarrow \nu & \Vert \\
/X & \xrightarrow{\epsilon^*} & /X^{\square} \times \square & = & /X^{\square} \times \square
\end{array}
$$

at $p\colon A \to X$ is the map $\delta \Rightarrow p\colon A^{\square} \times \square \to A_{\epsilon}$.

Constructing κ_ϵ

$$\begin{array}{ccc} A^{I} \times I & \xrightarrow{\epsilon} & A \\ {\scriptstyle p^{I} \times I}\downarrow & & \downarrow{\scriptstyle p} \\ X^{I} \times I & \xrightarrow[\epsilon]{} & X \end{array}$$

Constructing κ_ϵ

$$
\begin{array}{ccc}
/A^{\Box} \times \Box & \xrightarrow{\epsilon_!} & /A \\
{\scriptstyle (p^{\Box}\times\Box)_!}\downarrow & & \downarrow{\scriptstyle p_!} \\
/X^{\Box} \times \Box & \xrightarrow[\epsilon_!]{} & /X
\end{array}
$$

Constructing κ_ϵ

$$
\begin{array}{ccc}
/A^{\square} \times \square & \xrightarrow{\epsilon_!} & /A \\
{\scriptstyle (p^{\square}\times\square)^*}\uparrow & \Downarrow & \uparrow{\scriptstyle p^*} \\
/X^{\square} \times \square & \xrightarrow[\epsilon_!]{} & /X
\end{array}
$$

Constructing κ_ϵ

$$
\begin{array}{ccc}
/A^{\square} \times \square & \xleftarrow{\;\epsilon^*\;} & /A \\
{\scriptstyle (p^{\square}\times\square)^*}\big\uparrow & \cong & \big\uparrow{\scriptstyle p^*} \\
/X^{\square} \times \square & \xleftarrow[\;\epsilon^*\;]{} & /X
\end{array}
$$

Constructing κ_ϵ

$$
\begin{array}{ccc}
/A^{□} \times □ & \xleftarrow{\epsilon^*} & /A \\
{\scriptstyle (p^{□} \times □)_*} \big\downarrow & \Uparrow \kappa_\epsilon & \big\downarrow {\scriptstyle p_*} \\
/X^{□} \times □ & \xleftarrow[\epsilon^*]{} & /X
\end{array}
$$

Constructing κ_ϵ

$$
\begin{array}{ccc}
A^{\square} \times \square & \xrightarrow{\ \epsilon\ } & A \\
{\scriptstyle p^{\square}\times\square}\big\downarrow & & \big\downarrow{\scriptstyle p} \\
X^{\square} \times \square & \xrightarrow[\ \epsilon\ ]{} & X
\end{array}
\quad \rightsquigarrow \quad
\begin{array}{ccc}
/A^{\square} \times \square & \xrightarrow{\ \epsilon_!\ } & /A \\
{\scriptstyle (p^{\square}\times\square)_!}\big\downarrow & & \big\downarrow{\scriptstyle p_!} \\
/X^{\square} \times \square & \xrightarrow[\ \epsilon_!\ ]{} & /X
\end{array}
\quad \rightsquigarrow \quad
\begin{array}{ccc}
/A^{\square} \times \square & \xleftarrow{\ \epsilon^*\ } & /A \\
{\scriptstyle (p^{\square}\times\square)_*}\big\downarrow & \Uparrow \kappa_\epsilon & \big\downarrow{\scriptstyle p_*} \\
/X^{\square} \times \square & \xleftarrow[\ \epsilon^*\ ]{} & /X
\end{array}
$$

Constructing κ_ϵ

$$
\begin{array}{ccc}
A^{\square} \times \square & \xrightarrow{\epsilon} & A \\
{\scriptstyle p^{\square}\times\square}\downarrow & & \downarrow{\scriptstyle p} \\
X^{\square} \times \square & \xrightarrow[\epsilon]{} & X
\end{array}
\quad \rightsquigarrow \quad
\begin{array}{ccc}
/A^{\square} \times \square & \xrightarrow{\epsilon_!} & /A \\
{\scriptstyle (p^{\square}\times\square)_!}\downarrow & & \downarrow{\scriptstyle p_!} \\
/X^{\square} \times \square & \xrightarrow[\epsilon_!]{} & /X
\end{array}
\quad \rightsquigarrow \quad
\begin{array}{ccc}
/A^{\square} \times \square & \xleftarrow{\epsilon^*} & /A \\
{\scriptstyle (p^{\square}\times\square)_*}\downarrow & \Uparrow \kappa_\epsilon & \downarrow{\scriptstyle p_*} \\
/X^{\square} \times \square & \xleftarrow[\epsilon^*]{} & /X
\end{array}
$$

The component of κ_ϵ at $q\colon B \to A$ defines a map $\kappa_\epsilon\colon (\Pi_A B)_\epsilon \to \Pi_{A^{\square}\times\square} B_\epsilon$ over $X^{\square} \times \square$.

Constructing κ_ϵ

So far,

$$\begin{array}{ccc} (\Pi_A B)^{\square} \times \square & & \Pi_{A^{\square}\times\square} B^{\square} \times \square \\ {\scriptstyle \delta\Rightarrow p_* q}\Big\downarrow & & \Big\downarrow{\scriptstyle (p^{\square}\times\square)_*(\delta\Rightarrow q)} \\ (\Pi_A B)_\epsilon & \xrightarrow[\kappa_\epsilon]{\qquad} & \Pi_{A^{\square}\times\square}(B_\epsilon) \end{array}$$

Constructing κ_ϵ

So far,

$$\begin{array}{ccc}
(\Pi_A B)^{\square} \times \square & \overset{\kappa}{\dashrightarrow} & \Pi_{A^{\square} \times \square} B^{\square} \times \square \\
{\scriptstyle \delta \Rightarrow p_* q} \Big\downarrow & & \Big\downarrow {\scriptstyle (p^{\square} \times \square)_* (\delta \Rightarrow q)} \\
(\Pi_A B)_\epsilon & \underset{\kappa_\epsilon}{\longrightarrow} & \Pi_{A^{\square} \times \square} (B_\epsilon)
\end{array}$$

Next, we find the left top arrow.

Constructing κ from κ_ϵ

$$
\begin{array}{ccc}
/A^{\square} \times \square & \xleftarrow{\ \epsilon^*\ } & /A \\
\Big\downarrow (p^{\square} \times \square)_* & \Uparrow \kappa_\epsilon & \Big\downarrow p_* \\
/X^{\square} \times \square & \xleftarrow[\ \epsilon^*\ ]{} & /X
\end{array}
$$

Constructing κ from κ_ϵ

$$
\begin{array}{ccccc}
/A^{\square} & \xleftarrow{\ \pi_*\ } & /A^{\square} \times \square & \xleftarrow{\ \epsilon^*\ } & /A \\
{\scriptstyle p^{\square}_*}\downarrow & \cong & \downarrow{\scriptstyle (p^{\square} \times \square)_*} & \Uparrow \kappa_\epsilon & \downarrow{\scriptstyle p_*} \\
/X^{\square} & \xleftarrow[\ \pi_*\ ]{} & /X^{\square} \times \square & \xleftarrow[\ \epsilon^*\ ]{} & /X
\end{array}
$$

Constructing κ from κ_ϵ

$$
\begin{array}{ccccccc}
/A^{\square} \times \square & \xleftarrow{\pi^*} & /A^{\square} & \xleftarrow{\pi_*} & /A^{\square} \times \square & \xleftarrow{\epsilon^*} & /A \\
\Big\downarrow {\scriptstyle (p^{\square} \times \square)_*} & \cong & \Big\downarrow {\scriptstyle p^{\square}_*} & \cong & \Big\downarrow {\scriptstyle (p^{\square} \times \square)_*} & \Uparrow \kappa_\epsilon & \Big\downarrow {\scriptstyle p_*} \\
/X^{\square} \times \square & \xleftarrow[\pi^*]{} & /X^{\square} & \xleftarrow[\pi_*]{} & /X^{\square} \times \square & \xleftarrow[\epsilon^*]{} & /X
\end{array}
$$

Constructing κ from κ_ϵ

$$
\begin{array}{ccccccc}
/A^{\square} \times \square & \xleftarrow{\pi^*} & /A^{\square} & \xleftarrow{\pi_*} & /A^{\square} \times \square & \xleftarrow{\epsilon^*} & /A \\
\Big\downarrow (p^{\square} \times \square)_* & \cong & \Big\downarrow p^{\square}_* & \cong & \Big\downarrow (p^{\square} \times \square)_* & \Uparrow \kappa_\epsilon & \Big\downarrow p_* \\
/X^{\square} \times \square & \xleftarrow[\pi^*]{} & /X^{\square} & \xleftarrow[\pi_*]{} & /X^{\square} \times \square & \xleftarrow[\epsilon^*]{} & /X
\end{array}
$$

The component of the composite 2-cell at $q\colon B \to A$ defines a map

$$\kappa\colon (\Pi_A B)^{\square} \times \square \to \Pi_{A^{\square} \times \square}(B^{\square} \times \square)$$

over $X^{\square} \times \square$.

Constructing κ from κ_ϵ

The component of the composite 2-cell at $q\colon B \to A$ defines a map

$$\kappa\colon (\Pi_A B)^{\square} \times \square \to \Pi_{A^{\square}\times\square}(B^{\square} \times \square)$$

over $X^{\square} \times \square$.

$$\begin{array}{ccccccc}
/A^{\square} \times \square & \xleftarrow{\pi^*} & /A^{\square} & \xleftarrow{\pi_*} & /A^{\square} \times \square / X^{\square} \times \square & \xleftarrow{\epsilon^*} & /A \\
\downarrow{\scriptstyle (p^{\square}\times\square)_*} & & & \Uparrow \kappa & & & \downarrow{\scriptstyle p_*} \\
/X^{\square} \times \square & \xleftarrow[\pi^*]{} & /X^{\square} & \xleftarrow[\pi_*]{} & /X^{\square} \times \square & \xleftarrow[\epsilon^*]{} & /X
\end{array}$$

Constructing κ from κ_ϵ

B/A

Constructing κ from κ_ϵ

$$B_\epsilon / A^{\square} \times \square \overset{\epsilon^*}{\longleftarrow\!\!\!\shortmid} B/A$$

Constructing κ from κ_ϵ

$$B^{\square}/A^{\square} \xleftarrow{\;\pi_*\;} B_\epsilon/A^{\square} \times \square \xleftarrow{\;\epsilon^*\;} B/A$$

Constructing κ from κ_ϵ

$$B^{\square} \times \square / A^{\square} \times \square \overset{\pi^*}{\longmapsfrom} B^{\square} / A^{\square} \overset{\pi_*}{\longmapsfrom} B_\epsilon / A^{\square} \times \square \overset{\epsilon^*}{\longmapsfrom} B/A$$

Constructing κ from κ_ϵ

$$
\begin{array}{ccccccc}
B^{\square} \times \square / A^{\square} \times \square & \overset{\pi^*}{\longleftarrow\!\shortmid} & B^{\square}/A^{\square} & \overset{\pi_*}{\longleftarrow\!\shortmid} & B_\epsilon / A^{\square} \times \square & \overset{\epsilon^*}{\longleftarrow\!\shortmid} & B/A \\
\Big\downarrow {\scriptstyle (p^{\square} \times \square)_*} & & & & & & \\
\Pi_{A^{\square} \times \square}(B^{\square} \times \square)/X^{\square} \times \square & & & & & &
\end{array}
$$

Constructing κ from κ_ϵ

$$\begin{array}{c} B/A \\ \downarrow{\scriptstyle p_*} \\ \Pi_A B/X \end{array}$$

Constructing κ from κ_ϵ

$$
\begin{array}{ccc}
 & & B/A \\
 & & \Big\downarrow{\scriptstyle p_*} \\
(\Pi_A B)_\epsilon / X^{\Box} \times \Box & \xleftarrow[\epsilon^*]{} & \Pi_A B / X
\end{array}
$$

Constructing κ from κ_ϵ

$$\begin{array}{ccccc} & & & & B/A \\ & & & & \Big\downarrow{\scriptstyle p_*} \\ (\Pi_A B)^{\mathbb{I}}/X^{\mathbb{I}} & \xleftarrow[\pi_*]{} & (\Pi_A B)_\epsilon/X^{\mathbb{I}} \times \mathbb{I} & \xleftarrow[\epsilon^*]{} & \Pi_A B/X \end{array}$$

Constructing κ from κ_ϵ

$$
\begin{array}{ccccccc}
 & & & & & & B/A \\
 & & & & & & \Big\downarrow{\scriptstyle p_*} \\
(\Pi_A B)^{\square} \times \square / X^{\square} \times \square & \xleftarrow[\pi^*]{} & (\Pi_A B)^{\square} / X^{\square} & \xleftarrow[\pi_*]{} & (\Pi_A B)_\epsilon / X^{\square} \times \square & \xleftarrow[\epsilon^*]{} & \Pi_A B / X
\end{array}
$$

Constructing κ from κ_ϵ

We now have to verify that the square

$$\begin{array}{ccc}
(\Pi_A B)^{\square} \times \square & \xrightarrow{\kappa} & \Pi_{A^{\square}\times\square} B^{\square} \times \square \\
{\scriptstyle \delta \Rightarrow p_* q}\downarrow & & \downarrow{\scriptstyle (p^{\square}\times\square)_*(\delta\Rightarrow q)} \\
(\Pi_A B)_\epsilon & \xrightarrow[\kappa_\epsilon]{} & \Pi_{A^{\square}\times\square}(B_\epsilon)
\end{array}$$

commutes.

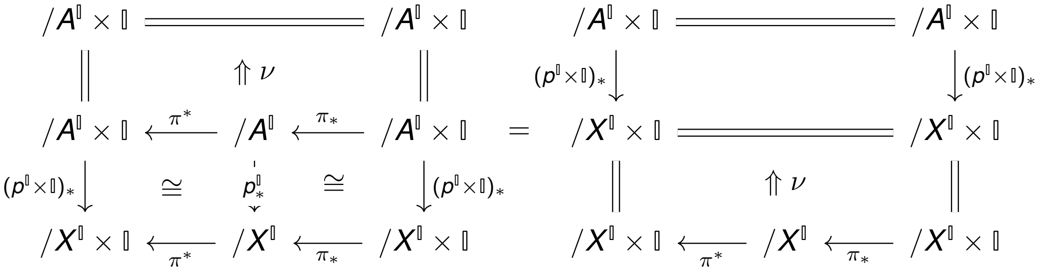
Constructing κ from κ_ϵ

$$
\begin{array}{ccc}
(\Pi_A B)^{\square} \times \square & & \\
\Big\downarrow{\scriptstyle \delta \Rightarrow p_* q} & & \\
(\Pi_A B)_\epsilon & \xrightarrow[\kappa_\epsilon]{} & \Pi_{A^{\square} \times \square}(B_\epsilon)
\end{array}
\qquad
\begin{array}{ccc}
(\Pi_A B)^{\square} \times \square & \xrightarrow{\kappa} & \Pi_{A^{\square} \times \square} B^{\square} \times \square \\
& & \Big\downarrow{\scriptstyle (p^{\square} \times \square)_*(\delta \Rightarrow q)} \\
& & \Pi_{A^{\square} \times \square}(B_\epsilon)
\end{array}
$$

$$
\begin{array}{ccccccc}
/A^{\square} \times \square & = & = & = & /A^{\square} \times \square & \xleftarrow{\epsilon^*} & /A \\
(p^{\square} \times \square)_* \downarrow & & & & (p^{\square} \times \square)_* \downarrow & \Uparrow \kappa_\epsilon & \downarrow p_* \\
/X^{\square} \times \square & = & = & = & /X^{\square} \times \square & \xleftarrow[\epsilon^*]{} & /X \\
\| & & \Uparrow \nu & & \| & & \| \\
/X^{\square} \times \square & \xleftarrow[\pi^*]{} & /X^{\square} & \xleftarrow[\pi_*]{} & /X^{\square} \times \square & \xleftarrow[\epsilon^*]{} & /X
\end{array}
\qquad
\begin{array}{ccccccc}
/A^{\square} \times \square & = & = & = & /A^{\square} \times \square & \xleftarrow{\epsilon^*} & /A \\
\| & & \Uparrow \nu & & \| & & \| \\
/A^{\square} \times \square & \xleftarrow{\pi^*} & /A^{\square} & \xleftarrow{\pi_*} & /A^{\square} \times \square & \xleftarrow{\epsilon^*} & /A \\
(p^{\square} \times \square)_* \downarrow & \cong & p^{\square}_* & \cong & (p^{\square} \times \square)_* & \Uparrow \kappa & \downarrow p_* \\
/X^{\square} \times \square & \xleftarrow[\pi^*]{} & /X^{\square} & \xleftarrow[\pi_*]{} & /X^{\square} \times \square & \xleftarrow[\epsilon^*]{} & /X
\end{array}
$$

Constructing κ from κ_ϵ

That reduces to asking for the equality of the pasted composites below:



Constructing κ from κ_ϵ

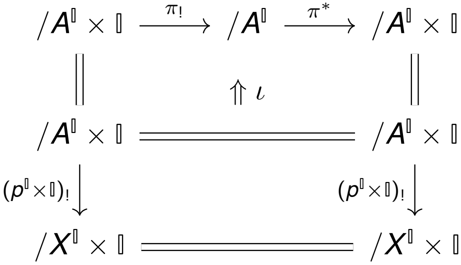
That reduces to asking for the equality of the pasted composites below:

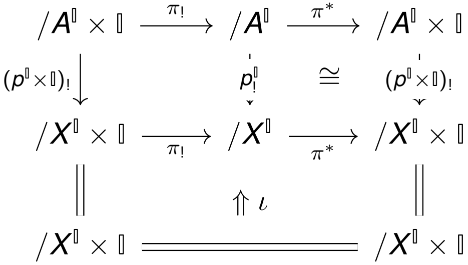
$$
\begin{array}{ccccc}
/A^{\square} \times \square & = & /A^{\square} \times \square & = & /A^{\square} \times \square \\
\Vert & \Uparrow \nu & & & \Vert \\
/A^{\square} \times \square & \xleftarrow{\pi^*} & /A^{\square} & \xleftarrow{\pi_*} & /A^{\square} \times \square \\
{\scriptstyle (p^{\square}\times\square)_*}\downarrow & \cong & \downarrow {\scriptstyle p^{\square}_*} & \cong & \downarrow{\scriptstyle (p^{\square}\times\square)_*} \\
/X^{\square} \times \square & \xleftarrow[\pi^*]{} & /X^{\square} & \xleftarrow[\pi_*]{} & /X^{\square} \times \square
\end{array}
\quad = \quad
\begin{array}{ccccc}
/A^{\square} \times \square & & = & & /A^{\square} \times \square \\
{\scriptstyle (p^{\square}\times\square)_*}\downarrow & & & & \downarrow{\scriptstyle (p^{\square}\times\square)_*} \\
/X^{\square} \times \square & & = & & /X^{\square} \times \square \\
\Vert & & \Uparrow \nu & & \Vert \\
/X^{\square} \times \square & \xleftarrow[\pi^*]{} & /X^{\square} & \xleftarrow[\pi_*]{} & /X^{\square} \times \square
\end{array}
$$

We simplify this verification by taking vertical mates, then horizontal mates, and then again the vertical mates of the two sides of the pasting equality (c.f. functoriality of mating).

Constructing κ from κ_ϵ

Finally, we need to verify that the following pasting diagrams are equal to each other.





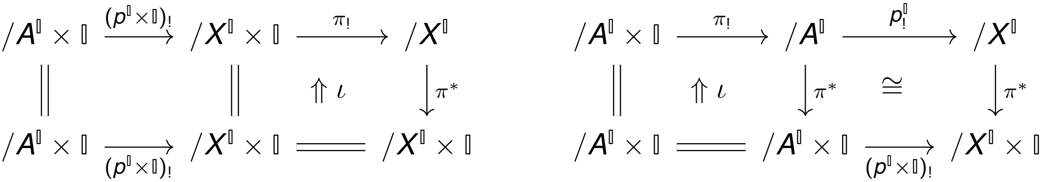
Constructing κ from κ_ϵ

The pasting diagrams above are equal if and only if

$$
\begin{array}{ccccc}
/A^{\square} \times \square & \xrightarrow{(p^{\square}\times\square)_!} & /X^{\square} \times \square & \xrightarrow{\pi_!} & /X^{\square} \\
\Vert & & \Vert & \Uparrow \iota & \downarrow \pi^* \\
/A^{\square} \times \square & \xrightarrow[(p^{\square}\times\square)_!]{} & /X^{\square} \times \square & = & /X^{\square} \times \square
\end{array}
\qquad
\begin{array}{ccccc}
/A^{\square} \times \square & \xrightarrow{\pi_!} & /A^{\square} & \xrightarrow{p^{\square}_!} & /X^{\square} \\
\Vert & \Uparrow \iota & \downarrow \pi^* & \cong & \downarrow \pi^* \\
/A^{\square} \times \square & = & /A^{\square} \times \square & \xrightarrow[(p^{\square}\times\square)_!]{} & /X^{\square} \times \square
\end{array}
$$

Constructing κ from κ_ϵ

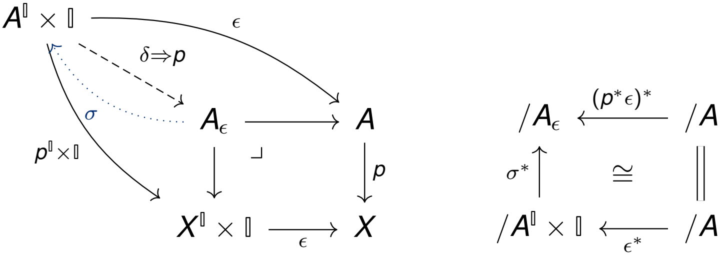
The pasting diagrams above are equal if and only if



Taking mates in the vertical direction reduces to a establishing a pasting equation between the same identity 2-cells.

Constructing the retract map ρ_ϵ

Since p is a fibration, $\delta \Rightarrow p$ has a section:



Constructing the retract map ρ_ϵ

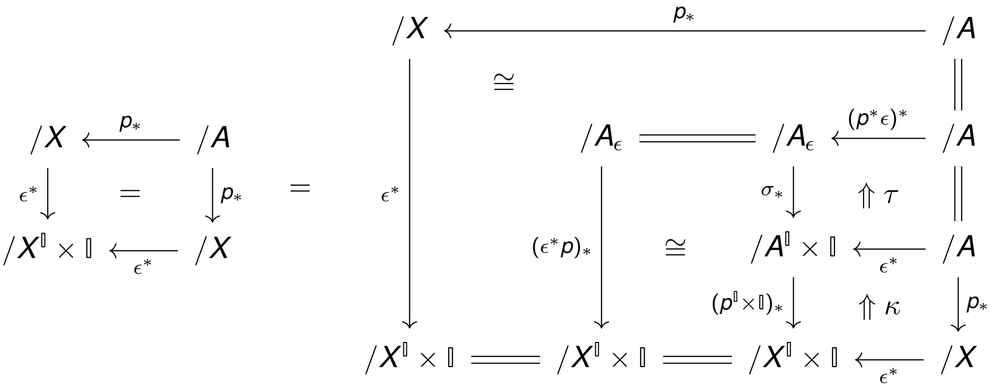
$$
\begin{array}{ccc}
/A_\epsilon & \xleftarrow{(p^*\epsilon)^*} & /A \\
{\scriptstyle (\epsilon^* p)^*}\uparrow & \cong & \uparrow{\scriptstyle p^*} \\
/X^{\square} \times \square & \xleftarrow[\epsilon^*]{} & /X
\end{array}
\quad = \quad
\begin{array}{ccccc}
/A_\epsilon & = & /A_\epsilon & \xleftarrow{(p^*\epsilon)^*} & /A \\
 & & {\scriptstyle \sigma^*}\uparrow & \cong & \| \\
{\scriptstyle (\epsilon^* p)^*}\uparrow & \cong & /A^{\square} \times \square & \xleftarrow[\epsilon^*]{} & /A \\
 & & {\scriptstyle (p^{\square} \times \square)^*}\uparrow & \cong & \uparrow{\scriptstyle p^*} \\
/X^{\square} \times \square & = & /X^{\square} \times \square & \xleftarrow[\epsilon^*]{} & /X
\end{array}
$$

Constructing the retract map ρ_ϵ

This gives rise to a pasting equation between the mates

$$
\begin{array}{ccc}
/A_\epsilon & \xleftarrow{(p^*\epsilon)^*} & /A \\
{\scriptstyle (\epsilon^* p)_*}\downarrow & \cong & \downarrow{\scriptstyle p_*} \\
/X^{\square} \times \square & \xleftarrow[\epsilon^*]{} & /X
\end{array}
\quad = \quad
\begin{array}{ccccc}
/A_\epsilon & = & /A_\epsilon & \xleftarrow{(p^*\epsilon)^*} & /A \\
 & & {\scriptstyle \sigma_*}\downarrow & \Uparrow \tau & \| \\
{\scriptstyle (\epsilon^* p)_*}\downarrow & \cong & /A^{\square} \times \square & \xleftarrow[\epsilon^*]{} & /A \\
 & & {\scriptstyle (p^{\square} \times \square)_*}\downarrow & \Uparrow \kappa & \downarrow{\scriptstyle p_*} \\
/X^{\square} \times \square & = & /X^{\square} \times \square & \xleftarrow[\epsilon^*]{} & /X
\end{array}
$$

Constructing the retract map ρ_ϵ



Constructing the retract map ρ_ϵ

The pasting equality above proves that ρ_ϵ is a retract of κ_{ep}.

$$
\begin{array}{ccccc}
(\Pi_A B)^{\square} \times \square & \xrightarrow{\kappa} & \Pi_{A^{\square} \times \square} B^{\square} \times \square & \overset{\rho}{\dashrightarrow} & (\Pi_A B)^{\square} \times \square \\
\Big\downarrow {\scriptstyle \delta \Rightarrow p_* q} & & \Big\downarrow {\scriptstyle (p^{\square} \times \square)_* (\delta \Rightarrow q)} & & \Big\downarrow {\scriptstyle \delta \Rightarrow p_* q} \\
(\Pi_A B)_\epsilon & \xrightarrow[\kappa_\epsilon]{} & \Pi_{A^{\square} \times \square} (B_\epsilon) & \xrightarrow[\rho_\epsilon]{} & (\Pi_A B)_\epsilon
\end{array}
$$

Constructing ρ from ρ_ϵ

Similar to the construction of κ from κ_ϵ we construct ρ from ρ_ϵ:

$$
\begin{array}{ccccccc}
/A^{\square} \times \square & \xleftarrow{\pi^*} & /A^{\square} & \xleftarrow{\pi_*} & /A^{\square} \times \square & \xleftarrow{\epsilon^*} & /A \\
(p^{\square}\times\square)_* \big\downarrow & \cong & p^{\square}_* \big\downarrow & \cong & (p^{\square}\times\square)_* \big\downarrow & \Downarrow \tau & \big\downarrow p_* \\
/X^{\square} \times \square & \xleftarrow[\pi^*]{} & /X^{\square} & \xleftarrow[\pi_*]{} & /X^{\square} \times \square & \xleftarrow[\epsilon^*]{} & /X
\end{array}
$$

Constructing ρ from ρ_ϵ

Similar to the commutativity of the square involving κ_ϵ and κ we show that the following square commutes:

$$
\begin{array}{ccc}
\Pi_{A^{\square} \times \square} B^{\square} \times \square & \xrightarrow{\rho} & (\Pi_A B)^{\square} \times \square \\
{\scriptstyle (p^{\square} \times \square)_* (\delta \Rightarrow q)} \Big\downarrow & & \Big\downarrow {\scriptstyle \delta \Rightarrow p_* q} \\
\Pi_{A^{\square} \times \square} (B_\epsilon) & \xrightarrow[\rho_\epsilon]{} & (\Pi_A B)_\epsilon
\end{array}
$$

Constructing ρ from ρ_ϵ

Similar to the commutativity of the square involving κ_ϵ and κ we show that the following square commutes:

$$\begin{array}{ccc} \Pi_{A^{\square} \times \square} B^{\square} \times \square & \xrightarrow{\rho} & (\Pi_A B)^{\square} \times \square \\ {\scriptstyle (p^{\square} \times \square)_*(\delta \Rightarrow q)} \downarrow & & \downarrow {\scriptstyle \delta \Rightarrow p_* q} \\ \Pi_{A^{\square} \times \square}(B_\epsilon) & \xrightarrow[\rho_\epsilon]{} & (\Pi_A B)_\epsilon \end{array}$$

That ρ is a retract of κ follows from the fact that ρ_ϵ is a retract of κ_ϵ and the iso 2-cells pasted to the left of κ_ϵ and ρ_ϵ, respectively, are pairwise inverses.

Carlo Angiuli, Guillaume Brunerie, Thierry Coquand, Robert Harper, Kuen-Bang Hou (Favonia), Daniel R. Licata
Syntax and models of Cartesian cubical type theory
Math. Struct. Comput. Sci., 2021.

Steve Awodey
Cartesian cubical model categories
arXiv:2305.00893, 2023

Thierry Coquand
Variation on Cubical sets
https://www.cse.chalmers.se/ coquand/diag.pdf, 2014

Nicola Gambino and Christian Sattler
The Frobenius condition, right properness, and uniform fibrations
J. Pure Appl. Algebra, 2017

Sina Hazratpour & Emily Riehl

A 2-Categorical Proof of Frobenius for Fibrations Defined From a Generic Point

Mathematical Structures in Computer Science, 2024

arXiv:2210.00078

G. M. Kelly & Ross Street

Review of the elements of 2-categories

Category Seminar Proc. Sem., Sydney, 1974

Andrew W. Swan

Definable and Non-definable Notions of Structure

arXiv:2206.13643, 2022