# Monoidal Indeterminates and Categories of Possible Worlds☆

C. Hermida, R. D. Tennent*

*School of Computing, Queen's University, Kingston, Canada K7L 3N6*

**Abstract**

Given any symmetric monoidal category $\boldsymbol{C}$, a small symmetric monoidal category $\boldsymbol{\Sigma}$ and a strong monoidal functor $j\colon \boldsymbol{\Sigma} \longrightarrow \boldsymbol{C}$, we construct $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$, the *polynomial* category with a system of (freely adjoined) monoidal indeterminates $\boldsymbol{x}\colon I \longrightarrow j(w)$, natural in $w \in \boldsymbol{\Sigma}$. As a special case, we construct the free co-affine category (symmetric monoidal category with initial unit) on a given small symmetric monoidal category. We then exhibit all the known categories of "possible worlds" used to treat languages that allow for dynamic creation of "new" variables, locations, or names as instances of this construction and explicate their associated universality properties. As an application of the resulting characterisation of $\boldsymbol{O}(\boldsymbol{W})$, Oles's category of possible worlds, we present an $\boldsymbol{O}(\boldsymbol{W})$-indexed Lawvere theory of many-sorted storage, generalizing the single-sorted one introduced by J. Power, and we describe explicitly an associated monad of (typed) block algebras for local storage.

*Key words:* indeterminates, symmetric monoidal categories, possible-world semantics, universality, indexed Lawvere theory

## Contents



☆A preliminary version of this work appears in the Proceedings of the 25th Conference on Mathematical Foundations of Programming Semantics, University of Oxford (April 3–7, 2009), Electronic Notes in Theoretical Computer Science, vol. 249, no. 8, pp. 39–60. The research was supported by a Discovery grant from the Natural Sciences and Engineering Research Council of Canada.

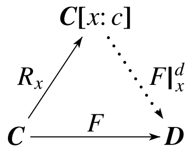
*Corresponding author.

*Email addresses:* `chermida@cs.queensu.ca` (C. Hermida), `rdt@cs.queensu.ca` (R. D. Tennent)

## 1. Introduction

The concept of a *polynomial algebra* $R[x]$, constructed from an algebra $R$ by freely adjoining an *indeterminate* element $x$, is familiar from algebra. Similarly, Lambek and Scott [1986, Part I, Section 5] show how to construct a cartesian (or cartesian closed) polynomial category $\boldsymbol{C}[x{:}\, c]$ from a base cartesian (closed) category $\boldsymbol{C}$ by freely adjoining an indeterminate arrow $x{:}\, 1 \longrightarrow c$.

The polynomial algebra $R[x]$ is the "most general" such extension of $R$. Similarly, the polynomial category $\boldsymbol{C}[x{:}\, c]$ is the most general cartesian (closed) extension of $\boldsymbol{C}$ containing indeterminate $x$. These properties follow from *universality* results. For example, consider the embedding $R_x{:}\, \boldsymbol{C} \longrightarrow \boldsymbol{C}[x]$ of $\boldsymbol{C}$ into $\boldsymbol{C}[x{:}\, c]$, any cartesian (closed) functor $F{:}\, \boldsymbol{C} \longrightarrow \boldsymbol{D}$, and any $d{:}\, 1 \longrightarrow F(c)$ in $\boldsymbol{D}$; then there exists a *unique* cartesian (closed) functor $F|^d_x$ from $\boldsymbol{C}[x{:}\, c]$ to $\boldsymbol{D}$ such that $(F|^d_x)(x) = d$ and $F|^d_x \cdot R_x = F$:




In this work, we develop comparable technology for *symmetric monoidal* categories [Mac Lane, 1971]. Given a symmetric monoidal category $\boldsymbol{C}$, a small symmetric monoidal category $\boldsymbol{\Sigma}$, and a strong monoidal functor $j{:}\, \boldsymbol{\Sigma} \longrightarrow \boldsymbol{C}$, we show how to construct $\boldsymbol{C}[\boldsymbol{x}{:}\, j\boldsymbol{\Sigma}]$, the symmetric monoidal polynomial category that results from freely adjoining, for every object $j(w)$ for $w \in \boldsymbol{\Sigma}$, indeterminates $x_{j(w)}{:}\, I \longrightarrow w$ satisfying a naturality constraint with respect to the arrows of $\boldsymbol{\Sigma}$. When $\boldsymbol{\Sigma}$ is the symmetric monoidal category freely generated by some set of $\boldsymbol{C}$ objects[1] the indeterminates are completely "free," as in the examples described above. Suitable choices of $\boldsymbol{\Sigma}$ and $j$ allow us to treat the cartesian situation as a special case (§2.9).

The *existence* of monoidal polynomial categories follows easily from the correspondence between deductive systems and (monoidal) categories studied by Lambek [1968, 1969, 1972], as well as the work by Kelly et al. [1989] on categorical universal algebra. From Lambek's work one can obtain *syntactic* descriptions of monoidal polynomial categories (as types and terms of suitable theories). But such a description does not allow one to recognize actual *semantic* instances of the construction, such as the categories of "possible worlds" that will be our primary focus in §4, nor to recognize additional structure and properties preserved by the construction, as shown in §3.

We believe our *semantic* construction has many applications. As our leading examples, we consider the categories of possible worlds that have been used in the semantics of imperative programming languages. Reynolds [1981b], Oles [1982, 1985, 1997] and O'Hearn and Tennent [1992] show how block-structured storage management in Algol-like languages [O'Hearn and Tennent, 1997] may be explicated using a semantics based on functor categories $\boldsymbol{W} \Rightarrow \boldsymbol{S}$, where $\boldsymbol{W}$ is a suitable category of "worlds" characterizing local aspects of storage structure, and $\boldsymbol{S}$ is a conventional semantic category of sets or domains. Every programming-language type $\theta$ is interpreted as a functor $[\![\theta]\!]{:}\, \boldsymbol{W} \longrightarrow \boldsymbol{S}$ and every programming-language term-in-context $\pi \vdash X{:}\, \theta$ is interpreted as a natural transformation $[\![\pi \vdash X{:}\, \theta]\!]{:}\, [\![\pi]\!] \stackrel{\cdot}{\longrightarrow} [\![\theta]\!]$.

Oles gives two presentations of his category of worlds and shows that they are equivalent. Reynolds presents what *seems* to be a different category of worlds; however, it has recently been shown [Hermida and Tennent, 2007] that, under reasonable closedness assumptions, it is in fact equivalent to Oles's category.

The functor-category framework has also been exploited to analyze noninterference in Reynolds's specification logic [Reynolds, 1981a,c; Tennent, 1990; O'Hearn, 1990; O'Hearn and Tennent, 1993], block expressions in Algol-like languages [Tennent, 1985], and passivity in a variant of Reynolds's Syntactic Control of Interference [Reynolds, 1978; O'Hearn et al., 1999]. These applications used a related but significantly different category of worlds, due to Tennent.

Several authors [Moggi, 1990; Pitts and Stark, 1993; Sieber, 1994; Stark, 1996; Fiore et al., 2002] have used *finite sets* (of locally available "locations" or "names") as worlds, with injections as the morphisms. In particular, Power [2006] uses this category to present an indexed Lawvere theory for single-sorted storage.

[1] i.e., the symmetric monoidal category consisting of all tensorings of the objects, with the morphisms being the relevant structural isomorphisms of $\boldsymbol{C}$.

What is noteworthy about all of this work is that the categories of worlds involved have been developed in *ad hoc* fashion and their properties have not been well understood. We show here that all of these categories of worlds are instances of our monoidal polynomial construction and have *universality* properties. In particular, we exploit universality to present a *many-sorted* version of Power's indexed Lawvere theory.

The construction of $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$ and its key properties, such as universality, and an important special case ($\boldsymbol{\Sigma}$ generated by a single object) are presented in §2. Our applications are discussed in §4. Some additional properties of $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$ not directly relevant to our applications are treated in §3; this may be skipped by readers more interested in the applications.

## 2. Monoidal Polynomial Categories

### 2.1. *The Categories $\boldsymbol{C}$ and $\boldsymbol{\Sigma}$*

Consider a symmetric monoidal category $\boldsymbol{C}$ with unit $I$ and structural isomorphisms

$$\lambda_x\colon I \otimes x \;\cong\; x$$

$$\rho_x\colon x \otimes I \;\cong\; x$$

$$\alpha_{x,y,z}\colon (x \otimes y) \otimes z \;\cong\; x \otimes (y \otimes z)$$

$$\sigma_{x,y}\colon x \otimes y \cong y \otimes x$$

subject to the usual coherence axioms [Mac Lane, 1971]. See [Kelly, 1974a] for explanations of additional monoidal-categorical concepts, such as *monoidal transformation* and *strong monoidal functor*, and [Joyal and Street, 1993] for detailed considerations of coherence issues.

We want to add *indeterminates* (generic global elements) to some objects of $\boldsymbol{C}$ subject to naturality constraints. To allow for different constraints, we parameterize our construction with an auxiliary *small* symmetric monoidal category $\boldsymbol{\Sigma}$ and a strong symmetric monoidal functor $j\colon \boldsymbol{\Sigma} \longrightarrow \boldsymbol{C}$, with structural isomorphisms $\delta_{v,w}\colon j(v) \otimes j(w) \longrightarrow j(v \otimes w)$ and $\gamma\colon I \longrightarrow j(I)$. Two boundary cases will be of particular interest:

(i) $\boldsymbol{\Sigma} = F(\star)$ the free symmetric monoidal category generated by one object and $j$ the canonical mapping picking out one object in $\boldsymbol{C}$; the only commutativity constraints are those imposed by structural isos.

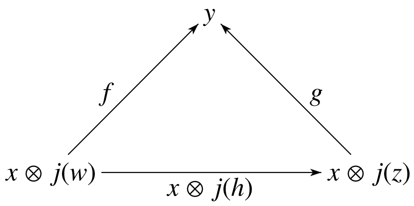
(ii) $\boldsymbol{\Sigma} = \boldsymbol{C}$, $j = \mathsf{id}$, when $\boldsymbol{C}$ is itself small; commutativity with *all* $\boldsymbol{C}$ morphisms will be required.

As explained by Lambek and Scott [1986], adding an indeterminate of type $\sigma \in \boldsymbol{C}$ to a cartesian category is achieved via the Kleisli category of the co-monad $- \times \sigma$; that is, the morphisms $h\colon a \longrightarrow b$ in $\boldsymbol{C}[x\colon \sigma]$ are morphisms $h\colon a \times \sigma \longrightarrow b$ in $\boldsymbol{C}$. Type-theoretically, this means that "terms" in the new theory $C[x\colon \sigma]$ are terms from the original theory $\boldsymbol{C}$ with an additional parameter $x\colon \sigma$.

If $\boldsymbol{C}$ is a *monoidal* category, $- \otimes \sigma$ does not in general support a co-monad structure, and so a more sophisticated construction is needed. But we continue to exploit the intuition that the new morphisms are to have their domains "expanded" by those objects $\sigma$ to which we are adding global elements. In this more general monoidal context, the morphisms naturally carry a higher-dimensional structure. This leads us to first set up a *bicategory* [Borceux, 1994a] $\boldsymbol{C}(\boldsymbol{x}\colon j\boldsymbol{\Sigma})$ whose morphisms are "parameterized" by additional objects and related by 2-cells. We can then obtain the desired *category* $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$ by taking equivalence classes of these parameterized morphisms. Readers who might not be interested in the details can turn directly to the description of the resulting category in §2.3.

### 2.2. *The Bicategory $\boldsymbol{C}(\boldsymbol{x}\colon j\boldsymbol{\Sigma})$*

- the *objects* are those of $\boldsymbol{C}$;
- for any $w$ an object in $\boldsymbol{\Sigma}$ and $f\colon x \otimes j(w) \longrightarrow y$ a morphism in $\boldsymbol{C}$, $(f, w)\colon x \longrightarrow y$ is a *morphism* in $\boldsymbol{C}(\boldsymbol{x}\colon j\boldsymbol{\Sigma})$;
- for any $\boldsymbol{C}$-morphisms $f\colon x \otimes j(w) \longrightarrow y$ and $g\colon x \otimes j(z) \longrightarrow y$, a *2-cell* $h\colon (f, w) \Rightarrow (g, z)$ is a morphism $h\colon w \longrightarrow z$ in $\boldsymbol{\Sigma}$ such that




– the *identity* for $x$ is $(\rho_x \cdot (x \otimes \gamma^{-1}),\ I)\colon x \longrightarrow x$.
– given morphisms $(f, w)\colon x \longrightarrow y$ and $(g, w')\colon y \longrightarrow z$ their *composite* is $(h, w \otimes w')\colon x \longrightarrow z$, where $h$ is defined as follows:

$$\begin{array}{c} x \otimes j(w \otimes w') \\ \big\downarrow {\scriptstyle x \otimes \delta^{-1}} \\ x \otimes j(w) \otimes j(w') \\ \big\downarrow {\scriptstyle \alpha^{-1}} \\ (x \otimes j(w)) \otimes j(w') \\ \big\downarrow {\scriptstyle f \otimes j(w')} \\ y \otimes j(w') \\ \big\downarrow {\scriptstyle g} \\ z \end{array}$$

– the *structural isomorphisms* are inherited from the monoidal structure of $\boldsymbol{C}$: given $(f, w)\colon x \longrightarrow y$, $(g, v)\colon y \longrightarrow z$ and $(h, u)\colon z \longrightarrow t$, define $\alpha_{(f,w),(g,v),(h,u)}$ to be $\alpha_{w,v,u} : (h, u) \cdot ((g, v) \cdot (f, w)) \Longrightarrow ((h, u) \cdot (g, v)) \cdot (f, w)$, and similarly for $\lambda$ and $\rho$.

### *2.3. The Category* $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$

To obtain a category from the bicategory, we will consider equivalence classes of morphisms so we can collapse the non-associative composition in the bicategory to a strictly associative composition, as needed for a proper category.

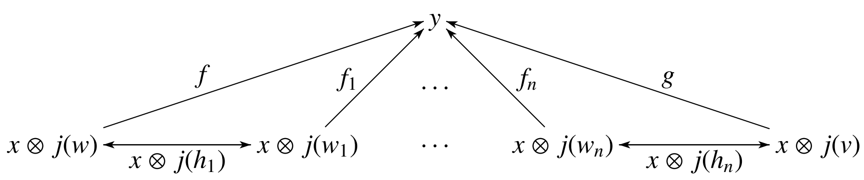
**Definition 2.1.** If $\boldsymbol{C}$ is any small category, an object $a \in \boldsymbol{C}$ is said to be *connected* to an object $b \in \boldsymbol{C}$ if it possible to get from $a$ to $b$ following a sequence of $\boldsymbol{C}$-arrows in either direction; i.e., there is an *undirected* ("zig-zag") path between $a$ and $b$. Connectedness is an equivalence relation on $\boldsymbol{C}$-objects and the equivalence classes are called the *connected components* of $\boldsymbol{C}$.

There is a functor $\pi_0$ from ***Cat*** to ***Set*** that maps any small category $\boldsymbol{C}$ to the set of its connected components and maps a functor $F\colon \boldsymbol{C} \longrightarrow \boldsymbol{D}$ to the function that maps a connected component $[a]$ of $\boldsymbol{C}$ to the connected component $[F(a)]$ of $\boldsymbol{D}$; this is independent of the choice of representative. The functor $\pi_0$ is left adjoint to the *discrete-category* functor from ***Set*** to ***Cat*** [Mac Lane, 1971, p. 88, Exercise 9], and so it preserves products. By applying it to the hom-categories of $\boldsymbol{C}(\boldsymbol{x}\colon j\boldsymbol{\Sigma})$, we collapse the structural-isomorphism 2-cells and composition becomes strictly associative and unitary.[2] This gives us our intended $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$:

$$\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}](x, y) \;=\; \pi_0(\boldsymbol{C}(\boldsymbol{x}\colon j\boldsymbol{\Sigma})(x, y)) \;\cong\; \coprod_{w \in |\boldsymbol{\Sigma}|} [\boldsymbol{C}(x \otimes j(w), y)]_{\simeq} \tag{1}$$

where $(f\colon x \otimes j(w) \longrightarrow y, w) \;\simeq\; (g\colon x \otimes j(v) \longrightarrow y, v)$ iff there is a undirected path of 2-cells between them in $\boldsymbol{C}(\boldsymbol{x}\colon j\boldsymbol{\Sigma})(x, y)$:

[2] In more detail: for any bicategory $\boldsymbol{B}$ there is an equivalent 2-category (Cat-enriched category) $\widehat{\boldsymbol{B}}$ with the same objects; i.e., for any objects $x, y \in \boldsymbol{B}$, there is an equivalence between $\boldsymbol{B}(x, y)$ and $\widehat{\boldsymbol{B}}(x, y)$. Since an equivalence is fully faithful and essentially surjective, $\boldsymbol{B}(x, y)$ and $\widehat{\boldsymbol{B}}(x, y)$ have essentially the "same" connected components; i.e., $\pi_0(\boldsymbol{B}(x, y))$ is isomorphic to $\pi_0(\widehat{\boldsymbol{B}}(x, y))$. So, given a bicategory $\boldsymbol{B}$, we obtain a category by applying $\pi_0$ to its hom-categories.



**Proposition 2.2.** *$\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$ has a symmetric monoidal structure.*

**Proof.** The tensor product of objects $x$ and $y$ is $x \otimes y$, as in $\boldsymbol{C}$, and the same is true for the unit $I$. The tensor product of morphisms $[f, w]\colon x \longrightarrow y$ and $[f', w']\colon x' \longrightarrow y'$ is the morphism $[g, w \otimes w']\colon x \otimes x' \longrightarrow y \otimes y'$ where $g$ is defined as follows:

$$\begin{array}{l} x \otimes x' \otimes j(w \otimes w') \\ \quad \downarrow \; x \otimes x' \otimes \delta \\ x \otimes x' \otimes j(w) \otimes j(w') \\ \quad \downarrow \; x \otimes \sigma_{x', j(w)} \otimes j(w') \\ x \otimes j(w) \otimes x' \otimes j(w') \\ \quad \downarrow \; f \otimes f' \\ y \otimes y' \end{array}$$

(omitting associativity isos). Verification that this action is functorial involves only functoriality of $\otimes$, naturality of $\sigma$, and the coherence conditions on $\sigma$. The structural isomorphisms are given by those in $\boldsymbol{C}$ suitably composed with $\rho$s to discard the unit parameter; e.g., associativity isomorphisms are of the form

$$[\alpha_{x,y,z} \cdot \rho_{(x \otimes y) \otimes z} \cdot ((x \otimes y) \otimes z) \otimes \gamma^{-1}\colon (x \otimes y) \otimes z \longrightarrow x \otimes (y \otimes z),\ I] \qquad \square$$

Note that a symmetry (or, more generally, a braiding) is needed to tensor morphisms as above.

*2.4. Raw morphisms*

There is a natural mapping of $\boldsymbol{C}$ into $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$ that takes $f\colon x \longrightarrow y$ into

$$[f \cdot \rho_x \cdot (x \otimes \gamma^{-1}), I]\colon x \longrightarrow y$$

As a consequence of the coherence axioms, $\rho_I = \lambda_I\colon I \times I \cong I$ [Joyal and Street, 1993, Prop. 1.1], and this mapping yields a functor $R_\Sigma\colon \boldsymbol{C} \longrightarrow \boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$.

A morphism $[f, w]\colon x \longrightarrow y$ with $w \cong I$ is termed *raw*. Raw morphisms yield a broad sub-category (i.e., with the same objects as the ambient category) of $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$, the essential image of $R_\Sigma$.

**Proposition 2.3.** *$R_\Sigma\colon \boldsymbol{C} \longrightarrow \boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$ is (strongly) symmetric monoidal; i.e., it preserves the structure up to coherent isomorphism.*

**Proof.** The coherent structural isomorphisms are the "raw" images of those in $\boldsymbol{C}$ under $R_\Sigma$ and functoriality ensures that the coherence axioms hold as well; this makes $R_\Sigma$ strongly symmetric monoidal. $\square$

**Remark 2.4.** With our construction, $R_\Sigma$ is in fact a *strict* monoidal functor.

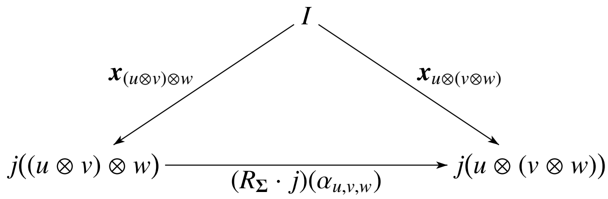
To clarify the presentation, we will write the raw images of $\alpha$, $\rho$, $\lambda$ and $\sigma$ *underlined*, so that, for example, $\underline{\alpha}$ will denote an associativity isomorphism in $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$.

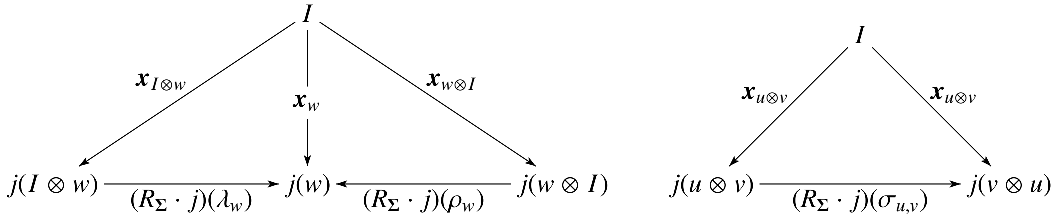
*2.5. Monoidal Indeterminates*

The most significant feature of $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$ is that it has, for every object $w$ in $\boldsymbol{\Sigma}$, a "global element" $\boldsymbol{x}_w = [\lambda_{j(w)}, w]\colon I \longrightarrow j(w)$. These morphisms will be termed *(constrained) monoidal elements*.

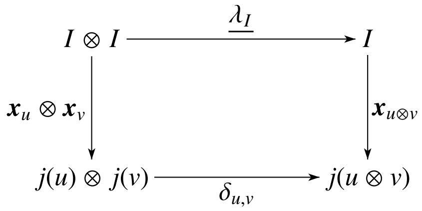
**Definition 2.5.** Given a strong monoidal functor $F\colon \boldsymbol{C} \longrightarrow \boldsymbol{D}$ and a small symmetric monoidal category $\boldsymbol{\Sigma}$ with a strong monoidal functor $j\colon \boldsymbol{\Sigma} \longrightarrow \boldsymbol{C}$, a ***system of monoidal elements for** $F$ **(constrained with respect to** $\boldsymbol{\Sigma}$**)*** is a monoidal transformation $\boldsymbol{d}\colon I^{\boldsymbol{D}}_{\boldsymbol{\Sigma}} \Longrightarrow F \cdot j$, where $I^{\boldsymbol{D}}_{\boldsymbol{\Sigma}}\colon \boldsymbol{\Sigma} \longrightarrow \boldsymbol{D}$ is the strong monoidal functor constantly $I$. When $\boldsymbol{\Sigma}$ is free on a set of objects, we talk simply of monoidal elements; these are *free* or unconstrained, as in our original scenario of polynomial algebras and categories in §1.

The $\boldsymbol{x}_w$ defined above form a system $\boldsymbol{x}\colon I^{\boldsymbol{C}[\boldsymbol{x}:j\boldsymbol{\Sigma}]}_{\boldsymbol{\Sigma}} \Longrightarrow R_{\boldsymbol{\Sigma}} \cdot j$ of monoidal elements for $R_{\boldsymbol{\Sigma}}$ constrained with respect to $\boldsymbol{\Sigma}$. Because $\boldsymbol{\Sigma}$ includes the relevant structural isomorphisms, naturality of $\boldsymbol{x}$ entails the following three commutativities:



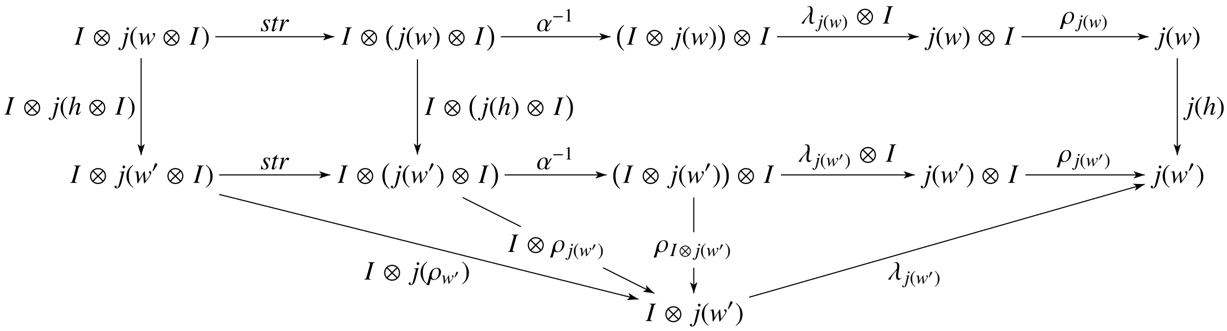




while the monoidal condition entails $\boldsymbol{x}_I = R_{\boldsymbol{\Sigma}} \cdot \gamma\colon I \longrightarrow j(I)$ and




**Proposition 2.6.** $\boldsymbol{x} = [\lambda_{j\_}, (\_)]\colon I^{\boldsymbol{C}[\boldsymbol{x}:j\boldsymbol{\Sigma}]}_{\boldsymbol{\Sigma}} \Longrightarrow R_{\boldsymbol{\Sigma}} \cdot j$ *is natural in* $\boldsymbol{\Sigma}$ *and monoidal.*

**Proof.** For naturality, consider $h\colon w \longrightarrow w'$ in $\boldsymbol{\Sigma}$ and the following diagram:



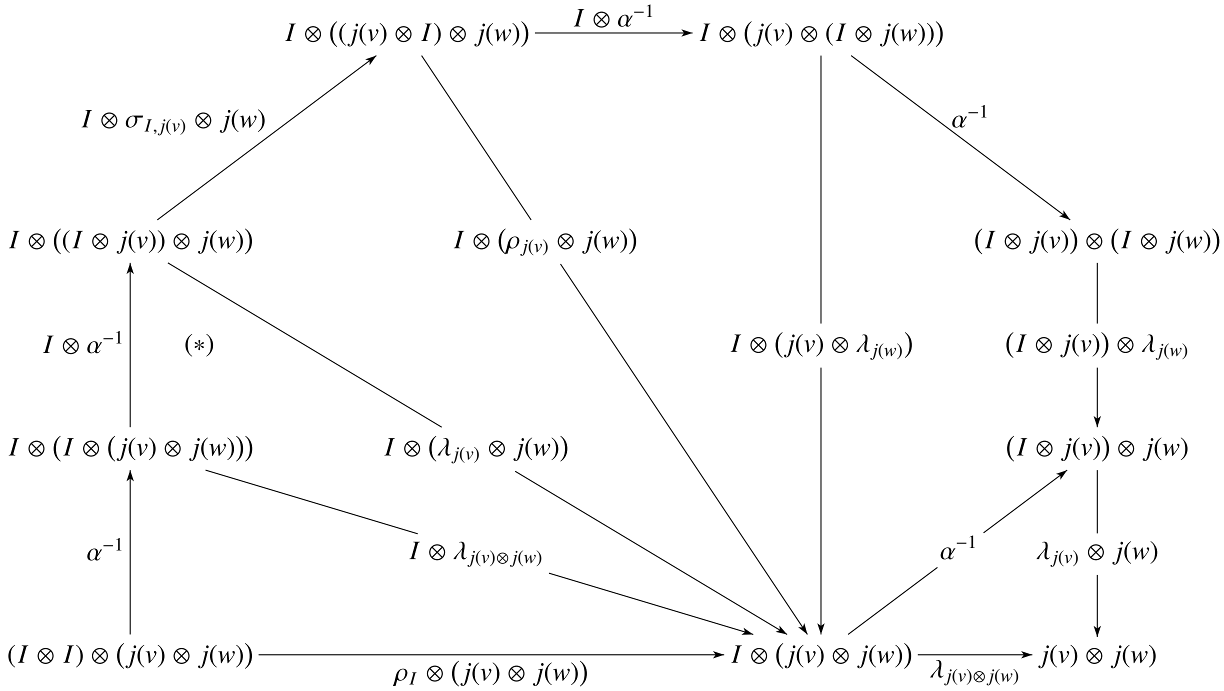



Figure 1: Monoidality Diagram

where *str* denotes the coherent structural isomorphism $I \otimes j(\_) \otimes \gamma^{-1} \cdot I \otimes \delta^{-1}$ associated with $j$. The middle-bottom triangle commutes by [Joyal and Street, 1993, Prop. 1.1] and the rest by naturality of the $\lambda$, $\rho$ and $\alpha$. Hence $\rho_{w'} \cdot (h \otimes I): w \otimes I \longrightarrow w'$ is a 2-cell in $\boldsymbol{C}(\boldsymbol{x}\colon j\boldsymbol{\Sigma})$ from $R_{\Sigma}(h) \cdot \lambda_w$ to $\lambda_{w'}$, and therefore $R_{\Sigma}(h) \cdot \boldsymbol{x}_w = \boldsymbol{x}_{w'}$ in $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$.

For monoidality, $\boldsymbol{x}_I = [\lambda_{j(I)}, I] = \gamma = \gamma \cdot \rho_I \cdot I \otimes \gamma^{-1}$, because $\lambda_I = \rho_I$ by [Joyal and Street, 1993, Prop. 1.1], and in the diagram in Figure 1 on Page 8, the $(*)$ triangle and the rightmost-bottom one commute by [Joyal and Street, 1993, Prop. 1.1]; the top triangle involving $\sigma$ commutes by [Joyal and Street, 1993, Prop. 2.1]; and the remaining ones commute by the coherence axiom relating $\alpha$, $\lambda$ and $\rho$, and naturality of $\alpha$. Using once again the coherence axiom relating $\alpha$, $\lambda$ and $\rho$ and naturality of $\lambda$ it may be concluded that $\lambda_{v,w}: I \otimes (v \otimes w) \longrightarrow v \otimes w$ realizes a 2-cell in $\boldsymbol{C}(\boldsymbol{x}\colon j\boldsymbol{\Sigma})$, which identifies $\delta_{v,w} \cdot (\boldsymbol{x}_v \otimes \boldsymbol{x}_w) \cdot R_{\Sigma}(\lambda_I^{-1})$ and $\boldsymbol{x}_{v\otimes w}$ in $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$. □

We will show below (Theorem 2.9) that $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$ is freely generated by this system of constrained monoidal elements. In other words, the $\boldsymbol{x}_{(\_)}$ form a *generic* such system; we call them ***monoidal indeterminates***.

**Definition 2.7.** For any object $y \in \boldsymbol{C}$, $e^w_y\colon [\mathsf{id}_{y\otimes j(w)}, w]\colon y \longrightarrow y \otimes j(w)$ is termed the ***expansion morphism at*** $y$ ***(with respect to*** $w$***)***.

The terminology will be justified in §4.1.1. Lemma 2.8 (part 4) generalizes the expansion-iso factorization of Oles [1982, 1997].

**Lemma 2.8 (Expansion–Raw Morphism Factorization).**

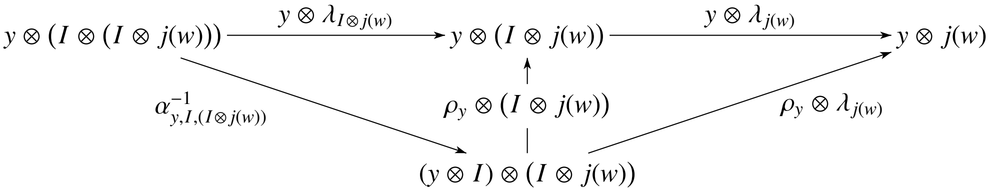
*(1)* Interdefinability of expansions and indeterminates*:*
$e^w_y = (y \otimes \boldsymbol{x}_w) \cdot \underline{\rho_y}^{-1}\colon y \longrightarrow y \otimes j(w)$ *and* $\boldsymbol{x}_w = \underline{\lambda_{j(w)}} \cdot e^w_I$*.*
*(2) The expansion morphisms* $e^w_y$ *are natural in* $y$*, and natural in* $w$ *with respect to* $\boldsymbol{\Sigma}$*-morphisms.*
*(3) Expansions compose:* $e^{v\otimes w}_y = \underline{\alpha_{y,j(v),j(w)}}^{-1} \cdot e^w_{y\otimes j(v)} \cdot e^v_y$*.*
*(4) Every* $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$ *map* $[f, w]\colon y \longrightarrow z$ *factors as an* expansion $e^w_y\colon y \longrightarrow y \otimes j(w)$*, followed by a* raw morphism $[f \cdot \rho_{y\otimes j(w)} \cdot (y \otimes j(w) \otimes \gamma^{-1}), I]\colon y \otimes j(w) \longrightarrow z$*. This factorization is unique in the following sense: if* $[f, w] = R_{\Sigma}(g) \cdot e^{w'}_y$ *for some* $g\colon y \otimes j(w') \longrightarrow z$*, then* $[f, w] = [g, w']$*.*

**Proof.**

(1) By [Joyal and Street, 1993, Prop.2.1], $\rho_I \cdot \sigma_{I,I} = \lambda_I$; hence, because $\lambda_I = \rho_I$, we have that $\sigma_{I,I} = \mathsf{id}_{I\otimes I}$. Because $j$ is strongly symmetric monoidal, $\sigma_{j(I),\,j(I)} = \mathsf{id}_{j(I)\otimes j(I)}$. Therefore, the composite $(y \otimes \boldsymbol{x}_w) \cdot \underline{\rho_y}^{-1}$ reduces to the morphism

$$\Big[(\rho_y \otimes \lambda_{j(w)}) \cdot \alpha^{-1}_{y,I,I\otimes j(w)} \cdot \Big(y \otimes \big(\gamma^{-1} \otimes (\gamma^{-1} \otimes j(w))\big)\Big) \cdot (y \otimes jI \otimes \delta^{-1}) \cdot (y \otimes \delta^{-1}),\ I \otimes (I \otimes w)\Big]$$

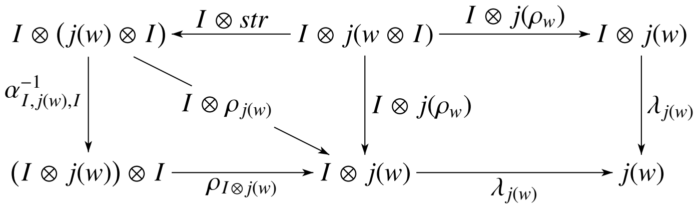
in $\boldsymbol{C}(\boldsymbol{x}\colon j\boldsymbol{\Sigma})$. Consider the following diagram:




The left triangle commutes by the coherence axioms for $\alpha$, $\lambda$ and $\rho$, and the right triangle by functoriality of $\otimes$. We conclude that $\lambda_w \cdot \lambda_{I\otimes w}\colon I \otimes (I \otimes w) \longrightarrow w$ is a 2-cell in $\boldsymbol{C}(\boldsymbol{x}\colon j\boldsymbol{\Sigma})$, which yields the identity

$$e^w_y = (y \otimes \boldsymbol{x}_w) \cdot \underline{\rho_y}^{-1}\colon y \longrightarrow y \otimes w$$

The diagram




in which the bottom left triangle commutes by [Joyal and Street, 1993, Prop.1.1], and the top left one by coherence for $j$, shows that $\rho_w\colon w \otimes I \longrightarrow w$ realizes a 2-cell in $\boldsymbol{C}(\boldsymbol{x}\colon j\boldsymbol{\Sigma})$, which identifies $\underline{\lambda_w} e^w_I$ with $\boldsymbol{x}_w$ in $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$.

(2) Follows from (1), because $\boldsymbol{x}_w$ is natural with respect to morphisms in $\boldsymbol{\Sigma}$.

(3) The morphism $v \otimes \rho_w\colon v \otimes (w \otimes I) \longrightarrow v \otimes w$ realizes a 2-cell in $\boldsymbol{C}(\boldsymbol{x}\colon j\boldsymbol{\Sigma})$ which identifies $(\underline{\alpha_{y,v,w}}^{-1} \cdot e^w_{y\otimes j(v)}) \cdot e^v_y$ with $e^{v\otimes w}_y$ in $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$.

(4) The morphism $\rho_w\colon w \otimes I \longrightarrow w$ realizes a 2-cell in $\boldsymbol{C}(\boldsymbol{x}\colon j\boldsymbol{\Sigma})$ from

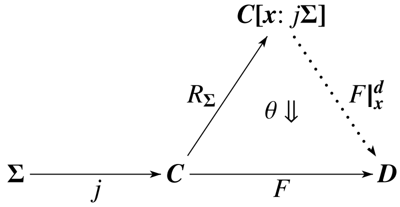
$$\Big(f \cdot \rho_{y\otimes j(w)} \cdot \big((y \otimes j(w)) \otimes \gamma^{-1}\big) \cdot \alpha^{-1}_{y,j(w),I} \cdot (y \otimes \delta^{-1}),\ w \otimes I\Big)$$

to $(f, w)$ which yields the required identification. Given another expansion-raw factorization, e.g., $[f, w] = R_{\boldsymbol{\Sigma}}(g) \cdot e^{w'}_y$, we have to argue by induction by the length of the undirected path of 2-cells in $\boldsymbol{C}(\boldsymbol{x}\colon j\boldsymbol{\Sigma})(y, z)$ realizing the identification. Clearly, it suffices to consider the case of a basic path of length one (the inductive step): assume a morphism $h\colon w' \otimes I \longrightarrow w$ such that $f(y \otimes j(h)) = g \cdot \rho_{y\otimes j(w')} \cdot str$. Then, setting $\overline{h} = h \cdot \rho^{-1}_{w'}$, we have $f \cdot (y \otimes j(\overline{h})) = g$, hence $[f, w] = [g, w']$. □

*2.6. Universality of* $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$

We will now show that $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$ is a *free* construction: given any other system of global elements, there is an essentially unique "interpretation" mapping the indeterminates $\boldsymbol{x}$ to those values.

**Theorem 2.9 (Universality).** *Given a symmetric monoidal category* $\boldsymbol{D}$*, a strong symmetric monoidal functor* $F\colon \boldsymbol{C} \longrightarrow \boldsymbol{D}$*, and a system* $\boldsymbol{d}\colon I^{\boldsymbol{D}}_{\boldsymbol{\Sigma}} \Longrightarrow F \cdot j$ *of monoidal elements for* $F$ *constrained with respect to* $\boldsymbol{\Sigma}$*, there exists a strong symmetric monoidal functor* $F|^{\boldsymbol{d}}_{\boldsymbol{x}}\colon \boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}] \longrightarrow \boldsymbol{D}$ *unique up to a unique isomorphism and a monoidal iso 2-cell* $\theta\colon (F|^{\boldsymbol{d}}_{\boldsymbol{x}} \cdot R_{\boldsymbol{\Sigma}}) \Longrightarrow F$ *such that* $F|^{\boldsymbol{d}}_{\boldsymbol{x}} \cdot \boldsymbol{x} \cong \boldsymbol{d}$*:*




The isomorphism $F|^{\boldsymbol{d}}_{\boldsymbol{x}} \cdot \boldsymbol{x} \cong \boldsymbol{d}$ here is a convenient abbreviation for the following commutativity of monoidal transformations on functors from $\boldsymbol{\Sigma}$ to $\boldsymbol{D}$:

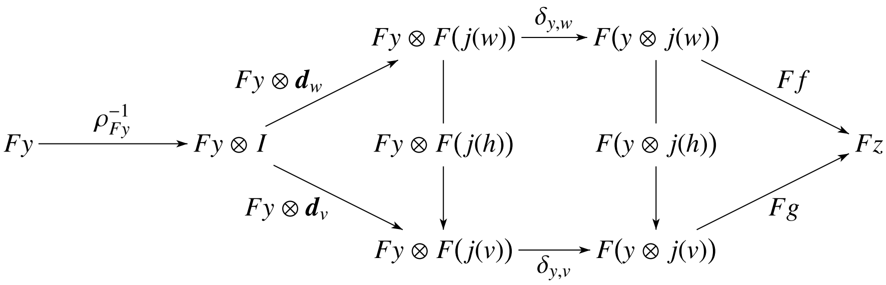
$$\begin{array}{ccc} F|^{\boldsymbol{d}}_{\boldsymbol{x}} \cdot I_{\Sigma}^{C[x: j\Sigma]} & \xrightarrow{\quad\gamma\quad} & I_{\Sigma}^{\boldsymbol{D}} \\ {\scriptstyle F|^{\boldsymbol{d}}_{\boldsymbol{x}} \cdot \boldsymbol{x}}\downarrow & & \downarrow{\scriptstyle \boldsymbol{d}} \\ F|^{\boldsymbol{d}}_{\boldsymbol{x}} \cdot R_{\Sigma} \cdot J_{\Sigma}^{\boldsymbol{C}} & \xrightarrow[\quad\theta j\quad]{} & F \cdot J_{\Sigma}^{\boldsymbol{C}} \end{array}$$

where $\gamma$ is the structural isomorphism associated with $F|^{\boldsymbol{d}}_{\boldsymbol{x}}$.

**Proof.** It is clear that the action of $F|^{\boldsymbol{d}}_{\boldsymbol{x}}$ on objects should be $(F|^{\boldsymbol{d}}_{\boldsymbol{x}})(y) = F(y)$. For a morphism $[f, w]: y \longrightarrow z$ factored as $[f \cdot \rho_{y \otimes jw} \cdot ((y \otimes jw) \otimes \gamma^{-1}), I] \cdot e^w_y$, with $e^w_y = (y \otimes x_w)\underline{\rho_y}^{-1}$ by Lemma 2.8.(1), we get $(F|^{\boldsymbol{d}}_{\boldsymbol{x}})[f, w] =$

$$Fy \xrightarrow{\rho_{Fy}^{-1}} Fy \otimes I \xrightarrow{Fy \otimes \boldsymbol{d}_w} Fy \otimes F(j(w)) \;\cong\; F(y \otimes j(w)) \xrightarrow{Ff} Fz$$

In order to show the value of $(F|^{\boldsymbol{d}}_{\boldsymbol{x}})[f, w]$ is independent of the choice of representative, consider $(f, w) \simeq (g, v)$ with $g: y \otimes j(v) \longrightarrow z$ via $h: w \longrightarrow v$ in $\boldsymbol{\Sigma}$ and the diagram



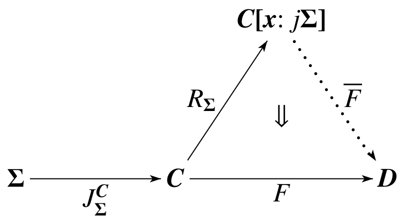

where the leftmost triangle commutes by naturality of $\boldsymbol{d}$, while the rightmost one commutes because $F$ is a functor.

Functoriality of $(F|^{\boldsymbol{d}}_{\boldsymbol{x}})$ follows from the coherence axioms for the structural isomorphisms associated with $F$ and the monoidality of the transformation $\boldsymbol{d}$. $F|^{\boldsymbol{d}}_{\boldsymbol{x}}$ is strong monoidal, with the same structural isomorphisms as $F$. We can take $\theta = \mathrm{id}$, but the general statement requires a general $\theta$ as we want $F|^{\boldsymbol{d}}_{\boldsymbol{x}}$ characterized only up to strong monoidal isomorphism. The coherence conditions on $F$ imply that $(F|^{\boldsymbol{d}}_{\boldsymbol{x}})(\boldsymbol{x}_w) = (F|^{\boldsymbol{d}}_{\boldsymbol{x}})[\lambda_{j(w)}, w] = d_w \cdot \gamma_w$ (with $\theta = \mathrm{id}$), and $(F|^{\boldsymbol{d}}_{\boldsymbol{x}})[f \cdot \rho_y \cdot (y \otimes \gamma^{-1}), I] = F(f)$ for any morphism $f: y \longrightarrow z$ in $\boldsymbol{C}$. □

**Remark 2.10.** There is a 2-dimensional aspect to the universality of $\boldsymbol{C}[\boldsymbol{x}: j\boldsymbol{\Sigma}]$: given strong symmetric monoidal functors $F, G: \boldsymbol{C} \longrightarrow \boldsymbol{D}$ with systems of monoidal elements $\boldsymbol{d}: I^{\boldsymbol{D}}_{\Sigma} \Longrightarrow F \cdot j$ and $\mathbf{e}: I^{\boldsymbol{D}}_{\Sigma} \Longrightarrow G \cdot j$, there is one-to-one correspondence between monoidal transformations $\overline{\beta}: (F|^{\boldsymbol{d}}_{\boldsymbol{x}}) \Longrightarrow (G|^{\mathbf{e}}_{\boldsymbol{x}})$ and monoidal transformations $\beta: F \Rightarrow G$ such that $\beta j \cdot \boldsymbol{d} = \mathbf{e}$. This aspect is illustrated in Example 4.8.

The following special case will prove useful in §4.5 in characterizing the "states" functor in the semantics of imperative languages:

**Corollary 2.11.** *If the unit* 0 *of the symmetric monoidal category* $\boldsymbol{D}$ *is an initial object, there is an essentially unique strong monoidal functor* $\overline{F}: \boldsymbol{C}[\boldsymbol{x}: j\boldsymbol{\Sigma}] \longrightarrow \boldsymbol{D}$ *extending a strong monoidal functor* $F: \boldsymbol{C} \longrightarrow \boldsymbol{D}$*:*



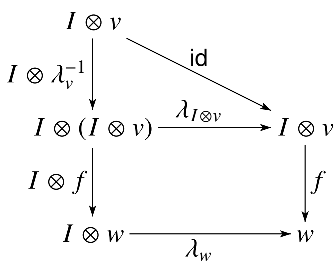

**Proof.** The unit 0 is initial, so there is a *unique* way to choose a global element $!_w\colon 0 \longrightarrow F(w)$ for any $w \in \Sigma$ (which is natural in $w$ with respect to $C$), and $\overline{F} = F\rceil^{!_w}_x$. □

*2.7. The Co-Affine Envelope of* $C$

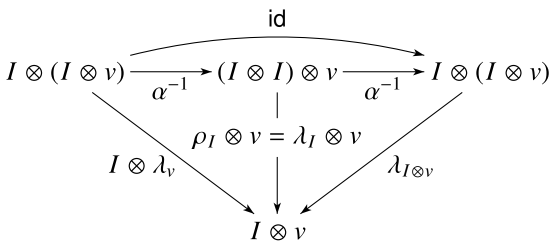
When $C$ is small, we can consider the important special case that $\Sigma = C$ and $j = \mathsf{id}$. The examples in §4 will be instances of $C[x\colon C]$ for suitable small categories $C$; to simplify the notation, we will use $\lceil C\rceil$ as an abbreviation for $C[x\colon C]$.

**Proposition 2.12.** *For any small symmetric monoidal* $C$*, the unit of* $\lceil C\rceil$ *is an initial object.*

**Proof.** For any object $w$ of $\lceil C\rceil$, we have a morphism $x_w = [\lambda_w, w]\colon I \longrightarrow w$. For any $[f\colon I \otimes v \longrightarrow w, v]$ in $\lceil C\rceil$, consider the following diagram:




The bottom part of the diagram commutes by naturality and the triangle commutes by the following:




where the left triangle commutes by the coherence axiom for $\alpha$, $\rho$ and $\lambda$, and the remaining two equalities are given in [Joyal and Street, 1993, Prop. 2.1]. We conclude that $f \cdot \lambda_v^{-1}\colon v \longrightarrow w$ is a 2-cell in $C(x\colon C)$, which identifies $x_w$ and $[f, v]$ in $\lceil C\rceil$. □

Alternatively, we can prove it as follows:

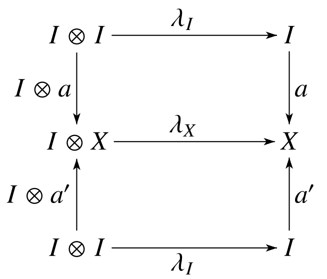
**Proof.** Recall that for a small category $D$, an initial object amounts to a limit of the identity functor $\mathsf{id}\colon D \longrightarrow D$, that is: a cone $\{\iota_d : I \longrightarrow d\}_{d\in|D|}$ such that $\iota_I = \mathsf{id}_I$. Using Lemma 2.8.(1) and monoidality of $x$, we conclude that our system of indeterminates is natural with respect to expansions: $e^w_v \cdot x_v = x_{v\otimes w}\colon I \longrightarrow v \otimes w$. Because they are natural with respect to all raw morphisms (by construction of $\lceil C\rceil$), the $\{x_v\colon I \longrightarrow v\}_{v\in|\lceil C\rceil|}$ form a cone, and by monoidality of $x$, $x_I = \mathsf{id}_I$. □

Combining Corollary 2.11 and Proposition 2.12, we conclude that the construction $C \longmapsto \lceil C\rceil$ provides the universal way of forcing the unit $I$ to be initial:

**Corollary 2.13.** *For* $C$ *a small symmetric monoidal category, functor* $R\colon C \longrightarrow \lceil C\rceil$ *is universal among strong symmetric monoidal functors into symmetric monoidal categories whose unit is an initial object.*

Symmetric monoidal categories with an initial unit are called *co-affine* in [Petrić, 2002]. Therefore the above corollary provides an explicit construction of the free co-affine category on a symmetric monoidal category, which we call the ***co-affine envelope*** of $\boldsymbol{C}$.

It is of interest to see what happens with existing distinct global elements $a\colon I \longrightarrow X$ and $a'\colon I \longrightarrow X$ in the transition from $\boldsymbol{C}$ to $\lceil \boldsymbol{C} \rceil$ via the functor $R\colon \boldsymbol{C} \longrightarrow \lceil \boldsymbol{C} \rceil$:




Naturality ensures that both parts of the diagram commute and therefore

$$R(a) = R(a') = \boldsymbol{x}_X$$

In contrast to the Lambek and Scott [1986] construction for cartesian categories, which only adds "unconstrained" global elements, our construction allows us to impose naturality constraints. We can thus enforce naturality of $\boldsymbol{x}\_$ and $\boldsymbol{x}_I = id_I$, which renders $I$ initial.

**Example 2.14 (Partial order on a monoid).** Given a commutative monoid $(M, \cdot, e)$, we can regard it as a *discrete symmetric monoidal category*. Its co-affine envelope $\lceil M \rceil$ has morphisms

$$\lceil M \rceil(m, n) = \{p \in M \mid m \cdot p = n\}$$

If $M$ is cancellative, i.e., $m \cdot (\_)$ is injective, for instance when $M$ is a group or a free monoid such as $\mathsf{Nat}$, $\lceil M \rceil$ is then a poset. Otherwise, we take the posetal reflection of $\lceil M \rceil$. Either way, we obtain the partial order which is conventionally associated with a monoid:

$$m \leq n \equiv \exists p \, . \, m \cdot p = n$$

Therefore, $\leq$ has the following universal characterisation: it is the least partial order on $M$ such that:

(i) $e$ is the least element
(ii) $\cdot$ is monotone

Notice that the definition of $\leq$ does not depend on the commutativity of $\cdot$. In case $\cdot$ is not commutative, we have to weaken (ii) in the above characterisation to the following: $m \cdot (\_)\colon M \longrightarrow M$ is monotone, for every $m \in M$.

□

*2.8. Indeterminate on a Single Object*

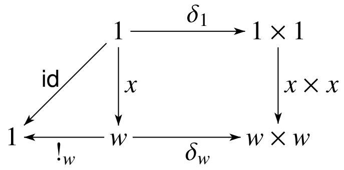
A special case of interest is the construction of the symmetric monoidal category generated by a category $\boldsymbol{C}$ and an indeterminate $x_w\colon I \longrightarrow w$ for a *single* object $w$. By tensoring such an indeterminate with itself and using the isomorphism $\lambda_I = \rho_I\colon I \otimes I \cong I$, one obtains indeterminates for all tensor powers $w^i$ of $w$, and more generally, all $i$-ary bracketings of $w$. We are led to consider $\boldsymbol{\Sigma}_\star$, the *free symmetric monoidal category on one generator*, and the strong symmetric monoidal functor $j_w\colon \boldsymbol{\Sigma}_\star \longrightarrow \boldsymbol{C}$, which takes $\star$ to $w$.

**Remark 2.15.** We recall that $\boldsymbol{\Sigma}_\star$ can be explicitly described as the category $\boldsymbol{F}_{\mathsf{bij}}$ of finite sets and bijections, see §4.6.

Given a symmetric monoidal category $\boldsymbol{D}$ and a strong symmetric monoidal functor $F\colon \boldsymbol{C} \longrightarrow \boldsymbol{D}$, a monoidal transformation $d\colon I^{\boldsymbol{D}}_{\boldsymbol{\Sigma}_\star} \Longrightarrow F \cdot j_w$ amounts precisely to an element $d_\star\colon I \longrightarrow Fw$. Therefore, $\boldsymbol{C}[\boldsymbol{x}\colon j_w \boldsymbol{\Sigma}_\star]$ is the *free symmetric monoidal category with an indeterminate* $x\colon I \longrightarrow w$.

*2.9. Monoidal Indeterminates in a Cartesian Setting*

When the monoidal structure on $\boldsymbol{C}$ is given by finite products so that $v \otimes w = v \times w$ and $I = 1$, each object $w$ carries a comonoid structure given by $!_w : w \longrightarrow 1$ and $\delta_w : w \longrightarrow w \times w$. Furthermore, each morphism in $\boldsymbol{C}$ is a comonoid morphism, by naturality of ! and $\delta$. In particular, a global element $x : 1 \longrightarrow w$ satisfies

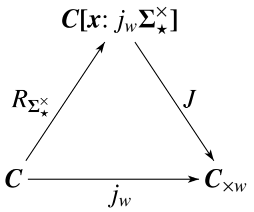



Therefore, if we want a monoidal indeterminate $x_w : 1 \longrightarrow w$ to be a cartesian one, we must enforce naturality with respect to $\boldsymbol{\Sigma}^{\times}_{\star}$, the *free symmetric monoidal category on one generator with a comonoid structure* $(\star, \delta_\star, !_\star)$. Equivalently, $\boldsymbol{\Sigma}^{\times}_{w}$ is the *free cartesian category on one generator*, since all tensor powers of $\star$ come equipped with natural comonoid structures, using repeatedly $\delta_\star$ and $!_\star$. Once again, we consider the strong symmetric monoidal functor $j_w : \boldsymbol{\Sigma}^{\times}_{\star} \longrightarrow \boldsymbol{C}$ which takes $(\star, \delta_\star, !_\star)$ to $(w, \delta_w, !_w)$. It is easy to see that $j_w$ is actually cartesian.

**Remark 2.16.** We recall that $\boldsymbol{\Sigma}^{\times}_{\star}$ can be explicitly described as $\boldsymbol{F}^{\mathsf{op}}$, the dual of the category of finite sets.

As we mentioned in our introduction, Lambek and Scott [1986, Part I, Section 5] show that $\boldsymbol{C}[x : 1 \longrightarrow w]$, the free cartesian category obtained from $\boldsymbol{C}$ by adjoining an indeterminate $x : 1 \longrightarrow w$, can be explicitly described by the Kleisli category of the comonad $(\_) \times w : \boldsymbol{C} \longrightarrow \boldsymbol{C}$, which we write $\boldsymbol{C}_{\times w}$, with associated functor $J_w : \boldsymbol{C} \longrightarrow \boldsymbol{C}_{\times w}$. Given a morphism $f : y \times w \longrightarrow z$ in $\boldsymbol{C}_{\times w}$, we write $J(f) = [f, \star]$ and interpret it as a morphism in $\boldsymbol{C}[\boldsymbol{x} : j_w \boldsymbol{\Sigma}^{\times}_{w}]$.

**Proposition 2.17.** *The assignment $f \longmapsto J(f)$ is an identity-on-objects isomorphism from $\boldsymbol{C}_{\times w}$ to $\boldsymbol{C}[\boldsymbol{x} : j_w \boldsymbol{\Sigma}^{\times}_{\star}]$ and the following diagram commutes:*




**Proof.** The isomorphism

$$\boldsymbol{C}(y \times w, z) \;\cong\; \Big( \coprod_{i \geq 0} \boldsymbol{C}(y \times w^i, z) \Big)_{\simeq}$$

induced by $J$ on homs is verified by setting up morphisms $\phi^i_\star : \star \longrightarrow \star^i$ which yield a 2-cell inducing the required identification in $\boldsymbol{C}[\boldsymbol{x} : j_w \boldsymbol{\Sigma}^{\times}_{\star}]$; the $\phi^i_\star$ are defined by induction on $i$:

$$\phi^0_\star = !_\star \qquad \phi^{i+1}_\star = (\delta^\star \times \star^{i-1}) \cdot \phi^i_\star$$

Functoriality of $J$ requires preservation of identities and composition, which is also achieved via $!_\star$ and $\delta_\star$ respectively. The identity $J \cdot j_w = R_{\boldsymbol{\Sigma}^{\times}_{\star}}$ requires identifying $\pi'_{y,w} : y \times w \longrightarrow w$ with $\pi'_{y,1} : y \times 1 \longrightarrow y$, via $!_\star : \star \longrightarrow 1$ (the same way in which $J$ preserves identities). □

## 3. Further Properties of $\boldsymbol{C}[\boldsymbol{x} : j\boldsymbol{\Sigma}]$

In this section, we describe additional properties of $\boldsymbol{C}[\boldsymbol{x} : j\boldsymbol{\Sigma}]$, with a view to the role this structure plays in categorical logic and semantics. Some readers might prefer to skip ahead to the applications in §4.

### 3.1. Closed Structure and Duals

**Proposition 3.1.** *If $\boldsymbol{C}$ is a* closed *symmetric monoidal category, so is $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$; furthermore, $R_\Sigma\colon \boldsymbol{C} \longrightarrow \boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$ preserves the closed structure.*

**Proof.** Given the formulation of the hom-sets of $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$ in equation (1), $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$ inherits closed structure from $\boldsymbol{C}$ via the isomorphism

$$\coprod_{w\in|\Sigma|} \boldsymbol{C}((x\otimes y)\otimes j(w), z) \;\cong\; \coprod_{w\in|\Sigma|} \boldsymbol{C}(x\otimes j(w), y\Rightarrow z)$$

which is compatible with the equivalence relation $\simeq$. It is then clear that $R_\Sigma$ preserves the closed structure. □

**Corollary 3.2.** *If $\boldsymbol{C}$ is* compact *closed (i.e., every object $c$ admits a dual $c^*$ such that $\boldsymbol{C}(x\otimes c, y) \cong \boldsymbol{C}(x, c^*\otimes y)$), so is $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$; furthermore, $R_\Sigma$ preserves duals.*

### 3.2. Traces

The notion of *trace* [Joyal et al., 1996; Hasegawa, 2004] in a monoidal category is also compatible with the addition of monoidal indeterminates.

**Proposition 3.3.** *If $\boldsymbol{C}$ admits a trace, so does $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$; furthermore, $R_\Sigma$ preserves traces.*

**Proof.** A trace function

$$Tr^u_{x,y}\colon \boldsymbol{C}(x\otimes u, y\otimes u) \longrightarrow \boldsymbol{C}(x,y)$$

for $\boldsymbol{C}$ is compatible with the equivalence $\simeq$ by dinaturality

$$\Big[\coprod_{w\in|\Sigma|} Tr^u_{x\otimes j(w),y}\Big]_\simeq : \Big[\coprod_{w\in|\Sigma|} \boldsymbol{C}((x\otimes j(w))\otimes u, y\otimes u)\Big]_\simeq \longrightarrow \Big[\coprod_{w\in|\Sigma|} \boldsymbol{C}(x\otimes j(w), y)\Big]_\simeq$$

and therefore induces a trace function on $\boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$, evidently preserved by $R_\Sigma$. □

## 4. Applications

### 4.1. Introduction

#### 4.1.1. The Oles Category of Possible Worlds

The following category is described in [Oles, 1982, 1997]. Let ***set*** be a small sub-category of the usual category of all sets and functions, interpreted as (products of) "data" (i.e., assignable) types. Note that procedures are not assignable in ALGOL-like languages [Reynolds, 1981b]. The objects of Oles's category are those of ***set***, interpreted as the sets of states *allowed* in each possible world, and morphisms from $X$ to $Y$ (termed "expansions") are pairs $f, Q$ such that

(i) $f$ is a function from $Y$ to $X$;

(ii) $Q$ is an equivalence relation on $Y$ with $Y/Q$ an object of ***set***; and

(iii) $X \xleftarrow{\;f\;} Y \xrightarrow{\;y\,\mapsto\,[y]_Q\;} Y/Q$ is a product diagram in ***set***.

Intuitively, $f$ extracts the small state embedded in a larger one, and $Q$ relates large states with identical "extensions." Note that the restriction of $f$ to any $Q$-equivalence class is bijective.

The identity morphism $\mathrm{id}_X$ on an object $X$ has as its two components: the identity function on $X$ and $\top_X$, the universally-true binary relation on $X$. The composition of morphisms $f, Q\colon X \longrightarrow Y$ and $g, R\colon Y \longrightarrow Z$ has as its two components: the functional composition of $f$ and $g$, and the equivalence relation on $Z$ that relates $z_0, z_1 \in Z$ just if they are $R$-related and $Q$ relates $g(z_0)$ and $g(z_1)$; in short, $R \cap g^{-1}(Q)$.

We will refer to this category as $\boldsymbol{O}(\boldsymbol{set})$. Oles gives another description which may be interpreted in any category $\boldsymbol{C}$ with finite products; see [O'Hearn and Tennent, 1995, Section 10]. So we have a *construction* $\boldsymbol{O}(\boldsymbol{C})$ that agrees with Oles's category when the ambient category $\boldsymbol{C}$ is ***set***.

*4.1.2. The Tennent Category of Possible Worlds*

To model noninterference in Reynolds's specification logic [Reynolds, 1981a,c; Tennent, 1990; O'Hearn and Tennent, 1993], the product condition on the $f$ component of morphisms $f, Q$ in Oles's category was weakened in [Tennent, 1990] to the requirement that it be *injective* on $Q$-equivalence classes (with the same definitions of identities and composites); we will refer to the resulting category as $\boldsymbol{T}(\boldsymbol{set})$.

*4.2. Universality of Tennent Categories*

We now apply our theory of monoidal indeterminates; we begin by characterizing $\boldsymbol{T}(\boldsymbol{set})$ as a polynomial category. The description may be re-formulated as follows. Recall that, for any function $f\colon X \longrightarrow Y$, $\mathsf{ker}(f)$, the *kernel* of $f$, is the binary relation $\{(x, x') \in X \times X \mid fx = fx'\}$.

**Proposition 4.1.** *Given sets X and Y, there is a one-to-one correspondence between the following sets of data:*

*(1) equivalence classes of pairs* $[m, W]_{\simeq}$ *where* $W$ *is an object of* **set**, $m\colon Y \rightarrowtail X \times W$ *is a monomorphism and* $(m, W) \simeq (n, V)$ *if* $\pi \cdot m = \pi \cdot n$ *and* $\mathsf{ker}(\pi' \cdot m) = \mathsf{ker}(\pi' \cdot n)$*, where* $\pi$ *and* $\pi'$ *denote the first and second projections of a product;*

*(2)* $\boldsymbol{T}(\boldsymbol{set})(X, Y)$*.*

**Proof.** From (1) to (2): Let $f\colon Y \longrightarrow X$ be the composite $Y \overset{m}{\rightarrowtail} X \times W \xrightarrow{\pi} X$ and $Q$ be the kernel of $Y \overset{m}{\rightarrowtail} X \times W \xrightarrow{\pi'} W$; i.e., $yQy'$ iff $\pi'(my) = \pi'(my')$. To show that $f$ is injective on each equivalence class, assume $yQy'$ and $f(y) = f(y')$; then $\pi(my) = \pi(my')$ and $\pi'(my) = \pi'(my')$ and so $my = my'$. But then $y = y'$ because $m$ is monic. Notice that, by construction, $f$ and $Q$ are independent of the choice of representative $(m, W)$.

From (2) to (1): Take $[m, W]_{\simeq}$, where $W$ is $Y/Q$ and $m\colon Y \longrightarrow X \times W$ maps $y$ to the pair $(fy, [y]_Q)$. To show $m$ is monic, assume $my = my'$; then $fy = fy'$ and $yQy'$, and so $y = y'$. □

**Corollary 4.2.** *The above correspondence restricts to one between* $\boldsymbol{O}(\boldsymbol{set})(X, Y)$ *and equivalence classes of pairs* $[i, W]_{\simeq}$ *where* $W$ *is an object of* **set** *and* $i\colon Y \cong X \times W$ *is an isomorphism.*

These correspondences are applicable to any category in which we can reason about "quotients of equivalence relations"; for instance, the argument can be carried out in any exact category[3]. We now give an equational characterization of the relation $\simeq$ in Proposition 4.1.

**Lemma 4.3.** *In any regular category*[3]*,*

*(i) given morphisms* $f\colon x \longrightarrow y$, $g\colon x \longrightarrow z$ *and a monomorphism* $m\colon z \rightarrowtail y$ *such that* $f = m \cdot g$*, we have that* $\mathsf{ker}(f) = \mathsf{ker}(g)$*;*

*(ii) given morphisms* $f\colon x \longrightarrow y$ *and* $g\colon x \longrightarrow z$*,* $\mathsf{ker}(f) = \mathsf{ker}(g)$ *iff there exists* $q\colon x \longrightarrow w$ *and monomorphisms* $m\colon w \rightarrowtail y$ *and* $n\colon w \rightarrowtail z$ *such that* $f = m \cdot q$ *and* $g = n \cdot q$*.*

**Proof.**

(i) Reasoning by elements, $\mathsf{ker}(g) \subseteq \mathsf{ker}(f)$. For the converse,

$$f \cdot x = f \cdot y \Longrightarrow m \cdot g \cdot x = m \cdot g \cdot y \Longrightarrow g \cdot x = g \cdot y$$

the last step justified by $m$ being a monomorphism.

(ii) Given $\mathsf{ker}(f) = \mathsf{ker}(g)$ take the (common) quotient of these kernels $q\colon x \longrightarrow w$. Both $f$ and $g$ factor through $q$ via monos $m\colon w \rightarrowtail y$ and $n\colon w \rightarrowtail z$. The converse follows from (i). □

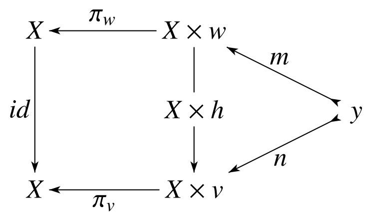
Let $\boldsymbol{set}_{\mathsf{mn}}$ be the sub-category of $\boldsymbol{set}$ consisting of the same objects but only monomorphisms. Finite products in $\boldsymbol{set}$ endow $\boldsymbol{set}_{\mathsf{mn}}$ with a symmetric monoidal structure, so we can apply our construction of constrained monoidal indeterminates to it. Note that the mono condition rules out maps into 1, and therefore $\boldsymbol{set}_{\mathsf{mn}}^{\mathsf{op}}$ does *not* have an initial unit.

[3] Regular categories support reasoning in the $(\top, \wedge, \exists)$ fragment of first-order logic with equality; for example, the theory of equivalence relations is regular. An exact category is a regular category in which there is one-to-one correspondence between equivalence relations and quotients (for every object). See [Borceux, 1994b, Chapter 2].

**Theorem 4.4.** $\boldsymbol{T}(\boldsymbol{set}) \equiv \lceil \boldsymbol{set}^{\mathsf{op}}_{\mathsf{mn}} \rceil$

where, as mentioned at the end of §2.9, $\lceil \boldsymbol{set}^{\mathsf{op}}_{\mathsf{mn}} \rceil$ is $\boldsymbol{set}^{\mathsf{op}}_{\mathsf{mn}}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$ when $\boldsymbol{\Sigma}$ is $\boldsymbol{set}^{\mathsf{op}}_{\mathsf{mn}}$ and $j$ is $\mathsf{id}$.

**Proof.** By Lemma 4.3, we see that the equivalence relation $(m, w) \simeq (n, v)$ involved in forming the hom-sets of $\boldsymbol{set}^{\mathsf{op}}_{\mathsf{mn}}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$ is exactly the equivalence of Proposition 4.1, part (2), because $\pi \cdot m = \pi \cdot n$ by the definition of 2-cells in $\boldsymbol{set}^{\mathsf{op}}_{\mathsf{mn}}(\boldsymbol{x}\colon j\boldsymbol{\Sigma})$. Notice that if there is a 2-cell from $(m, w)$ to $(n, v)$ then $\pi \cdot m = \pi \cdot n$:




This obviously extends to arbitrary undirected paths from $(m, w)$ to $(n, v)$.

Therefore, the underlying graphs of both categories are the same. All we need to verify is that the compositions in the two categories agree: given $[m, w]\colon x \to y$ and $[n, v]\colon y \to z$, i.e., $m\colon y \rightarrowtail x \times w$ and $n\colon z \rightarrowtail y \times v$, their composite is $[\alpha \cdot (m \times v) \cdot n, (w \times v)]$ and we verify that

$$\mathsf{ker}(\pi' \cdot \alpha \cdot (m \times v) \cdot n) \;=\; \mathsf{ker}(\pi' \cdot n) \,\cap\, (\pi \cdot n)^{-1}(\mathsf{ker}(\pi' \cdot m)) \qquad \Box$$

Having identified $\boldsymbol{T}(\boldsymbol{set})$ as a free addition of constrained monoidal indeterminates, it seems worthwhile to point out the ingredients of $\lceil \boldsymbol{set}^{\mathsf{op}}_{\mathsf{mn}} \rceil$ in the former:

- An indeterminate $x_W$ in $\boldsymbol{T}(\boldsymbol{set})$ is $(!\colon W \to 1, \Delta_W)$, where $\Delta_W$ is the equality relation on $W$.
- Raw morphisms are of the form $(m\colon W \to V, \top_W)$. By the injectivity requirement, $m$ must be a monomorphism.
- The naturality constraint for the indeterminates is satisfied: $(m, \top_W) \cdot x_V = x_W$ because $m^{-1}(\Delta_V) = \Delta_W$ by injectivity of $m$. Notice that this is a *necessary*, as well as a sufficient, condition on $m$ for commutativity with indeterminates.
- Any morphism $(f\colon Y \to X, Q)$ factors as the expansion $(\pi\colon X \times Y/Q \to X, \top_X \times \Delta_{Y/Q})$ followed by the raw monomorphism $(\langle f, q\rangle\colon Y \to X \times Y/Q, \top_Y)$.

*4.3. Universality of Oles Categories*

In a category with finite products, we say an object $X$ is *internally non-empty* if the unique arrow into a terminal object, $X \to 1$, is a regular epi (necessarily a coequalizer of the two projections $\pi, \pi'\colon X \times X \to X$).

For a small exact category $\boldsymbol{C}$, let $\boldsymbol{C}_{\mathsf{iso}}$ be the sub-category of $\boldsymbol{C}$ with the same objects but only isomorphisms as arrows; then,

**Theorem 4.5.** $\boldsymbol{O}(\boldsymbol{C}) \equiv \lceil \boldsymbol{C}^{\mathsf{op}}_{\mathsf{iso}} \rceil$, *provided every object of* $\boldsymbol{C}$ *is internally non-empty.*

**Proof.** Oles's category is essentially the sub-category of the Tennent category with the same objects but where we restrict the raw morphisms to be (equivalence classes of) *isos*, rather than monos.

Given isomorphisms $m\colon y \cong x \times w$ and $n\colon y \cong x \times v$, and a mono $h\colon w \rightarrowtail v$ such that $n = m \cdot x \times h$, it follows by cancellation that $x \times h$ is an isomorphism, and $x$ non-empty implies then that $h$ is itself an isomorphism. The result now follows from Theorem 4.4 and Corollary 4.2. $\Box$

The reason that we must restrict to *non-empty* objects above is that the identification in $\boldsymbol{O}(\boldsymbol{C})(x, y)$ should be achieved as in Proposition 4.1. This would require taking monos as the identifying arrows, but the base category of raw morphisms only provides isos. As argued in the above proof, *if* $x$ is non-empty, $(x \times m)$ iso implies $m$ is an iso, and hence isos suffice to provide the required identifications in this context.

Should it be needed in any application, an empty object may be added to $\boldsymbol{O}(\boldsymbol{C})$ above as a free terminal object[4]; such is the role of the empty set in Oles's original construction. The universality property should then be suitably extended by demanding that the target categories have terminal objects, and the mediating "substitution" functors between them preserve such.

[4] A free terminal object has no morphisms out of it.

*4.4. Symmetric Monoidal Generalizations of Oles and Tennent Categories*

Consider now any small symmetric monoidal category $\boldsymbol{C}$ where $x \otimes \_ : \boldsymbol{C} \longrightarrow \boldsymbol{C}$ preserves monomorphisms. For example, this is the case when $\boldsymbol{C}$ has cartesian monoidal structure. We may now describe $\boldsymbol{T}(\boldsymbol{C})$, a category of worlds with data types in $\boldsymbol{C}$, which agrees with Tennent's category when $\boldsymbol{C}$ is ***set*** with its cartesian monoidal structure, and can therefore be seen as a *symmetric monoidal* generalization of Tennent's construction.

Let $\boldsymbol{C}_{\mathsf{mn}}$ be the sub-category with the same objects as $\boldsymbol{C}$ and monomorphisms as arrows. It inherits the symmetric monoidal structure of $\boldsymbol{C}$ by our assumption on $x \otimes \_$; then define $\boldsymbol{T}(\boldsymbol{C}) = \lceil \boldsymbol{C}_{\mathsf{mn}}^{\mathsf{op}} \rceil$.

We may also describe an analogous symmetric-monoidal generalization of Oles's construction. For any small symmetric monoidal category $\boldsymbol{C}$, let $\boldsymbol{C}_{\mathsf{iso}}$ be the broad sub-category of isomorphisms, which retains the symmetric monoidal structure of $\boldsymbol{C}$. Then, $\lceil \boldsymbol{C}_{\mathsf{iso}}^{\mathsf{op}} \rceil$ agrees with $\boldsymbol{O}(\boldsymbol{C})$ when $\boldsymbol{C}$ is any small category of *non-empty* sets. Thus, we obtain a version of Oles's construction that applies to any symmetric monoidal category, in line with the later developments of O'Hearn and Reynolds [2000].

**Remark 4.6.** Although the Oles construction as specified above is a perfectly sensible one, the analysis of $\boldsymbol{F}_{\mathsf{inj}}$ in §4.6 will suggest we should really consider only free symmetric monoidal categories on *a set* $\mathcal{W}$ *of generating objects*, corresponding to the collection of basic data types. Thus, the only isomorphisms present will be those generated by symmetry and associativity. These are enough to build a symmetric monoidal category with *expansion morphisms*, the crucial feature of a category of storage-shapes. This version of the Oles construction is the one we will consider in §4.7 to obtain a many-sorted version of an indexed Lawvere-theory for storage.

*4.5. The States Functor*

O'Hearn and Tennent [1992] discuss a functor mapping worlds to the sets of states available in that world. We can show that the existence of this functor is a direct consequence of the *universality* of Tennent's and Oles's categories of worlds proved in §4.2 and §4.3, at the same time extending the construction to their monoidal generalizations in §4.4 when the monoidal structure is given by cartesian products.

Recall from Corollary 2.13 that $\lceil \boldsymbol{C} \rceil$ is the free co-affine category on a symmetric monoidal category $\boldsymbol{C}$. To give a strong monoidal functor $\overline{F}: \lceil \boldsymbol{C} \rceil \longrightarrow \boldsymbol{S}^{\mathsf{op}}$ (with the cartesian monoidal structure on $\boldsymbol{S}^{\mathsf{op}}$, which makes it coaffine) is equivalent to giving a strong monoidal functor $F: \boldsymbol{C} \longrightarrow \boldsymbol{S}^{\mathsf{op}}$. In the two cases of interest to us, namely $\boldsymbol{O}(\boldsymbol{C})$ and $\boldsymbol{T}(\boldsymbol{C})$, there is a natural choice for such an $F$; recall that a representable functor $\boldsymbol{C}(a, \_): \boldsymbol{C} \longrightarrow \boldsymbol{S}$ preserves all limits.

- For $\boldsymbol{O}(\boldsymbol{C}) = \lceil \boldsymbol{C}_{\mathsf{iso}}^{\mathsf{op}} \rceil$, the required $F^{\mathsf{op}}: \boldsymbol{C}_{\mathsf{iso}} \longrightarrow \boldsymbol{S}$ is $\boldsymbol{C}(1, \_) \cdot j : \boldsymbol{C}_{\mathsf{iso}} \longrightarrow \boldsymbol{C} \longrightarrow \boldsymbol{S}$, where $j: \boldsymbol{C}_{\mathsf{iso}} \longrightarrow \boldsymbol{C}$ is the inclusion, which is trivially strong monoidal.
- For $\boldsymbol{T}(\boldsymbol{C}) = \lceil \boldsymbol{C}_{\mathsf{mn}}^{\mathsf{op}} \rceil$, the required $F^{\mathsf{op}}: \boldsymbol{C}_{\mathsf{mn}} \longrightarrow \boldsymbol{S}$ is $\boldsymbol{C}(1, \_) \cdot k : \boldsymbol{C}_{\mathsf{mn}} \longrightarrow \boldsymbol{C} \longrightarrow \boldsymbol{S}$, where $k: \boldsymbol{C}_{\mathsf{mn}} \longrightarrow \boldsymbol{C}$ is the inclusion. Notice that $F^{\mathsf{op}}$ preserves the monoidal structure given by cartesian products, and sends the maps in $\boldsymbol{C}_{\mathsf{mn}}$ to injections (monomorphisms in $\boldsymbol{S}$).

In each case, the induced strong monoidal functor $\overline{F}$ is called the *states* functor and will be denoted $S: \boldsymbol{O}(\boldsymbol{C})^{\mathsf{op}} \longrightarrow \boldsymbol{S}$ and $S: \boldsymbol{T}(\boldsymbol{C})^{\mathsf{op}} \longrightarrow \boldsymbol{S}$. The states functor $S$ therefore sends a world $v$ to the corresponding set $\boldsymbol{C}(1, v)$ of global elements, a tensor product of worlds $v \otimes w$ to the cartesian product $S(v) \times S(w)$, and a world expansion $e_v^w: v \longrightarrow v \otimes w$ to the projection $\pi_0: S(v) \times S(w) \longrightarrow S(v)$.

It is worth pointing out that the remaining basic semantic functors for ALGOL, namely those corresponding to expressions, commands, and variables, are definable from $S$ and constant functors via the *contra-exponentiation* operation described in O'Hearn and Tennent [1992].

*4.6. The Category of Finite Sets and Injections*

Several authors [Moggi, 1990; Pitts and Stark, 1993; Sieber, 1994; Stark, 1996; Fiore et al., 2002] have used the category $\boldsymbol{F}_{\mathsf{inj}}$ of *finite sets* (of locally available "locations" or "names") with *injections* as the morphisms. Fiore [2005] and Power [2006] have observed that $\boldsymbol{F}_{\mathsf{inj}}$ is equivalent to the free symmetric (strict) monoidal category with an initial unit on one generator. We will exhibit this category as an instance of (our generalization of) the Oles construction described in §4.3.

Consider the category $\boldsymbol{F}_{\mathsf{bij}}$ of finite sets and bijections (or permutations). This is known to be the free symmetric monoidal category on one generator [Kelly, 1974b], the generator being any one-point set 1, and the monoidal structure being disjoint union (finite co-product). Applying the Oles construction to $\boldsymbol{F}_{\mathsf{bij}}$ freely adds a monoidal indeterminate $x_1 : \emptyset \longrightarrow 1$.

**Proposition 4.7.** *There is an identity-on-objects isomorphism* $\boldsymbol{F}_{\mathsf{inj}} \cong \lceil \boldsymbol{F}_{\mathsf{bij}} \rceil$ *and so* $(\boldsymbol{F}_{\mathsf{inj}}, +, \emptyset)$ *is the free symmetric monoidal category on one generator* 1 *with a monoidal indeterminate* $x_1 : \emptyset \longrightarrow 1$.

**Proof.** An injection $f : X \rightarrowtail Y$ corresponds to a identity-on-objects isomorphism $X + W \cong Y$ with $W = Y \setminus f(X)$ and this correspondence is compatible with permutations of $W$. The universal characterization of $(\boldsymbol{F}_{\mathsf{inj}}, +, \emptyset)$ now follows from those of $(\boldsymbol{F}_{\mathsf{bij}}, +, \emptyset)$ and the Oles construction. □

Although we used the $\lceil \cdot \rceil$ construction in the proof above, these indeterminates are in fact *free*, as $\boldsymbol{F}_{\mathsf{bij}}$ is a free symmetric monoidal category. In contrast to the characterization mentioned by ibid., we do not assume initiality of the unit, only the presence of a global element on the generator (to map the "monoidal indeterminate" given by the inclusion $\emptyset \longrightarrow 1$). In fact, initiality of the unit is a *consequence*, as explained in Proposition 2.12. The following example illustrates the different strengths of these two characterisations:

**Example 4.8.** Let $\mathcal{SMCAT}$ denote the large 2-category of symmetric monoidal categories, strong symmetric monoidal functors and monoidal transformations.

$$\mathcal{SMCAT}((\boldsymbol{F}_{\mathsf{inj}}, +, \emptyset), (\boldsymbol{Set}, \times, 1)) \cong 1/\boldsymbol{Set}$$

To give a strong symmetric monoidal functor $H : \boldsymbol{F}_{\mathsf{inj}} \longrightarrow \boldsymbol{Set}$ is to give a set and an element $x \in H(\{\star\})$, while a monoidal transformation $\beta : H \Longrightarrow H' : \boldsymbol{F}_{\mathsf{inj}} \longrightarrow \boldsymbol{Set}$ amounts to a function $h = \beta_{\{\star\}} : H(\{\star\}) \longrightarrow H'(\{\star\})$ such that $hx = x'$. Notice that the freeness of $\boldsymbol{F}_{\mathsf{inj}}$ as a co-affine category tells us nothing in this situation, since $(\boldsymbol{Set}, \times, 1)$ is not co-affine.

□

A straightforward consequence of our identification is that the formula

$$B^A(s) = \boldsymbol{set}^{\boldsymbol{F}_{\mathsf{inj}}}(A(s + \cdot), B(s + \cdot))$$

for functor exponentiation in [Stark, 1996, Section 5] is an instance of the Exponent Representation Lemma of [O'Hearn and Reynolds, 2000, Lemma 4], which in fact holds for any $\boldsymbol{O}(\boldsymbol{C})$ category.

### *4.7. Lawvere Theories for Storage*

Moggi [1990] analyzed dynamic and local storage allocation in his monad-based approach to computational effects. Recently, Plotkin and Power [2002, 2004]; Hyland et al. [2006] have developed an *algebraic* approach to computational effects based on (extended versions of) Lawvere theories [Hyland and Power, 2007]. The essential idea is to describe computational effects as algebraic operations or generic effects, which determine monads in standard ways. It is then feasible to study *combinations* of such effects, an important modularity issue which is problematic with monads. In particular, the addition of a global storage cell, or program variable, can be achieved using the tensor product of Lawvere theories. Tennent and Ghica [2000] give an historical account of earlier efforts to achieve an *abstract* theory of storage.

Our aim in this section is to use our theory of monoidal indeterminates to present a treatment of *many-sorted* storage as an indexed Lawvere theory. Following Moggi [1990, Exercise 4.1.15.2] and Power [2006], we introduce a typed version of their "block" construction to account for *local* variables. We take a leisurely route towards the theory of local storage. We start with a simple theory for a *single* global variable (Example 4.10), then extend it to a finite number of variables using a tensor product of theories (Example 4.11), then to a variable finite number via indexing (Example 4.18), and we conclude with one for local storage, which appeals to the semantic "block" operation (Examples 4.20 and 4.21).

#### *4.7.1. Lawvere Theories*

Traditionally, a Lawvere theory [Lawvere, 1963] is a category with finite products and a generator 1; i.e., every object is of the form $1^n$. This is the category-theoretic characterization of the universal-algebra notion of the "clone" of an algebraic theory. The distinctive feature of Lawvere's approach to universal algebra is that a conventional model of a theory is viewed as a finite-product preserving *functor* to ***Set***, the category of sets, so that the conventional notion of homomorphism of algebras amounts to a *natural transformation* of functorial models: a model $M$ is determined on objects by its value $M(1)$ at the generator 1, so that a natural transformation $\theta\colon M \dot{\longrightarrow} M'$ of models is determined by a function from $M(1)$ to $M'(1)$ that commutes with the interpretations of the operations by $M$ and $M'$. This re-formulation of the basic notions of universal algebra immediately generalizes to interpretations in arbitrary categories with finite products, such as ***ωCpo***, but, for simplicity, we will discuss only ***Set***-based models and will not consider *enriched* Lawvere theories.

The re-formulation of Moggi's monads for side-effects led Plotkin and Power [2002] to generalize traditional Lawvere theories to allow for non-finite but *countable* arities.

**Definition 4.9 (Countable Lawvere Theories).** Let $\boldsymbol{Set}_{\aleph_1}$ be the skeleton of the full subcategory of ***Set*** on the countable sets. It is the free category with countable products on the generator 1, with $1^w \cong w$ [Power, 1999].

- A ***countable Lawvere theory*** is a small category ***Th*** with countable products and $I\colon \boldsymbol{Set}_{\aleph_1}^{\mathrm{op}} \longrightarrow \boldsymbol{Th}$, a strict-countable-product preserving identity-on-objects functor. A morphism of countable Lawvere theories is a functor $F\colon \boldsymbol{Th} \longrightarrow \boldsymbol{Th}'$ commuting with the $I$ functors. We write ***Law*** for the resulting category of countable Lawvere theories.

- If $\boldsymbol{S}$ is a category with countable products, a ***model of Th in S*** is a countable-product preserving functor $M\colon \boldsymbol{Th} \longrightarrow \boldsymbol{S}$. A morphism of models $M, M'\colon \boldsymbol{Th} \longrightarrow \boldsymbol{S}$ is simply a natural transformation. We write $\boldsymbol{Mod}(\boldsymbol{Th}, \boldsymbol{S})$ for the category of models of ***Th*** in ***S***.

**Example 4.10 (Theory for a Global Variable).** For any countable set $V$ of storable values, the category $\boldsymbol{GV}(V)$, a countable Lawvere theory for a global variable taking values in $V$, is generated by operations

$$\mathsf{lku}\colon 1^{|V|} \longrightarrow 1$$

$$\mathsf{upd}\colon 1 \longrightarrow 1^{|V|}$$

for "lookup" and "update," respectively, subject to commutative diagrams given in [Plotkin and Power, 2002]. The constraints may also be given using algebraic equations. For every $v \in V$, let $\mathsf{upd}_v\colon 1 \longrightarrow 1$ be a unary function symbol so that, essentially, $\mathsf{upd} = \langle \mathsf{upd}_v \mid v \in V \rangle$, where $\langle \cdots v \cdots \mid v \in V \rangle$ denotes a countable sequence such that, for all $v \in V$, the $v$th component is $\cdots v \cdots$. Note that the equations are in "continuation" format; for example, in $\mathsf{upd}_v(\mathsf{upd}_{v'}(a))$, the update to $v'$ is the *second* update.

- $\mathsf{lku}\,\langle \mathsf{upd}_v(a) \mid v \in V \rangle = a$, where $a \in 1$ ; i.e., looking up the value of the variable and then updating the variable by that value is equivalent to doing nothing.

- $\mathsf{lku}\,\big\langle \mathsf{lku}\,\langle t(v, v') \mid v \in V \rangle \,\big|\, v' \in V \big\rangle = \mathsf{lku}\,\langle\, t(v, v) \mid v \in V \rangle$, where $t \in 1^{|V \times V|}$; i.e., looking up the value of the variable twice is equivalent to looking it up once and duplicating the value.

- $\mathsf{upd}_v(\mathsf{upd}_{v'}(a)) = \mathsf{upd}_{v'}(a)$, where $a \in 1$; i.e., updating the variable twice is equivalent to doing just the second update.

- $\mathsf{upd}_v(\mathsf{lku}(t)) = \mathsf{upd}_v(t(v))$, where $t \in 1^{|V|}$; i.e., the value of the variable immediately after updating it is the value just assigned to it.

Three further axioms that treat interactions involving *distinct* variables can be omitted because we have only one variable.

For a non-trivial model, suppose $R$ is any set with at least 2 elements (the possible final "results" or "answers") and interpret the objects and the basic operations of $\boldsymbol{GV}(V)$ as follows:

$$[\![1]\!] = R^V$$

$$[\![1^{|V|}]\!] = (R^V)^V$$

$$[\![\mathsf{lku}]\!]\colon\ (R^V)^V \longrightarrow R^V :\ (k)(v) \longmapsto k(v)(v)$$

$[\![\mathsf{upd}_v]\!]\colon R^V \longrightarrow R^V : (c)(v_0) \longmapsto c(v)$

It is routine to verify that these interpretations satisfy the four equations given previously.

□

*4.7.2. Tensor Product of Lawvere Theories*

Hyland et al. [2006] describe a tensor product $\boldsymbol{Th}\otimes\boldsymbol{Th'}$ of Lawvere theories $\boldsymbol{Th}$ and $\boldsymbol{Th'}$ such that, for any category $\boldsymbol{S}$ with countable products, there is a coherent equivalence of categories

$$\boldsymbol{Mod}(\boldsymbol{Th} \otimes \boldsymbol{Th'}, \boldsymbol{S}) \;\cong\; \boldsymbol{Mod}(\boldsymbol{Th}, \boldsymbol{Mod}(\boldsymbol{Th'}, \boldsymbol{S}))$$

i.e., each operation of one theory is interpreted as a homomorphism of models of the other theory. Intuitively, the construction is as follows: one takes all the operations and equations of each and insists that each operation of one commute with each operation of the other. A fact crucial to our intended application is that the *initial* countable Lawvere theory, $id\colon \boldsymbol{Set}^{\mathsf{op}}_{\aleph_1} \longrightarrow \boldsymbol{Set}^{\mathsf{op}}_{\aleph_1}$ is a unit for the tensor, so that $(\boldsymbol{Law}, \otimes, \boldsymbol{Set}^{\mathsf{op}}_{\aleph_1})$ is a *co-affine* category; cf., § 2.7.

**Example 4.11 (Theory for Two Global Variables).** Consider countable sets $V$ and $V'$ of storable values; the theory $\boldsymbol{GV}(V) \otimes \boldsymbol{GV}(V')$ has "selective" lookup and update operations as follows:

$$\begin{array}{ll} \mathsf{lku}^V\colon 1^{|V|} \longrightarrow 1 & \qquad \mathsf{upd}^V\colon 1 \longrightarrow 1^{|V|} \\ \mathsf{lku}^{V'}\colon 1^{|V'|} \longrightarrow 1 & \qquad \mathsf{upd}^{V'}\colon 1 \longrightarrow 1^{|V'|} \end{array}$$

axiomatised separately as above, plus the following "non-interference" (commutativity) axioms:

- $\mathsf{upd}^V_v(\mathsf{lku}^{V'}(t)) \;=\; \mathsf{lku}^{V'}\langle \mathsf{upd}^V_v(t(v')) \bigm| v' \in V'\rangle$, where $t \in 1^{|V'|}$; i.e., updating and looking up distinct variables will yield the same result if done in either order, (and, symmetrically, with $V$ and $V'$ swapped).
- $\mathsf{lku}^V\langle \mathsf{lku}^{V'}\langle t(v,v') \mid v' \in V'\rangle \bigm| v \in V\rangle \;=\; \mathsf{lku}^{V'}\langle \mathsf{lku}^V\langle t(v,v') \mid v \in V\rangle \bigm| v' \in V'\rangle$, where $t \in 1^{|V\times V'|}$; i.e., looking up the values of distinct variables will yield the same results if done in either order.
- $\mathsf{upd}^V_v(\mathsf{upd}^{V'}_{v'}(a)) \;=\; \mathsf{upd}^{V'}_{v'}(\mathsf{upd}^V_v(a))$, where $a \in 1$; i.e., updating distinct variables will have the same effects if done in either order.

Note that the three Plotkin and Power [2002] axioms for distinct variables (omitted in the preceding sub-section) re-appear here as non-interference axioms.

It follows from [Hyland et al., 2006, Theorem 2.4] that

$$\boldsymbol{GV}(V) \;\otimes\; \boldsymbol{GV}(V') \;\cong\; \boldsymbol{GV}(V \times V')$$

i.e., the tensor of the theories for *two* independent variables is equivalent to the theory of a *single* variable for *pairs* of values; this generalizes to any finite number of variables. This shows clearly one way in which Lawvere theories are *abstract*: two different presentations yield equivalent theories.

A non-trivial model may be obtained by setting

$[\![1]\!] = R^{V\times V'}$

$[\![1^{|V|}]\!] = (R^{V\times V'})^V$

$[\![\mathsf{lku}^V]\!]\colon (R^{V\times V'})^V \longrightarrow R^{V\times V'} : (k)(v,v') \longmapsto k(v)(v,v')$

$[\![\mathsf{upd}^V_v]\!]\colon R^{V\times V'} \longrightarrow R^{V\times V'} : (c)(v_0,v') \longmapsto c(v,v')$

where $R$ is again a non-trivial set of "results," and similarly for $[\![\mathsf{lku}^{V'}]\!]$ and $[\![\mathsf{upd}^{V'}_{v'}]\!]$.

□

*4.7.3. Co-Models of Lawvere Theories*

We shall need the following definition and results from [Power and Shkaravska, 2004; Power, 2006]:

**Definition 4.12 (Co-model of a Lawvere Theory).** If $\boldsymbol{Th}$ is a countable Lawvere theory and $\boldsymbol{S}$ is a category with countable co-products, a *co-model of* $\boldsymbol{Th}$ *in* $\boldsymbol{S}$ is a model of $\boldsymbol{Th}$ in $\boldsymbol{S}^{\mathsf{op}}$.

**Example 4.13 (Final Co-model of the Theory of a Global Variable).** Algebraic equations satisfied by co-models of $\boldsymbol{GV}(V)$ can be presented in a more "direct" way than those for models; for operations

$$\mathsf{lku}\colon \mathsf{A} \longrightarrow \mathsf{A} \times V \qquad \mathsf{upd}\colon \mathsf{A} \times V \longrightarrow \mathsf{A}$$

suitable axioms are, for $a \in \mathsf{A}$ and $v, v' \in V$:

- $\mathsf{upd}(\mathsf{lku}(a)) = a$; i.e., looking up the value of the variable and then updating the variable by that value is equivalent to doing nothing.
- $\mathsf{upd}(\mathsf{upd}(a, v), v') = \mathsf{upd}(a, v')$ i.e., updating the variable twice is equivalent to doing just the second update.
- $\mathsf{lku}(a') = \mathsf{lku}(a)$ where $(a', v) = \mathsf{lku}(a)$; i.e., looking up the value of the variable has no side-effects.
- $\mathsf{lku}(\mathsf{upd}(a, v)) = (\mathsf{upd}(a, v), v)$; i.e., the value of the variable immediately after updating it is the value just assigned to it.

The following is a co-model (in fact, the *final* co-model) for $\boldsymbol{GV}(V)$:

- $[\![\mathsf{A}]\!] = V$
- $[\![\mathsf{lku}]\!] : [\![\mathsf{A}]\!] \longrightarrow [\![\mathsf{A}]\!] \times V : v \longmapsto (v, v)$
- $[\![\mathsf{upd}]\!] : [\![\mathsf{A}]\!] \times V \longrightarrow [\![\mathsf{A}]\!] : (v_0, v) \longmapsto v$ □

The motivation for considering co-models is the following:

**Proposition 4.14 (Power06).** *If $A$ is a co-model of a theory $\boldsymbol{Th}$ in $\boldsymbol{S}$, and $M$ is a model of a theory $\boldsymbol{Th}'$ in $\boldsymbol{S}$, then $M^A$ is a model of $\boldsymbol{Th} \otimes \boldsymbol{Th}'$ in $\boldsymbol{S}$.*

**Example 4.15.** $V$ is the final co-model of $\boldsymbol{GV}(V)$; hence, if $M_u$ is a model of $\boldsymbol{GV}(U)$, ${M_u}^V$ is a model of $\boldsymbol{GV}(V) \otimes \boldsymbol{GV}(U)$ (which is equivalent to $\boldsymbol{GV}(V \times U)$). □

**Example 4.16.** As a special case of the Proposition, if $R$ is any set, $R^{(-)}\colon \boldsymbol{Mod}(\boldsymbol{Th}, \boldsymbol{S}^{\mathsf{op}}) \longrightarrow \boldsymbol{Mod}(\boldsymbol{Th}, \boldsymbol{S})$; the "continuation" model of $\boldsymbol{GV}(V)$ described in Example 4.10 can be obtained in this way from the "direct" co-model in Example 4.13. □

*4.7.4. Indexed Lawvere Theories*

To treat "local" (i.e., world-dependent) computational effects in the algebraic framework, Power [2006] introduced *indexed* Lawvere theories.

**Definition 4.17 (Indexed Lawvere Theories).** Let $\boldsymbol{D}$ be a small category;

- a ***D-indexed Lawvere theory*** is a functor $T\colon \boldsymbol{D} \longrightarrow \boldsymbol{Law}$.
- A ***model of a D-indexed Lawvere theory*** $T$ in a category $\boldsymbol{S}$ with countable products consists of, for every object $d$ in $\boldsymbol{D}$, a model $M_d\colon T(d) \longrightarrow \boldsymbol{S}$, and for every map $f\colon d \longrightarrow d'$ in $\boldsymbol{D}$, a natural transformation $M_f\colon M_d \overset{\cdot}{\longrightarrow} M_{d'} \cdot T(f)$, and this assignment is functorial in $\boldsymbol{D}$.

Given a model $M$ of a $\boldsymbol{D}$-indexed Lawvere theory $\boldsymbol{Th}$, evaluating each model $M_d$ at the generator $1$ of $\boldsymbol{Th}_d$ yields a forgetful functor $U\colon \boldsymbol{Mod}(\boldsymbol{Th}, \boldsymbol{S}) \longrightarrow [\boldsymbol{D}, \boldsymbol{S}]$. When $\boldsymbol{S}$ is a locally countably presentable category, this forgetful functor has a left adjoint given pointwise $F_d \dashv U_d\colon \boldsymbol{Mod}(\boldsymbol{Th}_d, \boldsymbol{S}) \longrightarrow \boldsymbol{S}$, and it is monadic. We write $T_{\boldsymbol{Th}}\colon [\boldsymbol{D}, \boldsymbol{S}] \longrightarrow [\boldsymbol{D}, \boldsymbol{S}]$ for the resulting monad.

**Example 4.18 (Theory for Many-Sorted Global Storage).** Given a collection $\boldsymbol{W}$ of sets $V_i$ of storable values, let $\Pi(\boldsymbol{W})$ be the *free symmetric monoidal category*[5] on $\boldsymbol{W}$: its objects are $n$-tuples of objects of $\boldsymbol{W}$ and its morphisms permutations of such tuples. The assignment $V_i \longmapsto \boldsymbol{GV}(V_i)$ therefore induces a strong symmetric monoidal functor $\boldsymbol{GV}(\_)\colon \Pi(\boldsymbol{W}) \longrightarrow \boldsymbol{Law}$, which is a $\Pi(\boldsymbol{W})$-indexed Lawvere theory for $\boldsymbol{W}$-sorted (global) storage. □

*4.7.5. Multi-Sorted Local Storage*

Consider the basic Oles construction on $\Pi(\boldsymbol{W})$; to simplify the notation, we will use $\boldsymbol{O}(\boldsymbol{W})$ as an abbreviation for $\boldsymbol{O}(\Pi(\boldsymbol{W}))$. The only morphisms in this category are *expansions* and *structural isomorphisms*. By virtue of the free co-affineness of $\boldsymbol{O}(\boldsymbol{W})$ (Theorem 4.5 and Corollary 2.13), we immediately obtain an indexed Lawvere theory $L^{\boldsymbol{W}}\colon \boldsymbol{O}(\boldsymbol{W}) \longrightarrow \boldsymbol{Law}$ for ($\boldsymbol{W}$-sorted) *local* storage. To construct a model, notice that, for every world $w \in \boldsymbol{O}(\boldsymbol{W})$, the theory $L^{\boldsymbol{W}}(w)$ is $\boldsymbol{GV}(S(w))$, where $S$ here is the contravariant states functor discussed in §4.5; so, we can define the functor $[\![1]\!]\colon \boldsymbol{O}(\boldsymbol{W}) \longrightarrow \boldsymbol{Set}$ as follows:

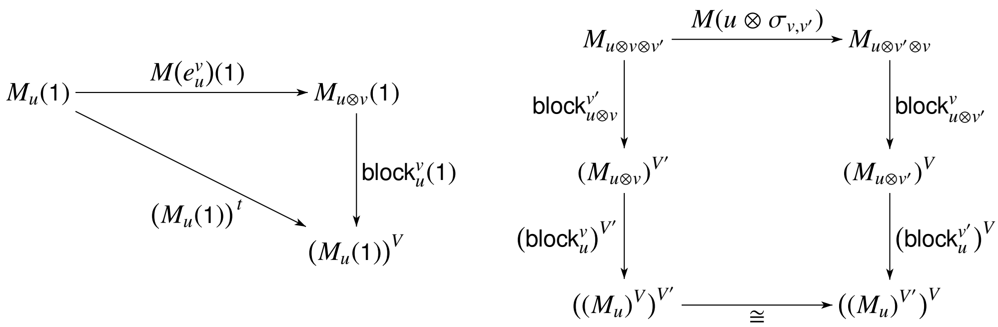
- $[\![1]\!]w = R^{S(w)}$ for every $w \in \boldsymbol{O}(\boldsymbol{W})$
- $[\![1]\!](e_w^y)$ is the model morphism $R^{S(e_w^y)} = R^{\pi_0} : R^{S(w)} \longrightarrow R^{S(w)\times S(y)}$ for every expansion $e_w^y\colon w \longrightarrow w \otimes y$ in $\boldsymbol{O}(\boldsymbol{W})$

To emulate the Reynolds-Oles treatment of *stack-allocatable* storage, Moggi [1990, Exercise 4.1.15.2] suggested introducing a "block" operation which would both bind a local identifier to new storage and allow for the de-allocation of the memory after execution of the block body. Similarly, Power [2006] introduces a *block* construct to add a new variable, while preserving the operations on the existing variables. Along the same lines, we define a *typed block algebra* as follows.

**Definition 4.19.** A ***typed block algebra*** is a model $M$ of the $\boldsymbol{O}(\boldsymbol{W})$-indexed Lawvere theory $L^{\boldsymbol{W}}$ together with a family of maps of models natural in $u$

$$\mathsf{block}_u^v\colon M_{u\otimes v} \longrightarrow (M_u)^V$$

where $V$ is the final co-model of $M_v$, subject to the following two equational axioms:




where $t\colon V \longrightarrow 1$ is the unique morphism into the terminal set and $V'$ is the final co-model of $M_{v'}$. The first diagram asserts that $\mathsf{block}_u^v$ only affects the newly-created variable; the second states that one can create and initialize two variables in either order.

**Example 4.20 (A Typed Block Algebra for Local State).** Consider the model $M$ of $L^{\boldsymbol{W}}\colon \boldsymbol{O}(\boldsymbol{W}) \longrightarrow \boldsymbol{Law}$ for $\boldsymbol{W}$-sorted local storage described above; maps $\mathsf{block}_u^v\colon M_{u\otimes v} \longrightarrow (M_u)^V$ for $V = S(v)$, the final co-model of $\boldsymbol{GV}(S(v))$, are evident because $M_{u\otimes v} = R^{S(u\otimes v)} = R^{S(u)\times V} \cong (R^{S(u)})^V = (M_u)^V$. □

**Example 4.21 (Another Typed Block Algebra for Local State).** The classic typed block algebra for local state was described by Oles [1982] and Reynolds [1981b]. Define $[\![1]\!]$ as follows:

- $[\![1]\!]w = S(w)^{S(w)}$ for every $w \in \boldsymbol{O}(\boldsymbol{W})$

[5] We have seen in §4.6 the example $\boldsymbol{F}_{\mathsf{bij}} = \Pi(\{*\})$.

- $[\![1]\!](e_w^y)(c)(w,y) = (c(w), y)$ for every expansion $e_w^y\colon w \longrightarrow w \otimes y$ in $\boldsymbol{O}(\boldsymbol{W})$

The interpretations of $\mathsf{lku}$ and $\mathsf{upd}$ are as in Example 4.13. Then, $\mathsf{block}_u^v\colon M_{u\otimes v} \longrightarrow (M_u)^V : (c)(v)(u) \longmapsto \pi_0(c(u,v))$; here, $v$ is the initial value of the new local variable, $c(u,v)$ is the execution in the local world, and projection $\pi_0$ discards the final value of the local variable. □

A morphism of typed block algebras is a morphism of models which commutes with the families of $\mathsf{block}$-maps in the evident fashion. We thus have a category ***TypedBlockAlg*** and a forgetful functor to $[\boldsymbol{O}(\boldsymbol{W}), \boldsymbol{Set}]$ (underlying the models of the $\boldsymbol{O}(\boldsymbol{W})$-indexed Lawvere theory); this functor is monadic. We conclude by spelling out the resulting monad $T_{\mathsf{LS}}$ on $[\boldsymbol{O}(\boldsymbol{W}), \boldsymbol{Set}]$. Recall from §4.5 the contravariant strong symmetric-monoidal *states* functor $S\colon \boldsymbol{O}(\boldsymbol{W})^{\mathsf{op}} \longrightarrow \boldsymbol{Set}$, which sends the object $w$ to itself, and expansions to projections. The action of $T_{\mathsf{LS}}$ on objects is as follows: given $X\colon \boldsymbol{O}(\boldsymbol{W}) \longrightarrow \boldsymbol{Set}$,

$$T_{\mathsf{LS}}(X)(w) \;=\; \left(\int^{f:w\longrightarrow w'} S(w') \times X(w')\right)^{S(w)}$$

where $\int$ stands for the *co-end* of a bivariant functor/diagram [Mac Lane, 1971, Section IX.6], evaluated over the slice category $w/\boldsymbol{O}(\boldsymbol{W})$.

For the action on morphisms, we consider the two kinds: on isomorphisms, it's quite evident. On expansions, given $e_w^u\colon w \longrightarrow w \otimes u$, recall $S(w \otimes u) = S(w) \times S(u)$, and $(\_) \times S(u)$ preserves co-ends (as they are colimits). Transposing across the exponential adjunction, the map $T_{\mathsf{LS}}(e_w^u)$ amounts to

$$\left(\int^{f:w\longrightarrow w'} S(w') \times X(w')\right)^{S(w)} \times S(w) \times S(u) \longrightarrow \left(\int^{f:w\longrightarrow w'} S(w') \times X(w')\right)$$

which evaluates at $S(w)$ and maps the $h\colon w \longrightarrow v$ component of the first co-end, to the $(h \otimes u)\colon (w \otimes u) \longrightarrow (v \otimes u)$ component of the second as follows:

$$S(v) \times X(v) \times S(u) \xrightarrow{\;\cong\;} S(v \otimes u) \times X(v) \xrightarrow{\;\mathsf{id} \times X(e_w^u)\;} S(w \otimes u) \times X(w \otimes u)$$

Note that naturality of this assignment of co-end components relies on the naturality of expansions $e_w^u$ on $w$, part (2) of our Lemma 2.8.

Notice that $[\![1]\!]$ in Example 4.21 is $T_{\mathsf{LS}}(1)$.

## 5. Discussion

We have described here the construction of a polynomial symmetric monoidal (closed) category, obtained from a symmetric monoidal (closed) category by freely adjoining a system of monoidal indeterminates. The construction was motivated by our desire to understand the categories of possible worlds that have been used in semantical analyses of languages allowing creation of "new" variables or names. These categories, though originally presented in fairly *ad hoc* fashion, have all been shown here to be polynomial monoidal categories, with corresponding universality properties. Intuitively, the indeterminates represent uninitialized "new" components of the state or name context; the substitution functor $F|_{\boldsymbol{x}}^{\boldsymbol{d}}$ then provides the means to produce an "expanded" state or context with *initialized* new variables, for any appropriate choice of initial values $\boldsymbol{d}$:

$$\boldsymbol{C} \underset{F|_{\boldsymbol{x}}^{\boldsymbol{d}}}{\overset{R_{\Sigma}}{\rightleftarrows}} \boldsymbol{C}[\boldsymbol{x}\colon j\boldsymbol{\Sigma}]$$

We expect that the methodology introduced here will be useful in other applications. For example, it is tempting to consider "contextual (or functional) completeness" [Hermida and Jacobs, 1995] in the symmetric-monoidal setting by requiring $R_{\Sigma}$ to have a left (resp. right) adjoint. However, we have not yet been able to identify reasonable conditions under which such adjunctions exist.

*Related work*

After our initial submission of this work, it came to our attention that the construction of a category generated by an indeterminate for a *single* object (cf. §2.8) in the *strict* symmetric monoidal case and its universal property were briefly described in the Appendix of Richard Wood's dissertation [Wood, 1976].

Pavlović [1997] considered an application of monoidal indeterminates in relation to Milner's action calculi. Only the evident "syntactic" construction is considered, together with the well-known special case when the object under consideration admits a comonoid structure, whereby the addition of an indeterminate can be realised by taking the Kleisli category of the resulting comonad, see §2.9. This latter identification is further analyzed in [Hermida and Jacobs, 1995], where it is shown that, in the cartesian setting, $\boldsymbol{C}_{\times w}$ has the universal property of $\boldsymbol{C}[x_w \colon w]$ based merely on its 2-categorical universal characterisation as a lax colimit, regardless of any explicit description.

The above mentioned "syntactic" construction corresponds to the fact that the categorical structures under consideration are monadic over the category of graphs [Burroni, 1981], and therefore admit presentations by generators and relations. Thus, given a symmetric monoidal category $\boldsymbol{C}$, we consider its underlying graph $G(\boldsymbol{C})$, add whichever elements $\mathcal{W}$ we require, freely generate a symmetric monoidal category on the extended graph $F(G(\boldsymbol{C}) + \mathcal{W})$, and then impose the existing relations in $\boldsymbol{C}$ so as to obtain a strong symmetric monoidal functor $R \colon \boldsymbol{C} \longrightarrow [F(G(\boldsymbol{C}) + \mathcal{W})]_{\simeq}$.

As far as the structure of categories of possible worlds is concerned, the prominent role of expansion morphisms and an associated notion of *quotient* are considered in [Levy, 2008]. In a quite different line of application, namely update strategies for databases, Johnson et al. [2009] exhibit the morphisms of Oles's category of possible worlds (as formulated in [Oles, 1985]) as algebras, with the same caveat of non-emptiness we considered in §4.3: an $\boldsymbol{O}(\boldsymbol{Set})(X, V)$ morphism is an algebra for the monad $(\_) \times V \dashv \mathsf{dom} \colon \boldsymbol{Set}/V \longrightarrow \boldsymbol{Set}$. Johnson et al. also exhibit the above mentioned quotients (implicit in [Oles, 1985]) via an isomorphism of the category of algebras with ***set***.

## Acknowledgements

We are grateful to Paul Levy for pointing out to us that the empty set was problematic in an earlier version of our re-interpretation of Oles's category of worlds using indeterminates; see §4.3. He also pointed out that considering $\boldsymbol{\Sigma}$ as merely a *subcategory* of $\boldsymbol{C}$ in order to obtain the addition of a single indeterminate was problematic; in this revised version we consider the more general situation with $j \colon \boldsymbol{\Sigma} \longrightarrow \boldsymbol{C}$, crucial in §2.8 and 2.9.

We gratefully acknowledge the referees' suggestions which helped us improve the exposition.

## References

Barr, M., 1971. Exact categories. In: Barr, M., Grillet, P. A., van Osdol, D. H. (Eds.), Exact Categories and Categories of Sheaves. Vol. 236 of Lecture Notes in Mathematics. Springer, pp. 1–120.

Borceux, F., 1994a. Handbook of Categorical Algebra 1, Basic Category Theory. Vol. 50 of Encyclopedia of Mathematics and its Applications. Cambridge University Press.

Borceux, F., 1994b. Handbook of Categorical Algebra 2, Categories and Structures. Vol. 51 of Encyclopedia of Mathematics and its Applications. Cambridge University Press.

Burroni, A., 1981. Algèbres graphiques: sur un concept de dimension dans les langages formels. Cahiers Topologie Géom. Différentielle 22 (3), 249–265, third Colloquium on Categories, Part IV (Amiens, 1980).

Fiore, M., Moggi, E., Sangiorgi, D., 2002. A fully abstract model for the $\pi$-calculus. Information and Computation 179, 76–117.

Fiore, M. P., 2005. Mathematical models of computational and combinatorial structures. In: Foundations of Software Science and Computational Structures, 8th International Conference, FOSSACS 2005. Vol. 3441 of Lecture Notes in Computer Science. Springer-Verlag, Berlin, Edinburgh, U.K., pp. 25–46.

Hasegawa, M., 2004. The uniformity principle on traced monoidal categories. Publications of the Research Institute for Mathematical Sciences, Kyoto University 40 (3), 991–1014.

Hermida, C., Jacobs, B., 1995. Fibrations with indeterminates: contextual and functional completeness for polymorphic lambda calculi. Math. Structures Comput. Sci. 5 (4), 501–531.

Hermida, C., Tennent, R. D., May 2007. A fibrational framework for possible-world semantics of ALGOL-like languages. Theoretical Computer Science 375, 3–19.

Hyland, J. M. E., Plotkin, G., Power, A. J., 2006. Combining effects: Sum and tensor. Theoretical Computer Science 357 (1-3), 70–99.

Hyland, M., Power, J., April 2007. The category theoretic understanding of universal algebra: Lawvere theories and monads. Electron. Notes Theor. Comput. Sci. 172, 437–458.

Johnson, M., Rosebrugh, R., Wood, R. J., 2009. Algebras and update strategies, Submitted for publication to the Journal of Universal Computer Science. Preprint available here: `http://www.cs.mq.edu.au/~mike/papers/70.pdf`.

Joyal, A., Street, R., 1993. Braided monoidal categories. Advances in Mathematics 102, 20–78.

Joyal, A., Street, R., Verity, D., 1996. Traced monoidal categories. Mathematical Proceedings of the Cambridge Philosophical Society 119 (3), 447–468.

Kelly, G. M., 1974a. Doctrinal adjunction. In: Category Seminar (Proc. Sem., Sydney, 1972/1973). Springer, Berlin, pp. 257–280. Lecture Notes in Math., Vol. 420.

Kelly, G. M., 1974b. On clubs and doctrines. In: Sydney Category Seminar. Vol. 420 of Lecture Notes in Mathematics. Springer-Verlag, Berlin, pp. 181–256.

Kelly, G. M., Blackwell, R., Power, A. J., 1989. Two dimensional monad theory. J. Pure Appl. Algebra 59, 1–41.

Lambek, J., 1968. Deductive systems and categories I. Syntactic calculus and residuated categories. Theory of Computing Systems 2, 287–318.

Lambek, J., 1969. Deductive systems and categories II. Standard constructions and closed categories. In: Category Theory, Homology Theory and their Applications I. Vol. 86 of Lecture Notes in Mathematics. Springer Berlin / Heidelberg, pp. 76–122.

Lambek, J., 1972. Deductive systems and categories III. Cartesian closed categories, intuitionist propositional calculus, and combinatory logic. In: Toposes, Algebraic Geometry and Logic. Vol. 274 of Lecture Notes in Mathematics. Springer Berlin / Heidelberg, pp. 57–82.

Lambek, J., Scott, P. J., 1986. Introduction to Higher-Order Categorical Logic. Vol. 7 of Cambridge Studies in Advanced Mathematics. Cambridge University Press, Cambridge, England.

Lawvere, F. W., 1963. Functorial semantics of algebraic theories. Ph.D. thesis, Columbia University, republished in: *Reprints in Theory and Applications of Categories*, No. 5 (2004) pp 1-121.

Levy, P. B., 2008. Global state considered helpful. In: Proceedings of the 24th Conference on the Mathematical Foundations of Programming Semantics. Vol. 218 of Electronic Notes in Theoretical Computer Science. Elsevier, pp. 241–259.

Mac Lane, S., 1971. Categories for the Working Mathematician. Vol. 5 of Graduate Texts in Mathematics. Springer-Verlag, second edition, 1998.

Moggi, E., 1990. An abstract view of programming languages. Tech. rep., Laboratory for Foundations of Computer Science, Department of Computer Science, University of Edinburgh, available here: `http://www.lfcs.informatics.ed.ac.uk/reports/90/ECS-LFCS-90-113`.

O'Hearn, P. W., 1990. The semantics of non-interference: A natural approach. Ph.D. thesis, Queen's University, Kingston, Canada.

O'Hearn, P. W., Power, A. J., Takeyama, M., Tennent, R. D., 1999. Syntactic control of interference revisited. Theoretical Computer Science 228, 211–252, preliminary version published as Chapter 18 of O'Hearn and Tennent [1997].

O'Hearn, P. W., Reynolds, J. C., January 2000. From Algol to polymorphic linear lambda-calculus. Journal of the ACM 47 (1), 167–223.

O'Hearn, P. W., Tennent, R. D., 1992. Semantics of local variables. In: Fourman, M. P., Johnstone, P. T., Pitts, A. M. (Eds.), Applications of Categories in Computer Science. Vol. 177 of London Mathematical Society Lecture Note Series. Cambridge University Press, Cambridge, England, pp. 217–238.

O'Hearn, P. W., Tennent, R. D., 1993. Semantical analysis of specification logic, 2. Information and Computation 107 (1), 25–57, reprinted as Chapter 14 of O'Hearn and Tennent [1997].

O'Hearn, P. W., Tennent, R. D., May 1995. Parametricity and local variables. J. ACM 42 (3), 658–709, reprinted as Chapter 16 of O'Hearn and Tennent [1997].

O'Hearn, P. W., Tennent, R. D. (Eds.), 1997. Algol-like Languages. Progress in Theoretical Computer Science. Birkhäuser, Boston, two volumes.

Oles, F. J., 1982. A category-theoretic approach to the semantics of programming languages. Ph.D. thesis, Syracuse University, Syracuse, N.Y.

Oles, F. J., 1985. Type algebras, functor categories and block structure. In: Nivat, M., Reynolds, J. C. (Eds.), Algebraic Methods in Semantics. Cambridge University Press, Cambridge, England, pp. 543–573.

Oles, F. J., 1997. Functor categories and store shapes. In: O'Hearn and Tennent [1997], Ch. 11, pp. 3–12 of Volume 2, two volumes.

Pavlović, D., 1997. Categorical logic of names and abstraction in action calculi. Math. Structures Comput. Sci. 7 (6), 619–637.

Petrić, Z., 2002. Coherence in substructural categories. Studia Logica 70 (2), 271–296.

Pitts, A., Stark, I., 1993. Observable properties of higher order functions that dynamically create local names. In: A. M. Borzyszkowski, Sokolowski, S. (Eds.), Mathematical Foundations of Computer Science. Vol. 711 of Lecture Notes in Computer Science. Springer-Verlag, Berlin, Gdansk, Poland, pp. 122–140.

Plotkin, G., Power, A. J., 2004. Computational effects and operations: An overview. Electronic Notes in Theoretical Computer Science 73, 149–163, Proceedings of the Workshop on Domains VI.

Plotkin, G., Power, J., 2002. Notions of computation determine monads. In: Foundations of software science and computation structures (Grenoble, 2002). Vol. 2303 of Lecture Notes in Comput. Sci. Springer, Berlin, pp. 342–356.

Power, A. J., 2006. Semantics for local computational effects. Electronic Notes in Theoretical Computer Science 158, 355–371, Proceedings of the 22nd Annual Conference on Mathematical Foundations of Programming Semantics.

Power, J., 1999. Enriched Lawvere theories. Theory Appl. Categ. 6, 83–93 (electronic), the Lambek Festschrift.

Power, J., Shkaravska, O., December 2004. From comodels to coalgebras: State and arrays. Electron. Notes Theor. Comput. Sci. 106, 297–314.

Reynolds, J. C., Jan. 1978. Syntactic control of interference. In: Conference Record of the Fifth Annual ACM Symposium on Principles of Programming Languages. ACM, New York, Tucson, Arizona, pp. 39–46, reprinted as Chapter 10 of O'Hearn and Tennent [1997].

Reynolds, J. C., 1981a. The Craft of Programming. Prentice Hall International, U.K.

Reynolds, J. C., Oct. 1981b. The essence of Algol. In: de Bakker, J. W., van Vliet, J. C. (Eds.), Algorithmic Languages, Proceedings of the International Symposium on Algorithmic Languages. North-Holland, Amsterdam, Amsterdam, pp. 345–372, reprinted as Chapter 3 of O'Hearn and Tennent [1997].

Reynolds, J. C., Dec. 1981c. Idealized Algol and its specification logic. In: Néel, D. (Ed.), Tools and Notions for Program Construction. Cambridge University Press, Cambridge, 1982, Nice, France, pp. 121–161, reprinted as Chapter 6 of O'Hearn and Tennent [1997].

Sieber, K., Aug. 1994. Full abstraction for the second order subset of an Algol-like language. In: Mathematical Foundations of Computer Science. Vol. 841 of Lecture Notes in Computer Science. Springer-Verlag, Berlin, Kǒsice, Slovakia, pp. 608–617, reprinted as Chapter 15 of O'Hearn and Tennent [1997].

Stark, I., Feb. 1996. Categorical models for local names. lisp and Symbolic Computation 9 (1), 77–107.

Tennent, R. D., Sep. 1985. Functor-category semantics of programming languages and logics. In: Pitt, D., Abramsky, S., Poigné, A., Rydeheard, D. (Eds.), Category Theory and Computer Programming. Vol. 240 of Lecture Notes in Computer Science. Springer-Verlag, Berlin (1986), Guildford, U.K., pp. 206–224.

Tennent, R. D., 1990. Semantical analysis of specification logic. Information and Computation 85 (2), 135–162, reprinted as Chapter 13 of O'Hearn and Tennent [1997].
Tennent, R. D., Ghica, D. R., 2000. Abstract models of storage. Higher-Order and Symbolic Computation 13 (1/2), 119–129.
Wood, R., 1976. Indicial methods for relative categories. Ph.D. thesis, Dalhousie University, Halifax, Nova Scotia.

Delimiter test: innermost delimiter pairs should be identical; then of gradually increasing size. Contact author RDT if the innermost delimiter pairs differ; i.e., if, in the fonts you're using, the normal-sized delimiters in math mode are *bigger* than the normal-sized delimiters in text mode.

$$\Biggl(\biggl(\Bigl(\bigl((())\bigr)\Bigr)\biggr)\Biggr)\Biggl[\biggl[\Bigl[\bigl[[[]]\bigr]\Bigr]\biggr]\Biggr]$$