UNDERSTANDING (DELTA) LENSES USING CATEGORY THEORY

BRYCE CLARKE

28 May 2025

GOALS OF TALK:

(1) Explain how delta lens generalise classical lenses.

(2) Give an overview of tools from category theory.

(3) Ask questions on applicability to other kinds of bx.

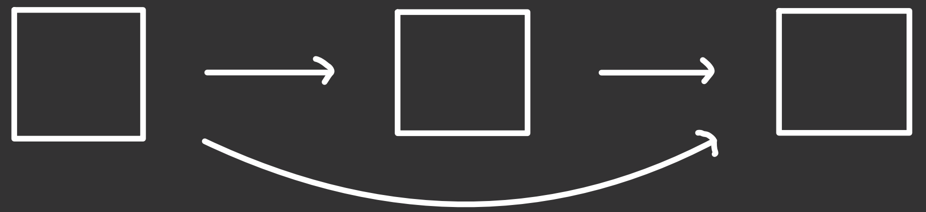
WHY CATEGORY THEORY ?

- Category theory is a framework for studying structures and mappings between them.
  - → sets and functions
  - → directed graphs and homomorphisms
  - → preorders and monotone maps

- Provides an abstract setting to focus on essential features of particular examples.

Category theory toolbox:

- (de) compositionality



- universal constructions with guaranteed properties
- diagrammatic reasoning

Are these useful tools ?

CLASSICAL STATE-BASED LENSES

<u>Assumption</u>: A system is a set of states.

A very well-behaved (state-based) lens $(f,p): A \to B$ is a pair of functions satisfying three axioms:

SOURCE

$$\begin{tikzcd} A \arrow[d, "f"'] & A \\ B & A \times B \arrow[u, "p"'] \end{tikzcd}$$

GET

VIEW

PUT

$$\begin{tikzcd} A \arrow[d, "f"'] & a \arrow[r, mapsto] \arrow[d, phantom, ":"] & p(a,b) \arrow[d, phantom, ":"] \\ B & fa \arrow[r, mapsto] & b = fp(a,b) \end{tikzcd}$$

$$\begin{tikzcd} & a \\ a \arrow[ur, mapsto] \arrow[r, mapsto] \arrow[d, phantom, ":"] & p(a,fa) \arrow[u, phantom, "\shortparallel"] \arrow[d, phantom, ":"] \\ fa \arrow[r, mapsto] & fa \end{tikzcd}$$

$$\begin{tikzcd} & & p(a,b') \\ a \arrow[urr, mapsto, bend left] \arrow[r, mapsto] \arrow[d, phantom, ":"] & p(a,b) \arrow[r, mapsto] \arrow[d, phantom, ":"] & p(p(a,b),b') \arrow[u, phantom, "\shortparallel"] \arrow[d, phantom, ":"] \\ fa \arrow[r, mapsto] \arrow[rr, mapsto, bend right] & b \arrow[r, mapsto] & b' \end{tikzcd}$$

SYSTEMS AS DIRECTED GRAPHS WITH STRUCTURE

Assumption: A system is a directed graph.

- states are vertices (set A_0)
- updates are edges (set A_1)
- for each state, an identity update $(i : A_0 \to A_1)$

$$a \longmapsto \begin{tikzcd} a \arrow[loop above, "1_a"] \end{tikzcd}$$

- for each sequential pair of updates, a composite update $(c : A_1 \times A_1 \nrightarrow A_1)$

$$\begin{tikzcd} & a' \arrow[dr, "v"] & \\ a \arrow[ur, "u"] & & a'' \end{tikzcd} \longmapsto a \xrightarrow{v \circ u} a''$$

A mapping between systems

$$(A_0, A_1) \xrightarrow{f} (B_0, B_1)$$

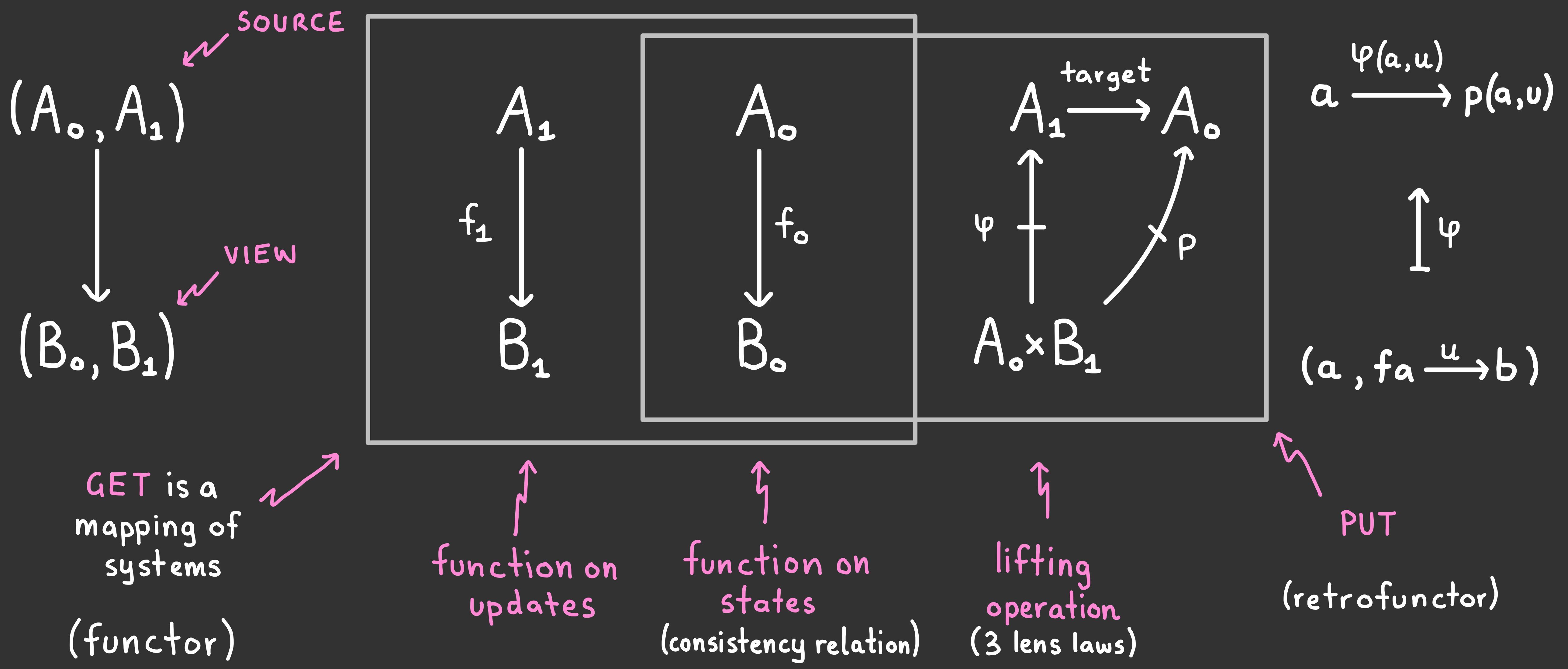
is a graph homomorphism which preserves identities & composites.

function on updates $\;\; A_1 \xrightarrow{f_1} B_1 \qquad A_0 \xrightarrow{f_0} B_0 \;\;$ function on states

Are these good assumptions for your application of bx?

DELTA LENSES

- A delta lens between directed graphs is a compatible GET and PUT.



COMPOSITION & COMPATIBILITY

Delta lenses compose **sequentially**.

$$\begin{tikzcd}
(A_0, A_1) \arrow[r, "{(f_0, f_1, \varphi)}"] \arrow[rr, bend right] & (B_0, B_1) \arrow[r, "{(g_0, g_1, \psi)}"] & (C_0, C_1)
\end{tikzcd}$$

Delta lenses compose in **parallel**.

$$\begin{tikzcd}
(A_0, A_1) \arrow[d] & & (C_0, C_1) \arrow[d] & & (A_0, A_1) \otimes (C_0, C_1) \arrow[d] \\
(B_0, B_1) & \otimes & (D_0, D_1) & \rightsquigarrow & (B_0, B_1) \otimes (D_0, D_1)
\end{tikzcd}$$

> Is <u>modularity</u> a desirable feature for your notion of bx?

mapping of systems

$$\begin{tikzcd}
(A_0, A_1) \arrow[r] \arrow[d] & (C_0, C_1) \arrow[d] \\
(B_0, B_1) \arrow[r] & (D_0, D_1)
\end{tikzcd}$$

delta lens

delta lens

mapping of systems

We can **tile** compatibility squares horizontally & vertically + **stack** them.

> Do you want to <u>compare</u> bx between different systems?

CONSTRUCTING FREE & COFREE DELTA LENSES

Delta lenses are algebras for a monad.

freely add updates to source

$$\begin{array}{c} (A_0, A_1) \\ \downarrow (f_0, f_1) \\ (B_0, B_1) \end{array} \quad \overset{\text{↯}}{\rightsquigarrow} \quad \begin{array}{c} \boxed{\text{source}} \\ \downarrow T(f_0, f_1) \\ (B_0, B_1) \end{array}$$

GET
(mapping of systems)

free delta lens

> Do you want to build bx from a GET?
>
> Is your bx specificied algebraically?

Delta lenses are coalgebras for a comonad.

delete/duplicate updates in source

$$\begin{array}{c} (A_0, A_1) \\ \downarrow (f_0, \varphi) \\ (B_0, B_1) \end{array} \quad \overset{\text{↯}}{\rightsquigarrow} \quad \begin{array}{c} \boxed{\text{source}} \\ \downarrow D(f_0, f_1) \\ (B_0, B_1) \end{array}$$

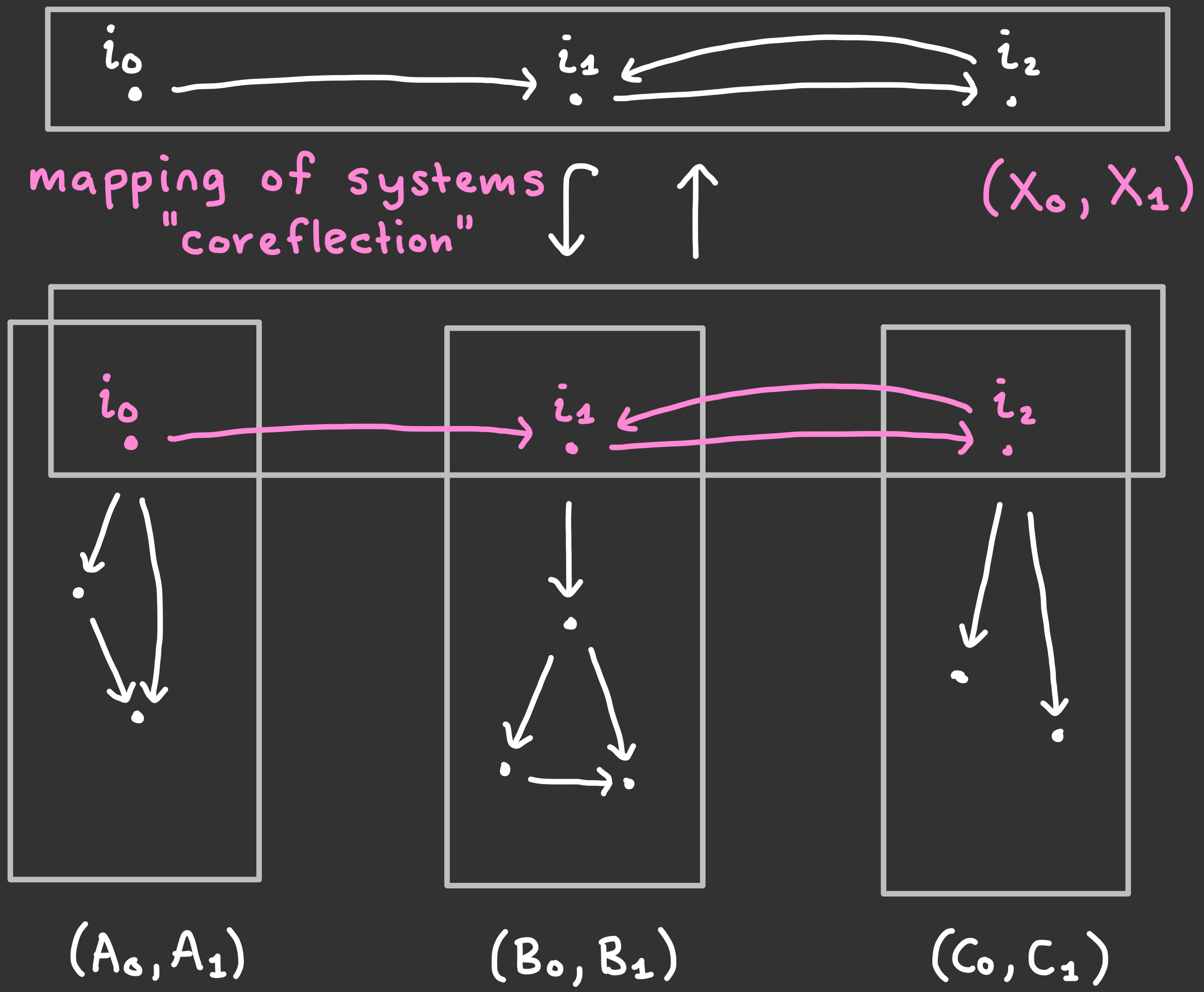
PUT
(consistency relation
& lifting operation)

cofree delta lens

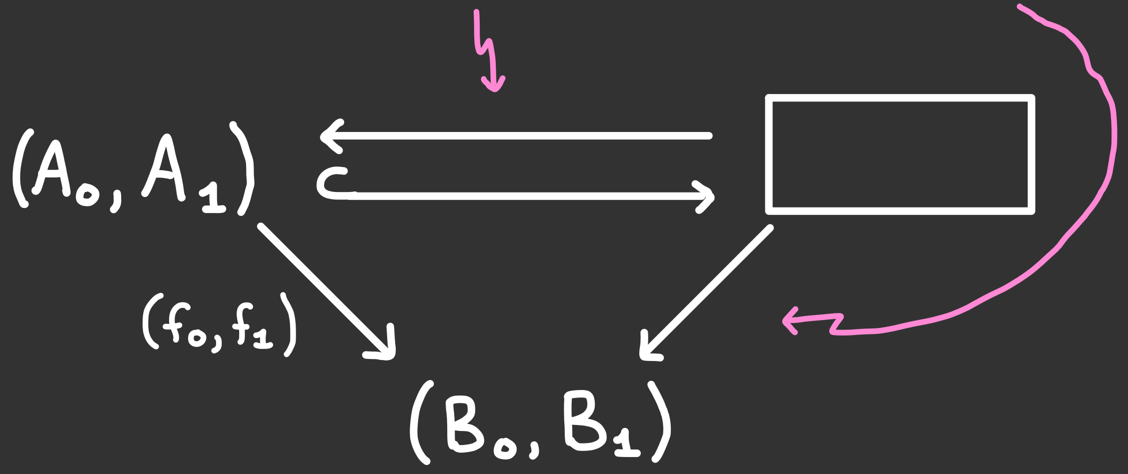
> Do you want to build bx from a PUT?
>
> Is your bx specificied coalgebraically?

FACTORISATION

Take a collection of systems with an initial state and glue these with updates



Each mapping of systems factors into a coreflection & a delta lens.



Delta lenses also factorise further.

In what situations would you like to factorise through a bx?

LIFTING & UNIVERSAL PROPERTIES

Examples of lifting problems

$$\begin{tikzcd}
A \arrow[r] \arrow[d, "\text{surjective function}"'] & C \arrow[d, tail, "\text{injective function}"] \\
B \arrow[r] \arrow[ur, dashed, "\text{function}", "\exists!"'] & D
\end{tikzcd}$$

$$\begin{tikzcd}
\{\bullet\} \arrow[r, "\text{chose a state } a \in A_0"] \arrow[d, "\text{inclusion of initial state}"'] & (A_0, A_1) \arrow[d, "\text{delta lens}"] \\
\{\bullet \to \bullet\} \arrow[r, "\text{choose an update } fa \xrightarrow{u} b \in B_1"'] \arrow[ur, dashed, "\varphi(a,u)", "\exists!"'] & (B_0, B_1)
\end{tikzcd}$$

Universal property of delta lenses

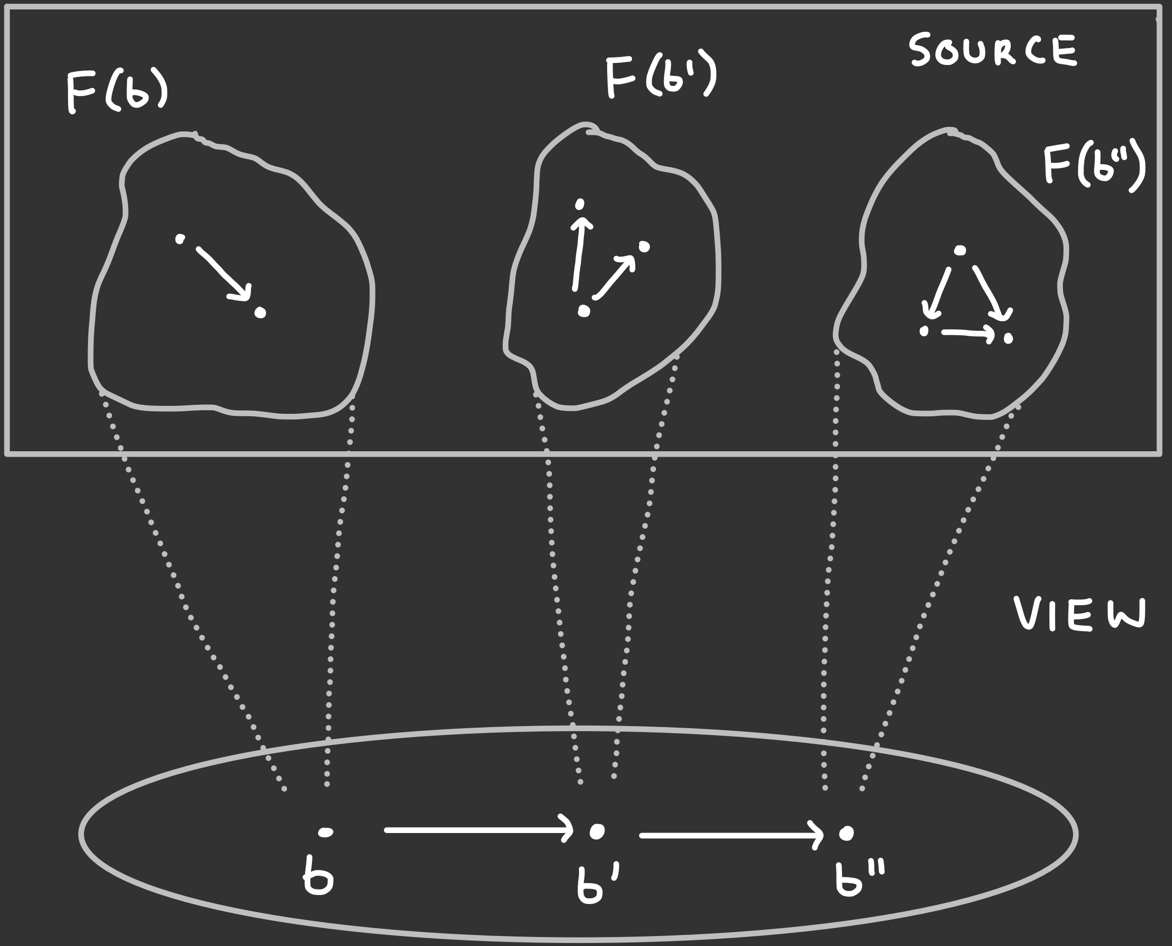
$$\begin{tikzcd}
(A_0, A_1) \arrow[r, "\text{mapping of systems}"] \arrow[d, "\text{coreflection}"'] & (C_0, C_1) \arrow[d, "\text{delta lens}"] \\
(B_0, B_1) \arrow[r, "\text{mapping of systems}"'] \arrow[ur, dashed, "\text{mapping}", "\exists!"'] & (D_0, D_1)
\end{tikzcd}$$

Most general lifting problem delta lenses solve.

Is the PUT of your bx a kind of lifting? What problems does it solve?

INDEXING AND CHANGE-OF-BASE

VIEW states index the SOURCE states.



A delta lens specifies multivalued functions between these indexed sets.

$$\begin{tikzcd}
(A_0, A_1) \arrow[r] \arrow[d, "\text{delta lens}"'] & \boxed{\text{source}} \arrow[d, "\text{Extension}"] \\
(B_0, B_1) \arrow[r, "\text{mapping of systems}"'] & (D_0, D_1)
\end{tikzcd}$$

$$\begin{tikzcd}
\boxed{\text{source}} \arrow[r] \arrow[d, "\text{Restriction}"'] & (C_0, C_1) \arrow[d, "\text{delta lens}"] \\
(B_0, B_1) \arrow[r, "\text{mapping of systems}"'] & (D_0, D_1)
\end{tikzcd}$$

Does VIEW of your bx index the SOURCE?

SUMMARY, QUESTIONS, & FUTURE WORK

- Many tools from category theory to construct and study delta lenses with guaranteed properties
- Many more tools I did not cover:
  - → double categories
  - → lenses between posets, metric spaces, etc.
  - → least-change bx

- How widely applicable are these tools to other kinds of bx?
- How can we implement these tools in specific delta lenses?
- Can category theory help us discover other useful approaches to study bx? And guide us in the questions we ask?