String Diagrams for Optics

Guillaume Boisseau
University of Oxford, UK

Abstract

Optics are a data representation for compositional data access, with lenses as a popular special case. Hedges has presented a diagrammatic calculus for lenses, but in a way that does not generalize to other classes of optic. We present a calculus that works for all optics, not just lenses; this is done by embedding optics into their presheaf category, which naturally features string diagrams. We apply our calculus to the common case of lenses, extend it to effectful lenses, and explore how the laws of optics manifest in this setting.

2012 ACM Subject Classification Software and its engineering → Visual languages; Software and its engineering → Data types and structures; Software and its engineering → Specialized application languages; Software and its engineering → Functional languages

Keywords and phrases Optic, string diagram, lens, category theory, Yoneda lemma

Digital Object Identifier 10.4230/LIPIcs.FSCD.2020.17

Related Version A preprint of this paper is available at [2], `https://arxiv.org/abs/2002.11480`. That version may be extended in the future.

Funding *Guillaume Boisseau*: Work funded by the EPSRC.

Acknowledgements I want to thank the reviewers for the SYCO6 workshop for their thorough reviews, and Jeremy Gibbons for his ever helpful comments.

1 Introduction

Optics are a versatile categorical structure. Their best-known special case, *lenses*, have found uses in a variety of contexts, from machine learning to game theory [5]. Their more general instantiations have been studied in the context of bidirectional data transformations [14]. In all cases, their main feature of interest is their composability and their peculiar bidirectional information flow.

In the interest of making them easier to represent and manipulate, authors often spontaneously use diagrams to construct instances of optics [13, 14]. These diagrams are usually informal, with one notable exception in the work of Hedges [4] on diagrams for lenses. Hedges' diagrammatic calculus however assumes a lot of structure on the underlying categories, in a way that doesn't extend to more general optics.

Here we propose instead a different approach that embeds optics into a larger space (namely its presheaf category) that naturally has string diagrams. Not only does this work for the most general optics, but all the diagrammatic gadgets follow naturally from the embedding, and it even allows for useful diagrams that would not be expressible in the category *Optic* alone.

2 Background

We fix a monoidal category $(M, \otimes, I, \lambda, \mu, a)$ throughout the paper.

We assume readers are familiar with *coends*. For an introduction to the material relevant to the study of optics, see [15, Chapter 2].

▶ Note. We will prefer diagrammatic order for composition, using the symbol $⨾$.

5th International Conference on Formal Structures for Computation and Deduction (FSCD 2020).
Editor: Zena M. Ariola; Article No. 17; pp. 17:1–17:18
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

2.1 Actegories

▶ **Definition 1** ([10]). *An M-actegory (contraction of "action" and "category") is a category C equipped with a functor $\odot_C : M \times C \to C$ (the "action") and two natural structure isomorphisms $\lambda_x : I \odot_C x \xrightarrow{\sim} x$ and $a_{m,n,x} : (m \otimes n) \odot_C x \xrightarrow{\sim} m \odot_C (n \odot_C x)$ that satisfy compatibility axioms with the monoidal structure of M.*

We will drop the subscripts when the relevant actegory is clear from context. The naming of the structure morphisms clashes with those of M on purpose:

▶ **Proposition 2.** *M has canonically the structure of an M-actegory, with $\odot_M = \otimes$, and λ and a as the actegory structure morphisms.*

In what follows, when we use M as an M-actegory, we assume this canonical structure.

2.2 Optics

▶ **Definition 3** ([15, Proposition 3.1.1]). *Given two M-actegories C and D, we construct the category $Optic_{C,D}$ as follows: objects are pairs $\binom{x}{u}$ where $x : C$ and $u : D$, and arrows are elements of the set*

$$Optic_{C,D}(\binom{x}{u}, \binom{y}{v}) := \int^{m:M} C(x, m \odot_C y) \times D(m \odot_D v, u)$$

Given $\alpha : C(x, m \odot_C y)$ and $\beta : D(m \odot_D v, u)$, we will denote the corresponding arrow by $\langle \alpha \mid \beta \rangle_m$. Composition and identities are defined componentwise in the expected way; see [15] for more details.

▶ Note. Expanding the definition of coends in Set, we get that the coend above denotes the set of pairs $\langle \alpha \mid \beta \rangle_m$ with $\alpha : C(x, m \odot_C y)$ and $\beta : D(m \odot_D v, u)$, quotiented by the equation $\langle \alpha \fatsemi (f \odot_C y) \mid \beta \rangle_m = \langle \alpha \mid (f \odot_D v) \fatsemi \beta \rangle_n$ for $f : M(n, m)$.

Except in special cases, this category is not monoidal. This prevents us from having string diagrams in the usual way. We will see how to work around this limitation in the rest of the paper.

▶ **Example 4.** The canonical example of optics are lenses. They arise when $C = D = M$ and the monoidal structure of C is cartesian. We get:

$$Lens_C(\binom{x}{u}, \binom{y}{v}) := \int^{c:C} C(x, c \times y) \times C(c \times v, u)$$

While this presentation is pleasantly symmetrical, lenses are usually described as a pair of functions without this unfamiliar coend. We can in fact calculate that both presentations are equivalent:

$$\begin{aligned} Lens_C(\binom{x}{u}, \binom{y}{v}) &= \int^{c:C} C(x, c \times y) \times C(c \times v, u) \\ &\cong \int^{c:C} C(x, y) \times C(x, c) \times C(c \times v, u) \\ &\cong C(x, y) \times \int^{c:C} C(x, c) \times C(c \times v, u) \\ &\cong C(x, y) \times C(x \times v, u) \end{aligned}$$

We recover the usual formulation: a lens from $\binom{x}{u}$ to $\binom{y}{v}$ is a pair of functions $get : x \to y$ and $put : x \times v \to u$. The intuition is that get extracts some y from a datum x, and put allows replacing that y by a new v, yielding an updated datum u. It is often the case that $x = u$ and $y = v$, making this intuition clearer, but having distinct types allows for more flexibility.

A concrete example of a lens that gives access to a field of a record can be written in Haskell:

```
data Lens x u y v = L (x -> y) (x -> v -> u)

data Person = P { name :: String, address :: String }
personName :: Lens Person Person String String
personName = L get put
   where get (P name _) = name
         put (P _ address) name = P name address
```

The case for distinct types is well illustrated on tuples:

```
tupleSnd :: Lens (a, b) (a, c) b c
tupleSnd = L get put
   where get (_, b) = b
         put (a, _) c = (a, c)
```

⌟

2.3 Tambara Modules

▶ **Definition 5** ([15, Proposition 5.1.1]). *Given two M-actegories C and D, we construct the category $Tamb_{C,D}$ as follows: objects are (pro)functors $P : C^{op} \times D \to Set$ equipped with a natural transformation $strength : \int_{m:M} P(a, b) \to P(m \odot_C a, m \odot_D b)$ compatible with the actegory structures; arrows are strength-preserving natural transformations.*

This generalizes the usual notion of strength for a profunctor.

▶ **Definition 6.** *We construct the bicategory $Tamb$ as follows: objects are M-actegories; Hom-categories are the categories $Tamb_{C,D}$.*

It inherits its bicategorical structure from the bicategory $Prof$ of profunctors: the identities are the hom-profunctors $C(-, =)$, and the tensor (horizontal composition) is profunctor composition, defined as usual as follows:

$$(P \otimes Q)(a, c) = \int^{b} P(a, b) \times Q(b, c)$$

▶ Note. $Prof$ and $Tamb$ share in fact a lot of structure. In a sense $Tamb$ is the analogue of $Prof$ for M-actegories, and we will see that like $Prof$ it supports a rich diagrammatic calculus.

Our interest in Tambara modules comes from the following strong relationship with optics:

▶ **Theorem 7** ([15, Proposition 5.5.2]). $[Optic^{op}_{C,D}, Set] \cong Tamb_{C,D}$

Proof. The proof can be found in [15, Proposition 5.5.2], but initially comes from [12, Proposition 6.1] in the special case where $M = C = D$, along with more results on the structure of both of those categories. ◀

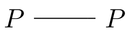
3 Diagrams for Tambara Modules

3.1 Basics

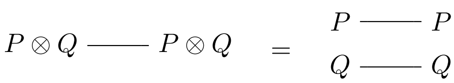
As in any bicategory, cells in *Tamb* can be represented as diagrams, as follows:

A 0-cell (an M-actegory) is represented as a planar region delimited by the other types of cells. For technical reasons we will not represent them in what follows, but it should be kept in mind that 1-cells can only be composed if their types match.

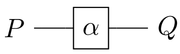
A 1-cell $P : \mathit{Tamb}_{C,D}$ is represented as a wire, with C above and D below:



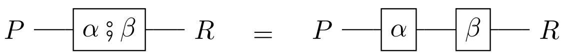
Tensoring (1-cell composition) is vertical juxtaposition (for $P : \mathit{Tamb}_{C,D}$ and $Q : \mathit{Tamb}_{D,E}$):



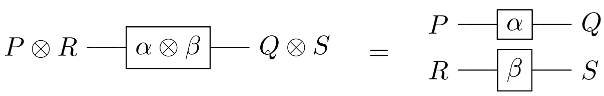
A 2-cell $\alpha : P \to Q$ (for $P, Q : \mathit{Tamb}_{C,D}$) is represented as:



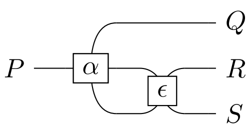
Composition is horizontal juxtaposition:



and tensoring is vertical juxtaposition:



For example, one could represent the following complex composition of cells diagrammatically:



The axioms of bicategories ensure that we can interchange boxes like we do in string diagrams for monoidal categories.

3.2 Oriented Wires

So far, this was common to any bicategory. We can now investigate gadgets specific to *Tamb*.

Let us fix an M-actegory C.

▶ **Definition 8.** *Given $x : C$, let us define two profunctors $R_x := C(-, = \odot_C x)$ and $L_x := C(- \odot_C x, =)$.*

▶ **Proposition 9.** *R_x is in $\mathit{Tamb}_{C,M}$ and L_x is in $\mathit{Tamb}_{M,C}$, where M is taken with its canonical M-actegory structure.*

Proof. R_x is a profunctor $C^{op} \times M \to Set$. The action of the $(m \odot_C -)$ functor provides it with a strength. The same works for L_x. ◀

▶ **Proposition 10.** *R_x extends to a functor $R : C \to Tamb_{C,M}$, and L_x extends to a functor $L : C^{op} \to Tamb_{M,C}$*

Proof. Straightforward from their definitions. ◀

▶ **Proposition 11.** *R and L respect the actegory structures: $R_I \cong L_I \cong M(-,=)$, $R_x \otimes R_m \cong R_{m\odot x}$, and $L_m \otimes L_x \cong L_{m\odot x}$.*

Proof. See appendix A.1. ◀

This justifies the following notation:

$$x \rightarrow\!\!\!- \; x \quad := \quad R_x \;\text{———}\; R_x \tag{1}$$

and

$$x \rightarrow\!\!\!- \boxed{f} \rightarrow\!\!\!- \; y \quad := \quad R_x \;\text{—}\boxed{R_f}\text{—}\; R_y \tag{2}$$

similarly

$$y \;-\!\!\!\leftarrow\; y \quad := \quad L_y \;\text{———}\; L_y \tag{3}$$

and

$$y \;\leftarrow\!\!\!- \boxed{f} \leftarrow\!\!\!- \; x \quad := \quad L_y \;\text{—}\boxed{L_f}\text{—}\; L_x \tag{4}$$

▶ Note. This choice of notation could create confusion as to whether a box on an oriented wire is meant to be seen as in the image of R/L or not. However we will see later that R and L are fully faithful, and thus this confusion fades away: all boxes on an oriented wire are arrows in C.

From the propositions above, we see that this notation respects composition in C as well as the M-actegory structures (note the inversion that happens when tensoring on a right-oriented wire):

$$\begin{array}{c} x \rightarrow\!\!\!- \; x \\ m \rightarrow\!\!\!- \; m \end{array} \quad = \quad m \odot x \rightarrow\!\!\!- \; m \odot x$$

$$\begin{array}{c} m \;-\!\!\!\leftarrow\; m \\ x \;-\!\!\!\leftarrow\; x \end{array} \quad = \quad m \odot x \;-\!\!\!\leftarrow\; m \odot x$$

$$I \rightarrow\!\!\!- \; I \quad = \quad \textit{empty diagram}$$

$$I \;-\!\!\!\leftarrow\; I \quad = \quad \textit{empty diagram}$$

▶ Note. Note that because of the types of the 1-cells (that are not shown in the diagrams), not all tensorings of the oriented wires are allowed. For example, it could be tempting to think that $R_x \otimes R_y \cong R_{y \otimes x}$ for $x, y : C$, but not only is C not monoidal in general, the tensoring doesn't even type-check since both R_x and R_y are objects of $Tamb_{C,M}$.

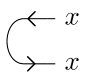
▶ Note. When C is chosen to be M, both R and L provide a monoidal embedding of M into $Tamb_{M,M}$; we will see later that it is also fully faithful. This means that the string diagrams in M have two full and faithful embeddings into the string diagrams of $Tamb$, using the oriented wires.

3.3 Bending Wires

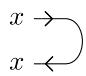
▶ **Proposition 12.** *For a given $x : C$, the modules R_x and L_x are adjoint. Moreover, the structure maps of the adjunction are dinatural in x.*

Proof. $R_x = C(-, = \odot x)$ and $L_x = C(- \odot x, =)$ are clearly adjoint in $Prof$. The adjunction lifts to $Tamb$; see appendix A.2. Dinaturality in x is straightforward from the definition of the unit and counit. ◀

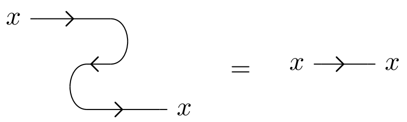
This means that there exist two 2-cells, that we will draw as:



and

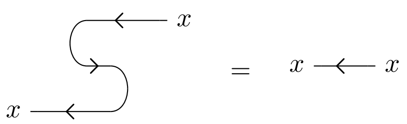


that satisfy the so-called "snake equations":



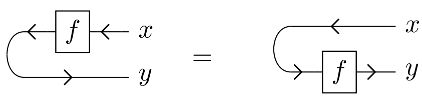
(5)

and



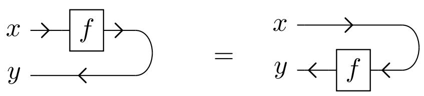
(6)

Those maps are additionally dinatural in x, which means we can also slide C-arrows around them:



(7)

and



(8)

We have discovered an additional property of the diagrammatic language: oriented arrows can be bent downwards. Note that bending upwards is not in general possible.

▶ Note. In the case of set-based lenses (i.e. $C = D = M = Set$ with the cartesian product), the second of those maps (the "cap") was featured in the calculus of [4]. The first map (the "cup") however cannot be expressed in that calculus.

4 Embedding Optics

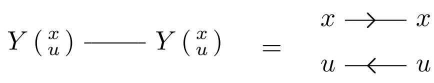
4.1 A Representation Theorem

We will now use this calculus to express optics. Recall from Theorem 7 that presheaves on optics are equivalent to Tambara modules. Consequently, the Yoneda embedding $Y : Optic_{C,D} \to [Optic^{op}_{C,D}, Set] \cong Tamb_{C,D}$ provides a fully faithful embedding of optics into $Tamb$. This is the crucial property that enables our calculus.

▶ **Lemma 13.** $Y\left(\begin{smallmatrix}x\\u\end{smallmatrix}\right) = R_x \otimes L_u$

Proof. By definition of Y, R and L, modulo the equivalence of Theorem 7. ◀

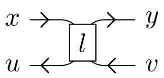
Thus $Y\left(\begin{smallmatrix}x\\u\end{smallmatrix}\right)$ has the following nice diagrammatic notation:



(9)

From this we deduce the main theorem of this paper:

▶ **Theorem 14** (Representation theorem)**.** *Optics $l : Optic_{C,D}(\left(\begin{smallmatrix}x\\u\end{smallmatrix}\right), \left(\begin{smallmatrix}y\\v\end{smallmatrix}\right))$ are in bijection with arrows in $Tamb_{C,D}$ of type:*

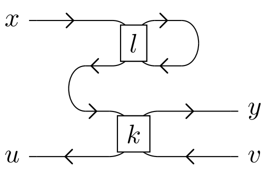


and moreover this bijection is functorial, i.e. composition of optics becomes horizontal composition of diagrams and the identity optic is the identity diagram.

Proof. By full-faithfulness and functoriality of the Yoneda embedding. ◀

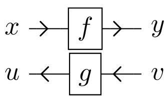
The consequences of this property need stressing: any diagram of this type represents an optic, *even if it is made of subcomponents that are not themselves optics.* A parallel can be drawn with complex numbers: a complex number with no imaginary part represents a real number, regardless of whether it was constructed (using complex operations like rotation) from complex numbers that were not themselves real numbers. In both cases, we can work in this more general space (complex numbers/Tambara modules) to reason more flexibly about the simpler objects (reals/optics).

For example, the following diagram is a valid optic, even though several of its subcomponents are not optics.



4.2 Simple Arrows

The simplest optic we can construct is made out of two simple arrows (i.e. arrows in the base M-actegories). This is sometimes called an *adapter*. Given $f : C(x, y)$ and $g : D(v, u)$, we can see from its type that $R_f \otimes L_g$ is an optic:

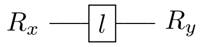


▶ **Lemma 15.** *The optic corresponding to this diagram is* $\langle f \,\text{;}\, \lambda_y^{-1} \mid \lambda_v \,\text{;}\, g\rangle_I$.

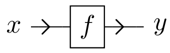
Proof. By a straightforward calculation; see appendix A.3. ◀

The special case of a single simple arrow is particularly interesting:

▶ **Theorem 16.** *All morphisms of type*



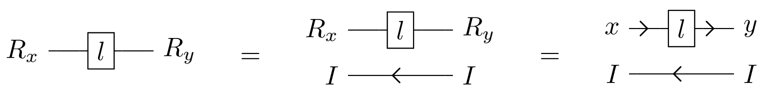
are of the form



for some unique $f : C(x, y)$.

Similarly for L *and wires going to the left.*

Proof. Since $L_I \cong M(-,=)$, we have (using a potentially confusing notation):



Thus by the representation theorem, l can be seen as an optic in $Optic_{C,M}((\begin{smallmatrix} x \\ I \end{smallmatrix}), (\begin{smallmatrix} y \\ I \end{smallmatrix}))$. We then calculate (see appendix A.4) that $Optic_{C,M}((\begin{smallmatrix} x \\ I \end{smallmatrix}), (\begin{smallmatrix} y \\ I \end{smallmatrix})) \cong C(x, y)$, with the reverse direction given by the action of R. The proof for L is identical. ◀

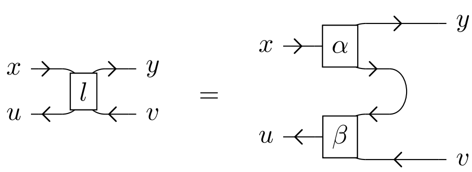
▶ **Corollary 17.** *R and L are fully faithful.*

▶ Note. As pointed out earlier, in the particular case where we choose $C = D = M$ (as in the case of lenses), then R and L both provide a fully-faithful *and monoidal* embedding of the arrows in M into diagrams.

4.3 Refining the Representation Theorem

Together, simple arrows and the cap are enough to represent any optic as a string diagram.

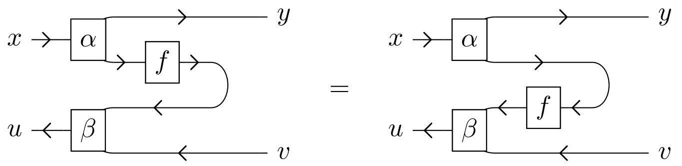
▶ **Theorem 18.** *Given* $\alpha : C(x, m \odot y)$ *and* $\beta : D(m \odot v, u)$*, the optic* $l := \langle \alpha \mid \beta \rangle_m$ *can be represented as follows:*



(10)

Proof. By calculating the composition of the pair of simple arrows with the cap; see appendix A.5. ◀

▶ Note. Recall that the pairs $\langle \alpha \,|\, \beta \rangle_m$ are defined modulo an equivalence relation. How is this compatible with the diagrammatic notation? The equivalence says that $\langle \alpha \mathbin{⨟} (f \odot y) \,|\, \beta \rangle_m = \langle \alpha \,|(f \odot v) \mathbin{⨟} \beta \rangle_n$; diagrammatically, this becomes:



(11)

Which we already know holds, by sliding f along the bent wire!

5 Applications

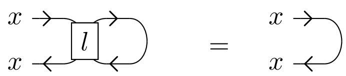
We present two examples of applications of the calculus that illustrate its expressivity.

5.1 Lawful Optics

One of the most striking consequences of this calculus (and the question that led to its discovery) is the neatness with which it can express optic laws.

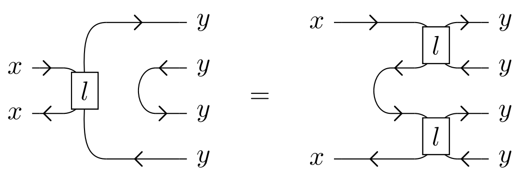
As originally constructed by the Haskell community [9], optics were required to abide by certain round-trip laws that ensure coherence of their operations. Those laws in particular coincide with very-well-behavedness [3] in the case of lenses, which we investigate in more detail in the next section. Riley formalized those laws in a general form [14, Section 3], but the result is rather hard to manipulate. The string calculus enables an alternative (and equivalent) description that is purely diagrammatic:

▶ **Definition 19.** *An optic $l : \left(\begin{smallmatrix} x \\ x \end{smallmatrix}\right) \to \left(\begin{smallmatrix} y \\ y \end{smallmatrix}\right)$ is said to be lawful when*



(12)

and



(13)

▶ Note. We can see that lawful optics are exactly the homomorphisms for the "pair-of-pants" comonoid made from pairs of oriented wires. Interestingly, if we view this comonoid as a procomonad on C, then lawful optics are in bijection with its coalgebras on the carrier R_x. This is a significant generalization of the result by O'Connor [11] that lawful lenses are the coalgebras for the store comonad: here the "pair-of-pants" procomonad precisely generalizes the store comonad.

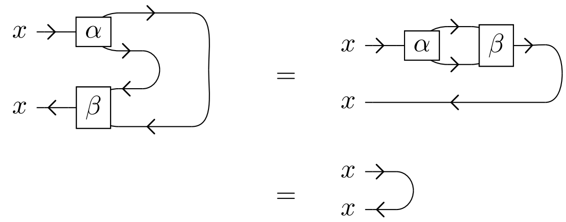
▶ **Theorem 20.** *This notion of lawfulness is equivalent to the one defined by Riley in [14, Section 3].*

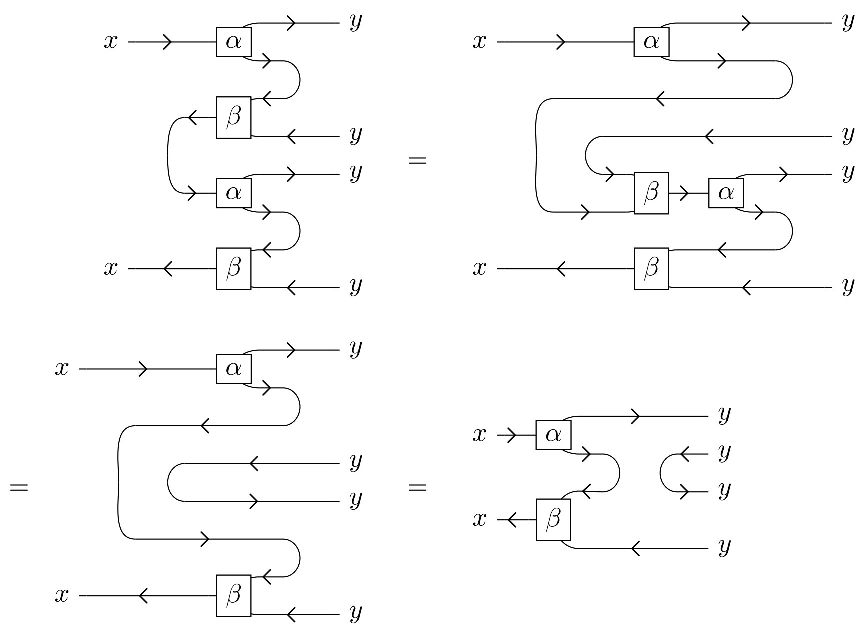
Proof. See appendix A.6. ◀

Thus this diagrammatic definition captures properly the useful and very general notion of lawfulness for optics. Using this theorem, many properties of lawfulness can be derived purely diagrammatically. As an example, let us reprove [14, Proposition 3.0.4]:

▶ **Proposition 21** ([14, Proposition 3.0.4])**.** *If α and β are mutual inverses, then the optic $\langle \alpha \mid \beta \rangle_m$ is lawful.*

Proof.



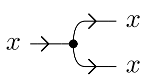
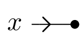


◀

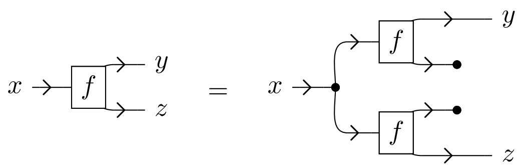
5.2 Cartesian Lenses

The canonical special case of optics, that we mentioned in Example 4, is cartesian lenses. They arise when we restrict ourselves to $C = D = M$ and the monoidal product of C is cartesian.

In this setting, we have two important gadgets in C: duplication and deletion, corresponding respectively to the diagonal map $C(x, x \times x)$ and the terminal map $C(x, I)$. Diagrammatically, we represent them as follows:

 and 

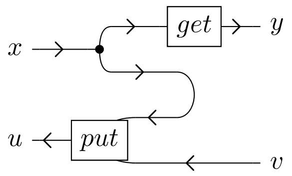
▶ **Lemma 22.** *Given* $f : C(X, Y \times Z)$*, we have*



(14)

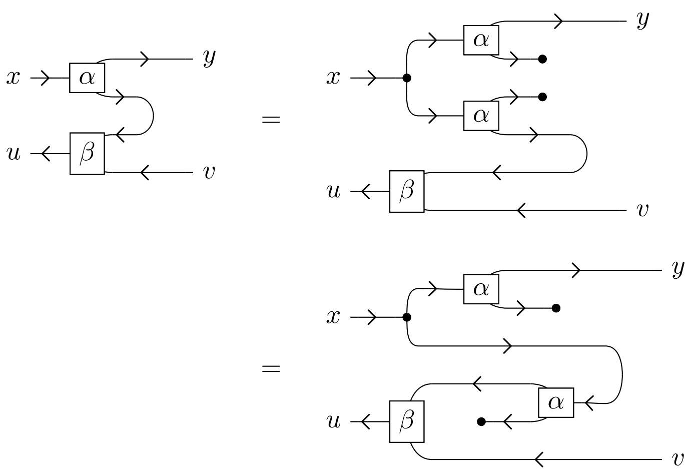
Proof. This corresponds to the standard fact that $f = \langle fst \circ f, snd \circ f \rangle$. ◀

▶ **Theorem 23.** *A lens* $l : \binom{x}{u} \to \binom{y}{v}$ *can be expressed as:*

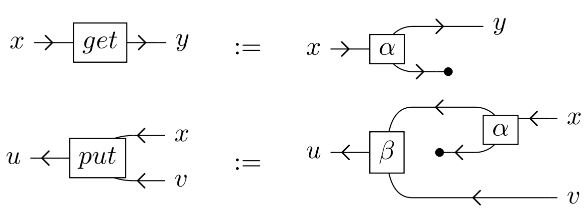


for some $get : C(x, y)$ *and* $put : C(x \times v, u)$*.*

Proof.



which has the required shape. We have:

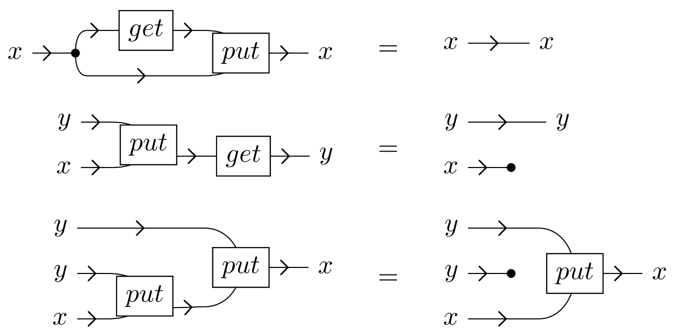


◀

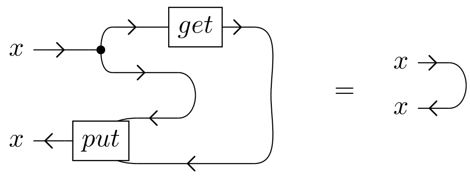
▶ Note. Observe that it is diagrammatically clear that the definition of *put* and *get* in terms of $\langle \alpha \,|\, \beta \rangle_m$ respects the equivalence relation induced by the coend.

We recovered purely diagrammatically the usual formulation of lenses in terms of *get* and *put*, that we had derived in Example 4. In this setting, various properties of lenses can be investigated purely diagrammatically. As an example, let us revisit [14, Proposition 3.0.3], which captures the fact that the general notion of lawfulness for optics coincides with the familiar PutGet, GetPut and PutPut laws [3] (together called "very-well-behavedness") in the case of lenses.

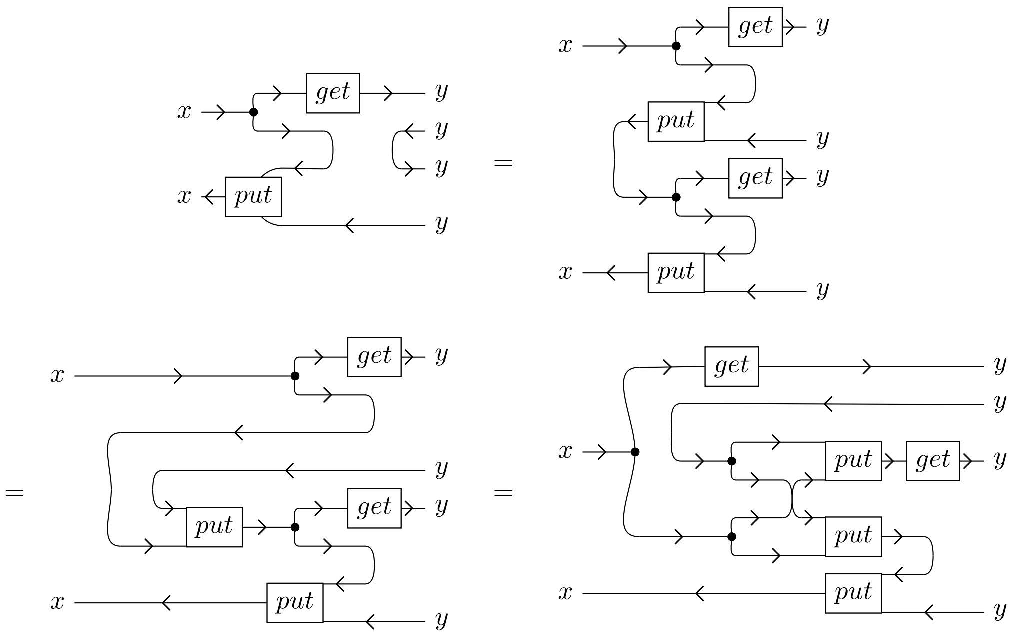
▶ **Proposition 24** ([14, Proposition 3.0.3]). *A lens $l : \left(\begin{smallmatrix} x \\ x \end{smallmatrix}\right) \to \left(\begin{smallmatrix} y \\ y \end{smallmatrix}\right)$ is lawful iff the following three laws (respectively called PutGet, GetPut and PutPut) hold in C:*



Proof. Diagrammatically, the fact that a lens is lawful reads:



which is exactly the PutGet law, and:



It is straightforward to see that the PutPut and the GetPut laws together entail this equality. By applying the deletion map successively to the outputs, one can also show that this equation entails those two laws, when y is inhabited. ◀

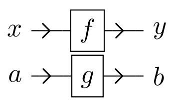
5.3 Effectful Lenses

We now turn to a less common example: effectful lenses. They stem from the desire to allow lenses to perform effects while retrieving or updating data. Various approaches have been proposed; see Abou-Saleh et al. [1] for an overview.

Let C be a cartesian category and T a monad on C. We would like an optic that resembles cartesian lenses from the previous section, but with effectful arrows. This means that we would like our arrows to live in the Kleisli category C_T. This category however is rarely monoidal, let alone cartesian: for it to be monoidal, the monad T would need to be commutative, which rules out large classes of effects that we might want to use. Thus we cannot reuse the results from the previous section. Here we can instead make good use of the generality of monoidal actions: C_T may not be monoidal, but when T is strong (which is rather common), the product of C extends to an action of C on C_T [14, Proposition 4.9.3]. This is enough to define an optic for monadic lenses:

$$MLens_T(\binom{x}{u}, \binom{y}{v}) := \int^{c:C} C_T(x, c \times y) \times C_T(c \times v, u)$$

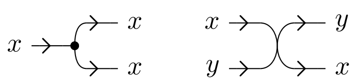
Let us now investigate the diagrams for such an optic. Recall the details of how oriented wires are typed. Here the acting category is C, which means that in a diagram like the following, the typing rules enforce that x, y and f can live in C_T, but a, b and g can only live in C.



This is why we don't need C_T to be monoidal: this calculus only allows an arrow in C_T to be tensored with arrows in C. This gives us a string diagram calculus where otherwise none would have been possible.

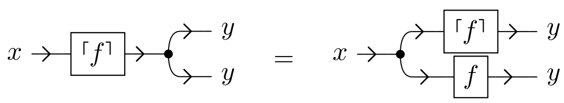
The distinction between effectful maps (in C_T) and pure maps (in C) is an important aspect of this calculus. Note that every pure map f can be lifted to an effectful map written $\ulcorner f \urcorner$, via a canonical functor. This functor also respects the actegory structures, and therefore allows us to embed the pure lenses from the previous section as monadic lenses.

This calculus even inherits some of the diagrammatic features of the previous section: the duplication map and the swap still exist and are represented as before.



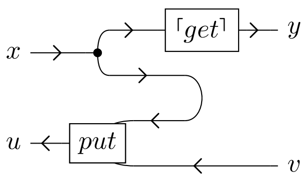
(15)

The difference is that the bottom wire can only carry maps living in C. Whereas before, all maps could be "slid through" the duplication map and swap, now only C-maps (aka pure maps) can:

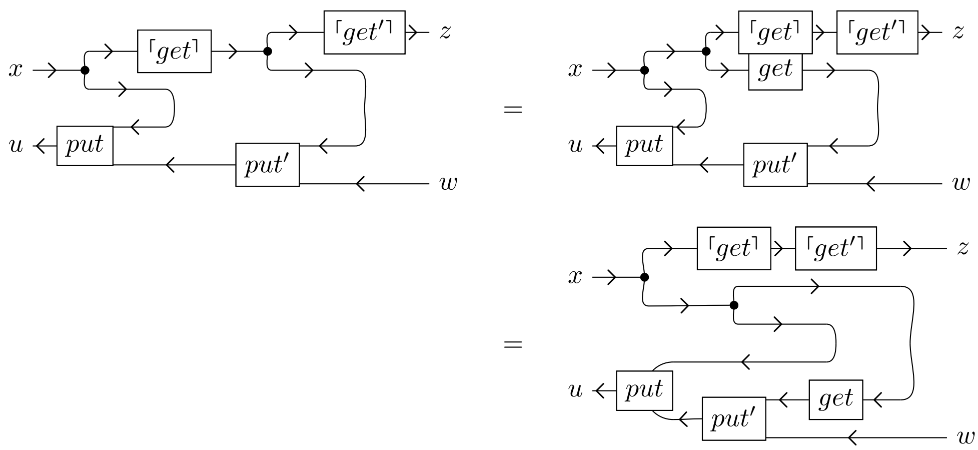


(16)

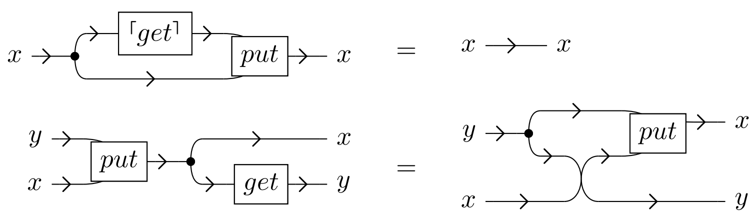
We now restrict ourselves to the monadic lenses proposed by Abou-Saleh et al. [1]. Those lenses are modeled closer to ordinary lenses, in particular their definition does not involve a coend. They are composed of $get : C(x, y)$ and $put : C_T(x \times v, u)$. Diagrammatically they look quite like cartesian lenses:



Note that *get* is required to be pure. This is important to ensure that composing two such lenses stays of that simplified shape. This non-trivial fact can be seen diagrammatically in what follows: if *get* was not pure, it couldn't be slid across the duplication map.



Finally, this new calculus can express the laws proposed by Abou-Saleh et al. [1], making them much easier to reason about:



6 Conclusion and Future Work

We have presented a calculus that flowed naturally from the Yoneda embedding of optics into Tambara modules. We have shown that it was well-suited for expressing common properties of optics and proving useful theorems generally, some of which would otherwise be painful to prove. This work however is only the start: it provides the basis of a calculus, whose expressive power hasn't yet been explored in the plethora of topics where optics have found a use. In particular, we expect new specific diagrammatic properties like those of lenses to arise for other kinds of optics like prisms or traversals.

Then, the calculus could be linked with related constructions, like the calculus for teleological categories from [4], or the Int construction from [7].

Properties of *Tamb* as a bicategory also seem worth exploring, in particular its strong similarity with *Prof*, and the link between the properties of M and those of *Tamb*.

Finally, diagrams in *Tamb* with multiple ingoing and outgoing legs seem to relate to combs as in [8] and dialogues in the style of [6]; there is potential for using *Tamb* to provide a basis for general diagrammatic descriptions of those objects.

References

1. Faris Abou-Saleh, James Cheney, Jeremy Gibbons, James McKinna, and Perdita Stevens. Reflections on Monadic Lenses. *A List of Successes That Can Change the World*, 9600:1–31, April 2016. `doi:10.1007/978-3-319-30936-1_1`.
2. Guillaume Boisseau. String Diagrams for Optics. *arXiv:2002.11480 [math]*, February 2020. `arXiv:2002.11480`.
3. J. Nathan Foster, Michael B. Greenwald, Jonathan T. Moore, Benjamin C. Pierce, and Alan Schmitt. Combinators for bi-directional tree transformations: A linguistic approach to the view update problem. In Jens Palsberg and Martín Abadi, editors, *Proceedings of the 32nd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2005, Long Beach, California, USA, January 12-14, 2005*, pages 233–246. ACM, 2005. `doi:10.1145/1040305.1040325`.
4. Jules Hedges. Coherence for lenses and open games. *arXiv:1704.02230 [cs, math]*, September 2017. `arXiv:1704.02230`.
5. Jules Hedges. Lenses for Philosophers, August 2018. URL: `https://julesh.com/2018/08/16/lenses-for-philosophers/`.
6. Jules Hedges. The game semantics of game theory. *arXiv:1904.11287 [cs]*, April 2019. `arXiv:1904.11287`.
7. André Joyal, Ross Street, and Dominic Verity. Traced monoidal categories. *Mathematical Proceedings of the Cambridge Philosophical Society*, 119(3):447–468, April 1996. `doi:10.1017/S0305004100074338`.
8. Aleks Kissinger and Sander Uijlen. A categorical semantics for causal structure. In *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017*, pages 1–12. IEEE Computer Society, 2017. `doi:10.1109/LICS.2017.8005095`.
9. Edward Kmett. Haskell lens library, February 2020. URL: `https://hackage.haskell.org/package/lens`.
10. nLab authors. Actegory. URL: `http://ncatlab.org/nlab/show/actegory`.
11. Russell O'Connor. Lenses Are Exactly the Coalgebras for the Store Comonad, November 2010. URL: `https://r6research.livejournal.com/23705.html`.
12. Craig Pastro and Ross Street. Doubles for monoidal categories. *Theory and Applications of Categories*, 21, November 2007. `arXiv:0711.1859`.
13. Matthew Pickering, Jeremy Gibbons, and Nicolas Wu. Profunctor optics: Modular data accessors. *Programming Journal*, 1(2):7, March 2017. `doi:10.22152/programming-journal.org/2017/1/7`.
14. Mitchell Riley. Categories of Optics. *arXiv:1809.00738 [math]*, September 2018. `arXiv:1809.00738`.
15. Mario Román. *Profunctor Optics and Traversals.* Msc Thesis, University of Oxford, January 2020. `arXiv:2001.08045`.

A Proofs

A.1 R Respects the Actegory Structure (Proposition 11)

Proof (Proposition 11).

$$\begin{aligned} R_I &= M(-, = \odot_M I) \\ &= M(-, = \otimes I) \\ &\cong M(-, =) \end{aligned}$$

$$\begin{aligned} R_x \otimes R_m &= \int^{n:M} C(-, n \odot_C x) \times M(n, = \odot_M m) \\ &= \int^{n:M} C(-, n \odot_C x) \times M(n, = \otimes m) \\ &\cong C(-, (= \otimes m) \odot_C x) \\ &\cong C(-, = \odot_C (m \odot_C x)) \\ &= R_{m \odot_C x} \end{aligned}$$

$$\begin{aligned} L_I &= M(- \odot_M I, =) \\ &= M(- \otimes I, =) \\ &\cong M(-, =) \end{aligned}$$

$$\begin{aligned} L_m \otimes L_x &= \int^{n:M} M(- \odot_M m, n) \times C(n \odot_C x, =) \\ &= \int^{n:M} M(- \otimes m, n) \times C(n \odot_C x, =) \\ &\cong C((- \otimes m) \odot_C x, =) \\ &\cong C(- \odot_C (m \odot_C x), =) \\ &= L_{m \odot_C x} \end{aligned}$$

It is easy to check that the corresponding strengths coincide as well. ◀

A.2 R and L Are Adjoint (Proposition 12)

Proof (Proposition 12). The counit $\varepsilon : R_x \otimes L_x \to C(-, =)$ of the adjunction in *Prof* is given by composition in C. We need it to commute with strength:

$$\begin{array}{ccc} \int^{b} C(a, b \odot x) \otimes C(b \odot x, c) & \xrightarrow{\fatsemi} & C(a, c) \\ {\scriptstyle strength}\downarrow & & \downarrow{\scriptstyle strength} \\ \int^{b'} C(m \odot a, b' \odot x) \otimes C(b' \odot x, m \odot c) & \xrightarrow{\fatsemi} & C(m \odot a, m \odot c) \end{array}$$

We inline the definition of *strength*, and move the coends out by continuity, to get an equivalent square:

$$\begin{array}{ccc}
C(a, b \odot x) \otimes C(b \odot x, c) & \xrightarrow{\fatsemi} & C(a,c) \\
\downarrow{\scriptstyle (m\odot -)\otimes(m\odot -)} & & \downarrow{\scriptstyle (m\odot -)} \\
C(m \odot a, m \odot (b \odot x)) \otimes C(m \odot (b \odot x), m \odot c) & \xrightarrow{\fatsemi} & C(m \odot a, m \odot c) \\
\downarrow{\scriptstyle C(id, a^{-1})\otimes C(a, id)} & & \downarrow{\scriptstyle id} \\
C(m \odot a, (m \otimes b) \odot x) \otimes C((m \otimes b) \odot x, m \odot c) & \xrightarrow{\fatsemi} & C(m \odot a, m \odot c)
\end{array}$$

The top square commutes by functoriality of $(m \odot -)$; the bottom one by the fact that $a^{-1} \fatsemi a = id$.

Similarly, the unit also lives in *Tamb*. This is enough for the adjunction to lift from *Prof* to *Tamb*. ◀

A.3 Diagram for Simple Arrows (Lemma 15)

Proof (Lemma 15). The diagram corresponds to the 2-cell $R_f \otimes L_g$.

It has type

$$\begin{aligned}
&R_f \otimes L_g : R_x \otimes L_u \to R_y \otimes L_v \\
&\qquad = \int_{ab} (\int^m R_x(a,m) \times L_u(m,b)) \to (\int^m R_y(a,m) \times L_v(m,b))
\end{aligned}$$

And value

$$\begin{aligned}
(R_f \otimes L_g)(\langle p \,|\, q\rangle_m) &= \langle R_f(p) \,|\, L_g(q)\rangle_m \\
&= \langle p \fatsemi (m \odot f) \,|(m \odot g) \fatsemi q\rangle_m
\end{aligned}$$

To get the preimage through Y, we apply this map to the identity optic.

$$\begin{aligned}
(R_f \otimes L_g)(id_{\binom{x}{u}}) &= (R_f \otimes L_g)(\langle \lambda_x^{-1} \,|\, \lambda_u\rangle_I) \\
&= \langle \lambda_x^{-1} \fatsemi (I \odot f) \,|(I \odot g) \fatsemi \lambda_u\rangle_I \\
&= \langle f \fatsemi \lambda_y^{-1} \,|\, \lambda_v \fatsemi g\rangle_I
\end{aligned}$$ ◀

A.4 Simple Arrows Embed Fully-Faithfully (Theorem 16)

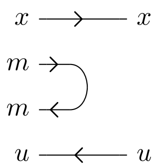
Proof (Theorem 16). We calculate:

$$\begin{aligned}
&Optic_{C,M}(\binom{x}{I}, \binom{y}{I}) \\
&= \int^m C(x, m \odot_C y) \times M(m \odot_M I, I) \\
&= \int^m C(x, m \odot_C y) \times M(m \otimes I, I) \\
&\cong \int^m C(x, m \odot_C y) \times M(m, I) \\
&\cong C(x, I \odot_C y) \\
&\cong C(x,y)
\end{aligned}$$

By following the isomorphisms, we get that the reverse direction is the function $f : C(x,y) \mapsto \langle f \fatsemi \lambda_y^{-1} \,|\, \lambda_I\rangle_I$, which as we saw previously corresponds to $f \mapsto \iota(f, id_I) = R_f \otimes L_{id_I} = R_f$. ◀

A.5 Representation Theorem (Theorem 18)

▶ **Lemma 25.** *The optic corresponding to this diagram is* $\langle id_{m\odot x} \,|\, id_{m\odot u}\rangle_m$.



Proof. Let us name the map corresponding to this diagram $\Lambda_{x,m,u}$.

Knowing the action of the cap ε, we obtain by a tedious calculation that we will omit here:

$$\Lambda_{x,m,u} : Y\binom{m\odot x}{m\odot u} \to Y\binom{x}{u}$$
$$= \langle \alpha \,|\, \beta\rangle_n \mapsto \langle \alpha \fatsemi a^{-1}_{n,m,x} \,|\, a_{n,m,u} \fatsemi \beta\rangle_{n\otimes m}$$

Thus the corresponding optic is:

$$\begin{aligned}\Lambda_{x,m,u}(id_{\binom{m\odot x}{m\odot u}}) &= \Lambda_{x,m,u}(\langle \lambda^{-1}_{m\odot x} \,|\, \lambda_{m\odot u}\rangle_I)\\ &= \langle \lambda^{-1}_{m\odot x} \fatsemi a^{-1}_{I,m,x} \,|\, a_{I,m,u} \fatsemi \lambda_{m\odot u}\rangle_{I\otimes m}\\ &= \langle \lambda^{-1}_m \odot x \,|\, \lambda_m \odot u\rangle_{I\otimes m}\\ &= \langle (\lambda^{-1}_m \fatsemi \lambda_m) \odot x \,|\, id_{m\odot u}\rangle_m\\ &= \langle id_{m\odot x} \,|\, id_{m\odot u}\rangle_m\end{aligned}$$ ◀

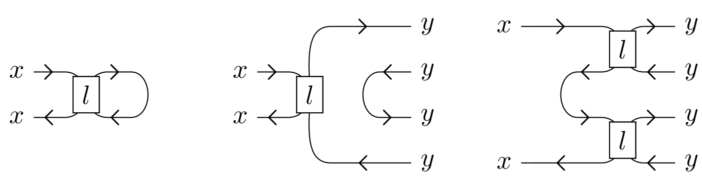
Proof (Theorem 18). The right-hand-side diagram is the composition of two optics of which we know the value: the first is $\langle \alpha \fatsemi \lambda^{-1}_{m\odot y} \,|\, \lambda_{m\odot v} \fatsemi \beta\rangle_I$; the second is $\langle id_{m\odot y} \,|\, id_{m\odot v}\rangle_m$.

The resulting optic is thus their composition:

$$\begin{aligned}&\langle \alpha \fatsemi \lambda^{-1}_{m\odot y} \,|\, \lambda_{m\odot v} \fatsemi \beta\rangle_I \fatsemi \langle id_{m\odot y} \,|\, id_{m\odot v}\rangle_m\\ &= \langle \alpha \fatsemi \lambda^{-1}_{m\odot y} \fatsemi (I \odot id_{m\odot y}) \fatsemi a^{-1}_{m,I} \,|\, a_{m,I} \fatsemi (I \odot id_{m\odot v}) \fatsemi \lambda_{m\odot v} \fatsemi \beta\rangle_{I\otimes m}\\ &= \langle \alpha \fatsemi \lambda^{-1}_{m\odot y} \fatsemi a^{-1}_{m,I} \,|\, a_{m,I} \fatsemi \lambda_{m\odot v} \fatsemi \beta\rangle_{I\otimes m}\\ &= \langle \alpha \fatsemi (\lambda^{-1}_I \odot y) \,|(\lambda_I \odot v) \fatsemi \beta\rangle_{I\otimes m}\\ &= \langle \alpha \fatsemi (\lambda^{-1}_I \odot y) \fatsemi (\lambda_I \odot y) \,|\, \beta\rangle_m\\ &= \langle \alpha \,|\, \beta\rangle_m\end{aligned}$$ ◀

A.6 Lawfulness in Diagrams (Theorem 20)

Proof (Theorem 20). Lawfulness in [14, Section 3] is based on three maps named *outside*, *once*, and *twice*. Unpacking the definitions, those three maps applied to an optic l correspond respectively to the three diagrams:



The interesting insight is that the complicated $Optic^2_M$ coend from Riley's paper can be easily constructed diagrammatically by tensoring oriented wires as above. The theorem then follows directly from Riley's definition of lawfulness. ◀