

---

# Internal Lenses as Functors and Cofunctors

Bryce Clarke

Centre of Australian Category Theory, Macquarie University, Australia

July 1, 2019

Lenses may be characterised as objects in the category of algebras over a monad, however they are often understood instead as morphisms, which propagate updates between systems. Working internally to a category with pullbacks, we define lenses as simultaneously functors and cofunctors between categories. We show that lenses may be canonically represented as a particular commuting triangle of functors, and unify the classical state-based lenses with both c-lenses and d-lenses in this framework. This new treatment of lenses leads to considerable simplifications that are important in applications, including a clear interpretation of lens composition.

## 1 Introduction

Lenses form a mathematical structure that aims to capture the fundamental aspects of certain synchronisations between pairs of systems. The central goal of such synchronisation is to coherently propagate updates in one system to updates in another, and vice versa. The precise nature of the synchronisation process depends closely on the type of system being studied, and thus many different kinds of lenses have been defined to characterise various applications and examples.

Although a relatively recent subject for detailed abstract study, lenses are an impressive example of applied category theory, playing major roles in database view updating, in Haskell programs of many kinds, and in diverse examples of Systems Interoperations, Data Sharing, and Model-Driven Engineering. Thus, further clarifying the category-theoretic status and systematising the use of lenses, as this paper aims to do, is an important part of applied category theory.

Lenses were originally introduced [8] to provide a solution to the view-update problem [3]. In treatments of the view-update problem systems are generally modelled as a set of states, where it is possible to update from one state of the system to any other, and the only information retained about this update are its initial and final states. Thus a system may be understood as a *codiscrete category* on its set of states  $S$  with set of updates  $S \times S$  given by a pair of initial and final states.

Lenses have long been recognised to be some kind of morphism between systems. An obvious notion of morphism between systems is simply a function  $g: S \rightarrow V$  between their sets of states. Since systems may be modelled as codiscrete categories, there is also an induced function  $g \times g: S \times S \rightarrow V \times V$  between the sets of updates of these systems. The map  $g: S \rightarrow V$  is called the **Get** function and provides the first component of a lens between the systems  $S$  and  $V$ , often called the *source* and *view*.

---

Bryce Clarke: [bryce.clarke1@hdr.mq.edu.au](mailto:bryce.clarke1@hdr.mq.edu.au)

---

The second component of a lens is called the **Put** function  $p: S \times V \rightarrow S$  whose role is less obvious. The set  $S \times V$  may be interpreted as the set of *anchored view updates* via the induced function  $g \times 1_V: S \times V \rightarrow V \times V$  which produces a view update whose initial state is given by the **Get** function. Therefore the induced function  $\langle \pi_0, p \rangle: S \times V \rightarrow S \times S$  may be regarded as the **Put** function propagating every anchored view update to a source update, illustrated in the diagram below.

$$\begin{array}{ccc}
 S & s & \dashrightarrow p(s, v) \\
 g \downarrow & \vdots & \vdots \\
 V & gs & \longrightarrow v
 \end{array}$$

Frequently the **Get** and **Put** functions of a lens are required to satisfy three additional axioms, called the *lens laws*, which ensure the synchronisation of updates between systems is well-behaved.

$$\begin{array}{ccc}
 S \times V \xrightarrow{p} S & S \xrightarrow{\langle 1_S, g \rangle} S \times V & S \times V \times V \xrightarrow{p \times 1_V} S \times V \\
 \searrow \pi_1 \quad \downarrow g & \searrow 1_S \quad \downarrow p & \searrow \pi_{0,2} \quad \downarrow p \\
 V & S & S \times V \xrightarrow{p} S
 \end{array}$$

In order from left to right: the **Put-Get** law ensures that the systems  $S$  and  $V$  are indeed synchronised under the **Get** and **Put** functions; the **Get-Put** law ensures that anchored view updates which are identities are preserved by the **Put** function; the **Put-Put** law ensures that composite anchored view updates are preserved under the **Put** function.

In summary, a *state-based lens* [8], denoted  $\Lambda: S \rightleftharpoons V$ , consists of a **Get** function  $g: S \rightarrow V$  and a **Put** function  $p: S \times V \rightarrow S$  satisfying the lens laws. Early mathematical work [13] characterised state-based lenses as algebras for a well-known monad,

$$\begin{array}{ccc}
 \mathbf{Set}/_V & \longrightarrow & \mathbf{Set}/_V \\
 g: S \rightarrow V & \longmapsto & \pi_1: S \times V \rightarrow V
 \end{array}$$

which may be generalised to any category with finite products. It was later shown that lenses are also coalgebras for a comonad [9] and may be defined inside any cartesian closed category. While these works took the first steps towards internalisation of lenses, they characterised lenses as objects in the category of Eilenberg-Moore (co)algebras, rather than morphisms between sets, and did not account for composition of lenses.

A significant shortcoming of state-based lenses in many applications is they only describe synchronisation between systems as a set of states, or codiscrete categories, ignoring the information on how states are updated. This motivated the independent development of both *c-lenses* [14] and *d-lenses* [7] between systems modelled as arbitrary categories. Making use of comma categories instead of products, c-lenses were defined as algebras for a classical KZ-monad [17], and may be also understood as split Grothendieck opfibrations. In contrast d-lenses were shown [11] to be more general, as split opfibrations *without* the usual universal property, and could only be characterised as algebras for a *semi-monad* satisfying an additional axiom.

Later work [12] showed that the category of state-based lenses (as morphisms) is a full subcategory of the category of d-lenses (which also contains a subcategory of c-lenses). Despite this unification of category-based lenses, composition was still defined in an ad hoc fashion, and there was no mathematical explanation as to why lenses characterised as algebras should be understood as morphisms.

---

## Summary of Paper

The contribution of this paper may be summarised as follows:

- Generalise the theory of lenses internal to any category  $\mathcal{E}$  with pullbacks.
- Characterise internal cofunctors as spans of internal functors with left leg identity-on-objects and right leg a discrete opfibration.
- Define an *internal lens* as an internal functor and an internal cofunctor, which provide the appropriate notion of **Get** and **Put**, respectively.
- Show there is a well-defined category  $\mathbf{Lens}(\mathcal{E})$  whose objects are internal categories and whose morphisms are internal lenses.
- Demonstrate state-based lenses, c-lenses, and d-lenses as examples of internal lenses.

## 2 Background

This section provides a brief review of the relevant internal category theory required for the paper, most of which can be found in standard references such as [4, 15, 16]. Throughout we work internal to a category  $\mathcal{E}$  with pullbacks, with the main examples being  $\mathcal{E} = \mathbf{Set}, \mathbf{Cat}$ .

The idea is that a system may be defined as an internal category with an *object of states* and an *object of updates*. An internal functor will later be interpreted as the **Get** component of an internal lens, while internal discrete opfibrations will also be central in defining the **Put** component of an internal lens. Codiscrete categories and arrow categories are presented as examples and will later be used to define internal versions of state-based lenses and c-lenses.

### 2.1 Internal Categories

**Definition 1.** An *internal category*  $\mathbf{V}$  consists of an object of objects  $V_0$  and an object of morphisms  $V_1$  together with a span,

$$\begin{array}{ccc} & V_1 & \\ l_0 \swarrow & & \searrow r_0 \\ V_0 & & V_0 \end{array} \quad (1)$$

where  $l_0: V_1 \rightarrow V_0$  is the *domain map* and  $r_0: V_1 \rightarrow V_0$  is the *codomain map*, and the pullbacks,

$$\begin{array}{ccc} & V_2 & \\ l_1 \swarrow & \downarrow & \searrow r_1 \\ V_1 & & V_1 \\ r_0 \searrow & & \swarrow l_0 \\ & V_0 & \end{array} \quad \begin{array}{ccc} & V_3 & \\ l_2 \swarrow & \downarrow & \searrow r_2 \\ V_2 & & V_2 \\ r_1 \searrow & & \swarrow l_1 \\ & V_1 & \end{array} \quad (2)$$

where  $V_2$  is the object of composable pairs and  $V_3$  is the object of composable triples, as well as an *identity map*  $i: V_0 \rightarrow V_1$  and *composition map*  $c: V_2 \rightarrow V_1$  satisfying the following commutative diagrams:

$$\begin{array}{ccccc} V_0 & \xrightarrow{i} & V_1 & & V_1 & \xleftarrow{l_1} & V_2 & \xrightarrow{r_1} & V_1 & & V_1 & \xrightarrow{\langle 1, ir_0 \rangle} & V_2 & & V_3 & \xrightarrow{c \times r_1} & V_2 \\ i \downarrow & \searrow 1_{V_0} & \downarrow l_0 & & l_0 \downarrow & & \downarrow c & & \downarrow r_0 & & \langle i l_0, 1 \rangle \downarrow & \searrow 1_{V_1} & \downarrow c & & l_1 \times c \downarrow & & \downarrow c \\ V_1 & \xrightarrow{r_0} & V_0 & & V_0 & \xleftarrow{l_0} & V_1 & \xrightarrow{r_0} & V_0 & & V_2 & \xrightarrow{c} & V_1 & & V_2 & \xrightarrow{c} & V_1 \end{array} \quad (3)$$

*Notation.* The commutative diagrams (3) include universal morphisms induced by the pullbacks (2) defined by the diagrams below:

$$\begin{array}{cccc}
 \begin{array}{ccc}
 & V_1 & \\
 l_0 \swarrow & \vdots & \searrow 1_{V_1} \\
 V_0 & \langle il_0, 1 \rangle & \\
 \downarrow i & \downarrow & \downarrow \\
 V_1 & V_2 & V_1 \\
 l_1 \swarrow & \downarrow & \searrow r_1 \\
 & \checkmark & \\
 & V_1 & \\
 r_0 \swarrow & & \searrow l_0 \\
 & V_0 & 
 \end{array}
 &
 \begin{array}{ccc}
 & V_1 & \\
 & \downarrow r_0 & \\
 & V_0 & \\
 1_{V_1} \swarrow & \downarrow & \searrow \\
 V_1 & V_2 & V_1 \\
 l_1 \swarrow & \downarrow & \searrow i \\
 & \checkmark & \\
 & V_1 & \\
 r_0 \swarrow & & \searrow l_0 \\
 & V_0 & 
 \end{array}
 &
 \begin{array}{ccc}
 & V_3 & \\
 l_2 \swarrow & \vdots & \searrow r_2 \\
 V_2 & l_1 \times c & V_2 \\
 \downarrow l_1 & \downarrow & \downarrow c \\
 V_1 & V_2 & V_1 \\
 l_1 \swarrow & \downarrow & \searrow r_1 \\
 & \checkmark & \\
 & V_1 & \\
 r_0 \swarrow & & \searrow l_0 \\
 & V_0 & 
 \end{array}
 &
 \begin{array}{ccc}
 & V_3 & \\
 l_2 \swarrow & \vdots & \searrow r_2 \\
 V_2 & c \times r_1 & V_2 \\
 \downarrow c & \downarrow & \downarrow r_1 \\
 V_1 & V_2 & V_1 \\
 l_1 \swarrow & \downarrow & \searrow r_1 \\
 & \checkmark & \\
 & V_1 & \\
 r_0 \swarrow & & \searrow l_0 \\
 & V_0 & 
 \end{array}
 \end{array} \quad (4)$$

**Example 1.** A *small category* is an internal category in **Set**. Thus a small category consists of a *set* of objects and a *set* of morphisms, together with *functions* specifying the domain, codomain, identity, and composition.

**Example 2.** A (small) *double category* is an internal category in **Cat**, the category of small categories and functors. Thus a double category consists of a *category* of objects and a *category* of morphisms, together with *functors* specifying the domain, codomain, identity, and composition.

**Example 3.** Assume  $\mathcal{E}$  has finite limits. A *codiscrete category* is an internal category whose object of morphisms is the product  $V \times V$  of its object of objects  $V$ , with domain and codomain maps given by the left and right projections:

$$\begin{array}{ccc}
 & V \times V & \\
 \pi_0 \swarrow & & \searrow \pi_1 \\
 V & & V
 \end{array}$$

The identity map is given by the diagonal  $\langle 1_V, 1_V \rangle: V \rightarrow V \times V$ , and object of composable pairs is given by the product  $V \times V \times V$ , with the composition map defined by the following universal morphism:

$$\begin{array}{ccccc}
 & & V \times V \times V & & \\
 & \pi_0 \swarrow & \vdots \pi_{0,2} & \searrow \pi_2 & \\
 V & \xleftarrow{\pi_0} & V \times V & \xrightarrow{\pi_1} & V
 \end{array}$$

**Example 4.** Let  $\mathbf{V}$  be an internal category. The *arrow category*  $\Phi(\mathbf{V})$  has an object of objects  $V_1$  and an object of morphisms  $V_{11}$  defined by the pullback,

$$\begin{array}{ccccc}
 & & V_{11} & & \\
 & \langle \sigma, \rho \rangle \swarrow & \downarrow & \searrow \langle \lambda, \tau \rangle & \\
 & V_2 & \checkmark & V_2 & \\
 l_1 \swarrow & \downarrow c & & \downarrow c & \searrow r_1 \\
 V_1 & & V_1 & & V_1
 \end{array}$$

with domain map  $\sigma: V_{11} \rightarrow V_1$  and codomain map  $\tau: V_{11} \rightarrow V_1$ . The pullback  $V_{11}$  may be understood as the object of commutative squares in  $\mathbf{V}$ . The identity and composition maps require tedious invention of notation to define precisely, however we note they are induced from the diagrams (3).

## 2.2 Internal Functors

**Definition 2.** Let  $\mathbf{S}$  and  $\mathbf{V}$  be internal categories. An *internal functor*  $G: \mathbf{S} \rightarrow \mathbf{V}$  consists of morphisms,

$$g_0: S_0 \longrightarrow V_0 \quad g_1: S_1 \longrightarrow V_1$$

satisfying the following commutative diagrams:

$$\begin{array}{ccccc} S_0 & \xleftarrow{l_0} & S_1 & \xrightarrow{r_0} & S_0 & & S_0 & \xrightarrow{i} & S_1 & & S_2 & \xrightarrow{c} & S_1 \\ g_0 \downarrow & & \downarrow g_1 & & \downarrow g_0 & & g_0 \downarrow & & \downarrow g_1 & & g_2 \downarrow & & \downarrow g_1 \\ V_0 & \xleftarrow{l_0} & V_1 & \xrightarrow{r_0} & V_0 & & V_0 & \xrightarrow{i} & V_1 & & V_2 & \xrightarrow{c} & V_1 \end{array} \quad (5)$$

*Notation.* The morphism  $g_2: S_2 \rightarrow V_2$  present in (5) is defined using the universal property of the pullback via the diagram below:

$$\begin{array}{ccccc} & & S_2 & & \\ & l_1 \swarrow & \vdots g_2 & \searrow r_1 & \\ S_1 & & & & S_1 \\ g_1 \downarrow & l_1 \swarrow & \downarrow & \searrow r_1 & \downarrow g_1 \\ V_1 & & V_2 & & V_1 \\ & r_0 \searrow & \downarrow & \swarrow l_0 & \\ & & V_0 & & \end{array} \quad (6)$$

*Remark.* Given an internal category  $\mathbf{V}$ , the *identity functor* consists of a pair of morphisms:

$$1_{V_0}: V_0 \longrightarrow V_0 \quad 1_{V_1}: V_1 \longrightarrow V_1$$

Given internal functors  $G: \mathbf{S} \rightarrow \mathbf{V}$  and  $H: \mathbf{V} \rightarrow \mathbf{U}$ , their *composite functor*  $HG: \mathbf{S} \rightarrow \mathbf{U}$  consists of a pair of morphisms:

$$h_0 g_0: S_0 \longrightarrow U_0 \quad h_1 g_1: S_1 \longrightarrow U_1$$

Composition of internal functors is both unital and associative, as it is induced by composition of morphisms in  $\mathcal{E}$ .

**Definition 3.** Let  $\text{Cat}(\mathcal{E})$  be the category whose objects are internal categories and whose morphisms are internal functors.

**Example 5.** The category of sets and functions  $\mathbf{Set}$  has pullbacks, thus we obtain the familiar example  $\mathbf{Cat} = \text{Cat}(\mathbf{Set})$  of small categories and functors between them.

**Example 6.** The category  $\mathbf{Cat}$  has pullbacks, so we obtain the category  $\mathbf{DbI} = \text{Cat}(\mathbf{Cat})$  of double categories and (double) functors between them.

*Remark.* The category  $\text{Cat}(\mathcal{E})$  has all pullbacks. Given internal functors  $G: \mathbf{S} \rightarrow \mathbf{V}$  and  $H: \mathbf{U} \rightarrow \mathbf{V}$ , their pullback is the category  $\mathbf{S} \times_{\mathbf{V}} \mathbf{U}$  constructed from the pullbacks,

$$\begin{array}{ccc} & S_0 \times_{V_0} U_0 & \\ & \swarrow \quad \searrow & \\ S_0 & & U_0 \\ & \searrow \quad \swarrow & \\ & V_0 & \end{array} \quad \begin{array}{ccc} & S_1 \times_{V_1} U_1 & \\ & \swarrow \quad \searrow & \\ S_1 & & U_1 \\ & \searrow \quad \swarrow & \\ & V_1 & \end{array}$$

which define the object of objects and object of morphisms, respectively. The rest of the structure is defined using the universal property of the pullback. Therefore *internal double categories* may be defined as categories internal to  $\mathcal{E}' = \text{Cat}(\mathcal{E})$ .

**Example 7.** An (internal) *discrete opfibration* is an internal functor  $G: \mathbf{S} \rightarrow \mathbf{V}$  such that the following diagram is a pullback:

$$\begin{array}{ccc} & S_1 & \\ l_0 \swarrow & \downarrow & \searrow g_1 \\ S_0 & & V_1 \\ g_0 \searrow & & \swarrow l_0 \\ & V_0 & \end{array}$$

Note the identity functor is a discrete opfibration, and the composite of discrete opfibrations is a discrete opfibration using the Pullback Pasting Lemma.

**Definition 4.** Let  $\mathbf{DOpf}(\mathcal{E})$  be the category whose objects are internal categories and whose morphisms are discrete opfibrations.

### 3 Internal Cofunctors

This section introduces the notion of an internal cofunctor and proves a useful representation of internal cofunctors as certain spans of internal functors. Since their introduction [1] there has been almost no work on cofunctors apart from the recent reference [2]. To avoid confusion, we explicitly note that a cofunctor is *not* a contravariant functor.

The idea of a cofunctor is to generalise discrete opfibrations, providing a way to lift certain morphisms while preserving identities and composition. Cofunctors are dual to functors in the sense that they *lift* morphisms in the opposite direction to the object assignment, while functors *push-forward* morphisms in the same direction. In the context of synchronisation, a cofunctor will later be interpreted as the **Put** component of an internal lens which lifts anchored view updates in the pullback  $\Lambda_1$  to source updates in  $S_1$ .

**Definition 5.** Let  $\mathbf{S}$  and  $\mathbf{V}$  be internal categories. An *internal cofunctor*  $\Lambda: \mathbf{S} \rightleftarrows \mathbf{V}$  consists of morphisms,

$$g_0: S_0 \longrightarrow V_0 \quad k_1: \Lambda_1 \longrightarrow S_1 \quad p_0: \Lambda_1 \longrightarrow S_0$$

where  $\Lambda_1 := S_0 \times_{V_0} V_1$  (defined below) along with the pullbacks,

$$\begin{array}{ccc} & \Lambda_1 & \\ l_0 \swarrow & \downarrow & \searrow \bar{g}_1 \\ S_0 & & V_1 \\ g_0 \searrow & & \swarrow l_0 \\ & V_0 & \end{array} \quad \begin{array}{ccc} & \Lambda_2 & \\ l_1 \swarrow & \downarrow & \searrow \bar{g}_2 \\ \Lambda_1 & & V_2 \\ \bar{g}_1 \searrow & & \swarrow l_1 \\ & V_1 & \end{array} \quad (7)$$

and satisfying the following commutative diagrams:

$$\begin{array}{ccccc} \Lambda_1 & \xrightarrow{p_0} & S_0 & & S_0 & \xleftarrow{l_0} & \Lambda_1 & \xrightarrow{p_0} & S_0 & & S_0 & \xrightarrow{i} & \Lambda_1 & & \Lambda_2 & \xrightarrow{c} & \Lambda_1 \\ \bar{g}_1 \downarrow & & \downarrow g_0 & & 1_{S_0} \downarrow & & \downarrow k_1 & & \downarrow 1_{S_0} & & 1_{S_0} \downarrow & & \downarrow k_1 & & k_2 \downarrow & & \downarrow k_1 \\ V_1 & \xrightarrow{r_0} & V_0 & & S_0 & \xleftarrow{l_0} & S_1 & \xrightarrow{r_0} & S_0 & & S_0 & \xrightarrow{i} & S_1 & & S_2 & \xrightarrow{c} & S_1 \end{array} \quad (8)$$

*Remark.* The pullback projections in (7) will play different roles which prompt different notational conventions. The projection  $l_0$  should be understood as the domain map for an internal category with object of morphisms  $\Lambda_1$  which will be defined in Proposition 1. The projection  $\bar{g}_1$  should be understood as morphism assignment for a discrete opfibration  $\bar{G}$  which will be defined in Theorem 2. The projections  $l_1$  and  $\bar{g}_2$  may be understood similarly.

*Notation.* The commutative diagrams (8) include morphisms defined using the universal property of the pullback via the diagrams below:

$$\begin{array}{cccc}
\begin{array}{c} S_0 \\ \downarrow i \\ \Lambda_1 \\ \downarrow \bar{g}_1 \\ V_1 \\ \downarrow i \\ V_0 \end{array} & \begin{array}{c} \Lambda_2 \\ \downarrow c \\ \Lambda_1 \\ \downarrow \bar{g}_1 \\ V_1 \\ \downarrow c \\ V_0 \end{array} & \begin{array}{c} \Lambda_2 \\ \downarrow p_1 \\ \Lambda_1 \\ \downarrow \bar{g}_1 \\ V_1 \\ \downarrow r_1 \\ V_0 \end{array} & \begin{array}{c} \Lambda_2 \\ \downarrow k_2 \\ S_2 \\ \downarrow r_1 \\ S_1 \\ \downarrow k_1 \\ S_0 \end{array} \\
\downarrow l_0 & \downarrow l_0 & \downarrow l_0 & \downarrow l_0 \\
S_0 & S_0 & S_0 & S_1 \\
\downarrow g_0 & \downarrow g_0 & \downarrow g_0 & \downarrow r_0 \\
V_0 & V_0 & V_0 & S_0
\end{array}
\tag{9}$$

*Remark.* Strictly speaking, the morphism  $p_0: \Lambda_1 \rightarrow S_0$  is not required for the definition of a cofunctor. Instead the two commutative diagrams in (8) which contain it may be replaced with the commutative diagram:

$$\begin{array}{ccc}
\Lambda_1 & \xrightarrow{k_1} & S_1 & \xrightarrow{r_0} & S_0 \\
\bar{g}_1 \downarrow & & & & \downarrow g_0 \\
V_1 & \xrightarrow{r_0} & & & V_0
\end{array}
\tag{10}$$

**Example 8.** An internal cofunctor with  $k_1: \Lambda_1 \cong S_1$  is a discrete opfibration.

**Example 9.** An internal cofunctor between monoids, as categories with one object, is a monoid homomorphism.

**Example 10.** An internal cofunctor with  $g_0 = 1_{S_0}$  is an identity-on-objects functor.

*Remark.* Given an internal category  $\mathbf{V}$ , the *identity cofunctor* consists of morphisms:

$$1_{V_0}: V_0 \longrightarrow V_0 \quad 1_{V_1}: V_1 \longrightarrow V_1 \quad r_0: V_1 \longrightarrow V_0$$

Given internal cofunctors  $\Lambda: \mathbf{S} \rightrightarrows \mathbf{V}$  and  $\Omega: \mathbf{V} \rightrightarrows \mathbf{U}$ , consisting of triples  $(g_0, k_1, p_0)$  and  $(h_0, j_1, q_0)$  respectively, their *composite cofunctor*  $\Lambda \odot \Omega: \mathbf{S} \rightarrow \mathbf{U}$  consists of the morphism,

$$h_0 g_0: S_0 \longrightarrow U_0$$

together with the pullback  $(\Lambda \odot \Omega)_1 := S_0 \times_{U_0} U_1$  and the morphisms,

$$k_1 \langle l_0, j(g_0 \times 1_{U_1}) \rangle: (\Lambda \odot \Omega)_1 \longrightarrow S_1 \quad p_0 \langle l_0, j(g_0 \times 1_{U_1}) \rangle: (\Lambda \odot \Omega)_1 \longrightarrow S_1 \tag{11}$$

where the universal morphisms are defined via the following commutative diagram:

$$\begin{array}{ccccc}
(\Lambda \odot \Omega)_1 & \xrightarrow{g_0 \times 1_{U_1}} & \Omega_1 & \xrightarrow{\bar{h}_1} & U_1 \\
\downarrow \langle l_0, j_1(g_0 \times 1_{U_1}) \rangle & \lrcorner & \downarrow j_1 & & \downarrow l_0 \\
\Lambda_1 & \xrightarrow{\bar{g}_1} & V_1 & & \\
\downarrow k_1 & & \downarrow l_0 & & \\
S_1 & & & & \\
\downarrow l_0 & & \downarrow l_0 & & \\
S_0 & \xrightarrow{g_0} & V_0 & \xrightarrow{h_0} & U_0
\end{array}
\tag{12}$$

Composition of cofunctors is both unital and associative, however we omit the diagram-chasing required for the proof.

**Definition 6.** Let  $\text{Cof}(\mathcal{E})$  be the category whose objects are internal categories and whose morphisms are internal cofunctors.

**Proposition 1.** *If  $\Lambda: \mathbf{S} \rightarrow \mathbf{V}$  is an internal cofunctor, then there exists an internal category  $\mathbf{\Lambda}$  with object of objects  $S_0$  and object of morphisms  $\Lambda_1$ , together with domain map  $l_0: \Lambda_1 \rightarrow S_0$ , codomain map  $p_0: \Lambda_1 \rightarrow S_0$ , identity map  $i: S_0 \rightarrow \Lambda_1$ , and composition map  $c: \Lambda_2 \rightarrow \Lambda_1$ .*

*Proof.* We give a partial proof and show the first pair of diagrams in (3) are satisfied. Using the relevant diagrams from Definition 1 and Definition 5 we have the following commutative diagram:

$$\begin{array}{ccccc}
 & & S_0 & & \\
 & & \downarrow i & \nearrow i & \\
 & & \Lambda_1 & \xrightarrow{k_1} & S_1 \\
 & \swarrow 1_{S_0} & & & \searrow r_0 \\
 S_0 & \xleftarrow{l_0} & \Lambda_1 & \xrightarrow{p_0} & S_0
 \end{array}$$

This shows that the identity map  $i: S_0 \rightarrow \Lambda_1$  is well-defined.

To show that  $\Lambda_2$  is well-defined as the pullback of the domain and codomain maps (left-most square below) we use the Pullback Pasting lemma, noting that the outer rectangles below are equal:

$$\begin{array}{ccccc}
 \Lambda_2 & \xrightarrow{p_1} & \Lambda_1 & \xrightarrow{\bar{g}_1} & V_1 \\
 l_1 \downarrow & & l_0 \downarrow & \lrcorner & \downarrow l_0 \\
 \Lambda_1 & \xrightarrow{p_0} & S_0 & \xrightarrow{g_0} & V_0
 \end{array}
 =
 \begin{array}{ccccc}
 \Lambda_2 & \xrightarrow{\bar{g}_2} & V_2 & \xrightarrow{r_1} & V_1 \\
 l_1 \downarrow & \lrcorner & l_1 \downarrow & \lrcorner & \downarrow l_0 \\
 \Lambda_1 & \xrightarrow{\bar{g}_1} & V_1 & \xrightarrow{r_0} & V_0
 \end{array}$$

Again using the relevant diagrams from Definition 1 and Definition 5 we have the following commutative diagram:

$$\begin{array}{ccccc}
 \Lambda_1 & \xleftarrow{l_1} & \Lambda_2 & \xrightarrow{p_1} & \Lambda_1 \\
 \downarrow l_0 & & \downarrow c & \nearrow k_2 & \downarrow p_0 \\
 & & & S_2 & \xrightarrow{r_1} & S_1 & \xleftarrow{k_1} & \Lambda_1 \\
 & & & \downarrow c & & \downarrow r_0 & & \\
 & & & S_1 & \xrightarrow{r_0} & S_0 & \xrightarrow{=} & S_0 \\
 & \swarrow l_0 & & \nearrow k_1 & & & & \\
 S_0 & \xleftarrow{l_0} & \Lambda_1 & \xrightarrow{p_0} & S_0
 \end{array}$$

This shows that the composition map  $c: \Lambda_2 \rightarrow \Lambda_1$  is well-defined.  $\square$

*Remark.* Proposition 1 may be understood as showing that a cofunctor induces a category whose objects are source states and whose morphisms are anchored view updates. The internal category  $\mathbf{\Lambda}$  is shown in Theorem 2 to mediate between the source and the view, and reduces the complexity of Definition 5 to a simple statement concerning internal categories and functors.

**Theorem 2.** *If  $\Lambda: \mathbf{S} \rightarrow \mathbf{V}$  is an internal cofunctor, then there is a discrete opfibration  $\bar{G}: \mathbf{\Lambda} \rightarrow \mathbf{V}$  consisting of the morphisms,*

$$g_0: S_0 \rightarrow V_0 \qquad \bar{g}_1: \Lambda_1 \rightarrow V_1$$

and an identity-on-objects functor  $K: \mathbf{\Lambda} \rightarrow \mathbf{S}$  consisting of morphisms:

$$1_{S_0}: S_0 \longrightarrow S_0 \quad k_1: \Lambda_1 \longrightarrow S_1$$

Thus every cofunctor  $\Lambda: \mathbf{S} \rightrightarrows \mathbf{V}$  may be represented as a span of functors:

$$\begin{array}{ccc} & \mathbf{\Lambda} & \\ K \swarrow & & \searrow \bar{G} \\ \mathbf{S} & & \mathbf{V} \end{array}$$

*Proof.* To show that  $\bar{G}: \mathbf{\Lambda} \rightarrow \mathbf{V}$  is a well-defined discrete opfibration, we note from (7), (8), and (9) that the following diagrams commute:

$$\begin{array}{ccccc} S_0 & \xleftarrow{l_0} & \Lambda_1 & \xrightarrow{p_0} & S_0 & & S_0 & \xrightarrow{i} & \Lambda_1 & & \Lambda_2 & \xrightarrow{c} & \Lambda_1 \\ g_0 \downarrow & & \downarrow \bar{g}_1 & & \downarrow g_0 & & g_0 \downarrow & & \downarrow \bar{g}_1 & & \bar{g}_2 \downarrow & & \downarrow \bar{g}_1 \\ V_0 & \xleftarrow{l_0} & V_1 & \xrightarrow{r_0} & V_0 & & V_0 & \xrightarrow{i} & V_1 & & V_2 & \xrightarrow{c} & V_1 \end{array}$$

To show that  $K: \mathbf{\Lambda} \rightarrow \mathbf{S}$  is a well-defined identity-on-objects functor, we again note from (7), (8), and (9) that the following diagrams commute:

$$\begin{array}{ccccc} S_0 & \xleftarrow{l_0} & \Lambda_1 & \xrightarrow{p_0} & S_0 & & S_0 & \xrightarrow{i} & \Lambda_1 & & \Lambda_2 & \xrightarrow{c} & \Lambda_1 \\ 1_{S_0} \downarrow & & \downarrow k_1 & & \downarrow 1_{S_0} & & 1_{S_0} \downarrow & & \downarrow k_1 & & k_2 \downarrow & & \downarrow k_1 \\ S_0 & \xleftarrow{l_0} & S_1 & \xrightarrow{r_0} & S_0 & & S_0 & \xrightarrow{i} & S_1 & & S_2 & \xrightarrow{c} & S_1 \end{array}$$

Thus every cofunctor may be represented as a span of functors, with right-leg a discrete opfibration, and left-leg an identity-on-objects functor.  $\square$

## 4 Internal Lenses

In this section we define an internal lens to consist of an internal Get functor and an internal Put cofunctor satisfying a simple axiom akin to the Put-Get law. An immediate corollary of Theorem 2 then is that every internal lens may be understood as a particular commuting triangle (15) of internal functors. We also construct a category whose objects are internal categories and whose morphisms are internal lenses. The section concludes with a unification of discrete opfibrations, state-based lenses, c-lenses, and d-lenses in this internal framework, based upon results in [5].

**Definition 7.** An *internal lens*  $(G, \Lambda): \mathbf{S} \rightrightarrows \mathbf{V}$  consists of an internal functor  $G: \mathbf{S} \rightarrow \mathbf{V}$  comprised of morphisms,

$$g_0: S_0 \longrightarrow V_0 \quad g_1: S_1 \longrightarrow V_1$$

and an internal cofunctor  $\Lambda: \mathbf{S} \rightrightarrows \mathbf{V}$  comprised of morphisms,

$$g_0: S_0 \longrightarrow V_0 \quad k_1: \Lambda_1 \longrightarrow S_1 \quad p_0: \Lambda_1 \longrightarrow S_0$$

such that the following diagram commutes:

$$\begin{array}{ccc} & \Lambda_1 & \\ k_1 \swarrow & & \searrow 1_{\Lambda_1} \\ S_1 & \xrightarrow{\langle l_0, g_1 \rangle} & \Lambda_1 \end{array} \quad (13)$$

*Remark.* Alternatively, the commutative diagram (13) for an internal lens may be replaced with the requirement that the following diagram commutes:

$$\begin{array}{ccc}
 & \Lambda_1 & \\
 k_1 \swarrow & & \searrow \bar{g}_1 \\
 S_1 & \xrightarrow{g_1} & V_1
 \end{array} \tag{14}$$

In either case, this axiom for an internal lens ensures that the functor and cofunctor parts interact as expected. Explicitly it states that *lifting* a morphism by the cofunctor then *pushing-forward* by the functor should return the original morphism.

**Corollary.** *Every lens  $(G, \Lambda): \mathbf{S} \rightleftharpoons \mathbf{V}$  may be represented as a commuting triangle of functors,*

$$\begin{array}{ccc}
 & \Lambda & \\
 K \swarrow & & \searrow \bar{G} \\
 \mathbf{S} & \xrightarrow{G} & \mathbf{V}
 \end{array} \tag{15}$$

where  $\bar{G}: \Lambda \rightarrow \mathbf{V}$  is a discrete opfibration, and  $K: \Lambda \rightarrow \mathbf{S}$  is an identity-on-objects functor.

**Corollary.** *Given a pair of lenses  $(G, \Lambda): \mathbf{S} \rightleftharpoons \mathbf{V}$  and  $(H, \Omega): \mathbf{V} \rightleftharpoons \mathbf{U}$ , their composite lens may be computed via the composition of the respective functor and cofunctor parts, and has a simple representation as a commuting triangle of functors:*

$$\begin{array}{ccccc}
 & & \Lambda \times_{\mathbf{V}} \Omega & & \\
 & & \swarrow & \searrow & \\
 & \Lambda & & & \Omega \\
 K \swarrow & & \searrow \bar{G} & & \swarrow J & \searrow \bar{H} \\
 \mathbf{S} & \xrightarrow{G} & \mathbf{V} & \xrightarrow{H} & \mathbf{U}
 \end{array} \tag{16}$$

**Definition 8.** Let  $\text{Lens}(\mathcal{E})$  be the category whose objects are internal categories and whose morphisms are internal lenses. Composition of lenses is determined by composition of the corresponding functor and cofunctor parts.

**Example 11.** Every discrete opfibration is both an internal functor and an internal cofunctor, hence also an internal lens. Therefore  $\text{DOPf}(\mathcal{E})$  is a subcategory of  $\text{Lens}(\mathcal{E})$ .

**Example 12.** If  $\mathcal{E} = \mathbf{Set}$ , then the category  $\text{Lens}(\mathbf{Set})$  is the category of d-lenses [7]. The *Get* of a d-lens is given by the functor  $G: \mathbf{S} \rightarrow \mathbf{V}$ , while the *Put* of a d-lens is given by the cofunctor  $\Lambda: \mathbf{S} \rightrightarrows \mathbf{V}$ .

In particular, the function  $k_1: \Lambda_1 \rightarrow S_1$  takes each pair  $(s, \alpha: gs \rightarrow v) \in \Lambda_1$  to an arrow  $k(s, \alpha): s \rightarrow p(s, \alpha) \in \mathbf{S}$ , as illustrated in the diagram below.

$$\begin{array}{ccc}
 \mathbf{S} & & s \xrightarrow{k(s, \alpha)} p(s, \alpha) \\
 G \downarrow \uparrow \Lambda & & \vdots \quad \quad \quad \vdots \\
 \mathbf{V} & & gs \xrightarrow{\alpha} v
 \end{array} \tag{17}$$

The *Put-Get* law is satisfied by (13), which corresponds in the above diagram to the morphism  $k(s, \alpha)$  being a genuine lift of  $\alpha: gs \rightarrow v$  with respect to the functor acting on morphisms. The *Get-Put* and *Put-Put* laws are satisfied as  $K: \Lambda \rightarrow \mathbf{S}$  is a functor, which by definition respects identities and composition.

**Example 13.** Every state-based lens consisting of **Get** function  $g: S \rightarrow V$  and **Put** function  $p: S \times V \rightarrow S$  induces a lens in  $\mathbf{Lens}(\mathbf{Set})$ .

Let  $\mathbf{S}$  and  $\mathbf{V}$  be the small codiscrete categories induced by the sets  $S$  and  $V$ , respectively, and let  $G: \mathbf{S} \rightarrow \mathbf{V}$  the canonical functor,

$$\begin{array}{ccccc} S & \xleftarrow{\pi_0} & S \times S & \xrightarrow{\pi_1} & S \\ g \downarrow & & \downarrow g \times g & & \downarrow g \\ V & \xleftarrow{\pi_0} & V \times V & \xrightarrow{\pi_1} & V \end{array}$$

induced by the **Get** function. Let  $\mathbf{A}$  be the category with domain and codomain maps described by the span:

$$\begin{array}{ccc} & S \times V & \\ \pi_0 \swarrow & & \searrow p \\ S & & S \end{array}$$

The category  $\mathbf{A}$  is well-defined by the lens laws. The functor  $\bar{G}: \mathbf{A} \rightarrow \mathbf{V}$  is induced using the **Put-Get** law,

$$\begin{array}{ccccc} S & \xleftarrow{\pi_0} & S \times V & \xrightarrow{p} & S \\ g \downarrow & & \downarrow g \times 1_V & & \downarrow g \\ V & \xleftarrow{\pi_0} & V \times V & \xrightarrow{\pi_1} & V \end{array}$$

while the functor  $K: \mathbf{A} \rightarrow \mathbf{S}$  is induced for free:

$$\begin{array}{ccccc} S & \xleftarrow{\pi_0} & S \times V & \xrightarrow{p} & S \\ 1_S \downarrow & & \downarrow \langle \pi_0, p \rangle & & \downarrow 1_S \\ S & \xleftarrow{\pi_0} & S \times S & \xrightarrow{\pi_1} & S \end{array}$$

This example may be instantiated internal to any category  $\mathcal{E}$  with finite limits.

**Example 14.** Given a pair of state-based lenses  $(g, p): S \rightleftharpoons V$  and  $(h, q): V \rightleftharpoons U$ , their composite is a lens whose **Get** function is simply given by  $hg: S \rightarrow U$  and whose **Put** function may be computed from the formula (11):

$$p\langle \pi_0, q(g \times 1_U) \rangle: S \times U \longrightarrow S$$

**Example 15.** Every c-lens (also known as a split opfibration) consisting of a **Get** functor  $G: \mathbf{S} \rightarrow \mathbf{V}$  and **Put** functor  $P: (G \downarrow \mathbf{V}) \rightarrow \mathbf{S}$  induces a lens in  $\mathbf{Lens}(\mathbf{Cat})$ .

Let  $\mathbb{V}$  be the double category of squares, whose category of objects is  $\mathbf{V}$  and whose category of morphisms is the arrow category  $\Phi(\mathbf{V})$ , together with domain and codomain functors  $L, R: \Phi(\mathbf{V}) \rightarrow \mathbf{V}$  given by,

$$\begin{array}{ccccc} V_1 & \xleftarrow{\sigma} & V_{11} & \xrightarrow{\tau} & V_1 \\ l_0 \downarrow & & \downarrow \lambda & & \downarrow l_0 \\ V_0 & \xleftarrow{l_0} & V_1 & \xrightarrow{r_0} & V_0 \end{array} \quad \begin{array}{ccccc} V_1 & \xleftarrow{\sigma} & V_{11} & \xrightarrow{\tau} & V_1 \\ r_0 \downarrow & & \downarrow \rho & & \downarrow r_0 \\ V_0 & \xleftarrow{l_0} & V_1 & \xrightarrow{r_0} & V_0 \end{array}$$

using the same notation from the diagram in Example 4; define  $\mathbb{S}$  similarly. Construct the functor  $\Phi(G): \Phi(\mathbf{S}) \rightarrow \Phi(\mathbf{V})$  between the arrow categories,

$$\begin{array}{ccccc} S_1 & \xleftarrow{\sigma} & S_{11} & \xrightarrow{\tau} & S_1 \\ g_1 \downarrow & & \downarrow g_2 \times g_2 & & \downarrow g_1 \\ V_1 & \xleftarrow{\sigma} & V_{11} & \xrightarrow{\tau} & V_1 \end{array}$$

induced by the `Get` functor, which forms a canonical double functor  $G: \mathbb{S} \rightarrow \mathbb{V}$ . Let  $\mathbb{A}$  be the double category with domain and codomain functors described by the span:

$$\begin{array}{ccc} & (G \downarrow \mathbf{V}) & \\ L \swarrow & & \searrow P \\ \mathbf{S} & & \mathbf{S} \end{array}$$

Note that the comma category  $(G \downarrow \mathbf{V})$  may be defined as the pullback,

$$\begin{array}{ccc} & & R \\ & \text{---} & \text{---} \\ (G \downarrow \mathbf{V}) & \longrightarrow & \Phi(\mathbf{V}) \xrightarrow{R} \mathbf{V} \\ L \downarrow & \lrcorner & \downarrow L \\ \mathbf{S} & \xrightarrow{G} & \mathbf{V} \end{array}$$

where  $L: (G \downarrow \mathbf{V}) \rightarrow \mathbf{S}$  and  $R: (G \downarrow \mathbf{V}) \rightarrow \mathbf{V}$  are the usual comma category projections. The double category  $\mathbb{A}$  is well-defined by the c-lens laws, and we may show with further reasoning that there exist unique double functors  $K: \mathbb{A} \rightarrow \mathbb{S}$  and  $\overline{G}: \mathbb{A} \rightarrow \mathbb{V}$ .

## 5 Conclusion and Future Work

In this paper it was shown that lenses may be defined internal to any category  $\mathcal{E}$  with pullbacks, providing a significantly generalised yet minimal framework to understand the notion of synchronisation between systems. It was demonstrated that the enigmatic `Put` of a lens may be understood as a cofunctor, which has a simple description as a span of a discrete opfibration and an identity-on-objects functor. The surprising characterisation of a lens as a functor/cofunctor pair both promotes the prevailing attitude of lenses as morphisms between categories, and yields a straightforward definition for composition in the category  $\text{Lens}(\mathcal{E})$ , which fits within an elegant diagram of forgetful functors.

$$\begin{array}{ccccc} & & & \text{Cof}(\mathcal{E}) & \\ & & & \nearrow & \\ \text{DOpf}(\mathcal{E}) & \longrightarrow & \text{Lens}(\mathcal{E}) & & \mathcal{E} \\ & & & \searrow & \\ & & & \text{Cat}(\mathcal{E}) & \end{array}$$

The success of internal lenses in unifying the known examples of state-based lenses, c-lenses, and d-lenses promotes the effectiveness of this perspective to be used in applications such as programming, databases, and Model-Driven Engineering, and also anticipates many future mathematical developments. Current work in progress indicates  $\text{Lens}(\mathcal{E})$  may be enhanced to a 2-category through incorporating natural transformations between lenses, while consideration of spans in  $\text{Lens}(\mathcal{E})$  leads towards a clarified understanding of symmetric lenses; both ideas which have been shown to be important in applications and the literature [6, 10]. Future work also intends to investigate examples of lenses internal to diverse range of categories, as well as taking steps towards a theory of lenses between enriched categories.

---

## References

- [1] Marcelo Aguiar. *Internal Categories and Quantum Groups*. PhD thesis, Cornell University, August 1997. URL <http://pi.math.cornell.edu/~maguiar/thesis2.pdf>.
- [2] Danel Ahman and Tarmo Uustalu. Taking updates seriously. In *Proceedings of the 6<sup>th</sup> International Workshop on Bidirectional Transformations*, volume 1827 of *CEUR Workshop Proceedings*, pages 59–73, 2017. URL <http://ceur-ws.org/Vol-1827/paper11.pdf>.
- [3] F. Bancilhon and N. Spyratos. Update semantics of relational views. *ACM Transactions on Database Systems*, 6(4):557–575, December 1981. DOI: [10.1145/319628.319634](https://doi.org/10.1145/319628.319634).
- [4] Francis Borceux. *Handbook of Categorical Algebra*, volume 1 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, 1994. DOI: [10.1017/CBO9780511525858](https://doi.org/10.1017/CBO9780511525858).
- [5] Bryce Clarke. Characterising asymmetric lenses using internal categories. Master’s thesis, Macquarie University, October 2018. URL <http://hdl.handle.net/1959.14/1268984>.
- [6] Zinovy Diskin. Compositionality of update propagation: Laxed putput. In *Proceedings of the 6<sup>th</sup> International Workshop on Bidirectional Transformations*, volume 1827 of *CEUR Workshop Proceedings*, pages 74–89, 2017. URL <http://ceur-ws.org/Vol-1827/paper12.pdf>.
- [7] Zinovy Diskin, Yingfei Xiong, and Krzysztof Czarnecki. From state- to delta-based bidirectional model transformations: the asymmetric case. *Journal of Object Technology*, 10:6:1–25, 2011. DOI: [10.5381/jot.2011.10.1.a6](https://doi.org/10.5381/jot.2011.10.1.a6).
- [8] J. Nathan Foster, Michael B. Greenwald, Jonathan T. Moore, Benjamin C. Pierce, and Alan Schmitt. Combinators for bidirectional tree transformations: A linguistic approach to the view-update problem. *ACM Transactions on Programming Languages and Systems*, 29(3), May 2007. DOI: [10.1145/1232420.1232424](https://doi.org/10.1145/1232420.1232424).
- [9] Jeremy Gibbons and Michael Johnson. Relating algebraic and coalgebraic descriptions of lenses. In *Proceedings of the 1<sup>st</sup> International Workshop on Bidirectional Transformations*, volume 49 of *Electronic Communications of the EASST*, March 2012. DOI: [10.14279/tuj.eceasst.49.726](https://doi.org/10.14279/tuj.eceasst.49.726).
- [10] Martin Hofmann, Benjamin Pierce, and Daniel Wagner. Symmetric lenses. *SIGPLAN Not.*, 46(1), January 2011. DOI: [10.1145/1925844.1926428](https://doi.org/10.1145/1925844.1926428).
- [11] Michael Johnson and Robert Rosebrugh. Delta lenses and opfibrations. *Electronic Communications of the EASST*, 57, 2013. DOI: [10.14279/tuj.eceasst.57.875](https://doi.org/10.14279/tuj.eceasst.57.875).
- [12] Michael Johnson and Robert Rosebrugh. Unifying set-based, delta-based and edit-based lenses. In *Proceedings of the 5<sup>th</sup> International Workshop on Bidirectional Transformations*, volume 1571 of *CEUR Workshop Proceedings*, pages 1–13, April 2016. URL [http://ceur-ws.org/Vol-1571/paper\\_13.pdf](http://ceur-ws.org/Vol-1571/paper_13.pdf).
- [13] Michael Johnson, Robert Rosebrugh, and Richard Wood. Algebras and update strategies. *Journal of Universal Computer Science*, 16(5):729–748, March 2010. DOI: [10.3217/jucs-016-05-0729](https://doi.org/10.3217/jucs-016-05-0729).
- [14] Michael Johnson, Robert Rosebrugh, and Richard Wood. Lenses, fibrations and universal translations. *Mathematical Structures in Computer Science*, 22(1):25–42, 2012. DOI: [10.1017/S0960129511000442](https://doi.org/10.1017/S0960129511000442).
- [15] Peter T Johnstone. *Sketches of an Elephant: A Topos Theory Compendium*, volume 1. Oxford University Press, 2002.

- 
- [16] Saunders Mac Lane. *Categories for the Working Mathematician*, volume 5 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 2 edition, 1998. ISBN 978-0-387-98403-2. DOI: [10.1007/978-1-4757-4721-8](https://doi.org/10.1007/978-1-4757-4721-8).
- [17] Ross Street. Fibrations and Yoneda's lemma in a 2-category. In *Category Seminar*, Lecture Notes in Mathematics, pages 104–133. Springer, 1974. DOI: [10.1007/BFb0063102](https://doi.org/10.1007/BFb0063102).