Cartesian closed bicategories: type theory and coherence

Philip James Saville

Department of Computer Science and Technology

Sidney Sussex College, University of Cambridge

This dissertation is submitted for the degree of Doctor of Philosophy

October 2019

Abstract

In this thesis I lift the Curry–Howard–Lambek correspondence between the simply-typed lambda calculus and cartesian closed categories to the bicategorical setting, then use the resulting type theory to prove a coherence result for cartesian closed bicategories. Cartesian closed bicategories—2-categories 'up to isomorphism' equipped with similarly weak products and exponentials—arise in logic, categorical algebra, and game semantics. However, calculations in such bicategories quickly fall into a quagmire of coherence data. I show that there is at most one 2-cell between any parallel pair of 1-cells in the free cartesian closed bicategory on a set and hence—in terms of the difficulty of calculating—bring the data of cartesian closed bicategories down to the familiar level of cartesian closed categories.

In fact, I prove this result in two ways. The first argument is closely related to Power's coherence theorem for bicategories with flexible bilimits. For the second, which is the central preoccupation of this thesis, the proof strategy has two parts: the construction of a type theory, and the proof that it satisfies a form of normalisation I call *local coherence*. I synthesise the type theory from algebraic principles using a novel generalisation of the (multisorted) abstract clones of universal algebra, called *biclones*. The result brings together two extensions of the simply-typed lambda calculus: a 2-dimensional type theory in the style of Hilken, which encodes the 2-dimensional nature of a bicategory, and a version of explicit substitution, which encodes a composition operation that is only associative and unital up to isomorphism. For products and exponentials I develop the theory of cartesian and cartesian closed biclones and pursue a connection with the representable multicategories of Hermida. Unlike preceding 2-categorical type theories, in which products and exponentials are encoded by postulating a unit and counit satisfying the triangle laws, the universal properties for products and exponentials are encoded using T. Fiore's biuniversal arrows.

Because the type theory is extracted from the construction of a free biclone, its syntactic model satisfies a suitable 2-dimensional freeness universal property generalising the classical Curry–Howard–Lambek correspondence. One may therefore describe the type theory as an ‘internal language’. The relationship with the classical situation is made precise by a result establishing that the type theory I construct is the simply-typed lambda calculus up to isomorphism.

This relationship is exploited for the proof of local coherence. It is has been known for some time that one may use the normalisation-by-evaluation strategy to prove the simply-typed lambda calculus is strongly normalising. Using a bicategorical treatment of M. Fiore’s categorical analysis of normalisation-by-evaluation, I prove a normalisation result which entails the coherence theorem for cartesian closed bicategories. In contrast to previous coherence results for bicategories, the argument does not rely on the theory of rewriting or strictify using the Yoneda embedding. I prove bicategorical generalisations of a series of well-established category-theoretic results, present a notion of *glueing of bicategories*, and bicategorify the folklore result providing sufficient conditions for a glueing category to be cartesian closed. Once these prerequisites have been met, the argument is remarkably similar to that in the categorical setting.

Declaration

This dissertation is the result of my own work and includes nothing which is the outcome of work done in collaboration except as declared in the Preface and specified in the text. It is not substantially the same as any that I have submitted, or am concurrently submitting, for a degree or diploma or other qualification at the University of Cambridge or any other University or similar institution except as declared in the Preface and specified in the text. I further state that no substantial part of my dissertation has already been submitted, or is being concurrently submitted, for any such degree, diploma or other qualification at the University of Cambridge or any other University or similar institution except as declared in the Preface and specified in the text. This dissertation does not exceed the prescribed limit of 60 000 words.

Acknowledgements

First and foremost, I have to thank my supervisor Marcelo Fiore. Besides the many hours of technical discussions, I owe an intellectual debt to his precise and thoughtful approach to problems. I am particularly grateful for the patient way he dealt with my (sometimes egregious) errors. I also owe thanks to Martin Hyland and Steve Awodey for examining this thesis, and to André Joyal for suggesting Power's coherence proof for bicategories with finite bilimits could be adapted to cc-bicategories.

Thank you to the fellow occupants of FE14 and to the students of the PLS group, with whom I enjoyed many lunches and pub trips. Thank you especially to Ian Orton, Dylan McDermott, Hugo Paquet, Matthew Daggitt, and Michael Schaarschmidt, who all put up with me for three years, and to Alex Hickey for many chats over afternoon tea. Ian discovered a bug in an early version of the type theory that forms the first part of this thesis which, despite by panic at the time, greatly improved the end result. As with so many of my technical problems over the last few years, I was lucky to have Ian, Hugo and Dylan patiently spending their time to sanity-check my ideas and explain the basic concepts I was missing. Thank you also to Ohad Kammar for his forbearance and financial support as my intended submission date slipped back and back.

I seem to have spent most of my time in Cambridge either doing mathematics or rowing. Sidney Sussex Boat Club was my pressure release valve and a source of great friendships, and I am incredibly grateful to everyone who made the club special. A special shout-out goes to 'my' Lents 2019 crew.

I cannot do justice to the friends and family who have helped me over the last four years: to all of you, thank you.

Finally, and most importantly, thank you to Aijing Wang for her love and care over the last four years. I wouldn't have done it without you.

Lay introduction

This introduction is for the friends and family who have occasionally asked *what it is I actually do*, and to whom I don't think I've ever managed a satisfactory answer. I hope this goes some way to explaining what the next 200-odd pages are about.

Here's the three-sentence explanation. This thesis is about using *category theory* and *type theory* together to prove a *coherence theorem*. I construct a type theory—a kind of mathematical language—to describe a category-theoretic structure which turns up in algebra and logic. Then, by proving a property of the type theory, I deduce the category-theoretic structure has a property called *coherence*.

Let's flesh that out a bit more. Part I of the thesis is about *syntax*, while Part II is about *semantics*. The distinction between the two is one we are used to in our day-to-day lives. If you read a message from me and judge me for spelling 'life' as 'liffe', you are judging the syntax: the string of symbols that make up the message. If you nonetheless grasped what I meant by the whole phrase 'what have I been doing with my liffe', you understood the semantics: the meaning I was trying to convey. When a translator translates a sentence from English to Mandarin, they change the syntax (from Roman letters to Chinese characters), but maintain the semantics: a Chinese reader should finish the Chinese sentence understanding the same thing as an English reader who has just read the English sentence.

The syntactic-semantic distinction is central to the study of programs and programming languages. On the syntactic side, there is the literal string of characters making up a program. If I write `print`('hello world'), the computer has to break this up into the command (`print`) and the string that I'm telling it to print (*hello world*), and act accordingly. If I write $((3 + 6) \times 7)^2$, it has to break it up into the series of instructions

1. Add 3 to 6, *then*
2. Multiply the result by 7, *then*
3. Multiply this result by itself.

Anyone who has sat down to write a program will know that a fair amount of time is spent chasing down the little syntactic mistakes (such as missing a crucial ';') that, as far as the computer is concerned, make what you have written unreadable.

Comparing programs only by their syntax is not very helpful, however. Here are three different programs that take in a number x and give back another number:

$$\frac{(\frac{x}{2}+5)\times 6}{3} \qquad (\frac{x}{2}+5)\times 2 \qquad x+10 \tag{1}$$

The string of symbols in each case is different, so syntactically they are different programs. But, as we learn in secondary school algebra, these all mean the same thing: they evaluate to the same answer. Intuitively, we can think of all these programs as the same. From the programmer's perspective, writing any one of these is as good as the other. So if the computer transforms between them (for example, because one of them is quicker to run), then the programmer doesn't care. But if the computer transforms one of these programs into $x + 1$, then they most certainly will.

This suggests that we should study programming languages not just by thinking about the syntax, but by making precise our intuitive idea of what a program 'says'. First we provide a mathematical description of what each part of a program means. For example, the command `add(2)(3)` 'means' $2 + 3$. Then we say that two programs are the same if they have the same mathematical description. The idea is that the mathematics captures the meaning of the program (its semantics), and allows us to abstract away from its syntax. We can then prove all kinds of useful guarantees. For example, we can show that every syntactically correct program will eventually stop, and that the answer it will give is the one you would expect.

What does this have to do with category theory, type theory, or coherence? It turns out that type theory can be thought of as the logic of programs, and that category theory is one of the best ways of describing what these programs mean.

Type theory grew up in the early 20th century in response to problems in logic, most famously Russell's paradox. One formulation of the paradox is this. Imagine you are a very organised person, and are constantly making lists: to-do lists, shopping lists, and so on. But one day you worry that you might be missing something, so you sit down to enumerate all the things that do not appear on any of your lists. Do you add *this list* to this new list? If you do, it appears on a list, so shouldn't be on the list. If you don't, it doesn't appear on any list, so should be on the list. It seems neither choice is correct! The solution suggested by Russell is to stratify objects: at the first level are things that may appear in a list (things you need to do, food you need to buy), at the second level are lists of things in the first level, at the third level are lists of things at the second level, and so on. Every list has a level, and a list can only contain things at lower levels, so you never encounter the question of whether a list must contain the entry *this list*.

This kind of logic is governed by the principle that everything has a type, and a thing's type determines how it can behave. So you have a type of *things that go in lists*, a type of *lists of things that go in lists*, a type of lists of these lists, and so on. Similarly, you might have a type `nat` of natural (counting) numbers, and the numbers $0, 1, \ldots$ all have type `nat`. From this point of view, the expression $0 = 1$ is false, but expressions like $\frac{2}{0}$ or $\texttt{print} + 2$ are ruled to be nonsense: the language of type theory simply doesn't allow you to form such expressions. With enough types and enough ways of forming new types, one can go a long way to formulating all of mathematics in a type theory.

This way of thinking has been absorbed into computer science as a way of structuring programs. When a programmer sits down to write a program, they have in mind some kind of input (say, a list of numbers) and an output (say, the highest number in the list). One can therefore think of a program as something that takes in something of some type, and gives out something of another type. For example, I can tell the computer that I want it to treat `add(2)(3)` as something of type `int`—as a whole number, obtained by adding 2 to 3—or as something of type `string`—as a list of nine characters that happen to look like a command to add two numbers. If I declare `add(2)(3)` to be of type `string`, I can't treat it as a number: I can ask for its length (9), but can't multiply it by two. The more types you have, and the more constructions for new types you allow, the more precise you can make these restrictions.

Type theory, then, can be viewed in two ways. As a kind of logic, in which every true or false statement is attached to a type. Or as a programming language, in which the statements I can write down correspond to programs with a set input type and a set output type.

Thinking of programs as processes which take an input and return an output helps clarify the connection with category theory. Category theorists are mathematicians who truly believe that *it's not about the destination, it's about the journey.* Instead of asking about particular objects, category theorists study the way things are related. The diagrams that you'll see if you flick through this thesis say exactly this: if you walk around the diagram following the arrows in one direction, and then walk around the diagram following the arrows in the other direction, the two walks will be equal. The fundamental idea is that, if I know all the ways to get into an object, and all the ways to get out of it, then I can discover everything I need to know. More than this: I can discover other, seemingly unrelated, objects that are related to the things around them in the same way. For example, the 'if ... then' construction of logic, the collection of ways to assign an object of a set B to every object of a set A, and the notion of group from algebra—which axiomatises the ways of rotating and reflecting shapes like triangles, squares, and cubes—are all examples of the same categorical construction.

The categorical perspective has unearthed unexpected relationships between geometry, algebra, and logic, but it also plays an important role as a mathematical description for programming languages: category theory is the *semantics* for the *syntax* of type theory. For a type theorist, a program is a particular way of constructing objects of a certain type. For the category theorist, this is exactly a way of getting from one object (the input type) to another (the output type). Type theory and category theory are intertwined: by carefully choosing our categories, we can provide constructions that correspond exactly to the allowed type-theoretic expressions. By studying these categories, we can learn about type theory; by studying type theories, we can learn about their corresponding categories. Broadly speaking, this is the what I do in this thesis: I construct a type theory, show it corresponds to a special class of categories, and then—by proving something about the type theory—solve a problem about the class of categories.

The problem is called *coherence.* The special categories I work with—the 'cartesian closed bicategories' of the title—have uses in other areas of category theory, as well as in algebra and in the study of programming languages, but they are intricate. As well as the ways of getting from A to B, they include the routes between these routes. Imagine A and B are Cambridge and Oxford. Then the routes between them might be walking directions for the various routes, and the routes-between-routes might be the ways you can change one set of directions into the other: change 'left' for 'right' at this junction, replace '100 yards' with '2 miles', and so on. Or you can imagine studying programs, and the ways of transforming them stage-by-stage into something that you can run in 0s and 1s on your hardware. In this example, you might have two programs with the same input type and the same output type—such as those in (1) above—and think about the ways of transforming one into another: replacing $\frac{y\times 6}{3}$ by $y \times 2$, and $\frac{x}{2} \times 2$ by just x, and so on.

Precisely describing these two levels, and the ways they must interact, requires many axioms and many checks at every stage of a calculation. This quickly becomes tedious, and leads to proofs that are so long it is hard to check they are correct, let alone fit them onto a page so that they can be verified by the community. In this thesis I show that cartesian closed bicategories have the property that any equation you can write down for any cartesian closed bicategory (not relying on any special properties of a specific one) must hold. This means that those long tedious calculations are dramatically simplified: all those things that you had to check before are now guaranteed to hold by the theorem.

In Part I, then, I construct a type theory for describing cartesian closed bicategories. If a type theory is a logic for programs, this is a logic for programs *and ways of transforming programs into one another.* I show that expressions in this type theory correspond exactly to data in any cartesian closed bicategory, so that a proof about the type theory is a proof about every cartesian bicategory. Then, in Part II, I prove a property of the type theory that guarantees that every cartesian closed bicategory is coherent. If you want to see what it all looks like, the type theory is in Appendix C, and the big theorem is Theorem 8.4.6.

Contents

Chapter 1

Introduction

The Curry–Howard–Lambek correspondence and beyond

The simply-typed lambda calculus lives a remarkable double life. It can be seen as a term calculus for intuitionistic logic, or as the syntax of cartesian closed categories—a class of algebraic structures encompassing many important examples. This two-fold relationship, known as the *Curry–Howard–Lambek correspondence*, is fundamental to the study of logic, type theory, and programming language theory.

In this thesis we are largely concerned with the relationship between type theory and category theory. In the context of the simply-typed lambda calculus the crucial observation is due to Lambek [Lam80, Lam86], who showed that the simply-typed lambda calculus may be interpreted in any cartesian closed category, that any cartesian closed category gives rise to a simply-typed lambda calculus, and moreover that these two operations are—in a suitable sense—mutually inverse. For a computer scientist, this says that cartesian closed categories capture the meaning, or *semantics*, of the simply-typed lambda calculus: to give a model of the simply-typed lambda calculus is to give a cartesian closed category. For a category theorist, this says that one may use the simply-typed lambda calculus as a convenient syntax or *internal language* for constructing proofs in cartesian closed categories.

The simply-typed lambda calculus is just the starting point. Internal languages are a key tool in topos theory [MR77, Joh02], and there are well-known versions of Lambek's correspondence for linear logic [BBdPH93] (see *e.g.* [Mel09] for an overview) and Martin-Löf type theory [See84, CD14]. Meanwhile, categorical constructions such as monads have become standard for semantic descriptions of so-called 'effectful programs', which display behaviours beyond merely computing some result [Mog89, Mog91].

Latent within each of these developments is the notion of *reduction* or *rewriting*. In a Lambek-style semantics one begins with a type theory together with rules specifying how terms reduce to one another. These reduction rules generate an equational theory, and one identifies terms modulo this theory with morphisms in a suitable category. This is generally sufficient for type-theoretic applications, despite the loss of intensional information. To study the behaviour of reductions, however, this information must be retained.

One way to retain this information is through *2-categories*. A 2-category consists of objects, morphisms, and *2-cells* relating morphisms, subject to the usual unit and associativity laws. In the late 1980s multiple authors suggested 2-categories as a semantics for rewriting (*e.g.* [RS87, Pow89a]). In particular, Seely [See87] sketched a connection between 2-categories equipped with a (lax) cartesian closed structure and the $\beta\eta$-rewriting rules of the simply-typed lambda calculus. In this model, η-expansion and β-reduction form the unit and counit of the adjunction defining 2-categorical cartesian closed structure. Hilken [Hil96] then took the identification between cartesian closed 2-categories and the rewriting theory of the simply-typed lambda calculus a step further by introducing a '2λ-calculus' consisting of types, terms, and *rewrites* between terms. Syntactically, rewrites model reduction rules—for example, the $\beta\eta$-rules of the simply-typed lambda calculus—while semantically they play the role of 2-cells.

Since Hilken's work, *2-dimensional type theories* consisting of types, terms and rewrites have been employed for a range of applications, from rewriting theory [Hir13] to the study of Martin-Löf type theory and its connections to homotopy theory and higher category theory (*e.g.* [Gar09, LH11, LH12]). In this thesis I also connect 2-dimensional type theory to higher category theory, but with different aims. Here, the focus is on a class of higher categories of recent importance for applications in logic [FGHW07, GJ17, Oli20], the semantics of programming languages [Paq20], and the study of category theory itself [FJ15, Fio16] known as *cartesian closed bicategories*. The copious data required to define a cartesian closed bicategory makes calculations within them a demanding undertaking: the aim of this thesis is to drastically reduce those demands.

'The technical nightmares of bicategories'

Suppose given a pair of spans $(A \leftarrow B \rightarrow C)$ and $(C \leftarrow D \rightarrow E)$ in a category with finite limits. By analogy with the category of sets, these could be thought of as 'relations' $A \rightsquigarrow C$ and $C \rightsquigarrow E$. How should the composite $A \rightsquigarrow E$ be defined? A natural suggestion is to take the pullback of $(B \rightarrow C \leftarrow D)$ and use the associated projection maps, thus:

$$\begin{tikzcd}
& & B \times_C D \arrow[dl] \arrow[dr] \arrow[dd, phantom, "\lrcorner", very near start] & & \\
& B \arrow[dl] \arrow[dr] & & D \arrow[dl] \arrow[dr] & \\
A & & C & & E
\end{tikzcd}$$

Because limits are only unique up to unique isomorphism, this definition does not satisfy the unit and associativity laws of a 2-category. However, such laws do hold up to specified isomorphism, and these isomorphisms satisfy coherence axioms. The resulting structure is called a *bicategory*. Bicategories are rife in mathematics and theoretical computer science, arising for instance in algebra [Bén67, Str95], semantics of computation [GFW98, CCRW17], datatype models [Abb03, DM13], categorical logic [FGHW07, GK13], and categorical algebra [FJ15, GJ17, FGHW17]. More generally, one may (loosely) consider *weak n-categories* to have k-cells relating $(k-1)$-cells for $k = 1, \dots, n$, such that the coherence axioms for k-cells are themselves witnessed by a specified $(k+1)$-cell.

Weak higher category theory entails layers of complexity that do not exist at the 1-categorical level. Morphisms (more generally, k-cells) satisfying axioms up to some higher cell may exist in new relationships; specifying their behaviour leads to intimidating lists of axioms, for which the intuitive content is not immediately obvious. Proofs become purgatorial exercises in drawing pasting diagram after pasting diagram, or diagram chases in which an intuitively-clear kernel is dominated by endless structural isomorphisms shifting data back and forth. Even at the level $k = 2$, Lack—certainly a member of the higher-categorical *cognoscenti*—refers to (strict) 2-category theory as a “middle way”, avoiding “some of the technical nightmares of bicategories” [Lac10].

A small example highlights how the step from categories to bicategories blows up the length of a proof. Consider the following lemma, which is an elementary exercise in working with cartesian closed categories.

Lemma 1.1.

1. Every object X in a category with finite products $(\mathbb{C}, \times, 1)$ has a canonical structure as a commutative comonoid, namely $\left(1 \xleftarrow{!} X \xrightarrow{\Delta} X \times X\right)$.
2. Every endo-exponential $[X \Rightarrow X]$ in a cartesian closed category $(\mathbb{C}, \times, 1, \Rightarrow)$ has a canonical structure as a monoid, namely

$$1 \xrightarrow{\mathrm{Id}_X} [X \Rightarrow X] \xleftarrow{\circ} [X \Rightarrow X] \times [X \Rightarrow X]$$

□

Following the principle that higher categories behave in roughly the same manner as 1-categories so long as care is taken to specify the behaviour of the higher cells, one expects a version of this result to hold for cartesian closed bicategories. The bicategorical notion of monoid is called a *pseudomonoid* [DS97]. In a bicategory $\mathcal{B}$ with finite products $(\times, 1)$, this is a structure $(1 \xrightarrow{e} M \xleftarrow{m} M \times M)$ equipped with invertible 2-cells α, λ and ρ witnessing the categorical unit and associativity laws:

$$\begin{tikzcd}
1 \times M \arrow[r, "e \times M"] \arrow[dr, bend right, "\simeq"'] & M \times M \arrow[d, "m"] \arrow[dl, phantom, "\overset{\lambda}{\cong}"] \arrow[dr, phantom, "\overset{\rho}{\cong}"] & M \times 1 \arrow[l, "M \times e"'] \arrow[dl, bend left, "\simeq"] \\
& M &
\end{tikzcd}
\qquad
\begin{tikzcd}
(M \times M) \times M \arrow[r, "\simeq"] \arrow[d, "m \times M"'] & M \times (M \times M) \arrow[r, "M \times m"] \arrow[d, phantom, "\overset{\alpha}{\cong}"] & M \times M \arrow[d, "m"] \\
M \times M \arrow[rr, "m"'] & & M
\end{tikzcd}$$

These 2-cells are required to satisfy two coherence laws, corresponding to the triangle and pentagon axioms for a monoidal category. Indeed, the prototypical example—obtained by instantiating the definition in **Cat**—is of monoidal categories. Comparing with our categorical lemma suggests the following.

Conjecture 1.2.

1. Every object X in a bicategory with finite products $(\mathcal{B}, \times, 1)$ has a canonical structure as a commutative pseudocomonoid, with 1-dimensional structure $\left(1 \xleftarrow{!} X \xrightarrow{\Delta} X \times X\right)$.
2. Every endo-exponential $[X \Rightarrow X]$ in a cartesian closed bicategory $(\mathcal{B}, \times, 1, \Rightarrow)$ has a canonical structure as a pseudomonoid, with 1-dimensional structure

$$1 \xrightarrow{\mathrm{Id}_X} [X \Rightarrow X] \xleftarrow{\circ} [X \Rightarrow X] \times [X \Rightarrow X]$$

Moreover, in each case the 2-cells witnessing the 1-categorical axioms are canonical choices arising from the cartesian (closed) structure of $\mathcal{B}$. ◂

Constructing the witnessing 2-cells α, λ and ρ is relatively straightforward: roughly speaking, one can translate each equality used in the categorical proof into a 2-cell, and then compose these together. The difficulty arises in checking the coherence laws, which entails a series of long diagram chases unfolding the properties of these composites. It is this extra work that makes bicategorical calculations more burdensome than their strict counterparts: it is not enough to merely witness the axioms—which corresponds to checking them in a strict setting—one must also check the witnesses are themselves *coherent*.

Not only do these checks entail extra work, they are often extremely tedious. Generally one does not have to apply clever tricks or techniques, only plough through diagram chases until the result falls out. This is the case, for example, when one sits down to verify the coherence laws for Conjecture 1.2. This leads to a false sense of security: it is tempting to believe that the coherence axioms 'must' work out as expected, and that these extra checks may be omitted. As Power put it as long ago as 1989 [Pow89b]:

> The verification is almost always routine, and one's intuition is almost always vindicated; but to check the detail is often a very tedious job. Of course, one should still do it. . . [ignoring such details] can be dangerous, as illustrated in [Bén85], because on rare occasions, one's intuition fails. . .

Despite these difficulties, higher categories—either as ∞-categories or as bicategories and tricategories—present an intuitively appealing and technically rich setting for studying phenomena arising throughout mathematics and theoretical computer science. Examples arise in topology [Lei04], categorical logic [FGHW07], categorical algebra [Bén67], semantics of computation [CFW98], and datatype semantics [Abb03], to name but a few. The success of the 'Australian school' of the 1970s and 1980s highlights especially the fruitfulness of studying categorical constructions in the bicategorical setting (*e.g.* [Str72, Str80, BKP89]).

One is, therefore, caught between interest and difficulty: one wants to be able to work in higher categories, but the technicalities of doing so are formidable. And the squeeze only becomes tighter as the structure becomes richer. The question then becomes: how can one construct a way out?

Coherence laws and coherence theorems

One solution to the difficulties of working in a higher category is to develop a formal calculus that provides a pragmatic language for constructing and presenting proofs. In recent years there has been a great deal of work along these lines (*e.g.* [RS17, CHTM19, Shu19]), generally motivated by applications to ∞-categories (although not always, see *e.g.* [Fre19]). Much of the impetus stems from the connections between type theory, homotopy theory, and ∞-categories (*e.g.* [Gar09, LH11]), particularly the versions of Martin-Löf type theory known as *homotopy type theory* or *univalent type theory* (*e.g.* [The13]). The type theory is generally strict—allowing for simpler reasoning—but satisfies an up-to-equivalence universal property interpreting it in the weak structure in question; this is analogous to the relationship between Martin-Löf type theory with extensional identity types and locally cartesian closed categories [CD14]. A related strand of research is the development of *computer-aided* systems such as Globular [BKV18], which aim to provide interactive theorem-proving tools for certain weak n-categories.

An alternative approach is to show that the weak structure in question is (weakly) equivalent to a strict structure: the so-called *coherence* property. To paraphrase Jane Austen:

> It is a truth universally acknowledged, that a higher category in possession of a good structure, must be in want of a coherence theorem.

So long as equivalences are injective-on-cells in the appropriate sense, one can then parley this into a result proving that classes of diagrams always commute. Since Mac Lane's first coherence theorem for monoidal categories, together with its pithy slogan *all diagrams commute* [Mac63], a cottage industry has sprung up proving coherence results in various forms (notable examples include *e.g.* [MP85, Pow89b, Pow89c, JS93, GPS95]). Coherence proofs often rely on the Yoneda embedding, which allows one to embed a weak structure (such as a bicategory) into a strict structure (such as the 2-category of **Cat**-valued pseudofunctors), or on the sophisticated machinery of 2-dimensional universal algebra. Rewriting theory provides an alternative, syntactic, approach (*e.g.* [Hou07, FM18]).

However, coherence turns out to be a subtle property. Certainly, one can not always show that *all* diagrams commute: consider, for instance, the case of braided monoidal categories. In general, the dividing line between 'coherent' and 'non-coherent' definitions may not be where one would naïvely hope it to be, and the exact line can be surprising. Tricategories are not generally triequivalent to strict 3-categories [GPS95], and the tricategory **Bicat** is not triequivalent to the tricategory **Gray** of 2-categories, 2-functors, *pseudo*natural transformations and modifications [Lac07].

The difficulty, therefore, is twofold: first, to identify the boundaries between commutativity and its failure, and second, to prove that all diagrams within a conjectured boundary do in fact commute.

Coherence for cartesian closed bicategories

In this thesis I prove a coherence theorem for bicategories equipped with products and exponentials in an 'up to equivalence' fashion. As far as I am aware, these were first studied in [Mak96], and the coherence result I prove was first conjectured by Ouaknine [Oua97]. It is an unfortunate accident of terminology that there is no connection to the 'cartesian bicategories' of Carboni & Walters [CW87, CKWW08], nor to the 'closed cartesian bicategories' of Frey [Fre19]. Precisely, the theorem is the following.

Theorem. The free cartesian closed bicategory on a set of 0-cells has at most one 2-cell between any parallel pair of 1-cells. □

Note that this is a particularly concrete statement of coherence. In terms of Conjecture 1.2, it goes further than showing that, once one has constructed witnessing 2-cells such as α, λ and ρ using only the axioms of a cartesian closed bicategory, then the coherence laws will hold. The theorem also guarantees that there is a unique choice of witnessing 2-cells. Using this in tandem with a precise connection between the 2-cells of the free cartesian closed bicategory and equality in the free cartesian closed category (Section 5.4), we shall be able to show further that it suffices to calculate completely 1-categorically.

This work was initially motivated by the difficulty of proving statements such as Conjecture 1.2 and the corresponding obstruction to the development of a theory of ∞-categories [Fio16] in the cartesian closed bicategories of generalised species [FGHW07] and cartesian distributors [FJ15]. However, cartesian closed bicategories appear more widely, for example in categorical algebra [GJ17] and game semantics [YA18, Paq20].

The strategy has two parts. First, I develop a type theory $\Lambda_{\text{ps}}^{\times,\rightarrow}$ for cartesian closed bicategories and show that it satisfies a suitable 2-dimensional freeness property. This extends the classical

Curry–Howard–Lambek correspondence to the bicategorical setting. The shape of the type theory follows the tradition of 2-dimensional type theory instigated by Seely [See87] and Hilken [Hil96]. The up-to-isomorphism nature of bicategorical composition is captured through an *explicit substitution* operation (*c.f.* [ACCL90]). Second, I adapt the *normalisation-by-evaluation* technique introduced by Berger & Schwichtenberg [BS91] for proving normalisation of the simply-typed lambda calculus to extract the theorem above. Here I closely follow Fiore's categorical treatment of the proof [Fio02].

Of course, for $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$ to be a type theory for cartesian closed bicategories, one must impose some constraints. I stipulate the following three desiderata.

> **Internal language.** The syntactic model of the type theory must be free, in an appropriately bicategorical sense. From a logical perspective, this corresponds to a soundness and completeness property. We shall not go so far as, say, constructing a triadjunction between a tricategory of signatures and the tricategory of cartesian closed bicategories. Instead, we prove strict universal properties (*c.f.* [Gur06]) wherever possible. As well as being readily verifiable, these properties are often easier to work with.
>
> **Relationship to STLC.** The type theory we construct must have the 'flavour' of type theory. In particular, one should be able to recover the simply-typed lambda calculus (STLC) as some kind of fragment: following the intuition that cartesian closed bicategories are cartesian closed categories up-to-isomorphism, a corresponding property should relate the simply-typed lambda calculus to $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$. This also imposes restrictions on the form of judgements and derivations: they should be presented in a style recognisable as type theory.
>
> **Usability.** This is connected to the preceding point. There is no gain in constructing a syntactic calculus that merely re-phrases the axioms of a cartesian closed bicategory. Instead, the type theory ought to be a reasonable tool for constructing proofs. Its equational theory ought to be kept small, and express requirements that are natural from the semantic perspective.

These desiderata are not merely stylistic: they will play a key part in our eventual proof of coherence. The precise correspondence with the simply-typed lambda calculus, for example, will allow us to leverage the categorical arguments of [Fio02] in a particularly direct way. Moreover, they should also make the type theory amenable to deep embedding in proof assistants such as Agda [Agd], and to extension with further structure in future work.

Outline

The thesis is in two parts. Part I is devoted to the construction of $\Lambda_{\text{ps}}^{\times,\to}$ and a proof of its free property. Part II covers the normalisation-by-evaluation proof.

In Chapter 2 I present an overview of the basic theory of bicategories. Much of the theory is well-known, but I take the opportunity to develop it with a focus on T. Fiore's *biuniversal arrows* [Fio06, Chapter 9]. This bicategorification of universal arrows encompasses both biadjunctions and bilimits, and is particularly amenable to being translated into type theory.

Chapter 3 constructs the core part of $\Lambda_{\text{ps}}^{\times,\to}$, namely a type theory for mere bicategories. This type theory is synthesised from an algebraic description of bicategorical substitution, called a *biclone*, which generalises the *abstract clones* of universal algebra (*e.g.* [Coh81, Plo94]). We also establish a coherence theorem for this fragment of the type theory, generalising the Mac Lane-Paré coherence theorem for bicategories [MP85].

In Chapter 4 we extend the type theory with finite products. We pursue a connection between the *representable multicategories* of Hermida [Her00], introducing the notion of *representable (bi)clone* and showing that it coincides with a notion of (bi)clone with cartesian structure. Thereafter we synthesise a type theory from the free such biclone, and show that its syntactic model is free.

Chapter 5 follows a similar pattern: we define cartesian closed biclones and extract a type theory from the construction of the free such. Establishing the free property for cc-bicategories throws up more complications than the preceding two chapters, so we spend some time over this. Thereafter we establish that the simply-typed lambda calculus embeds into $\Lambda_{\text{ps}}^{\times,\to}$ and that, modulo the existence of invertible rewrites (2-cells), this restricts to a bijection on $\beta\eta$-equivalence classes of terms. We also observe that Power's coherence theorem for bicategories with flexible bilimits [Pow89b] may be adapted to the case of cc-bicategories (Proposition 5.1.10).

In each of Chapters 3–5, the development is motivated by the construction of a version of the following diagram. This provides a technical statement of the intuitive fact that, in order to construct a type theory for cartesian or cartesian closed (bi)categories, it suffices to construct a type theory for the corresponding (bi)clones. As a slogan: *(bi)clones are the right intermediary between syntax and semantics.*

We then move to the normalisation-by-evaluation proof. In Chapter 6 we prove bicategorical correlates of three well-known facts about presheaf categories, namely:

1. Every presheaf category is complete,

2. Every presheaf category is cartesian closed,

3. For any presheaf P and representable presheaf $\mathrm{y}(X)$ on a small category with binary products, the exponential $[\mathrm{y}X, P]$ is, up to isomorphism, the presheaf $P(- \times X)$.

$$\begin{tikzcd}
& \substack{\text{structured (bi)clones} \\ \text{many-in one-out morphisms}} \arrow[dl, bend left] \arrow[dr, bend left, "\text{restriction}"] & \\
\text{signatures} \arrow[ur, bend left, "\text{free}"] \arrow[ur, phantom, "\top"] \arrow[dr, bend left, "\text{restriction}"] \arrow[dr, phantom, "\perp"] & & \substack{\text{structured (bi)categories} \\ \text{one-in one-out morphisms}} \arrow[ul, bend left] \arrow[ul, phantom, "\top"] \arrow[dl, bend left] \arrow[dl, phantom, "\perp"] \\
& \text{unary signatures} \arrow[ul, bend left, "\text{inclusion}"] \arrow[ur, bend left, "\text{free}"] &
\end{tikzcd}$$

The reader willing to believe versions of these results for every 2-category Hom($\mathcal{B}$, **Cat**) of **Cat**-valued pseudofunctors may safely skip this chapter.

Chapter 7 introduces the notion of *glueing of bicategories* and establishes mild conditions for the glueing bicategory to be cartesian closed. In the 1-categorical setting, this implies the so-called *fundamental lemma* of logical relations [Plo73, Sta85].

In Chapter 8 we complete the proof of the main result via a bicategorical adaptation of Fiore's [Fio02]. Much of the apparatus required is contained in the preceding two chapters. Finally, Chapter 9 briefly lays out some applications and suggestions for further work.

Appendices A–C contain an index of the bicategorical free constructions and syntactic models throughout this thesis, an overview of the cartesian closed structures we construct, and the complete set of rules for $\Lambda_{\mathrm{ps}}^{\times,\to}$ together with their semantic interpretation.

Previous publication. The type theory $\Lambda_{\mathrm{ps}}^{\times,\to}$ was presented in the paper *A type theory for cartesian closed bicategories* [FS19]. This is available online at `https://ieeexplore.ieee.org/document/8785708`.

Contributions

The most obvious contribution is the coherence theorem for cartesian closed bicategories. In fact, we prove this in three different ways: two closely-related arguments using the Yoneda lemma (Proposition 5.1.10 and Theorem 8.5.2) and the third by normalisation-by-evaluation (Theorem 8.4.6). In each case the strategy is of interest in its own right. The arguments from the Yoneda argument extend Power's coherence argument for bicategories with flexible bilimits [Pow89b] to closed structure for the first time. On the other hand, the normalisation-by-evaluation argument shows potential for further development. First, it is plausible that, by further refining the normalisation-by-evaluation one would be able to extract a normalisation algorithm computing the canonical 2-cell between any given 1-cells in the free cartesian closed bicategory. Second, the combination of syntactic and semantic methods employed here is a novel approach to proving higher-categorical coherence theorems (although Licata & Harper have gone some way in this direction, using a groupoidal model to prove canonicity for their 2-dimensional type theory [LH12]). This approach may extend to situations where other proofs of coherence—employing either syntactic approaches or the apparatus of 2-dimensional universal algebra—are less successful.

From the type-theoretic perspective, I believe the view taken here—namely, that the appropriate mediator between syntax and semantics is some version of abstract clones—is a fruitful one. Indeed, the definition of the type theory $\Lambda_{\mathrm{ps}}^{\times,\to}$ follows automatically from the definition of cartesian closed biclones. As far as I am aware, this is the first attempt to construct a type theory describing higher categories from such universal-algebraic grounds, and the first to exploit the machinery of explicit substitution (although Curien's diagrammatic calculus for locally cartesian closed categories shows similar ideas [Cur93]).

The theoretical development required for the normalisation proof—such as the work on bicategorical glueing in Chapter 7—lays important foundations for further work. For instance, the machinery of Part II is the groundwork for proving a conservative extension result for cartesian closed bicategories over bicategories with finite products in the style of [Laf87, FDCB02].

Finally, this thesis contains moderately detailed proofs of results that one would certainly *expect* but I have not seen *proved* in the literature, such as the cartesian closure of the 2-category $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$ of **Cat**-valued pseudofunctors, pseudonatural transformations and modifications. At the very least, I hope this saves others the work of reproducing the extensive calculations required.

Notation and prerequisites

I have tried to keep the presentation self-contained and accessible to type theorists with a categorical bent, as well as to (higher) category theorists with less experience in type theory. I recap the bicategory theory we shall need, and do not employ any heavyweight results without proof. Similarly, I take the simply-typed lambda calculus and its semantics (as in *e.g.* [LS86, Cro94]) as known, but do not assume familiarity with strategies such as glueing or normalisation-by-evaluation. This occasionally requires recapitulating folklore or standard results, but I hope in these cases the presentation is original enough to be of interest in itself.

I have attempted to generally (but not universally) maintain the following typographical conventions:

- Named 1-categories are written in Roman font (*e.g.* Set); named higher categories are in **bold** font (*e.g.* **Cat**). Arbitrary categories are written in blackboard bold ($\mathbb{C}, \mathbb{D}, \dots$) and arbitrary bicategories in calligraphic font ($\mathcal{B}, \mathcal{C}, \dots$).

- 2-cells are denoted either by lower-case Greek letters ($\alpha, \beta, \tau, \sigma, \dots$) or given suggestive names in sans-serif (*e.g.* push).

An index of notation covering most of the recurring 1- and 2-cells is on page 330.

I have also borrowed the convention of Troelstra & Schwichtenberg [TS00] for denoting the end of environments. The end of a proof is marked by a white square (□) and the end of a remark, definition or example by a black triangle (◂).

Chapter 2

Bicategories, bilimits and biadjunctions

This chapter introduces the basic theory of bicategories, bilimits and biadjoints. Much of the content is well-known, and many excellent overviews of the material are available (*e.g.* [Bén67, Str80, Bor94, Str95, Lei04]). The intention behind recapitulating it here is two-fold. Firstly, to fix notation. Second, to introduce concepts in a style that is convenient for later chapters. There are many equivalent ways of formulating basic notions such as adjunction, adjoint equivalence and universal arrow. In the categorical setting, translating between the various formulations is generally straightforward. Bicategorically, however, such translations can require extensive checking of coherence data. We avoid this by taking the most convenient definition for our purposes as primitive, and by focussing on the *biuniversal arrows* of [Fio06, Chapter 9]. These capture both bicategorical limits and adjunctions—and thereby cartesian closed structure—in a uniform way. We therefore spend some time developing the theory of biuniversal arrows before showing how it specialises to standard results about bilimits and biadjunctions.

2.1 Bicategories

The fundamental notion is that of a *bicategory*, due to Bénabou [Bén67]. These structures often arise when one defines composition by a universal property. Such an operation will generally not be associative and unital up to equality, only up to some mediating isomorphisms. A classical example is the bicategory of spans over a category $\mathbb{C}$ with pullbacks. The objects are those of $\mathbb{C}$, the morphisms $A \rightsquigarrow B$ are spans $A \xleftarrow{f} X \xrightarrow{g} B$, and composition is given by pullback.

Definition 2.1.1. A *bicategory* $\mathcal{B}$ consists of

- A class of objects $ob(\mathcal{B})$,
- For every $X, Y \in ob(\mathcal{B})$ a *hom-category* $\big(\mathcal{B}(X,Y), \bullet, \mathrm{id}\big)$ with objects *1-cells* $f : X \to Y$ and morphisms *2-cells* $\alpha : f \Rightarrow f' : X \to Y$; composition of 2-cells is called *vertical composition*,
- For every $X, Y, Z \in ob(\mathcal{B})$ an *identity* functor $\mathrm{Id}_X : \mathbb{1} \to \mathcal{B}(X,X)$ (for $\mathbb{1}$ the terminal category) and a *horizontal composition* functor $\circ_{X,Y,Z} : \mathcal{B}(Y,Z) \times \mathcal{B}(X,Y) \to \mathcal{B}(X,Z)$,
- Invertible 2-cells

$$\begin{aligned} \mathsf{a}_{h,g,f} &: (h \circ g) \circ f \Rightarrow h \circ (g \circ f) : W \to Z \\ \mathsf{l}_f &: \mathrm{Id}_X \circ f \Rightarrow f : W \to X \\ \mathsf{r}_g &: g \circ \mathrm{Id}_X \Rightarrow g : X \to Y \end{aligned}$$

for every $f : W \to X$, $g : X \to Y$ and $h : Y \to Z$, natural in each of their arguments and satisfying a *triangle law* and a *pentagon law* analogous to those for monoidal categories:

$$\begin{tikzcd}
\big((k \circ h) \circ g\big) \circ f \arrow[rr, "\mathsf{a}_{k,h,g} \circ f"] \arrow[d, "\mathsf{a}_{k \circ h,g,f}"'] & & \big(k \circ (h \circ g)\big) \circ f \arrow[d, "\mathsf{a}_{k,h \circ g,f}"] \\
(k \circ h) \circ (g \circ f) \arrow[dr, "\mathsf{a}_{k,h,g \circ f}"'] & & k \circ \big((h \circ g) \circ f\big) \arrow[dl, "k \circ \mathsf{a}_{h,g,f}"] \\
& k \circ \big(h \circ (g \circ f)\big) &
\end{tikzcd}$$

$$\begin{tikzcd}
(g \circ \mathrm{Id}_X) \circ f \arrow[rr, "\mathsf{a}_{g,\mathrm{Id},f}"] \arrow[dr, "\mathsf{r}_g \circ f"'] & & g \circ (\mathrm{Id}_X \circ f) \arrow[dl, "g \circ \mathsf{l}_f"] \\
& g \circ f &
\end{tikzcd}$$

The functorality of horizontal composition gives rise to the so-called *interchange law*: for suitable 2-cells $\tau, \tau', \sigma, \sigma'$ we have $(\tau' \bullet \tau) \circ (\sigma' \bullet \sigma) = (\tau' \circ \sigma') \bullet (\tau \circ \sigma)$. ◂

Notation 2.1.2. In the preceding we employ the standard notation for the *whiskering* operations. For a 1-cell $f : X \to Y$ and 2-cells $\sigma : h \Rightarrow h' : W \to X$ and $\tau : g \Rightarrow g' : Y \to Z$ we write $f \circ \sigma$ and $\tau \circ f$ for $\mathrm{id}_f \circ \sigma : f \circ h \Rightarrow f \circ h'$ and $\tau \circ \mathrm{id}_f : g \circ f \Rightarrow g' \circ f$, respectively. ◂

The category Rel of sets and relations may be viewed as a *locally posetal* bicategory—*i.e.* a bicategory in which each hom-category is a poset—by stipulating that $R \leqslant S : A \to B$ if and only if aRb implies aSb for all $a \in A$ and $b \in B$. A relation $R : A \to B$ is equivalently a map $A \times B \to \{0, 1\}$. Replacing sets by categories, one obtains the bicategory **Prof**: this has objects categories, 1-cells $\mathbb{C} \twoheadrightarrow \mathbb{D}$ the functors $\mathbb{D}^{\mathrm{op}} \times \mathbb{C} \to \mathrm{Set}$, and 2-cells natural transformations. The identity on $\mathbb{C}$ is the hom-functor $\mathrm{Hom}(-, =)$, and composition is given using the universal property of a presheaf category (see *e.g.* [Bén00]).

Remark 2.1.3. The coherence theorem for monoidal categories [Mac98, Chapter VII] generalises to bicategories: any bicategory is biequivalent to a 2-category [MP85]. Loosely speaking, then, any diagram constructed from only the identity and the structural constraints $\mathsf{a}, \mathsf{l}, \mathsf{r}$ with the operations of horizontal and vertical composition must commute (see [Lei04] for a readable summary of the argument). We are therefore justified in treating a, l and r as though they were the identity, and we will sometimes denote such 2-cells merely by $\cong$. ◂

Every bicategory $\mathcal{B}$ has three duals. Following the notation of [Lac10, §1.6], these are

- $\mathcal{B}^{\mathrm{op}}$, obtained by reversing the 1-cells,
- $\mathcal{B}^{\mathrm{co}}$, obtained by reversing the 2-cells,
- $\mathcal{B}^{\mathrm{coop}}$, obtained by reversing both.

We call the first option the *opposite bicategory.* This is the only form of dual we shall employ in this thesis.

A morphism of bicategories is called a *pseudofunctor* (or *homomorphism*) [Bén67]. It is a mapping on objects, 1-cells and 2-cells that preserves horizontal composition up to isomorphism. Vertical composition is preserved strictly.

Definition 2.1.4. A *pseudofunctor* $F : \mathcal{B} \to \mathcal{C}$ between bicategories $\mathcal{B}$ and $\mathcal{C}$ consists of

- A mapping $F : ob(\mathcal{B}) \to ob(\mathcal{C})$,
- A functor $F_{X,Y} : \mathcal{B}(X, Y) \to \mathcal{C}(FX, FY)$ for every $X, Y \in ob(\mathcal{B})$,
- An invertible 2-cell $\psi_X : \mathrm{Id}_{FX} \Rightarrow F(\mathrm{Id}_X)$ for every $X \in ob(\mathcal{B})$,
- An invertible 2-cell $\phi_{f,g} : F(f) \circ F(g) \Rightarrow F(f \circ g)$ for every $g : X \to Y$ and $f : Y \to Z$, natural in f and g,

subject to two unit laws and an associativity law:

$$\begin{tikzcd}
\mathrm{Id}_{FX'} \circ Ff \arrow[r, "\psi_{X'} \circ Ff"] \arrow[d, "\mathsf{l}_{Ff}"'] & F(\mathrm{Id}_{X'}) \circ F(f) \arrow[d, "\phi_{\mathrm{Id}_{X'},f}"] \\
Ff & F(\mathrm{Id}_{X'} \circ f) \arrow[l, "F\mathsf{l}_f"]
\end{tikzcd}
\qquad
\begin{tikzcd}
Ff \circ \mathrm{Id}_{FX} \arrow[r, "F(f) \circ \psi_X"] \arrow[d, "\mathsf{r}_{Ff}"'] & F(f) \circ F(\mathrm{Id}_X) \arrow[d, "\phi_{f,\mathrm{Id}_X}"] \\
Ff & F(f \circ \mathrm{Id}_X) \arrow[l, "F\mathsf{r}_f"]
\end{tikzcd}$$

$$\begin{tikzcd}
\big(Fh \circ Fg\big) \circ Ff \arrow[r, "\mathsf{a}_{Fh,Fg,Ff}"] \arrow[d, "\phi_{h,g} \circ Ff"'] & Fh \circ \big(Fg \circ Ff\big) \arrow[r, "F(h) \circ \phi_{g,h}"] & F(h) \circ F(g \circ f) \arrow[d, "\phi_{h,g \circ f}"] \\
F(h \circ g) \circ Ff \arrow[r, "\phi_{h \circ g,f}"'] & F\big((h \circ g) \circ f\big) \arrow[r, "F\mathsf{a}_{h,g,f}"'] & F\big(h \circ (g \circ f)\big)
\end{tikzcd}$$

A pseudofunctor for which ψ and ϕ are both the identity is called *strict.* ◂

We often abuse notation by leaving ψ and ϕ implicit when denoting a pseudofunctor.

Example 2.1.5.

1. A monoidal category is equivalently a one-object bicategory; a monoidal functor is equivalently a pseudofunctor between one-object bicategories,
2. A 2-category is equivalently a bicategory in which a, l and r are all the identity. A strict pseudofunctor $F : \mathcal{B} \to \mathcal{C}$ between 2-categories $\mathcal{B}$ and $\mathcal{C}$ is equivalently a 2-functor.
3. For every *locally small* bicategory $\mathcal{B}$ (see Notation 2.1.10) and $X \in \mathcal{B}$ there exists the *Yoneda pseudofunctor* $\mathrm{Y}X : \mathcal{B} \to \mathbf{Cat}$, defined by $\mathrm{Y}X := \mathcal{B}(X, -)$. The 2-cells ϕ and ψ are structural isomorphisms. ◂

Morphisms of pseudofunctors are called *pseudonatural transformations* [Gra74]. These are 2-natural transformations (Cat-enriched natural transformations) in which every naturality square commutes up to a specified 2-cell. Morphisms of pseudonatural transformations are called *modifications* [Bén67, Str80].

Definition 2.1.6. A *pseudonatural transformation* $(\mathsf{k}, \overline{\mathsf{k}}) : F \Rightarrow G : \mathcal{B} \to \mathcal{C}$ between pseudofunctors (F, ψ^F, ϕ^F) and (G, ψ^G, ϕ^G) consists of the following data:

1. A 1-cell $\mathsf{k}_X : FX \to GX$ for every $X \in \mathcal{B}$,
2. An invertible 2-cell $\overline{\mathsf{k}}_f : \mathsf{k}_Y \circ Ff \Rightarrow Gf \circ \mathsf{k}_X : FX \to GY$ for every $f : X \to Y$ in $\mathcal{B}$, natural in f and satisfying the following unit and associativity laws for every $X \in \mathcal{B}$, $f : X' \to X''$ and $g : X \to X'$ in $\mathcal{B}$. :

$$\begin{tikzcd}
& (Gf \circ \mathsf{k}_{X'}) \circ Fg \arrow[dr, "\mathsf{a}_{Gf,\mathsf{k},Fg}"] & \\
(\mathsf{k}_{X''} \circ Ff) \circ Fg \arrow[ur, "\overline{\mathsf{k}}_f \circ Fg"] \arrow[d, "\mathsf{a}_{\mathsf{k},Ff,Fg}"'] & & Gf \circ (\mathsf{k}_{X'} \circ Fg) \arrow[d, "G(f) \circ \overline{\mathsf{k}}_g"] \\
\mathsf{k}_{X''} \circ (Ff \circ Fg) \arrow[d, "\mathsf{k}_{X''} \circ \phi^F_{f,g}"'] & & Gf \circ (Gg \circ \mathsf{k}_X) \arrow[d, "\mathsf{a}^{-1}_{Gf,Gg,\mathsf{k}}"] \\
\mathsf{k}_{X''} \circ F(f \circ g) \arrow[dr, "\overline{\mathsf{k}}_{fg}"'] & & (Gf \circ Gg) \circ \mathsf{k}_X \arrow[dl, "\phi^G_{f,g} \circ \mathsf{k}_X"] \\
& G(f \circ g) \circ \mathsf{k}_X &
\end{tikzcd}$$

$$\begin{tikzcd}
& \mathsf{k}_X \arrow[dr, "\mathsf{l}^{-1}_{\mathsf{k}}"] & \\
\mathsf{k}_X \circ \mathrm{Id}_{FX} \arrow[ur, "\mathsf{r}_{\mathsf{k}}"] \arrow[d, "\mathsf{k}_X \circ \psi^F_X"'] & & \mathrm{Id}_{GX} \circ \mathsf{k}_X \arrow[d, "\psi^G_X \circ \mathsf{k}_X"] \\
\mathsf{k}_X \circ F\mathrm{Id}_X \arrow[rr, "\overline{\mathsf{k}}_{\mathrm{Id}_X}"'] & & G\mathrm{Id}_X \circ \mathsf{k}_X
\end{tikzcd}$$

A pseudonatural transformation for which every $\overline{\mathsf{k}}_f$ is the identity is called *strict* or *2-natural.* ◂

Remark 2.1.7. Note that we orient the 2-cells of a pseudonatural transformation as in the following diagram:

$$\begin{tikzcd} FX \arrow[r, "Ff"] \arrow[d, "\mathsf{k}_X"'] \arrow[dr, phantom, "\overset{\overline{\mathsf{k}}_f}{\Leftarrow}"] & FY \arrow[d, "\mathsf{k}_Y"] \\ GX \arrow[r, "Gf"'] & GY \end{tikzcd}$$

This is the reverse of [Lei98] but follows the direction of [Bén67, Str80]. Of course, since we require each $\overline{\mathsf{k}}_f$ to be invertible, the two choices are equivalent. ◂

Definition 2.1.8. A *modification* $\Xi : (\mathsf{k}, \overline{\mathsf{k}}) \to (\mathsf{j}, \overline{\mathsf{j}})$ between pseudonatural transformations $(\mathsf{k}, \overline{\mathsf{k}}), (\mathsf{j}, \overline{\mathsf{j}}) : F \Rightarrow G : \mathcal{B} \to \mathcal{C}$ is a family of 2-cells $\Xi_X : \mathsf{k}_X \Rightarrow \mathsf{j}_X$, such that the following commutes for every $f : X \to X'$ in $\mathcal{B}$:[1]

$$\begin{tikzcd} \mathsf{k}_{X'} \circ Ff \arrow[r, "\overline{\mathsf{k}}_f"] \arrow[d, "\Xi_{X'} \circ Ff"'] & Gf \circ \mathsf{k}_X \arrow[d, "Gf \circ \Xi_X"] \\ \mathsf{j}_{X'} \circ Ff \arrow[r, "\overline{\mathsf{j}}_f"'] & Gf \circ \mathsf{j}_X \end{tikzcd}$$

◂

Example 2.1.9. For every pair of bicategories $\mathcal{B}$ and $\mathcal{C}$ there exists a bicategory $\mathrm{Hom}(\mathcal{B}, \mathcal{C})$ of pseudofunctors, pseudonatural transformations and modifications. If $\mathcal{C}$ is a 2-category, so is $\mathrm{Hom}(\mathcal{B}, \mathcal{C})$. In particular, for every bicategory $\mathcal{B}$ there exists a 2-category $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$, which one might view as a bicategorical version of the covariant presheaf category $\mathrm{Set}^{\mathbb{C}}$. Where $\mathbb{C}$ is a mere category, pseudofunctors $\mathbb{C} \to \mathbf{Cat}$ are called *indexed categories* [MP85]. ◂

Bicategories, pseudofunctors, pseudonatural transformations and modifications organise themselves into a *tricategory* (weak 3-category, see [GPS95, Gur06, Gur13]) we denote **Bicat** [GPS95].

Notation 2.1.10. A bicategory $\mathcal{B}$ (resp. pseudofunctor F) is said to be *locally* P if the property P holds for each hom-category $\mathcal{B}(X, Y)$ (resp. functor $F_{X,Y}$). In particular, a bicategory is *locally small* if every hom-category is a set, and *small* if it is locally small and its class of objects is a set. We shall use **Cat** to denote the 2-category of small categories and stipulate that, whenever we write $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$, then it is assumed that $\mathcal{B}$ is small. As usual, such issues can be avoided using technical devices such as Groethendieck universes (see *e.g.* [Shu08]). ◂

The *bicategorical Yoneda Lemma* takes the following form, due to Street [Str80].[2]

[1] Leinster [Lei04] requires both the above coherence law and that the family of 2-cells Ξ_X be natural in X; this appears to be an oversight, as neither Leinster's own [Lei98] nor Street's [Str95] mention naturality.

[2] The bicategorical Yoneda Lemma is an example of a result that one would certainly expect to hold—and is generally only ever stated in the literature—but for which the proof actually requires a significant amount of work: see [Bak] for the gory details.

Lemma 2.1.11. For any bicategory $\mathcal{B}$ and pseudofunctor $F : \mathcal{B} \to \mathbf{Cat}$, evaluating at the identity for each $B \in \mathcal{B}$ provides the components $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})\big(\mathcal{B}(B, -), F\big) \xrightarrow{\simeq} FB$ of an equivalence in $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$. Hence, the Yoneda pseudofunctor $\mathrm{Y} : \mathcal{B} \to \mathrm{Hom}(\mathcal{B}, \mathbf{Cat}) : X \mapsto \mathcal{B}(X, -)$ is locally an equivalence. □

Bicategories provide a convenient setting for abstractly describing many categorical concepts (*e.g.* [Law17]); this perspective that has been used to particular effect by the Australian school (see for instance [LS12, LS14]). The following definition is a small example of this general phenomenon.

Definition 2.1.12. Let $\mathcal{B}$ be a bicategory.

1. An *adjunction* $(A, B, f, g, \mathsf{v}, \mathsf{w})$ in $\mathcal{B}$ is a pair of objects (A, B) with arrows $f : A \leftrightarrows B : g$ and 2-cells $\mathsf{v} : \mathrm{Id}_A \Rightarrow g \circ f$ and $\mathsf{w} : f \circ g \Rightarrow \mathrm{Id}_B$ such that the bicategorical triangle laws hold (*e.g.* [Gur12]):

$$\begin{tikzcd}
f \arrow[r, "\mathsf{r}_f^{-1}"] \arrow[d, equal] & f \circ \mathrm{Id}_X \arrow[r, "f \circ \mathsf{v}"] & f \circ (g \circ f) \arrow[d, "\mathsf{a}_{f,g,f}^{-1}"] \\
f & \mathrm{Id}_Y \circ f \arrow[l, "\mathsf{l}_f"'] & (f \circ g) \circ f \arrow[l, "\mathsf{w} \circ f"']
\end{tikzcd}
\qquad
\begin{tikzcd}
g \arrow[r, "\mathsf{l}_g^{-1}"] \arrow[d, equal] & \mathrm{Id}_Y \circ g \arrow[r, "\mathsf{v} \circ g"] & (g \circ f) \circ g \arrow[d, "\mathsf{a}_{g,f,g}"] \\
g & g \circ \mathrm{Id}_X \arrow[l, "\mathsf{r}_g"'] & g \circ (f \circ g) \arrow[l, "g \circ \mathsf{w}"']
\end{tikzcd}$$

2. An *equivalence* $(A, B, f, g, \mathsf{v}, \mathsf{w})$ in $\mathcal{B}$ is a pair of objects (A, B) with arrows $f : A \leftrightarrows B : g$ and invertible 2-cells $\mathsf{v} : \mathrm{Id}_A \overset{\cong}{\Rightarrow} g \circ f$ and $\mathsf{w} : f \circ g \overset{\cong}{\Rightarrow} \mathrm{Id}_B$,

3. An *adjoint equivalence* is an adjunction that is also an equivalence.

If 1-cells f and g are part of an equivalence, we refer to g as the *pseudoinverse* of f. Pseudoinverses are unique up to invertible 2-cell. ◂

In **Cat**, an (adjoint) equivalence is exactly an (adjoint) equivalence of categories. Moreover, just as in **Cat**, every equivalence induces an adjoint equivalence with the same 1-cells (see *e.g.* [Lei98]). The appropriate notion of equivalence between bicategories is called *biequivalence* [Str80].

Definition 2.1.13. A *biequivalence* between bicategories $\mathcal{B}$ and $\mathcal{C}$ consists of pseudofunctors $F : \mathcal{B} \leftrightarrows \mathcal{C} : G$ and chosen equivalences $G \circ F \simeq \mathrm{id}_{\mathcal{B}}$ and $F \circ G \simeq \mathrm{id}_{\mathcal{C}}$ in the bicategories $\mathrm{Hom}(\mathcal{B}, \mathcal{B})$ and $\mathrm{Hom}(\mathcal{C}, \mathcal{C})$, respectively. ◂

By a result of Gurski [Gur12], one may assume without loss of generality that a biequivalence is an *adjoint biequivalence*, in which F and G also form a *biadjunction* (see Definition 2.4.1).

Notation 2.1.14. Following standard practice from **Cat**, we shall sometimes refer to a pair of arrows $f : A \leftrightarrows B : g$ as an *(adjoint) equivalence*, leaving the 2-cells implicit. When we wish to emphasise that these 2-cells are given as data, we refer to a *chosen* or *specified* equivalence.

Similarly, we may sometimes leave most of the data implicit and refer to the pseudofunctor F on its own as a biequivalence. Unlike the 1-categorical case, however, we shall always assume this biequivalence to be chosen. ◂

Example 2.1.15.

1. A biequivalence between one-object bicategories is exactly an equivalence of monoidal categories (that is, an equivalence in the 2-category **MonCat** of monoidal categories, monoidal functors and monoidal natural transformations).
2. **Prof** is biequivalent to its opposite bicategory [DS97, Section 7] (*c.f.* the fact that the category Rel is isomorphic to its opposite). ◂

Loosely speaking, an equivalence of categories relates objects that are the same up to isomorphism, and a biequivalence of bicategories relates objects that are the same up to equivalence. Indeed, since every pseudofunctor preserves (adjoint) equivalences, an (adjoint) equivalence $A \simeq B$ in a bicategory $\mathcal{B}$ induces an (adjoint) equivalence $\mathcal{B}(A, -) \simeq \mathcal{B}(B, -)$ in $\mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$ and hence an (adjoint) equivalence $\mathcal{B}(A, X) \simeq \mathcal{B}(B, X)$ for every $X \in \mathcal{B}$. One consequence is that, if the pseudofunctor $F : \mathcal{B} \to \mathcal{C}$ is a biequivalence, then

1. For every $C \in \mathcal{C}$ there exists an object $B \in \mathcal{B}$ and an equivalence $C \simeq FB$,
2. F is *locally an equivalence*: for every $B, B' \in \mathcal{B}$ the functor $F_{B,B'}$ is part of an equivalence of categories $\mathcal{B}(B, B') \simeq \mathcal{C}(FB, FB')$; in particular, every $F_{B,B'}$ is fully faithful and essentially surjective.

In the presence of the Axiom of Choice, this formulation is equivalent to the definition given above (*e.g.* [Lei04, Proposition 1.5.13]).

In the categorical setting it is elementary to check that a natural isomorphism—as an iso in a functor category—is exactly a natural transformation for which every component is invertible. The bicategorical version of this result is the following.

Lemma 2.1.16. Let $F, G : \mathcal{B} \to \mathcal{C}$ be pseudofunctors and suppose $(\mathsf{k}, \overline{\mathsf{k}}) : F \Rightarrow G$ is a pseudonatural transformation such that every $\mathsf{k}_X : FX \to GX$ is part of a specified adjoint equivalence $(\mathsf{k}_X, \mathsf{k}^\star_X, \mathsf{w}_X : \mathsf{k}^\star_X \circ \mathsf{k}_X \Rightarrow \mathrm{Id}_{FX}, \mathsf{v}_X : \mathrm{Id}_{FX} \Rightarrow \mathsf{k}_X \circ \mathsf{k}^\star_X)$. Then:

1. The family of 1-cells $\mathsf{k}^\star_X : GX \to FX$ are the components of a pseudonatural transformation $(\mathsf{k}^\star, \overline{\mathsf{k}}^\star) : G \Rightarrow F$, where for $f : X \to Y$ the 2-cell $\overline{\mathsf{k}}^\star_f$ is defined by commutativity of the following diagram:

$$\begin{tikzcd}
\mathsf{k}^\star_Y \circ Gf \arrow[rr, "\overline{\mathsf{k}}^\star_f"] \arrow[d, "\cong"'] & & Ff \circ \mathsf{k}^\star_X \\
\mathsf{k}^\star_Y \circ (Gf \circ \mathrm{Id}_{GX}) \arrow[d, "\mathsf{k}^\star_Y \circ Gf \circ \mathsf{v}_X"'] & & \mathrm{Id}_{FY} \circ (Ff \circ \mathsf{k}^\star_X) \arrow[u, "\cong"'] \\
\mathsf{k}^\star_Y \circ (Gf \circ (\mathsf{k}_X \circ \mathsf{k}^\star_X)) \arrow[d, "\cong"'] & & (\mathsf{k}^\star_Y \circ \mathsf{k}_Y) \circ (Ff \circ \mathsf{k}^\star_X) \arrow[u, "\mathsf{w}_Y \circ Ff \circ \mathsf{k}^\star_X"'] \\
\mathsf{k}^\star_Y \circ ((Gf \circ \mathsf{k}_X) \circ \mathsf{k}^\star_X) \arrow[rr, "\mathsf{k}^\star_Y \circ \overline{\mathsf{k}}_f^{-1} \circ \mathsf{k}^\star_X"'] & & \mathsf{k}^\star_Y \circ ((\mathsf{k}_Y \circ Ff) \circ \mathsf{k}^\star_X) \arrow[u, "\cong"']
\end{tikzcd}$$

2. The pseudonatural transformations $(\mathsf{k}, \overline{\mathsf{k}}) : F \leftrightarrows G : (\mathsf{k}^\star, \overline{\mathsf{k}}^\star)$ are the 1-cells of an equivalence $F \simeq G$ in $\mathrm{Hom}(\mathcal{B}, \mathcal{C})$.

Proof. To see that $(\mathsf{k}^\star, \overline{\mathsf{k}}^\star)$ is a pseudonatural transformation, the naturality and the unit laws follow from the corresponding laws for $\overline{\mathsf{k}}_f$. For the associativity law the process is similar, except one also applies the triangle law relating v and w.

For the second claim we construct invertible modifications $(\mathsf{k}^\star, \overline{\mathsf{k}}^\star) \circ (\mathsf{k}, \overline{\mathsf{k}}) \cong \mathrm{Id}_F$ and $\mathrm{Id}_G \cong (\mathsf{k}, \overline{\mathsf{k}}) \circ (\mathsf{k}^\star, \overline{\mathsf{k}}^\star)$. The obvious choices for the components are $\mathsf{w}_X : \mathsf{k}^\star_X \circ \mathsf{k}_X \Rightarrow \mathrm{Id}_{FX}$ and $\mathsf{v}_X : \mathrm{Id}_{GX} \Rightarrow \mathsf{k}_X \circ \mathsf{k}^\star_X$. It remains to check the modification axiom. To this end, observe that for every $f : X \to Y$ in $\mathcal{B}$, is the composite

$$(\mathsf{k}^\star_Y \circ \mathsf{k}_Y) \circ Ff \xRightarrow{\mathsf{w}_Y \circ Ff} \mathrm{Id}_{FY} \circ Ff \overset{\cong}{\Rightarrow} Ff \circ \mathrm{Id}_{FX} \xRightarrow{Ff \circ \mathsf{w}_X^{-1}} Ff \circ (\mathsf{k}^\star_X \circ \mathsf{k}_X)$$

Similarly, $\overline{(\mathsf{k} \circ \mathsf{k}^\star)}_f$ is the composite

$$(\mathsf{k}_Y \circ \mathsf{k}^\star_Y) \circ Gf \xRightarrow{\mathsf{v}_Y^{-1} \circ Gf} \mathrm{Id}_{GY} \circ Gf \overset{\cong}{\Rightarrow} Gf \circ \mathrm{Id}_{GX} \xRightarrow{Gf \circ \mathsf{v}_X} Gf \circ (\mathsf{k}_Y \circ \mathsf{k}^\star_Y)$$

One then sees that

$$\begin{tikzcd}
(\mathsf{k}^\star_Y \circ \mathsf{k}_Y) \circ Ff \arrow[r, "\mathsf{w}_Y \circ Ff"] \arrow[d, "\mathsf{w}_Y \circ Ff"'] \arrow[ddd, bend right=60, "\overline{(\mathsf{k}^\star_Y \circ \mathsf{k}_Y)}_f"'] & \mathrm{Id}_{FY} \circ Ff \arrow[ddd, "\cong"] \\
\mathrm{Id}_{FY} \circ Ff \arrow[ur, equal] \arrow[d, "\cong"'] & \\
Ff \circ \mathrm{Id}_{FX} \arrow[d, "Ff \circ \mathsf{w}_X^{-1}"'] \arrow[dr, equal] & \\
Ff \circ (\mathsf{k}^\star_X \circ \mathsf{k}_X) \arrow[r, "Ff \circ \mathsf{w}_X"'] & Ff \circ \mathrm{Id}_{FX}
\end{tikzcd}$$

so that $(\mathsf{w}_X)_{X \in \mathcal{B}}$ does indeed form a modification. The proof for v is similar. □

This lemma is particularly useful when it comes to constructing a biequivalence: to construct an equivalence $F \circ G \simeq \mathrm{id}$ it suffices to construct a pseudonatural transformation for which each component is an equivalence.

The lemma also justifies the following terminology. We call a pseudonatural transformation $(\mathsf{k}, \overline{\mathsf{k}})$ a *pseudonatural equivalence* if every component k_X is an equivalence, and a *pseudonatural isomorphism* if every k_X is invertible.

2.2 Biuniversal arrows

In his famous textbook [Mac98], Mac Lane makes precise the notion of universal property by introducing *universal arrows*. The Yoneda Lemma, limits and adjunctions are then all characterised in these terms. We adopt a similar approach, focussing on T. Fiore's *biuniversal arrows* [Fio06]. As well as providing a uniform way to describe bicategorical limits and bicategorical adjunctions, this perspective is particularly amenable to syntactic description. Biuniversal arrows are fundamental to the type theoretic description of bicategorical products and exponentials we shall see in Chapters 4 and 5.

A detailed development of the relationship between biuniversal arrows and biadjoints, complete with proofs, is available in [Fio06, Chapter 9]. The other results in what follows are implicit in much historical work on bicategory theory (*e.g.* [Str80]), but—as far as I am aware—have not previously been collected together in this form.

We begin by recapitulating the notion of universal arrow and its bicategorical counterpart.

Definition 2.2.1. Let $F : \mathbb{B} \to \mathbb{C}$ be a functor and $C \in \mathbb{C}$. A *universal arrow from F to C* is a pair $(R \in \mathbb{B}, u : FR \to C)$ such that, for any $B \in \mathbb{B}$ and $f : FB \to C$, there exists a unique $f^\dagger : B \to R$ such that $u \circ Ff^\dagger = f$. ◂

It is an exercise to show that every universal arrow (R, u) from F to C is equivalently a chosen family of natural isomorphisms $\mathbb{B}(-, R) \cong \mathbb{C}(F(-), C)$, or—equivalently again—a terminal object in the comma category $(F \downarrow C)$. It follows that a right adjoint to $F : \mathbb{B} \to \mathbb{C}$ is specified by a choice of universal arrow $\varepsilon_C : FUC \to C$ for every $C \in \mathbb{C}$. The mapping U extends to a functor with $Uf := (f \circ \varepsilon_C)^\dagger$ for $f : C \to C'$. The counit is then ε and the unit η arises by applying the universal property to the identity: $\eta_B := (\mathrm{id}_{FB})^\dagger : B \to UFB$. If both ε and η are invertible, the result is an adjoint equivalence.

To define biuniversal arrows, one weakens the isomorphisms defining a universal arrow to equivalences. We take particular care in choosing how we spell these out. It is generally convenient to require adjoint equivalences; by the well-known lifting theorem (*e.g.* [Lei04, Proposition 1.5.7]) this entails no loss of generality, while providing a more structured object to work with. We also go beyond T. Fiore's definition by requiring that each adjoint equivalence is determined by a choice of universal arrow.

Definition 2.2.2 (*c.f.* [Fio06])**.** Let $F : \mathcal{B} \to \mathcal{C}$ be a pseudofunctor and $C \in \mathcal{C}$. A *biuniversal arrow from F to C* consists of a pair $(R \in \mathcal{B}, u : FR \to C)$ and, for every $B \in \mathcal{B}$, a chosen adjoint equivalence of categories

$$\mathcal{B}(B, R) \xrightarrow{\simeq} \mathcal{C}(FB, C)$$
$$(B \xrightarrow{h} R) \mapsto (FB \xrightarrow{Fh} FR \xrightarrow{u} C)$$

specified by choosing a family of invertible universal 2-cells as the counit.

Explicitly, a biuniversal arrow from F to C consists of the following data:

- A pair $(R \in \mathcal{B}, u : FR \to C)$,
- For every $B \in \mathcal{B}$ and $h : FB \to C$, a map $\psi_B(h) : B \to R$ and an invertible 2-cell $\varepsilon_{B,h} : u \circ F\psi_B(h) \Rightarrow h$, universal in the sense that for any map $f : B \to R$ and 2-cell $\tau : u \circ Ff \Rightarrow h$ there exists a 2-cell $\tau^\dagger : f \Rightarrow \psi_B(h)$, unique such that

$$\begin{tikzcd} FB \arrow[r, bend left, "Ff"] \arrow[r, bend right, "F\psi_B(h)"'] \arrow[r, phantom, "\Downarrow F\tau^\dagger"] \arrow[dr, bend right=60, "h"'] \arrow[dr, phantom, "\Downarrow \varepsilon_{B,h}"] & FR \arrow[d, "u"] \\ & C \end{tikzcd} = \begin{tikzcd} & FR \arrow[dr, "u"] \arrow[d, phantom, "\Downarrow \tau"] & \\ FB \arrow[ur, "Ff"] \arrow[rr, "h"'] & {} & C \end{tikzcd} \tag{2.1}$$

such that the 2-cell $(\mathrm{id}_{u \circ Ff})^\dagger : f \Rightarrow \psi_B(u \circ Ff)$ is invertible for every $f : B \to R$. ◂

Remark 2.2.3. Pictorial representations such as (2.1) are known as *pasting diagrams*. It is a consequence of the coherence theorem for bicategories that, once a choice of bracketing is made for the source and target 1-cells, a pasting diagram identifies a unique 2-cell (*c.f.* [Gur06, Remark 3.1.16]; for a detailed exposition, see [Ver92, Appendix A]). ◂

On the face of it, a biuniversal arrow is only local structure: the data imposes a requirement on each hom-category, but no global constraints. This property will be particularly useful for our later work synthesising a type theory, where we shall encode bicategorical products and exponentials as biuniversal arrows. Global structure arises in the following way (*c.f.* [Mac98, III.2]).

Lemma 2.2.4. Let $F : \mathcal{B} \to \mathcal{C}$ be a pseudofunctor and $C \in \mathcal{C}$. There exists a biuniversal arrow (R, u) from F to C if and only if there exists an equivalence of pseudofunctors $\mathcal{B}(-, R) \simeq \mathcal{C}(F(-), C)$ in $\mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$,

Proof. For every equivalence of pseudofunctors $\mathcal{B}(-, R) \xrightarrow{\gamma} \mathcal{C}(F(-), C)$ one obtains from the Yoneda Lemma an arrow $\gamma_R(\mathrm{Id}_R) : FR \to C$. This arrow is biuniversal: indeed, the image of $\gamma_R(\mathrm{Id}_R)$ under the pseudofunctor $\mathcal{C}(FR, C) \to \mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})\big(\mathcal{B}(-, R), \mathcal{C}(F(-), C)\big)$ given by the Yoneda Lemma is isomorphic to γ, and hence an equivalence. The converse is [Fio06, Theorem 9.5]. □

Remark 2.2.5. In Chapter 7 we shall see that a biuniversal arrow from $F : \mathcal{B} \to \mathcal{C}$ to $C \in \mathcal{C}$ is equivalently a terminal object in the bicategorical comma category $(F \downarrow \mathrm{const}_C)$, for const_C the constant pseudofunctor at C. ◂

Elementary properties of biuniversal arrows. Many standard properties of universal arrows—such as those in [Mac98]—extend to biuniversal arrows. Biuniversal arrows are unique up to equivalence, and the $(-)^{\dagger}$ operation preserves both invertibility and naturality.

Notation 2.2.6. In the next lemma, and throughout, we shall abuse notation by writing just $\cong$ for the invertible 2-cell filling a square. Unless marked otherwise, it is assumed this 2-cell is oriented right-to-left (*c.f.* Remark 2.1.7). ◂

Lemma 2.2.7 ([Fio06, Lemma 9.7])**.** Let $F : \mathcal{B} \to \mathcal{C}$ be a pseudofunctor and $C \in \mathcal{C}$. For any two biuniversal arrows (R, u) and (R', u') from F to C there exists an equivalence $e : R \to R'$ and an invertible 2-cell κ filling

$$\begin{tikzcd}
FR \arrow[r, "u"] \arrow[d, "Fe"'] \arrow[dr, phantom, "\overset{\kappa}{\cong}"] & C \arrow[d, equal] \\
FR' \arrow[r, "u'"'] & C
\end{tikzcd}$$

Moreover, for any other pair $(f : R \to R', \lambda : u' \circ Fe \overset{\cong}{\Rightarrow} u)$ filling the above diagram, e and f are isomorphic via $\lambda^{\dagger}$. □

It follows from the essential uniqueness of equivalences that, if $u : FR \to C$ is a biuniversal arrow from F to C and $u' \cong u$, then u' is also a biuniversal arrow from F to C. The next lemma follows from further standard facts about adjoint equivalences of categories.

Lemma 2.2.8. Let $F : \mathcal{B} \to \mathcal{C}$ be a pseudofunctor and (R, u) a biuniversal arrow from F to $C \in \mathcal{C}$. For every object $B \in \mathcal{B}$,

1. If $f : B \to R$ is any morphism and $\alpha : u \circ Ff \Rightarrow h$ is invertible, then so is $\alpha^\dagger$.
2. If the 1-cells $h, h' : FB \to C$ and $f, f' : B \to R$ and 2-cells $\alpha : u \circ Ff \Rightarrow h$ and $\beta : u \circ Ff' \Rightarrow h'$ are related by a commutative diagram of 2-cells as on the left below

$$\begin{tikzcd} u \circ Ff \arrow[r, "\alpha_f"] \arrow[d, "u \circ F\sigma"'] & h \arrow[d, "\tau"] \\ u \circ Ff' \arrow[r, "\alpha_{f'}"'] & h' \end{tikzcd} \qquad \begin{tikzcd} f \arrow[r, "(\alpha_f)^\dagger"] \arrow[d, "\sigma"'] & \psi_B(h) \arrow[d, "\psi_B(\tau)"] \\ f' \arrow[r, "(\alpha_{f'})^\dagger"'] & \psi_B(h') \end{tikzcd}$$

then the diagram on the right above commutes. In particular, if $\alpha : u \circ F(-) \Rightarrow \mathrm{id}_{\mathcal{C}(FB,C)}$ is a natural transformation, then so is $\alpha^\dagger : \mathrm{id}_{\mathcal{B}(B,R)} \Rightarrow \psi_B(-)$. □

It is sometimes convenient, for example when working with bilimits, to work with the notion of *birepresentable pseudofunctor*.

Definition 2.2.9 ([Str80])**.** Let $F : \mathcal{B} \to \mathbf{Cat}$ be a pseudofunctor. A *birepresentation* (R, ρ) for F consists of an object $R \in \mathcal{B}$ and an equivalence $\rho : \mathcal{B}(R, -) \xrightarrow{\simeq} H$ in $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$. ◂

Representable functors $F : \mathcal{B} \to \mathrm{Set}$ correspond to universal arrows from the terminal object to F. Similarly, to relate biuniversal arrows to birepresentable functors we employ the dual notion of a biuniversal arrow from an object to a pseudofunctor.

Lemma 2.2.10 (*c.f.* [Mac98, Proposition III.2.2])**.** A pseudofunctor $F : \mathcal{B} \to \mathbf{Cat}$ is birepresentable if and only if there exists a biuniversal arrow from the terminal category $\mathbb{1}$ to F.

Proof. It is certainly the case that $\mathbf{Cat}(\mathbb{1}, F(-)) \simeq F$ in $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$. From birepresentability and the closure of equivalences under composition one obtains $\mathbf{Cat}(\mathbb{1}, F(-)) \simeq F \simeq \mathcal{B}(R, -)$, so the result follows from Lemma 2.2.4. □

2.2.1 Preservation of biuniversal arrows

Preservation of biuniversal arrows will provide a systematic way to define preservation of bilimits and preservation of biadjoints. We begin by examining preservation of universal arrows. Using the fact that a right adjoint to $F : \mathbb{B} \to \mathbb{C}$ is completely specified by a choice of universal arrow $(UC, F(UC) \to C)$ for each $C \in \mathbb{C}$—namely, the counit—it is reasonable to define morphisms of universal arrows analogously to morphisms of adjunctions [Mac98, Chapter IV].

Definition 2.2.11. Let $F : \mathbb{B} \to \mathbb{C}$ and $F' : \mathbb{B}' \to \mathbb{C}'$ be functors and suppose (R, u) is a universal arrow from F to $C \in \mathbb{C}$. A pair of functors (K, L) *preserves the universal arrow* (R, u) if the following diagram commutes

$$\begin{tikzcd}
\mathbb{B} \arrow[r, "F"] \arrow[d, "L"'] & \mathbb{C} \arrow[d, "K"] \\
\mathbb{B}' \arrow[r, "F'"'] & \mathbb{C}'
\end{tikzcd}$$

and $F'LR = KFR \xrightarrow{Ku} KC$ is a universal arrow from F' to KR. ◂

Equivalently, we ask that the functor $(F \downarrow C) \to (F' \downarrow KC)$ defined by $(B, h : FB \to C) \mapsto (LB, F'LB = KFB \xrightarrow{Kh} KC)$ preserves the terminal object. This is a slight weakening of the definition of transformation of adjunctions given in [Mac98]: Mac Lane asks that the unit (or counit) be *strictly* preserved.

The bicategorical translation is as one would expect.

Definition 2.2.12. Let $F : \mathcal{B} \to \mathcal{C}$ and $F' : \mathcal{B}' \to \mathcal{C}'$ be pseudofunctors and suppose (R, u) is a biuniversal arrow from F to $C \in \mathcal{C}$. A triple of pseudofunctors and pseudonatural transformations (K, L, ρ) as in the diagram

$$\begin{tikzcd}
\mathcal{B} \arrow[r, "F"] \arrow[d, "L"'] \arrow[dr, phantom, "\overset{\rho}{\Rightarrow}"] & \mathcal{C} \arrow[d, "K"] \\
\mathcal{B}' \arrow[r, "F'"'] & \mathcal{C}'
\end{tikzcd} \tag{2.2}$$

preserves the biuniversal arrow (R, u) if $F'LR \xrightarrow{\rho_R} KFR \xrightarrow{Ku} KC$ is a biuniversal arrow from F' to KC. ◂

By Lemma 2.2.4, if (K, L, ρ) preserves the universal arrow (R, u) as in (2.2) then one obtains a pseudonatural family of equivalences $\mathcal{B}'(B', LR) \simeq \mathcal{C}'(F'B', KC)$.

Just as an equivalence of categories preserves all 'categorical' properties, so a biequivalence preserves all 'bicategorical' properties. In particular, a biequivalence preserves all biuniversal arrows.

Lemma 2.2.13. Let $H : \mathcal{C} \to \mathcal{D}$ be a biequivalence and $F : \mathcal{B} \to \mathcal{C}$ be a pseudofunctor. If (R, u) is a biuniversal arrow from F to $C \in \mathcal{C}$, then Hu is a biuniversal arrow from HF to HX. Hence, the triple $(H, \mathrm{id}_{\mathcal{B}}, \mathrm{id})$ preserves the biuniversal arrow.

Proof. Since H is locally an equivalence, for every $B \in \mathcal{B}$ there exists a composite adjoint equivalence of categories $\mathcal{B}(B, R) \simeq \mathcal{C}(FB, C) \overset{H_{FB,C}}{\simeq} \mathcal{D}(HFB, HC)$ taking $h : B \to R$ to $H(u \circ Fh)$. Since $H(u) \circ HF(-)$ is naturally isomorphic to this adjoint equivalence, it is an adjoint equivalence itself. □

There are two ways of formulating that a functor F preserves limits: one can either ask that the image of the terminal cone is also a terminal cone, or that the canonical map $F(\lim H) \to \lim(FH)$ is an isomorphism. Similar considerations apply to preservation of biuniversal arrows.

Lemma 2.2.14. Consider a square of pseudofunctors K, L, F, F' related by a pseudonatural transformation $(\rho, \overline{\rho}) : KF \Rightarrow F'L$ as in (2.2), thus:

$$\begin{tikzcd}
\mathcal{B} \arrow[r, "F"] \arrow[d, "L"'] \arrow[dr, phantom, "\overset{\rho}{\Rightarrow}"] & \mathcal{C} \arrow[d, "K"] \\
\mathcal{B}' \arrow[r, "F'"'] & \mathcal{C}'
\end{tikzcd}$$

For every pair of biuniversal arrows (R, u) and (R', u') from F to $C \in \mathcal{C}$ and F' to $KC \in \mathcal{C}'$, respectively, the following are equivalent:

1. (K, L, ρ) preserves the biuniversal arrow (R, u),
2. The canonical map $\psi'_{LR}(Ku \circ \rho_R) : LR \to R'$ is an equivalence, where we write ψ'_{LR} for the chosen pseudo-inverse to $u' \circ F'(-) : \mathcal{B}'(LR, R') \to \mathcal{C}'(F'LR, KC)$.

Proof. Suppose first that $\psi'_{LR}(Ku \circ \rho_R)$ is an equivalence. Since pseudofunctors preserve equivalences, the composite $\mathcal{B}'(B', LR) \xrightarrow{\psi'_{LR}(Ku\circ\rho_R)\circ(-)} \mathcal{B}'(B', R') \xrightarrow{u'\circ F'(-)} \mathcal{C}'(F'C', KC)$ is an equivalence. Hence $u' \circ F'(\psi'_{LR}(Ku \circ \rho_R))$ is a biuniversal arrow. But then the 2-cell $\varepsilon'_{LR}(Ku \circ \rho_R)$ provides a natural isomorphism $u' \circ F'(\psi'_{LR}(Ku \circ \rho_R)) \overset{\cong}{\Rightarrow} Ku \circ \rho_R$, so $Ku \circ \rho_R$ is also a biuniversal arrow.

The converse is a straightforward application of universality (*c.f.* also Lemma 2.2.7): if $(LR, Ku \circ \rho_R)$ and (R', u') are both biuniversal arrows from F' to KC, then the canonical arrows $LR \to R'$ and $R' \to LR$ obtained from the universal property must form an equivalence. □

It will be useful to define *strict preservation* of biuniversal arrows. This strictness will play an important role in later chapters, where we will ask that the syntactic models of our type theories satisfy a strict freeness property. The aim of this definition is to ensure that the chosen structure witnessed by a biuniversal arrow (*e.g.* a bilimit) is taken to exactly the chosen structure in the target.

Definition 2.2.15. Let $F : \mathcal{B} \to \mathcal{C}$ and $F' : \mathcal{B}' \to \mathcal{C}'$ be pseudofunctors and suppose (R, u) and (R', u') are biuniversal arrows from F to $C \in \mathcal{C}$ and from F' to $C' \in \mathcal{C}'$, respectively. A pair of pseudofunctors (K, L) is a *strict morphism of biuniversal arrows* from (R, u) to (R', u') if

1. K and L are strict pseudofunctors such that $KF = F'L$,
2. The data of the biuniversal arrow is preserved: $LR = R'$, $KC = C'$ and $Ku = u'$,
3. The mappings $\psi_B : \mathcal{C}(FB, C) \to \mathcal{B}(B, R)$ and $\psi'_{B'} : \mathcal{C}'(F'B', C') \to \mathcal{B}'(B', R')$ are preserved, so that $L\psi_B(f) = \psi'_{LB}K(f)$ for every $f : FB \to C$,

4. For every $B \in \mathcal{B}$ and equivalence $u \circ F(-) : \mathcal{B}(B,R) \leftrightarrows \mathcal{C}(FB,C) : \psi_B$ the universal arrow $\varepsilon_{B,h} : u \circ F\psi_B(h) \Rightarrow h$ is strictly preserved, in the sense that $K_{FB,C}(\varepsilon_{B,h}) = \varepsilon'_{LB,Kh}$. ◂

In bicategory theory it is usually good practice to specify data up to equivalence, as pseudofunctors preserve equivalences but may not preserve isomorphisms or equalities. The preceding definition abuses this convention, and so is not 'bicategorical' in style. A consequence is that an arbitrary biequivalence may not strictly preserve biuniversal arrows (*c.f.* the proof of Lemma 2.2.13). This level of strictness does, however, provide a way to talk about free bicategories-with-structure using the language of 1-category theory (*c.f.* [Gur06, Proposition 2.10]).

Remark 2.2.16. We distinguish between *preservation* of biuniversal arrows in the sense of Definition 2.2.12 and a *morphism* of biuniversal arrows as in the preceding definition on the following basis. In Definition 2.2.12 we require that the image of the given biuniversal arrow is a biuniversal arrow, but do not specify its exact nature. In the preceding definition, by contrast, we require that the pair (K, L) takes the biuniversal arrow specified in the source to exactly the biuniversal arrow specified in the target. ◂

Strict preservation of a biuniversal arrow is sufficient to imply preservation of the corresponding universal property, in the following sense.

Lemma 2.2.17. Let $F : \mathcal{B} \to \mathcal{C}$ and $F' : \mathcal{B}' \to \mathcal{C}'$ be pseudofunctors and suppose (R, u) and (R', u') are biuniversal arrows from F to $C \in \mathcal{C}$ and from F' to $C' \in \mathcal{C}'$, respectively. If (K, L) is a strict morphism from (R, u) to (R', u'), then for every $B \in \mathcal{B}$, $h : B \to R$ and $\tau : u \circ Fh \Rightarrow f$, $L\tau^\dagger = (K\tau)^\dagger$.

Proof. It suffices to show that $L\tau^\dagger$ satisfies the universal property of $(K\tau)^\dagger$. For this one observes that

$$\begin{aligned} \varepsilon'_{LB,Kf} \bullet F'L\tau^\dagger &= K(\varepsilon_{B,f}) \bullet KF(\tau^\dagger) \qquad \text{by strict preservation} \\ &= K(\varepsilon_{B,f} \bullet F\tau^\dagger) \\ &= K\tau \end{aligned}$$

as required. □

A strict morphism of biuniversal arrows (K, L) defines a morphism of adjunctions (in the sense of Mac Lane) at every hom-category. Indeed, it follows directly from the definition that for every $B \in \mathcal{B}$ the following square commutes:

$$\begin{tikzcd}
\mathcal{B}(B,R) \arrow[rrr, "u_C \circ F(-)"] \arrow[d, "L_{B,R}"'] & & & \mathcal{C}(FB,C) \arrow[d, "K_{FB,C}"] \\
\mathcal{B}'(LB,LR) \arrow[r, equal] & \mathcal{B}'(LB,R') \arrow[r, "u'_{LB} \circ F'(-)"'] & \mathcal{C}'(F'LB,C') \arrow[r, equal] & \mathcal{C}'(KFB,KC)
\end{tikzcd}$$

and $K_{FB,C}$ preserves the counit by assumption.

2.3 Bilimits

We are now in a position to introduce bilimits and preservation of bilimits. The formulation in terms of biuniversal arrows is pleasingly concise. For every pair of bicategories $\mathcal{J}, \mathcal{B}$ one has a *diagonal pseudofunctor* $\Delta : \mathcal{B} \to \mathrm{Hom}(\mathcal{J}, \mathcal{B})$ taking $B \in \mathcal{B}$ to the constant pseudofunctor at B. Explicitly, $\Delta B : \mathcal{J} \to \mathcal{B}$ takes a 2-cell $\tau : h \Rightarrow h' : j \to j'$ to the identity 2-cell $\mathrm{id}_B : \mathrm{Id}_B \Rightarrow \mathrm{Id}_B : B \to B$. The 2-cell $\psi_j : \mathrm{Id}_{(\Delta B)(j)} \Rightarrow (\Delta B)(\mathrm{Id}_j)$ is the identity and for a composite $j \xrightarrow{g} j' \xrightarrow{f} j''$ in $\mathcal{J}$ the 2-cell $\phi_{f,g} : (\Delta B)(f) \circ (\Delta B)(g) \Rightarrow (\Delta B)(f \circ g)$ is $\mathfrak{l}_{\mathrm{Id}_B} : \mathrm{Id}_B \circ \mathrm{Id}_B \Rightarrow \mathrm{Id}_B$. A bilimit is then a biuniversal arrow.

Definition 2.3.1. A *bilimit* for $F : \mathcal{J} \to \mathcal{B}$ is a biuniversal arrow from the diagonal pseudofunctor $\Delta : \mathcal{B} \to \mathrm{Hom}(\mathcal{J}, \mathcal{B})$ to F. ◂

Unwrapping the definition, we require a pair $(\mathrm{bilim}\, F, \lambda : \Delta(\mathrm{bilim}\, F) \Rightarrow F)$ such that for every object $B \in \mathcal{B}$ and cone (pseudonatural transformation) $\kappa : \Delta B \Rightarrow F$ there exists a map $u_\kappa : B \to \mathrm{bilim}\, F$ and an invertible modification $\varepsilon_{B,\kappa}$ filling

$$\begin{tikzcd}
\Delta B \arrow[rr, "\Delta(u_\kappa)"] \arrow[dr, "\kappa"'] & {} \arrow[d, phantom, "\overset{\varepsilon_{B,\kappa}}{\Leftarrow}" near start] & \Delta(\mathrm{bilim}\, F) \arrow[dl, "\lambda"] \\
& F &
\end{tikzcd}$$

This modification is required to be universal in the sense that, for any 1-cell $v : B \to \mathrm{bilim}\, F$ and 2-cell $\beta : \lambda \circ \Delta v \Rightarrow \kappa$, there exists a unique $\beta^\dagger : v \Rightarrow u_\kappa$ such that

$$\begin{tikzcd}
\Delta B \arrow[r, bend left, "\Delta v"] \arrow[r, bend right, "\Delta u_\kappa"'] \arrow[r, phantom, "\Downarrow \Delta\beta^\dagger"] \arrow[drr, bend right=50, "\kappa"'] & \Delta(\mathrm{bilim}\, F) \arrow[dr, "\lambda"] & & = & & \Delta(\mathrm{bilim}\, F) \arrow[dr, "\lambda"] \arrow[d, phantom, "\Downarrow \beta"] & \\
& \Downarrow \varepsilon_{B,\kappa} & F & & \Delta B \arrow[ur, "\Delta v"] \arrow[rr, "\kappa"'] & {} & F
\end{tikzcd}$$

Finally, we require that for every $w : B \to \mathrm{bilim}\, F$ the 2-cell $(\mathrm{id}_{\lambda \circ \Delta w})^\dagger : w \Rightarrow u_{\lambda \circ \Delta w}$ is invertible.

By Lemma 2.2.4 this definition can be rephrased as a pseudonatural family of adjoint equivalences $\mathcal{B}(B, \mathrm{bilim}\, F) \simeq \mathrm{Hom}(\mathcal{J}, \mathcal{B})(\Delta B, F)$. It therefore coincides with that of Street [Str80] in terms of birepresentations. We say that a bicategory $\mathcal{B}$ is *bicomplete* or *admits all bilimits* if for every small bicategory $\mathcal{J}$ and pseudofunctor $F : \mathcal{J} \to \mathcal{B}$ the bilimit $\mathrm{bilim}\, F$ exists in $\mathcal{B}$.

Preservation of bilimits. We define preservation of bilimits as preservation of the corresponding biuniversal arrows, via the following lemma.

Lemma 2.3.2. For any bicategory $\mathcal{J}$ and pseudofunctor $H : \mathcal{B} \to \mathcal{C}$ the following diagram commutes up to canonical isomorphism:

$$\begin{tikzcd}
\mathcal{B} \arrow[r, "\Delta^{\mathcal{B}}"] \arrow[d, "H"'] \arrow[dr, phantom, "\overset{\cong}{\Leftarrow}"] & \mathrm{Hom}(\mathcal{J},\mathcal{B}) \arrow[d, "H\circ(-)"] \\
\mathcal{C} \arrow[r, "\Delta^{\mathcal{C}}"'] & \mathrm{Hom}(\mathcal{J},\mathcal{C})
\end{tikzcd} \tag{2.3}$$

Proof. Let us write $H_* := H \circ (-)$. Unwinding the respective definitions, $(H_* \circ \Delta^{\mathcal{B}})B : \mathcal{J} \to \mathcal{C}$ is the pseudofunctor sending every $j \in \mathcal{J}$ to HB, every $p : j \to j'$ to $H\mathrm{Id}_B$ and every 2-cell $\sigma : p \Rightarrow p'$ to the identity. This coincides with $(\Delta^{\mathcal{C}} \circ H)B$ everywhere except that $(\Delta^{\mathcal{C}} \circ H)(B)(j \xrightarrow{p} j') = \mathrm{Id}_{HB}$. So for every $B \in \mathcal{B}$ there exists a pseudonatural isomorphism $\alpha_B := (H_* \circ \Delta^{\mathcal{B}})B \Rightarrow (\Delta^{\mathcal{C}} \circ H)B$ with components $\alpha_B(j) := \mathrm{Id}_{HB}$ for all $j \in \mathcal{J}$. The witnessing 2-cell is the evident composite of ψ^H with structural isomorphisms. Thus one obtains an invertible 1-cell α_B in $\mathrm{Hom}(\mathcal{J},\mathcal{C})$ for every $B \in \mathcal{B}$. To extend this to a pseudonatural isomorphism, one takes $\overline{\alpha}_f : \alpha_{B'} \circ H_*(\Delta^{\mathcal{B}} f) \Rightarrow \Delta^{\mathcal{C}}(Hf) \circ \alpha_B$ (for $f : B \to B'$) to be the invertible modification with components given by the structural isomorphism $\mathrm{Id}_{HB'} \circ Hf \overset{\cong}{\Rrightarrow} Hf \circ \mathrm{Id}_{HB}$. Then $(\alpha, \overline{\alpha})$ is the required isomorphism. □

Thus, assuming the bilimit exists in $\mathcal{C}$, we say that H *preserves the bilimit* of $F : \mathcal{J} \to \mathcal{B}$ if $(H_*, H, (\alpha, \overline{\alpha}))$ preserves the biuniversal arrow $(\mathrm{bilim}\, F, \lambda)$. By Lemma 2.2.14, this condition is equivalent to requiring that the canonical map $H(\mathrm{bilim}\, F) \to \mathrm{bilim}(HF)$ is an equivalence.

The general perspective of biuniversal arrows leads to a straightforward proof that biequivalences preserve all bilimits.

Corollary 2.3.3. For any biequivalence $H : \mathcal{B} \leftrightarrows \mathcal{B}' : G$,

1. H preserves all bilimits that exist in $\mathcal{B}$,
2. If $\mathcal{B}$ has all $\mathcal{J}$-bilimits then $\mathcal{B}'$ has all $\mathcal{J}$-bilimits.

Proof. For (1), suppose $F : \mathcal{J} \to \mathcal{B}$ has a bilimit. By Lemma 2.2.13 one obtains a biuniversal arrow from $H_* \circ \Delta$ to $H_*(F)$, which by (2.3) is biuniversal from $\Delta^{\mathcal{B}'} H$ to HF. So the bilimit is preserved.

For (2), suppose $F : \mathcal{J} \to \mathcal{B}'$. Then $GF : \mathcal{J} \to \mathcal{B}$ has a bilimit and hence, by the previous part, so does $HGF : \mathcal{J} \to \mathcal{B}'$. Since $HG \simeq \mathrm{id}_{\mathcal{B}'}$, it follows that F has a bilimit. □

Two other classes of pseudofunctors that one would certainly expect to preserve bilimits are right biadjoints (see Definition 2.4.1) and birepresentables. This is indeed the case.

Lemma 2.3.4.

1. If the pseudofunctor $F : \mathcal{B} \to \mathcal{C}$ has a left biadjoint, then F preserves all bilimits that exist in $\mathcal{B}$.
2. If $F : \mathcal{B} \to \mathbf{Cat}$ is a birepresentable pseudofunctor, then F preserves all bilimits that exist in $\mathcal{B}$.

Proof. These are [Str80, §1.32] and [Str80, §1.20], respectively. □

2.4 Biadjunctions

Recalling that an adjunction is specified by a choice of universal arrows, we define a biadjunction by a choice of biuniversal arrows (*c.f.* [Pow98]).

Definition 2.4.1. Let $F : \mathcal{B} \to \mathcal{C}$ be a pseudofunctor. To specify a *right biadjoint* to F is to specify a biuniversal arrow $(UC, u_C : FUC \to C)$ from F to C for every $C \in \mathcal{C}$. ◂

Spelling out the definition, to give a right biadjoint $U : \mathcal{C} \to \mathcal{B}$ to F is to give:

- A mapping $U : ob(\mathcal{C}) \to ob(\mathcal{B})$,
- A family of 1-cells $(u_C : FUC \to C)_{C \in \mathcal{C}}$,
- For every $B \in B$ and $h : FB \to C$ a 1-cell $\psi_B(h) : B \to UC$ and an invertible 2-cell $\varepsilon_{B,h} : u_C \circ F\psi_B(h) \Rightarrow h$ that is universal in the sense of (2.1) (p. 22), such that the unit $\eta_h := (\mathrm{id}_{u_C \circ Fh})^{\dagger} : h \Rightarrow \psi_B(u_C \circ Fh)$ is invertible for every h.

One thereby obtains a pseudofunctor $U : \mathcal{C} \to \mathcal{B}$ by setting $U(C) := UC$ on objects, $U(C \xrightarrow{g} C') := \psi_{UC}(g \circ u_C)$ and $U(g \overset{\sigma}{\Rightarrow} g') := ((\sigma \circ u_C) \bullet \varepsilon_{UC,g})^{\dagger}$. By Lemma 2.2.4, this definition is equivalent to asking for a pair of pseudofunctors $F : \mathcal{B} \leftrightarrows \mathcal{C} : U$ together with a pseudonatural family of equivalences $\mathcal{B}(B, UC) \simeq \mathcal{C}(FB, C)$. For detailed proofs of this and related results, see [Fio06, Chapter 9].

The biuniversal arrow formulation of biadjoints, relying as it does on universal properties at each level, is perhaps easiest to work with when it comes to calculations (*c.f.* [FGHW07]). As we shall see in Chapters 4 and 5, it is also particularly amenable to being expressed syntactically.

Remark 2.4.2. The definition of bilimit can now be rephrased in the following fashion: the pseudofunctor bilim : $\mathrm{Hom}(\mathcal{J}, \mathcal{B}) \to \mathcal{B}$, when it exists, is right biadjoint to the diagonal pseudofunctor (*c.f.* [Fio06, Remark 9.2.1]). ◂

We have chosen to place bilimits and biadjoints on a similar footing by presenting them both as instances of biuniversal arrows. The preceding remark indicates that the theory of bilimits could alternatively be phrased using biadjoints. For example, one may use the fact that a right biadjoint preserves all bilimits, together with the observation that every biequivalence can be 'upgraded' to an adjoint biequivalence [Gur12], to obtain an alternative proof of Corollary 2.3.3(1).

Preservation of biadjunctions. We shall use the notion of preservation of biadjunctions to define preservation of exponentials.

Definition 2.4.3. For any biadjoint pair $F : \mathcal{B} \leftrightarrows \mathcal{C} : U$ and pseudofunctor $F' : \mathcal{B}' \to \mathcal{C}'$, we say that the triple (K, L, ρ) as below

$$\begin{tikzcd} \mathcal{B} \arrow[r, "F"] \arrow[d, "L"'] \arrow[dr, phantom, "\overset{\rho}{\Rightarrow}"] & \mathcal{C} \arrow[d, "K"] \\ \mathcal{B}' \arrow[r, "F'"'] & \mathcal{C}' \end{tikzcd} \tag{2.4}$$

preserves the biadjunction if (K, L, ρ) preserves each biuniversal arrow $u_C : FUC \to C$. ◂

A triple (K, L, ρ) preserving a biadjunction preserves the corresponding counits up to isomorphism. By definition, whenever (K, L, ρ) preserves the biadjunction $F \dashv U$ as in (2.4), then $F'LUC \xrightarrow{\rho_{UC}} KFUC \xrightarrow{Ku_C} KC$ is a biuniversal arrow from $F'L$ to KC. The next lemma entails that, if F' has a right adjoint U', then

$$F'U'KC \xrightarrow{\simeq} F'LUC \xrightarrow{\rho_{UC}} KFUC \xrightarrow{Ku_C} KC$$

is another such biuniversal arrow. By Lemma 2.2.7, this must be canonically isomorphic to the biuniversal arrow u'_{KC} witnessing the biadjunction $F' \dashv U'$.

Lemma 2.4.4. Let (K, L, ρ) preserve the biadjunction $F \dashv U$ as in (2.4) and suppose F' has a right biadjoint U'. Then $U'K \simeq LU$.

Proof. The definition of preservation of a biuniversal arrow, together with the definition of a biadjunction, entails that for any $B \in \mathcal{B}$ and $C \in \mathcal{C}$:

$$\mathcal{B}'(B, LUC) \simeq \mathcal{C}'(F'B, KC) \simeq \mathcal{B}'(B, U'KC)$$

By Lemma 2.2.4 these equivalences may equally be expressed as equivalences of pseudofunctors. Hence, $\mathrm{Y} \circ (LU) \simeq \mathrm{Y} \circ (U'K)$, for $\mathrm{Y} : \mathcal{B}' \to \mathrm{Hom}\left((\mathcal{B}')^{\mathrm{op}}, \mathbf{Cat}\right)$ the Yoneda embedding. The Yoneda Lemma then entails that $LU \simeq U'K$, as claimed. □

We end this chapter by instantiating Lemma 2.2.13 in the particular case of biadjunctions.

Lemma 2.4.5. Suppose that $F : \mathcal{B} \to \mathcal{C}$ has a right biadjoint U and that $H : \mathcal{C} \leftrightarrows \mathcal{C}' : G$ is a biequivalence. Then $HF : \mathcal{B} \leftrightarrows \mathcal{C}' : UG$ is a biadjunction.

Proof. By Lemma 2.2.13, each biuniversal arrow $u_C : FUC \to C$ defining the biadjunction $F \dashv U$ is preserved. In particular, taking $C' \in \mathcal{C}'$ such that $GC' \simeq C$ and the biuniversal arrow $u_{GC'} : FUGC' \to GC'$, one obtains a biuniversal arrow $HFUGC' \to HGC'$ from HF to HGC'. But from the biequivalence one has an adjoint equivalence $HG \simeq \mathrm{id}_{\mathcal{C}'}$ for which the component at C' is an adjoint equivalence $HGC' \simeq C'$. Composing, there exists a biuniversal arrow $(HF)(UG)C' \to C'$ from HF to C', as required. □

Part I

A type theory for cartesian closed bicategories

Chapter 3

A type theory for biclones

In this chapter we begin our construction of the type theory $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$ for cartesian closed bicategories. We focus on the bicategorical fragment: we construct a type theory $\Lambda_{\mathrm{ps}}^{\mathrm{bicat}}$ for bicategories and use it to recover a version of the Mac Lane-Paré coherence theorem for bicategories [MP85].

The work is driven by the theory of *biclones*, a bicategorification of the *abstract clones* of universal algebra [Coh81]. Abstract clones axiomatise the notion of equational theory with variables and a substitution operation, and provide a natural intermediary between syntax (in the form of the set of terms generated from operators over a set of variables) and semantics (in the form of categorical algebraic theories) (see *e.g.* [Plo94, p.129]). Biclones will play the same role in our construction, axiomatising syntax with an up-to-isomorphism substitution operation. We shall then synthesise the rules of our type theory $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$ from biclone structure.

The resulting type theory varies from classical type theories such as the simply-typed lambda calculus in two important respects. First, we make use of a form of *explicit substitution* [ACCL90]; second, it is *2-dimensional* in the sense that judgements relate types, terms and *rewrites* between terms.

These two developments both arise in the study of rewriting in the lambda calculus, but have previously only been studied independently. Explicit substitution calculi were first studied as versions of the lambda calculus closer to machine implementation [ACCL90] and have found applications in proof theory [RPW00] and programming language theory [LM99]. Much recent research (*e.g.* [DK97, Rit99]) has focussed on Melliès' observation that, contrary to what one might expect from the lambda calculus, such calculi may not be strongly normalising [Mel95] (see *e.g.* [RBL11] for an overview).

Two-dimensional type theories, on the other hand, first arose from Seely's observation [See87] that η-expansion and β-reduction form the unit and counit of a *lax* (directed) cartesian closed structure, a perspective advocated further by Jay & Ghani [Gha95, JG95] and put to use by Hilken [Hil96] for a proof-theoretic account of rewriting. In the strict setting, Hirschowitz [Hir13] and Tabereau [Tab11] have constructed 2-dimensional type theories to describe 2-categorical structures in rewriting theory and programming language design, respectively. The connection with intensional equality, meanwhile, has recently sparked significant interest in type theories with a notion of 'rewrite' or 'equality' motivated by the connection between higher category theory, topology and type theory. Examples include Licata & Harper's 2-dimensional directed type theory [LH11, LH12], Riehl & Shulman's type theory for synthetic ∞-categories [RS17], and Garner's 2-dimensional type theory [Gar09].

The type theory we shall construct brings together a novel combination of explicit substitution and 2-dimensional judgements. Following Hilken, we relate terms by separate syntactic entities called *rewrites*, and interpret these as 2-cells. This contrasts with many type theories motivated by connections with homotopy type theory (*e.g.* the Riehl-Shulman and Garner type theories), which capture 2-cells using Martin-Löf style identity types. The relationship between the two approaches remains to be explored.

Outline. The chapter breaks up into three parts. In Section 3.1 we consider the appropriate form of signature for a 2-dimensional type theory and construct the free biclone over such a signature. This drives the second part (Section 3.2), where we synthesise the type theory $\Lambda_{\text{ps}}^{\text{bicl}}$ and show that it is the internal language of biclones; as a corollary, we obtain an internal language for bicategories. Finally, in Section 3.3 we use $\Lambda_{\text{ps}}^{\text{bicl}}$ to prove a coherence result for biclones, amounting to a form of normalisation for the corresponding type theory.

3.1 Bicategorical type theory

3.1.1 Signatures for 2-dimensional type theory

A signature for the simply-typed lambda calculus is specified by a choice of base types and constants (sometimes called a *$\lambda\times$-signature* [Cro94]). A natural way of packaging such data, exemplified by Lambek & Scott [LS86], is as a graph. Taking inspiration from Lambek's notion of multicategories as models of *deductive systems* [Lam69, LS86], one may extend this using a *multigraph* (*c.f.* [Lam89, Her00, Lei04]). Here, one thinks of a judgement $(x_1 : A_1, \dots, x_n : A_n \vdash t : B)$ as corresponding to an edge with source $(A_1, \dots, A_n)$ and target B.[1]

Definition 3.1.1. A *multigraph* $\mathcal{G}$ consists of a set $\mathcal{G}_0$ of *nodes* together with a set $\mathcal{G}(A_1, \dots, A_n; B)$ of *edges* from $(A_1, \dots, A_n)$ to B for every $A_1, \dots, A_n, B \in \mathcal{G}_0$ (we allow $n = 0$). A *homomorphism* of multigraphs $h = (h, h_{A_1,\dots,A_n;B}) : \mathcal{G} \to \mathcal{G}'$ consists of a function $h : \mathcal{G}_0 \to \mathcal{G}'_0$ together with functions $h_{A_1,\dots,A_n;B} : \mathcal{G}(A_1, \dots, A_n; B) \to \mathcal{G}'(hA_1, \dots, hA_n; hB)$ for every $A_1, \dots, A_n, B \in \mathcal{G}_0$ $(n \in \mathbb{N})$. We denote the category of multigraphs and multigraph homomorphisms by MGrph. The full

[1] This should not be confused with the terminology in graph theory, where a multigraph sometimes refers to a graph in which there are allowed to be multiple edges between nodes (*e.g.* [Har69, p.10]).

subcategory Grph of *graphs* has objects those multigraphs $\mathcal{G}$ such that $\mathcal{G}(A_1, \ldots, A_n; B) = \varnothing$ whenever $n \neq 1$. ◂

Example 3.1.2. Every graph freely generates a *typed λ-calculus* [LS86] with types the nodes and a unary operator for each edge. Conversely, the simply-typed lambda calculus over a fixed set of base types determines a multigraph with nodes the types and edges $(A_1, \ldots, A_n) \to B$ the derivable terms $x_1 : A_1, \ldots, x_n : A_n \vdash t : B$ up to α-equivalence (we assume a fixed enumeration of variables $x_1, x_2, \ldots$ determining the name of the ith variable in the context). ◂

In this vein, the appropriate notion of signature for a 2-dimensional type theory is a form of '2-multigraph' (*c.f.* [Gur13, Chapter 2]).

Notation 3.1.3. In the following definition, and throughout, we write $A_\bullet$ for a finite sequence $\langle A_1, \ldots, A_n \rangle$.[2] Following Example 3.1.2, we use Greek letters $\Gamma, \Delta, \ldots$ to denote sequences $\langle A_1, \ldots, A_n \rangle$ in which the names of the terms A_i are not of importance. We use Γ_1, Γ_2 or $\Gamma_1 \mathbin{@} \Gamma_2$ to denote the concatenation of Γ_1 and Γ_2, and write $|\Gamma|$ for the length of Γ. ◂

Definition 3.1.4. A *2-multigraph* $\mathcal{G}$ is a set of *nodes* $\mathcal{G}_0$ equipped with a multigraph $\mathcal{G}(A_\bullet; B)$ of *edges* and *surfaces* for every $A_1, \ldots, A_n, B \in \mathcal{G}_0$ (we allow $n = 0$). A *homomorphism* of 2-multigraphs $h = (h, h_{A_\bullet, B}, h_{f,g}) : \mathcal{G} \to \mathcal{G}'$ is a map $h : \mathcal{G}_0 \to \mathcal{G}'_0$ together with functions

$$\begin{aligned} h_{A_1, \ldots, A_n; B} &: \mathcal{G}(A_\bullet; B) \to \mathcal{G}'(hA_1, \ldots, hA_n; hB) \\ h_{f,g} &: \mathcal{G}(A_\bullet; B)(f, g) \to \mathcal{G}'(hA_1, \ldots, hA_n; hB)(hf, hg) \end{aligned}$$

for every $A_1, \ldots, A_n, B \in \mathcal{G}_0$ $(n \in \mathbb{N})$ and $f, g \in \mathcal{G}(A_\bullet; B)$. We denote the category of 2-multigraphs by 2-MGrph. The full subcategory 2-Grph of *2-graphs* is formed by restricting to 2-multigraphs $\mathcal{G}$ such that $\mathcal{G}(A_1, \ldots, A_n; B) = \varnothing$ whenever $n \neq 1$. ◂

Example 3.1.5.

1. Every category determines a graph; every bicategory determines a 2-graph.
2. Every monoidal category $(\mathbb{C}, \otimes, I)$ determines a multigraph $\mathcal{G}_\mathbb{C}$ with nodes $(\mathcal{G}_\mathbb{C})_0 := ob(\mathbb{C})$ and $\mathcal{G}_\mathcal{C}(X_1, \ldots, X_n; Y) := \mathbb{C}(X_1 \otimes \ldots \otimes X_n, Y)$ (for some chosen bracketing of the n-ary tensor product).
3. More generally, every *multicategory* [Lam69] determines a multigraph. ◂

We shall see in Chapter 4 that every bicategory with finite products determines a *bi-multicategory* and every bi-multicategory determines a 2-multigraph.

[2] This notation is adopted from homological algebra, where one writes $X_\bullet$ for a chain complex $X_1 \to X_2 \to \cdots$ (*e.g.* [Wei94]).

3.1.2 Biclones

We turn to constructing bicategorical substitution structure over a 2-multigraph. As indicated above, our approach is to bicategorify the notion of *abstract clone* [Coh81].

Abstract clones. Abstract clones provide a presentation-independent description of (algebraic) equational theories with variables and substitution. A leading example is the *clone of operations* given by the set of terms over a fixed signature, subject to the substitution operation. We shall recall only the basic properties we require: for an introduction to the theory of clones from the perspective of universal algebra, see *e.g.* [Plo94, Tay99].

Definition 3.1.6. A *(sorted) abstract clone* $(S, \mathbb{C})$ consists of a set S of *sorts* with

- A set $\mathbb{C}(X_1, \dots, X_n; Y)$ of *operations* $t : X_1, \dots, X_n \to Y$ for each $X_1, \dots, X_n, Y \in S$ $(n \in \mathbb{N})$,
- Distinguished *projections* $\mathsf{p}^{(i)}_{X_\bullet} \in \mathbb{C}(X_1, \dots, X_n; X_i)(i = 1, \dots, n)$ for each $X_1, \dots, X_n \in S$ $(n \in \mathbb{N})$,
- For all sequences of sorts Γ and sorts $Y_1, \dots, Y_n, Z$ $(n \in \mathbb{N})$ a *substitution* function

$$\mathsf{sub}_{\Gamma, Y_\bullet, Z} : \mathbb{C}(Y_\bullet; Z) \times \textstyle\prod_{i=1}^{n} \mathbb{C}(\Gamma; Y_i) \to \mathbb{C}(\Gamma; Z)$$

  we denote by $\mathsf{sub}\big(f, (g_1, \dots, g_n)\big) := f[g_1, \dots, g_n]$,

such that

1. $t\Big[\mathsf{p}^{(1)}_{X_\bullet}, \dots, \mathsf{p}^{(n)}_{X_\bullet}\Big] = t$ for all $t \in \mathbb{C}(X_\bullet; Y)$,
2. $\mathsf{p}^{(k)}_{Y_\bullet}[t_1, \dots, t_n] = t_k$ $(k = 1, \dots, n)$ for all $(t_i \in \mathbb{C}(\Gamma; Y_i))_{i=1,\dots,n}$,
3. $t[u_\bullet][v_\bullet] = t[u_\bullet[v_\bullet]]$ for all $v_j \in \mathbb{C}(W_\bullet; X_j)$, $u_i \in \mathbb{C}(X_\bullet; Y_i)$ and $t \in \mathbb{C}(Y_\bullet; Z)$ $(i = 1, \dots, n$ and $j = 1, \dots, m)$.

We write $(t[u_\bullet])[v_\bullet]$ for the iterated substitution $t[u_1, \dots, u_n][v_1, \dots, v_m]$; by default, we bracket substitution to the left. An operation of form $t : X \to Y$ is called *unary.*

A morphism $h = (h, h_{X_\bullet;Y}) : (S, \mathbb{C}) \to (S', \mathbb{C}')$ of abstract clones is a map $h : S \to S'$ together with functions $h_{X_\bullet;Y} : \mathbb{C}(X_1, \dots, X_n; Y) \to \mathbb{C}'(hX_1, \dots, hX_n; hY)$ for each $X_1, \dots, X_n, Y \in S$, such that the projections and substitution operation are preserved. We denote the category of clones by Clone. ◂

Following the terminology for multicategories, we occasionally refer to the operations $t : X_1, \dots, X_n \to Y$ of a clone as *multimaps* or *arrows.* Where the context is unambiguous, we refer to a sorted clone $(S, \mathbb{C})$ simply as an *S-clone* and denote it by $\mathbb{C}$; a clone with a single sort is called *mono-sorted.*

Example 3.1.7.

1. Every clone $(S, \mathbb{C})$ defines a category $\overline{\mathbb{C}}$ by restricting to the unary operations. We call this the *nucleus* of $(S, \mathbb{C})$. Composition is given by substitution in $(S, \mathbb{C})$ and the identity on $X \in S$ is $\mathsf{p}_X^{(1)}$.
2. Any small category $\mathbb{C}$ with finite products defines an $ob(\mathbb{C})$-clone $\mathrm{Cl}(\mathbb{C})$ with

$$\mathrm{Cl}(\mathbb{C})(X_1, \ldots, X_n; Y) := \mathbb{C}(X_1 \times \cdots \times X_n, Y)$$

The projections are the projections in $\mathbb{C}$; the substitution $t[u_1, \ldots, u_n]$ is the composite $t \circ \langle u_1, \ldots, u_n \rangle$. ◂

One may read the two cases just presented as follows: every Lawvere theory defines a mono-sorted clone, and every mono-sorted clone defines a Lawvere theory. In fact, the full subcategory of Clone consisting of just the mono-sorted clones is equivalent to the category of Lawvere theories (see *e.g.* [Plo94]). This makes precise the sense in which clones capture a notion of algebraic theory. In the next chapter we shall explore the relationship between multi-sorted clones and cartesian categories more generally.

Clones and type-theoretic syntax. The definition of abstract clone isolates three axioms sufficient to describe substitution. The next example shows how a clone augments a graph with a notion of substitution (*c.f.* Example 3.1.2).

Example 3.1.8. For a chosen set of base types $\mathfrak{B}$ and multigraph $\mathcal{G}$ with nodes generated by the grammar

$$X, Y ::= B \mid X \times Y \mid X \Rightarrow Y \qquad (B \in \mathfrak{B})$$

the corresponding lambda calculus may be equipped with a simultaneous substitution operation $(t, (u_1, \ldots, u_n)) \mapsto t[u_1/x_1, \ldots, u_n/x_n]$ which respects the typing in the sense that the following rule is admissible:

$$\frac{x_1 : A_1, \ldots, x_n : A_n \vdash t : B \qquad (\Delta \vdash u_i : A_i)_{i=1,\ldots,n}}{\Delta \vdash t[u_1/x_1, \ldots, u_n/x_n]}$$

One therefore obtains a clone with sorts the types and multimaps $X_1, \ldots, X_n \to Y$ the α-equivalence classes of derivable terms $x_1 : X_1, \ldots, x_n : X_n \vdash t : Y$. The three axioms encapsulate the following standard properties of simultaneous substitution (*c.f.* the *syntactic substitution lemma* [Bar85, p.27]):

$$x_k[u_1/x_1, \ldots, u_n/x_n] = u_k \qquad t[x_1/x_1, \ldots, x_n/x_n] = t \qquad t[u_i/x_i][v_j/y_j] = t\big[u_i[v_j/y_j]/x_i\big]$$

One still obtains a clone if one takes $\alpha\beta\eta$-equivalence classes of terms; we denote this by $\mathbb{C}_{\Lambda^{\times,\to}(\mathcal{G})}$. ◂

Example 3.1.8 exemplifies the way in which clones provide an algebraic description of (type-theoretic) syntax. The tradition of categorical algebra, on the other hand, describes such syntax through the construction of a *syntactic category*, for which one aims to prove a freeness universal property. Generally some massage is required to account for the fact that categorical morphisms take a single object as their domain, but terms may exist in contexts of arbitrary length. For example, one may take contexts as objects and morphisms as lists of terms (*e.g.* [Pit00]), or restrict to unary contexts and take morphisms to be single terms (*e.g.* [Cro94]). It turns out that, if one employs the latter strategy, the relationship between the clone-theoretic and category-theoretic perspectives is particularly tight.

Lemma 3.1.9.

1. The inclusion Grph $\hookrightarrow$ MGrph has a right adjoint given by restricting to edges of the form $X \to Y$.
2. The forgetful functor Clone $\to$ MGrph taking a clone to its underlying multigraph has a left adjoint.
3. The functor $\overline{(-)}$: Clone $\to$ Cat restricting a clone to its nucleus has a left adjoint.

Proof. For (1) define a functor $\mathcal{L}$: MGrph $\to$ Grph by taking $\mathcal{LG}$ to be the graph with nodes exactly the nodes of $\mathcal{G}$ and edges $(\mathcal{LG})(X, Y) := \mathcal{G}(X, Y)$. The action on homomorphisms is similar: for $h : \mathcal{G} \to \mathcal{G}'$ one obtains $\mathcal{L}(h)$ by restricting to edges of the form $X \to Y$. Then, where ι : Grph $\hookrightarrow$ MGrph denotes the obvious embedding, a multigraph homomorphism $h : \iota(\mathcal{G}) \to \mathcal{G}'$ is a map on nodes $h : (\iota\mathcal{G})_0 \to \mathcal{G}'_0$ together with maps $h_{X_\bullet;Y} : (\iota\mathcal{G})(X_1, \dots, X_n; Y) \to \mathcal{G}'(hX_1, \dots, hX_n; hY)$ for each $X_1, \dots, X_n, Y \in (\iota\mathcal{G})_0$ $(n \in \mathbb{N})$. Since $(\iota\mathcal{G})(X_1, \dots, X_n; Y)$ is empty except when $n = 0$, this is equivalently a graph homomorphism $\mathcal{G} \to \mathcal{LG}'$.

For (2) we construct the free clone $\mathbb{FC}\mathrm{l}(\mathcal{G})$ on a multigraph $\mathcal{G}$. The construction is similar to that for the free multicategory on a multigraph (*c.f.* [Lei04, Chapter 2]). The sorts are the nodes of $\mathcal{G}$, and the operations are given by the following deductive system:

$$\frac{c \in \mathcal{G}(X_1, \dots, X_n; Y)}{c \in \mathbb{FC}\mathrm{l}(\mathcal{G})(X_1, \dots, X_n; Y)} \qquad \frac{X_i \in \{X_1, \dots, X_n\}}{\mathsf{p}^{(i)}_{X_1, \dots, X_n} \in \mathbb{FC}\mathrm{l}(\mathcal{G})(X_1, \dots, X_n; X_i)}$$

$$\frac{f \in \mathbb{FC}\mathrm{l}(\mathcal{G})(X_1, \dots, X_n; Y) \qquad \big(g_i \in \mathbb{FC}\mathrm{l}(\mathcal{G})(\Gamma; X_i)\big)_{i=1,\dots,n}}{f[g_1, \dots, g_n] \in \mathbb{FC}\mathrm{l}(\mathcal{G})(\Gamma; Y)}$$

subject to the equational theory requiring the three axioms of a clone. To see this is free, observe that for any clone $(S, \mathbb{C})$ and multigraph homomorphism $h : \mathcal{G} \to \mathbb{C}$ from $\mathcal{G}$ to the multigraph underlying $(S, \mathbb{C})$, the unique clone homomorphism $h^{\#} : \mathbb{FC}\mathrm{l}(\mathcal{G}) \to \mathbb{C}$ extending h must be defined by

$$h^{\#}(c) := h(c) \qquad h^{\#}(\mathsf{p}^{(i)}_{A_\bullet}) := \mathsf{p}^{(i)}_{h^{\#}A_\bullet} \qquad h^{\#}(f[g_1, \dots, g_n]) := (h^{\#}f)\big[(h^{\#}g_1), \dots, (h^{\#}g_n)\big]$$

For (3), let $\mathbb{C}$ be a category. Define a clone $\mathcal{P}\mathbb{C}$ with sorts the objects of $\mathbb{C}$ and hom-sets constructed as follows:

$$\frac{f \in \mathbb{C}(X, Y)}{f \in (\mathcal{P}\mathbb{C})(X; Y)} \qquad\qquad \frac{X_i \in \{X_1, \dots, X_n\}}{\mathsf{p}^{(i)}_{X_1, \dots, X_n} \in (\mathcal{P}\mathbb{C})(X_1, \dots, X_n; X_i)}$$

$$\frac{f \in (\mathcal{P}\mathbb{C})(X_1, \dots, X_n; Y) \qquad \big(g_i \in (\mathcal{P}\mathbb{C})(\Gamma; X_i)\big)_{i=1,\dots,n}}{f[g_1, \dots, g_n] \in (\mathcal{P}\mathbb{C})(\Gamma; Y)}$$

The equational theory $\equiv$ is the three laws of a clone, augmented by

$$\frac{}{\mathsf{p}^{(1)}_X \equiv \mathrm{id}_X \in (\mathcal{P}\mathbb{C})(X; X)} \qquad\qquad \frac{f \in \mathbb{C}(Y, Z) \qquad g \in \mathbb{C}(X, Y)}{f \circ g \equiv f[g] \in (\mathcal{P}\mathbb{C})(X; Z)}$$

For any clone $(T, \mathbb{D})$, a clone homomorphism $h : \mathcal{P}\mathbb{C} \to \mathbb{D}$ consists of a map of objects $ob(\mathbb{C}) \to T$ together with substitution-preserving mappings $(\mathcal{P}\mathbb{C})(X_1, \dots, X_n; Y) \to \mathbb{D}(X_1, \dots, X_n; Y)$ for each $X_1, \dots, X_n, Y \in ob(\mathbb{C})$ $(n \in \mathbb{N})$. Restricting to unary operations, this is exactly a functor $\mathbb{C} \to \overline{\mathbb{D}}$. Conversely, since any clone homomorphism is fixed on the projections, a functor $\mathbb{C} \to \overline{\mathbb{D}}$ corresponds uniquely to a clone homomorphism $\mathcal{P}\mathbb{C} \to \mathbb{D}$. □

In the light of the preceding lemma one obtains the diagram below. The adjunction between the 1-category Cat and Grph is the usual free-forgetful adjunction, and the functor $\overline{(-)} : \mathrm{Clone} \to \mathrm{Cat}$ restricts a clone $(S, \mathbb{C})$ to its unary operations (*i.e.* its nucleus). The outer square commutes on the nose and hence the inner square commutes up to natural isomorphism.

$$\begin{tikzcd}
& \mathrm{Clone} \arrow[dl, bend right=40, "\text{forget}"'] \arrow[dr, bend left=40, "\overline{(-)}"] & \\
\mathrm{MGrph} \arrow[ur, bend right=15, "\mathbb{F}\mathrm{Cl}(-)"'] \arrow[ur, phantom, "\top"] \arrow[dr, bend right=40, "\mathcal{L}"'] \arrow[dr, phantom, "\perp"] & & \mathrm{Cat} \arrow[ul, bend left=15, "\mathcal{P}"'] \arrow[ul, phantom, "\top"] \arrow[dl, bend left=40, "\text{forget}"] \arrow[dl, phantom, "\perp"] \\
& \mathrm{Grph} \arrow[ul, hook, bend left=15] \arrow[ur, bend right=15, "\mathbb{F}\mathrm{Cat}"] &
\end{tikzcd} \tag{3.1}$$

Indeed, examining the constructions one sees that $\overline{(-)} \circ \mathcal{P} \cong \mathrm{id}_{\mathrm{Cat}}$ and hence that

$$\mathrm{Cat}(\mathbb{F}\mathrm{Cat}(\mathcal{G}), \mathbb{C}) \cong \mathrm{Cat}\Big(\overline{\mathcal{P}(\mathbb{F}\mathrm{Cat}(\mathcal{G}))}, \mathbb{C}\Big) \cong \mathrm{Cat}(\overline{\mathbb{F}\mathrm{Cl}(\iota\mathcal{G})}, \mathbb{C}) \tag{3.2}$$

For our purposes, the moral is the following: to provide a type-theoretic description of the free category on a graph, it is sufficient to describe the free clone on a multigraph. One thereby obtains a more natural type theory—one does not need to restrict the rules to unary contexts—and the commutativity of this diagram guarantees that, when one does perform such a restriction, the result is (up to isomorphism) as intended.

Our aim in what follows is to lift this story to the bicategorical setting, and use it to extract a type theory for bicategories. We begin by defining a bicategorified notion of clone.

Biclones. Abstract clones may be defined in any cartesian category (and much more generally, see [Sta13, Fio17]). The bicategorified version arises by instantiating this definition in **Cat** and weakening the axioms to natural isomorphisms.

Definition 3.1.10. A *(sorted) biclone* $(S, \mathcal{C})$ is a set S of *sorts* equipped with the following data:

- For all $X_1, \ldots, X_n, Y \in S$ $(n \in \mathbb{N})$ a category $\mathcal{C}(X_1, \ldots, X_n; Y)$ with objects *multimaps* $f : X_\bullet \to Y$ and morphisms *2-cells* $\alpha : f \Rightarrow g : X_\bullet \to Y$, subject to a *vertical composition* operation,
- Distinguished *projection* functors $\mathsf{p}^{(i)}_{X_\bullet} : \mathbb{1} \to \mathcal{C}(X_1, \ldots, X_n; X_i)$ $(i = 1, \ldots, n)$ for all $X_1, \ldots, X_n \in S$ $(n \in \mathbb{N})$,
- For all sequences of sorts Γ and sorts $Y_1, \ldots, Y_n, Z$ $(n \in \mathbb{N})$ a *substitution* functor

  $$\mathsf{sub}_{\Gamma, Y_\bullet, Z} : \mathcal{C}(Y_\bullet; Z) \times \textstyle\prod_{i=1}^{n} \mathcal{C}(\Gamma; Y_i) \to \mathcal{C}(\Gamma; Z)$$

  we denote by $\mathsf{sub}\big(f, (g_1, \ldots, g_n)\big) := f[g_1, \ldots, g_n]$,
- Natural families of invertible *structural isomorphisms*

  $$\begin{aligned} \mathsf{assoc}_{t, u_\bullet, v_\bullet} : t[u_1, \ldots, u_n][v_\bullet] &\Rightarrow t[u_1[v_\bullet], \ldots, u_n[v_\bullet]] \\ \iota_u : u &\Rightarrow u\Big[\mathsf{p}^{(1)}_{X_\bullet}, \ldots, \mathsf{p}^{(n)}_{X_\bullet}\Big] \\ \varrho^{(k)}_{u_1, \ldots, u_n} : \mathsf{p}^{(k)}_{Y_\bullet}[u_1, \ldots, u_n] &\Rightarrow u_k \quad (k = 1, \ldots, n) \end{aligned}$$

  for every $t \in \mathcal{C}(Y_\bullet, Z)$, $u_j \in \mathcal{C}(X_\bullet, Y_j)$, $v_i \in \mathcal{C}(W_\bullet, X_i)$ and $u \in \mathcal{C}(X_\bullet, Y)$ $(i = 1, \ldots, n$ and $j = 1, \ldots, m)$,

This data is subject to coherence laws corresponding to the triangle and pentagon laws of a bicategory:

$$\begin{tikzcd} t[v_\bullet] \arrow[r, "\iota_t{[v_\bullet]}"] \arrow[d, equal] & t\big[\mathsf{p}^{(1)}, \ldots, \mathsf{p}^{(n)}\big][v_\bullet] \arrow[d, "\mathsf{assoc}_{t;\mathsf{p}^{(\bullet)};v_\bullet}"] \\ t[v_\bullet] & t\big[\mathsf{p}^{(1)}[v_\bullet], \ldots, \mathsf{p}^{(n)}[v_\bullet]\big] \arrow[l, "t\big[\varrho^{(1)}_{v_\bullet}, \ldots, \varrho^{(n)}_{v_\bullet}\big]"'] \end{tikzcd}$$

$$\begin{tikzcd}
t[u_\bullet][v_\bullet][w_\bullet] \arrow[r, "\mathsf{assoc}_{t;u_\bullet;v_\bullet}{[w_\bullet]}"] \arrow[d, "\mathsf{assoc}_{t[u_\bullet];v_\bullet;w_\bullet}"'] & t[u_\bullet[v_\bullet]][w_\bullet] \arrow[r, "\mathsf{assoc}_{t;u_\bullet[v_\bullet];w_\bullet}"] & t[u_\bullet[v_\bullet][w_\bullet]] \arrow[d, "{t[\mathsf{assoc}_{u_\bullet;v_\bullet;w_\bullet}]}"] \\
t[u_\bullet][v_\bullet[w_\bullet]] \arrow[rr, "\mathsf{assoc}_{t;u_\bullet;v_\bullet[w_\bullet]}"'] & & t[u_\bullet[v_\bullet[w_\bullet]]]
\end{tikzcd}$$

◂

Remark 3.1.11. Note that an invertible 2-cell is simply an iso in the relevant hom-category, but the definition of invertible multimap is more subtle (see Definition 4.2.15). ◂

We direct the 2-cells to match the definition of a *skew monoidal category* [Szl12]; the definition should therefore generalise to the lax setting. When we wish to emphasise the set of sorts, we call a biclone $(S, \mathcal{C})$ an *S-biclone*; where the set of sorts is clear from context, we refer to a biclone $(S, \mathcal{C})$ simply by $\mathcal{C}$. One obtains a *2-clone*—a clone enriched over Cat—when all the structural isomorphisms $\mathsf{assoc}, \iota, \varrho^{(i)}$ $(i = 1, \dots, n)$ are the identity. The second half of this chapter will be devoted to a coherence theorem showing that every freely-generated biclone is suitably equivalent to a 2-clone.

Example 3.1.12 (*c.f.* Example 3.1.7)**.**

1. Every clone defines a *locally discrete* biclone, in which each hom-category is discrete.
2. Every bicategory $\mathcal{B}$ with finite products defines a biclone; if $\mathcal{B}$ is a 2-category with strict (2-categorical) products, this is a 2-clone.
3. Every biclone $(S, \mathcal{C})$ gives rise to a bicategory $\overline{\mathcal{C}}$ by taking the *unary* hom-categories, *i.e.* by taking $\overline{\mathcal{C}}(X, Y) := \mathcal{C}(X; Y)$. We call this the *nucleus* of $(S, \mathcal{C})$. ◂

One may think of a biclone as a generalised deductive system in which the multimaps $f : A_1, \dots, A_n \to B$ are judgements $A_1, \dots, A_n \vdash f : B$, related by proof transformations $\tau : f \Rightarrow f'$ (*c.f.* [See87]). Conversely, Example 3.1.12(3) shows that a type theory for biclones would encompass bicategories as a special case. In Lemma 3.1.18 we shall see that the type theory describing the free biclone on a 2-graph restricts to a type theory for the free bicategory on a 2-graph (*c.f.* diagram (3.1)).

Remark 3.1.13. Biclones are objects worthy of further study in their own right. Thinking of them as 'bicategorified clones' suggests a connection—to be fleshed out—with some notion of 'bicategorical Lawvere theory', and with pseudomonads. On the other hand, biclones provide a categorical description of certain kinds of explicit substitution; possible connections with the categorical semantics of the simply-typed lambda calculus with explicit substitution (*e.g.* [GdR99]) remain to be explored. ◂

Free biclones and free bicategories. Defining a free biclone requires an appropriate notion of morphism. The definitions are natural extensions of those for bicategories.

Definition 3.1.14. A *pseudofunctor* $F : (S, \mathcal{C}) \to (S', \mathcal{C}')$ between biclones consists of a mapping $F : ob(\mathcal{C}) \to ob(\mathcal{C}')$ equipped with:

- A functor $F_{X_\bullet;Y} : \mathcal{C}(X_1, \ldots, X_n; Y) \to \mathcal{C}'(FX_1, \ldots, FX_n; FY)$ for all $X_1, \ldots, X_n, Y \in S (n \in \mathbb{N})$,
- Invertible 2-cells $\psi^{(i)}_{X_\bullet} : \mathsf{p}^{(i)}_{FX_\bullet} \Rightarrow F(\mathsf{p}^{(i)}_{X_\bullet})$ $(i = 1, \ldots, n)$ for each $X \in S$,
- An invertible 2-cell $\phi_{t,u_\bullet} : (Ft)[Fu_1, \ldots, Fu_n] \Rightarrow F(t[u_1, \ldots, u_n])$ for every $(u_j : X_\bullet \to Y_i)_{j=1,\ldots,n}$ and $t : Y_\bullet \to Z$, natural in t and $u_1, \ldots, u_n$,

subject to the following three coherence laws for $i = 1, \ldots, n$:

$$\begin{tikzcd}
\mathsf{p}^{(i)}_{FX_\bullet}[Fu_1, \ldots, Fu_n] \arrow[r, "\varrho^{(i)}_{Fu_\bullet}"] \arrow[d, "\psi^{(i)}_{X_\bullet}[Fu_\bullet]"'] & Fu_i \\
(F\mathsf{p}^{(i)}_{X_\bullet})[Fu_\bullet] \arrow[r, "\phi_{\mathsf{p}^{(i)},u_\bullet}"'] & F(\mathsf{p}^{(i)}_{X_\bullet}[u_\bullet]) \arrow[u, "F\varrho^{(i)}_{u_\bullet}"']
\end{tikzcd} \tag{3.3}$$

$$\begin{tikzcd}
F(t) \arrow[r, "F\iota_t"] \arrow[d, "\iota_{Ft}"'] & F\left(t[\mathsf{p}^{(1)}_{X_\bullet}, \ldots, \mathsf{p}^{(n)}_{X_\bullet}]\right) \\
(Ft)[\mathsf{p}^{(1)}_{FX_\bullet}, \ldots, \mathsf{p}^{(n)}_{FX_\bullet}] \arrow[r, "(Ft)[\psi^{(1)}, \ldots, \psi^{(1)}]"'] & (Ft)[F\mathsf{p}^{(1)}_{X_\bullet}, \ldots, F\mathsf{p}^{(n)}_{X_\bullet}] \arrow[u, "\phi_{t;\mathsf{p}(\bullet)}"']
\end{tikzcd} \tag{3.4}$$

$$\begin{tikzcd}
F(t)[Fu_\bullet][Fv_\bullet] \arrow[r, "\mathsf{assoc}_{Ft;Fu_\bullet;Fv_\bullet}"] \arrow[d, "\phi_{t;u_\bullet}[Fv_\bullet]"'] & F(t)[Fu_\bullet[Fv_\bullet]] \arrow[d, "F(t)[\phi_{u_\bullet;v_\bullet}]"] \\
F(t[u_\bullet])[Fv_\bullet] \arrow[d, "\phi_{t[u_\bullet];v_\bullet}"'] & F(t)[F(u_\bullet[v_\bullet])] \arrow[d, "\phi_{t;u_\bullet[v_\bullet]}"] \\
F(t[u_\bullet][v_\bullet]) \arrow[r, "F\mathsf{assoc}_{t;u_\bullet;v_\bullet}"'] & F(t[u_\bullet[v_\bullet]])
\end{tikzcd} \tag{3.5}$$

A pseudofunctor for which ϕ and every $\psi^{(1)}, \ldots, \psi^{(n)}$ is the identity is called *strict*. ◀

Example 3.1.15. Every pseudofunctor of biclones $F : (S, \mathcal{C}) \to (T, \mathcal{D})$ restricts to a pseudofunctor of bicategories $\overline{F} : \overline{\mathcal{C}} \to \overline{\mathcal{D}}$ between the nucleus of $(S, \mathcal{C})$ and the nucleus of $(T, \mathcal{D})$ (recall Example 3.1.12(3)). ◀

The construction of the free biclone on a 2-multigraph follows the pattern of its 1-categorical counterpart.

Construction 3.1.16 (Free biclone on a 2-multigraph)**.** Let $\mathcal{G}$ be a 2-multigraph. Define a biclone $\mathcal{FC}l(\mathcal{G})$ as follows. The sorts are nodes of $\mathcal{G}$ and the hom-categories are defined by the following deductive system:

$$\frac{c \in \mathcal{G}(A_1, \dots, A_n; B)}{c \in \mathcal{FC}l(\mathcal{G})(A_1, \dots, A_n; B)} \qquad \frac{\kappa \in \mathcal{G}(A_1, \dots, A_n; B)(c, c')}{\kappa \in \mathcal{FC}l(\mathcal{G})(A_1, \dots, A_n; B)}$$

$$\frac{}{\mathsf{p}^{(i)}_{A_1, \dots, A_n} \in \mathcal{FC}l(\mathcal{G})(A_1, \dots, A_n; A_i)}\ (1 \leqslant i \leqslant n)$$

$$\frac{f \in \mathcal{FC}l(\mathcal{G})(A_1, \dots, A_n; B) \qquad \big(g_i \in \mathcal{FC}l(\mathcal{G})(X_\bullet; A_i)\big)_{i=1,\dots,n}}{f[g_1, \dots, g_n] \in \mathcal{FC}l(\mathcal{G})(X_\bullet; B)}$$

$$\frac{\tau \in \mathcal{FC}l(\mathcal{G})(A_1, \dots, A_n; B)(f, f') \qquad \big(\sigma_i \in \mathcal{FC}l(\mathcal{G})(X_\bullet; A_i)(g_i, g_i')\big)_{i=1,\dots,n}}{\tau[\sigma_1, \dots, \sigma_n] \in \mathcal{FC}l(\mathcal{G})(X_\bullet; B)(f[g_1, \dots, g_n], f'[g_1', \dots, g_n'])}$$

$$\frac{f \in \mathcal{FC}l(\mathcal{G})(A_\bullet; B)}{\mathrm{id}_f \in \mathcal{FC}l(\mathcal{G})(A_\bullet; B)(f, f)} \qquad \frac{\tau \in \mathcal{FC}l(\mathcal{G})(A_\bullet; B)(f', f'') \qquad \sigma \in \mathcal{FC}l(\mathcal{G})(A_\bullet; B)(f, f')}{\tau \bullet \sigma \in \mathcal{FC}l(\mathcal{G})(A_\bullet; B)(f, f'')}$$

$$\frac{f \in \mathcal{FC}l(\mathcal{G})(B_\bullet; C) \qquad \big(g_i \in \mathcal{FC}l(\mathcal{G})(A_\bullet; B_i)\big)_{i=1,\dots,n} \qquad \big(h_j \in \mathcal{FC}l(\mathcal{G})(X_\bullet; B_j)\big)_{j=1,\dots,m}}{\mathsf{assoc}_{f,g_\bullet,h_\bullet} \in \mathcal{FC}l(\mathcal{G})(X_\bullet; C)(f[g_\bullet]\,[h_\bullet], f[g_\bullet[h_\bullet]])}$$

$$\frac{f \in \mathcal{FC}l(\mathcal{G})(A_1, \dots, A_n; B)}{\iota_f \in \mathcal{FC}l(\mathcal{G})(A_\bullet; B)\Big(f, f[\mathsf{p}^{(1)}_{A_\bullet}, \dots, \mathsf{p}^{(n)}_{A_\bullet}]\Big)}$$

$$\frac{\big(g_i \in \mathcal{FC}l(\mathcal{G})(X_\bullet; A_i)\big)_{i=1,\dots,n}}{\varrho^{(i)}_{A_1, \dots, A_n} \in \mathcal{FC}l(\mathcal{G})(X_\bullet; A_i)(\mathsf{p}^{(i)}_{A_1, \dots, A_n}[g_1, \dots, g_n], g_i)}\ (1 \leqslant i \leqslant n)$$

The equational theory $\equiv$ requires that

- Every $\mathcal{FC}l(\mathcal{G})(A_1, \dots, A_n; B)$ forms a category with composition the $\bullet$ operation and identity on $f \in \mathcal{FC}l(\mathcal{G})(A_1, \dots, A_n; B)$ given by id_f,
- The operation $\big(f, (g_1, \dots, g_n)\big) \mapsto f[g_1, \dots, g_n]$ is functorial with respect to this category structure,
- The families of 2-cells assoc, ι and $\varrho^{(i)}$ $(i = 1, \dots, n)$ are invertible, natural and satisfy the triangle and pentagon laws of a biclone. ◂

It is clear that this construction yields a biclone. Indeed, Lambek's definition of the internal language of a multicategory [Lam89] transfers readily to clones, and the preceding construction may be used to extend this definition to biclones. The only adjustment is that the operation symbols $f : A_1, \dots , A_n \to B$ are now related by transformations $\tau : f \Rightarrow f'$. The judgements in our type theory $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}$ will match these sequents precisely.

We shall, so far as possible, phrase the free properties we prove in terms of a unique strict pseudofunctor of biclones (*c.f.* [Gur13, Proposition 2.10]): this obviates the need to work with uniqueness up to 2-cell, in which the 2-cells may themselves only be unique up to a unique 3-cell. In particular, we bicategorify diagram (3.1) by using 1-categories of bicategorical objects (biclones and bicategories) in which the morphisms are *strict* pseudofunctors. Write Biclone and Bicat for these two categories. The relevant freeness universal property of Construction 3.1.16 is therefore the following.

Lemma 3.1.17. The forgetful functor Biclone → 2-MGrph taking a biclone to its underlying 2-multigraph has a left adjoint.

Proof. Let $\mathcal{G}$ be a 2-multigraph and $(T, \mathbb{D})$ be a biclone. We show that for every 2-multigraph morphism $h : \mathcal{G} \to \mathbb{D}$ there exists a unique strict pseudofunctor of biclones $h^\sharp : \mathcal{FCl}(\mathcal{G}) \to \mathcal{G}$ such that $h^\sharp \circ \iota = h$, for $\iota : \mathcal{G} \to \mathcal{FCl}(\mathcal{G})$ the inclusion.

Define $h^\#$ by induction as follows:

$$\begin{aligned} h^\#(c) &:= h_{A_\bullet;B}(c) && \text{for } c \in \mathcal{G}(A_1, \dots , A_n; B) \\ h^\#(\kappa) &:= h_{A_\bullet;B}(\kappa) && \text{for } \kappa \in \mathcal{G}(A_1, \dots , A_n; B)(c, c') \\ h^\#(\mathrm{id}_f) &:= \mathrm{id}_{h^\#(f)} \\ h^\#(\tau \bullet \sigma) &:= h^\#(\tau) \bullet h^\#(\sigma) \end{aligned}$$

We then require that $h^\#$ strictly preserves the projections, the substitution operations and the structural isomorphisms. This is a strict pseudofunctor $\mathcal{FCl}(\mathcal{G}) \to \mathbb{D}$ extending h. Uniqueness follows because any strict pseudofunctor must strictly preserve projections and the substitution operations, and so also strictly preserve the structural isomorphisms. □

The proof of Lemma 3.1.9 extends straightforwardly to an adjunction between 2-Grph and 2-MGrph. The following lemma therefore completes our bicategorical adaptation of diagram (3.1).

Lemma 3.1.18.

1. The forgetful functor Bicat → 2-Grph taking a bicategory to its underlying 2-graph has a left adjoint (*c.f.* [Gur13, Proposition 2.10]).
2. The functor $\overline{(-)}$: Biclone → Bicat restricting a biclone to its nucleus (recall Example 3.1.12) has a left adjoint.

Proof. For (1) we define the free bicategory $\mathcal{FB}ct(\mathcal{G})$ on a 2-graph $\mathcal{G}$ as the following deductive system (*c.f.* the description of bicategories as a generalised algebraic theory [Oua97]):

$$\frac{c \in \mathcal{G}(A,B)}{c \in \mathcal{FB}ct(\mathcal{G})(A,B)} \qquad \frac{\kappa \in \mathcal{G}(A,B)(c,c')}{\kappa \in \mathcal{FB}ct(\mathcal{G})(A,B)} \qquad \frac{}{\mathrm{Id}_A \in \mathcal{FB}ct(\mathcal{G})(A,A)}$$

$$\frac{f \in \mathcal{FB}ct(\mathcal{G})(A,B) \qquad g \in \mathcal{FB}ct(\mathcal{G})(X;A)}{f \circ g \in \mathcal{FB}ct(\mathcal{G})(X;B)} \qquad \frac{\tau \in \mathcal{FB}ct(\mathcal{G})(A,B)(f,f') \qquad \sigma \in \mathcal{FB}ct(\mathcal{G})(X,A)(g,g')}{\tau \circ \sigma \in \mathcal{FB}ct(\mathcal{G})(X;B)(f \circ g, f' \circ g')}$$

$$\frac{f \in \mathcal{FB}ct(\mathcal{G})(A,B)}{\mathrm{id}_f \in \mathcal{FB}ct(\mathcal{G})(A,B)(f,f)} \qquad \frac{\tau \in \mathcal{FB}ct(\mathcal{G})(A,B)(f',f'') \qquad \sigma \in \mathcal{FB}ct(\mathcal{G})(A,B)(f,f')}{\tau \bullet \sigma \in \mathcal{FB}ct(\mathcal{G})(A,B)(f,f'')}$$

$$\frac{f \in \mathcal{FB}ct(\mathcal{G})(B,C) \qquad g \in \mathcal{FB}ct(\mathcal{G})(A,B) \qquad h \in \mathcal{FB}ct(\mathcal{G})(X,B)}{\mathsf{a}_{f;g;h} \in \mathcal{FC}l(\mathcal{G})(X;C)(f[g]\,[h], f[g[h]])}$$

$$\frac{f \in \mathcal{B}(A,B)}{\mathsf{l}_f \in \mathcal{FB}ct(\mathcal{G})(A,B)(\mathrm{Id}_B \circ f, f)} \qquad \frac{f \in \mathcal{FB}ct(\mathcal{G})(A,B)}{\mathsf{r}_f \in \mathcal{FB}ct(\mathcal{G})(A,B)\,(f \circ \mathrm{Id}_A, f)}$$

subject to an equational theory requiring

- Every $\mathcal{FB}ct(\mathcal{G})(A,B)$ forms a category with composition the $\bullet$ operation and identity on $f \in \mathcal{FB}ct(\mathcal{G})(A,B)$ given by id_f,
- The operation $(f,g) \mapsto f \circ g$ is functorial with respect to this category structure,
- The families of 2-cells a, l and r are invertible, natural and satisfy the triangle and pentagon laws of a bicategory.

Since strict pseudofunctors are determined on all the structural data, any 2-graph homomorphism $h : \mathcal{G} \to \mathcal{C}$ to the 2-graph underlying a bicategory $\mathcal{C}$ determines a unique strict pseudofunctor $h^{\#} : \mathcal{FC}l(\mathcal{G}) \to \mathcal{C}$ restricting to h on $\mathcal{G}$.

For (2), let $\mathcal{B}$ be any bicategory. Define a biclone $\mathcal{PB}$ as follows. The sorts are objects of $\mathcal{B}$ and the hom-categories $(\mathcal{PB})(X_1, \dots, X_n; Y)$ are those given by the deductive system of Construction 3.1.16, adapted by replacing the first two rules by

$$\frac{f \in \mathcal{B}(X,Y)}{f \in (\mathcal{PB})(X;Y)} \qquad \frac{\kappa \in \mathcal{B}(X,Y)(f,f')}{\kappa \in (\mathcal{PB})(X;Y)(f,f')}$$

and augmenting the equational theory with rules ensuring the biclone and bicategory structures coincide wherever possible:

$$\frac{}{\mathsf{p}^{(1)}_X \equiv \mathrm{Id}_X \in (\mathcal{PB})(X;X)} \qquad \frac{f \in \mathcal{B}(Y,Z) \qquad g \in \mathcal{B}(X,Y)}{f \circ g \equiv f[g] \in (\mathcal{PB})(X;Z)}$$

$$\frac{f \in \mathcal{B}(X,Y)}{(\mathrm{id}_f)_{\mathcal{B}} \equiv (\mathrm{id}_f)_{\mathcal{PB}} \in (\mathcal{PB})(X;Y)}$$

$$\frac{\tau \in \mathcal{B}(Y,Z)(f,f') \qquad \sigma \in \mathcal{B}(X,Y)(g,g')}{\tau \circ \sigma \equiv \tau[\sigma] \in (\mathcal{PB})(X;Z)(f[g], f'[g'])} \qquad \frac{\tau \in \mathcal{B}(X,Y)(f,f') \qquad \sigma \in \mathcal{B}(X,Y)(f',f'')}{\tau \bullet_{\mathcal{B}} \sigma \equiv \tau \bullet_{\mathcal{PB}} \sigma \in (\mathcal{PB})(X;Y)(f,f'')}$$

$$\frac{f \in \mathcal{FB}ct(\mathcal{G})(B,C) \qquad g \in \mathcal{FB}ct(\mathcal{G})(A,B) \qquad h \in \mathcal{FB}ct(\mathcal{G})(X,B)}{\mathsf{assoc}_{f,g,h} \equiv \mathsf{a}_{f,g,h} \in \mathcal{FB}ct(\mathcal{G})(X,C)}$$

$$\frac{f \in \mathcal{B}(X,Y)}{\iota_f \equiv \mathsf{r}_f^{-1} : (\mathcal{PB})(X,Y)(f, f[\mathsf{p}^{(1)}_X)]} \qquad \frac{f \in \mathcal{B}(X,Y)}{\varrho^{(1)}_f \equiv \mathsf{l}_f : (\mathcal{PB})(X,Y)(\mathsf{p}^{(1)}_Y[f], f)}$$

The free property is a simple extension of that for clones (Lemma 3.1.9(3)). □

One therefore obtains the following diagram of adjunctions, generalising diagram (3.1). As for (3.1), the outer diagram commutes on the nose so the inner diagram commutes up to isomorphism.

$$\begin{tikzcd}
 & \text{Biclone} \arrow[dl, bend right=40, "\text{forget}"'] \arrow[dr, bend left=40, "\overline{(-)}"] & \\
\text{2-MGrph} \arrow[ur, bend right=20, "\mathcal{FC}l(-)"'] \arrow[dr, bend right=40, "\mathcal{L}"'] \arrow[ur, phantom, "\top"] & & \text{Bicat} \arrow[ul, bend left=20, "\mathcal{P}"] \arrow[dl, bend left=40, "\text{forget}"] \arrow[ul, phantom, "\top"] \\
 & \text{2-Grph} \arrow[ul, bend right=20] \arrow[ur, bend left=20, "\mathcal{FB}ct(-)"] \arrow[ul, phantom, "\perp"] \arrow[ur, phantom, "\perp"] &
\end{tikzcd} \tag{3.6}$$

It follows that, modulo a natural isomorphism, the free bicategory on a 2-graph $\mathcal{G}$ is obtained as the nucleus of the free biclone on $\mathcal{G}$ (regarded as a 2-multigraph). Indeed, examining the constructions one sees that $\overline{(-)} \circ \mathcal{P} \cong \mathrm{id}_{\mathrm{Bicat}}$, yielding the following chain of natural isomorphisms (*c.f.* equation (3.2)):

$$\mathrm{Bicat}(\mathcal{FB}ct(\mathcal{G}), \mathcal{B}) \cong \mathrm{Bicat}\Big(\overline{\mathcal{P}(\mathcal{FB}ct(\mathcal{G}))}, \mathcal{B}\Big) \cong \mathrm{Bicat}(\overline{\mathcal{FC}l(\iota\mathcal{G})}, \mathcal{B}) \tag{3.7}$$

For us, the moral is the following: Construction 3.1.16 gives precisely the rules required to freely define bicategorical substitution structure. In Section 3.2, we shall use this to construct a type theory for bicategories. Before that, we finish giving the definitions required to specify an equivalence of biclones. These will be a key part of the coherence result at the end of this chapter.

Relating biclone pseudofunctors. The definition of transformation between biclone homomorphisms is rather involved. There is a well-known notion of transformation between maps of multicategories (*e.g.* [Lei04, Definition 2.3.5]), but the cartesian nature of biclone substitution means the definition is not directly applicable. However, every clone canonically gives rise to a multicategory—we discuss this in some detail in Section 4.2—and this suggests the definition of transformation should be a bicategorical adaptation of that for multicategory maps. The definition of modification is then fixed.

The following notation is intended to be reminiscent of the notation $f \times g$ for the action of the categorical cartesian product on morphisms.

Notation 3.1.19. For multimaps $(f_i : \Gamma_i \to Y_i)_{i=1,\dots,n}$ and in a (bi)clone, one obtains the composite

$$\Gamma_1, \dots, \Gamma_n \xrightarrow{[\mathsf{p}^{(1+\Sigma_{i=1}^{k-1}|\Gamma_i|)}, \dots, \mathsf{p}^{(|\Gamma_k|+\Sigma_{i=1}^{k-1}|\Gamma_i|)}]} \Gamma_k \xrightarrow{f_k} Y_k$$

for $k = 1, \dots, n$. For $h : Y_1, \dots Y_n \to Z$ we therefore define $h[\boxtimes_{i=1}^{n} f_i] = h[f_1 \boxtimes \cdots \boxtimes f_n] : \Gamma_1, \dots, \Gamma_n \to Z$ to be the composite

$$h\Big[f_1\big[\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(|\Gamma_1|)}\big], \dots, f_n\Big[\mathsf{p}^{(1+\Sigma_{i=1}^{n-1}|\Gamma_i|)}, \dots, \mathsf{p}^{(|\Gamma_n|+\Sigma_{i=1}^{n-1}|\Gamma_i|)}\Big]\Big]$$

◀

In particular, for $(g_j : \Gamma \to X_j)_{j=1,\dots,n}$, $(f_i : X_i \to Y_i)_{i=1,\dots,n}$ and $h : Y_1, \dots, Y_n \to Z$ there exists a canonical isomorphism

$$\mathsf{f}_{h;f_\bullet;g_\bullet} : h[f_1 \boxtimes \cdots \boxtimes f_n]\,[g_1, \dots, g_n] \Rightarrow h[f_1[g_1], \dots, f_n[g_n]]$$

given by applying assoc twice and then the projections $\varrho^{(i)}$.

Definition 3.1.20. Let $F, G : (\mathcal{C}, S) \to (\mathcal{C}', S')$ be pseudofunctors of biclones. A *transformation* $(\alpha, \overline{\alpha}) : F \Rightarrow G$ consists of the following data:

1. A multimap $\alpha_X : FX \to GX$ for every $X \in S$,
2. An invertible 2-cell

$$\overline{\alpha}_t : \alpha_Y[Ft] \Rightarrow G(t)[\alpha_{X_1} \boxtimes \cdots \boxtimes \alpha_{X_n}] : FX_1, \dots, FX_n \to GY \tag{3.8}$$

   for every $t : X_1, \dots, X_n \to Y$ in $\mathcal{C}$, natural in t and satisfying the following two laws for $k = 1, \dots, n$:

$$\begin{tikzcd}
\alpha_Y[F(t)]\,[Fu_\bullet] \arrow[r, "\mathsf{assoc}_{\alpha;Ft;Fu_\bullet}"] \arrow[d, "\overline{\alpha}_t{[Fu_\bullet]}"'] & \alpha_Y[F(t)[Fu_\bullet]] \arrow[r, "\alpha_Y{[\phi_{t;u_\bullet}]}"] & \alpha_Y[F(t[u_\bullet])] \arrow[r, "\overline{\alpha}_{t[u_\bullet]}"] & G(t[u_\bullet])[\boxtimes_{i=1}^{n}\alpha_{X_i}] \\
G(t)[\boxtimes_{i=1}^{n}\alpha_{X_i}]\,[Fu_\bullet] \arrow[d, "\mathsf{f}_{Gt;\alpha_\bullet;Fu_\bullet}"'] & & & \\
G(t)[\alpha_{X_1}[Fu_1],\,\dots\,,\alpha_{X_n}[Fu_n]] \arrow[d, "G(t){\left[\overline{\alpha}_{u_1},\dots,\overline{\alpha}_{u_n}\right]}"'] & & & \\
G(t)[G(u_\bullet)[\boxtimes_{i=1}^{n}\alpha_{X_i}]] \arrow[rrr, "\mathsf{assoc}^{-1}_{Gt;Gu_\bullet;\boxtimes_i\alpha_{X_i}}"'] & & & G(t)[G(u_\bullet)]\,[\boxtimes_{i=1}^{n}\alpha_{X_i}] \arrow[uuu, "\phi_{t;u_\bullet}{\left[\boxtimes_{i=1}^{n}\alpha_{X_i}\right]}"']
\end{tikzcd}$$

$$\begin{tikzcd}
\mathsf{p}^{(k)}_{GX_\bullet}[\alpha_{X_1}\boxtimes\cdots\boxtimes\alpha_{X_n}] \arrow[r, "\psi^{(k)}_{X_\bullet}{\left[\alpha_{X_1}\boxtimes\cdots\boxtimes\alpha_{X_n}\right]}"] \arrow[d, "\varrho^{(k)}_{(\boxtimes_i\alpha_{X_i})}"'] & G(\mathsf{p}^{(k)}_{X_\bullet})[\alpha_{X_1}\boxtimes\cdots\boxtimes\alpha_{X_n}] \\
\alpha_{X_k}\left[\mathsf{p}^{(k)}_{FX_\bullet}\right] \arrow[r, "\alpha_{X_k}{\left[\psi^{(k)}_{X_\bullet}\right]}"'] & \alpha_{X_k}\left[F\mathsf{p}^{(k)}_{X_\bullet}\right] \arrow[u, "\overline{\alpha}_{(\mathsf{p}^{(k)}_{X_\bullet})}"']
\end{tikzcd}$$

◀

Definition 3.1.21. Let $(\alpha,\overline{\alpha}), (\beta,\overline{\beta}) : F \Rightarrow G$ be transformations of pseudofunctors $(S,\mathcal{C}) \to (S',\mathcal{C}')$. A *modification* $\Xi : (\alpha,\overline{\alpha}) \to (\beta,\overline{\beta})$ consists of a 2-cell $\Xi_X : \alpha_X \Rightarrow \beta_X$ for every $X \in S$, such that the following diagram commutes for every $t : X_1,\,\dots\,,X_n \to Y$:

$$\begin{tikzcd}
\alpha_Y[Ft] \arrow[r, "\Xi_Y{[Ft]}"] \arrow[d, "\overline{\alpha}_t"'] & \beta_Y[Ft] \arrow[d, "\overline{\beta}_t"] \\
G(t)[\alpha_{X_1}\boxtimes\cdots\boxtimes\alpha_{X_n}] \arrow[r, "G(t){\left[\Xi_{X_1}\boxtimes\cdots\boxtimes\Xi_{X_n}\right]}"'] & G(t)[\beta_{X_1}\boxtimes\cdots\boxtimes\beta_{X_n}]
\end{tikzcd}$$

◀

It is natural to conjecture that biclones together with their pseudofunctors, transformations and modifications form a tricategory **Biclone** into which **Bicat** embeds as a sub-tricategory. We do not pursue such considerations here, but we do give the definition of equivalence they would suggest.

Definition 3.1.22. A *biequivalence* between biclones $(S,\mathcal{C})$ and $(S',\mathcal{C}')$ consists of

- Pseudofunctors $F : \mathcal{C} \leftrightarrows \mathcal{C}' : G$,
- Pairs of transformations $(\alpha,\overline{\alpha}) : F \circ G \leftrightarrows \mathrm{id}_{\mathcal{C}'} : (\alpha',\overline{\alpha'})$ and $(\beta,\overline{\beta}) : G \circ F \leftrightarrows \mathrm{id}_{\mathcal{C}} : (\beta',\overline{\beta'})$,
- Invertible modifications $\Xi : \alpha \circ \alpha' \to \mathrm{id}_{\mathrm{id}_{\mathcal{C}'}}$, $\Xi' : \mathrm{id}_{FG} \to \alpha' \circ \alpha$, $\Psi : \beta \circ \beta' \to \mathrm{id}_{\mathrm{id}_{\mathcal{C}}}$ and $\Psi' : \mathrm{id}_{GF} \to \beta' \circ \beta$. ◀

Lemma 3.1.23. For any biequivalence $F : (S, \mathcal{C}) \leftrightarrows (S', \mathcal{C}') : G$ of biclones,

1. The pseudofunctor F is a *local equivalence*, *i.e.* every $F_{X_1,\ldots,X_n;Y} : \mathcal{C}(X_1, \ldots , X_n; Y) \to \mathcal{C}'(FX_1, \ldots , FX_n; FY)$ is full, faithful and essentially surjective,

2. For every $X' \in S'$ there exists $X \in S$ such that $FX \simeq X'$ in $\mathcal{C}'$.

Proof. Just as for categories and for bicategories, *c.f.* [Awo10, p. 173]. □

3.2 The type theory $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$

We now turn to constructing the type theory $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$ that will be the internal language of biclones. Following the general philosophy of Lambek's internal language for multicategories [Lam89], our approach is to define a term calculus for the rules of Construction 3.1.16. Thus, for every rule in the construction we postulate an introduction rule in the type theory. These rules are collected in Figures 3.3–3.5. Note that we slightly abuse notation by simultaneously introducing the structural isomorphisms (corresponding to assoc, ι and $\varrho^{(k)}$) and their inverses.

The equational theory $\equiv$ is derived directly from the axioms of a biclone; the rules are collected together in Figures 3.6–3.11. The typing rules respect this equational theory in the following sense.

Lemma 3.2.1. For any 2-multigraph $\mathcal{G}$ and derivable judgements $\Gamma \vdash \tau \equiv \tau' : t \Rightarrow t' : B$ in $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}(\mathcal{G})$, the judgements $\Gamma \vdash \tau : t \Rightarrow t' : B$ and $\Gamma \vdash \tau' : t \Rightarrow t' : B$ are derivable. □

We denote the type theory over a fixed 2-multigraph $\mathcal{G}$ by $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}(\mathcal{G})$; when we do not wish to specify a particular choice of signature, we simply write $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$.

In what follows we provide a more leisurely introduction to $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$ and establish some basic meta-theoretic properties.

Judgements. We must capture the fact that a biclone has both 1-cells and 2-cells: for this we follow the tradition of 2-dimensional type theories consisting of types, terms and *rewrites* (*c.f.* [See87, Hil96, Hir13]). Accordingly, there are two forms of typing judgement. Alongside the usual $\Gamma \vdash t : A$ to indicate 'term t has type A in context Γ', we write $\Gamma \vdash \tau : t \Rightarrow t' : A$ to indicate 'τ is a rewrite from term t of type A to term t' of type A, in context Γ'.

Contexts are finite lists of (variable, type) pairs in which variable names must not occur more than once: the relevant rules are given in Figure 3.1. Writing *Var* for the set of variables, any context Γ determines a finite partial function from variables to types; we write $\mathrm{dom}(\Gamma)$ for the domain of this function. The concatenation of contexts Γ and Δ satisfying $\mathrm{dom}(\Gamma) \cap \mathrm{dom}(\Delta) = \varnothing$ is denoted $\Gamma \mathbin{@} \Delta$.

$$\frac{}{\diamond\ \mathrm{ctx}} \qquad\qquad \frac{\Gamma\ \mathrm{ctx} \qquad x \notin \mathrm{dom}(\Gamma)}{\Gamma, x : A\ \mathrm{ctx}}\ (A \in \mathcal{G}_0)$$

Figure 3.1: Context-formation rules for $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}(\mathcal{G})$.

Raw terms. Following the template provided by clones, we may capture constants in a signature—that is, edges in a 2-multigraph—by constants in the type theory, and projections by variables. The outstanding question is how to model the substitution operation of a biclone. This cannot be the standard meta-operation of substitution: Construction 3.1.16 requires that substitution is not associative on the nose, only up to the `assoc` 2-cell. Our solution is to model the substitution operation of the free biclone by a form of *explicit substitution* [ACCL90]. For every family of terms $u_1, \ldots, u_n$ and term t with free variables among $x_1, \ldots, x_n$ we postulate a term $t\{x_1 \mapsto u_1, \ldots, x_n \mapsto u_n\}$; this is the formal analogue of the term $t[u_1/x_1, \ldots, u_n/x_n]$ defined by the meta-operation of capture-avoiding substitution (*c.f.* [ACCL90, RdP97]). The variables $x_1, \ldots, x_n$ are bound by this operation. For a fixed 2-multigraph $\mathcal{G}$ the *raw terms* are therefore variables, constant terms and explicit substitutions, as in the grammar

$$t, u_1, \ldots, u_n ::= x \mid c(x_1, \ldots, x_n) \mid t\{x_1 \mapsto u_1, \ldots, x_n \mapsto u_n\} \qquad (c \in \mathcal{G}(A_1, \ldots, A_n; B))$$

One may think of constants $c(x_1, \ldots, x_n)$ as n-ary operators: indeed, for every sequence of n terms $(u_1, \ldots, u_n)$ explicit substitution defines a mapping

$$(u_1, \ldots, u_n) \mapsto c(x_1, \ldots, x_n)\{x_1 \mapsto u_1, \ldots, x_n \mapsto u_n\}$$

This is emphasised by the following notational convention.

Notation 3.2.2. We adopt the following abuses of notation:

1. Writing $t\{x_i \mapsto u_i\}$ or just $t\{u_i\}$ for $t\{x_1 \mapsto u_1, \ldots, x_n \mapsto u_n\}$,
2. Writing $c\{u_1, \ldots, u_n\}$ for the explicit substitution $c(x_1, \ldots, x_n)\{x_i \mapsto u_i\}$ whenever c is a constant. ◂

Remark 3.2.3. Alternative notations for explicit substitution include $t\langle x := u\rangle$ and the let-binding operation `let` $x = u$ `in` t (*e.g.* [RdP97, DL11]). ◂

α-equivalence on terms. We work with terms up to α-equivalence defined in the standard way (*c.f.* [RdP97]).

Definition 3.2.4. For any 2-multigraph $\mathcal{G}$ we define the *α-equivalence relation* $=_\alpha$ on raw terms by the rules

$$\frac{}{t =_\alpha t}\,\text{refl} \qquad \frac{t =_\alpha t'}{t' =_\alpha t}\,\text{symm} \qquad \frac{t =_\alpha t' \qquad t' =_\alpha t''}{t =_\alpha t''}\,\text{trans}$$

$$\frac{t[y_i/x_i] =_\alpha t'[y_i/x'_i] \qquad (u_i =_\alpha u'_i)_{i=1,\dots,n} \qquad y_1, \dots, y_n \text{ fresh}}{t\{x_1 \mapsto u_1, \dots, x_n \mapsto u_n\} =_\alpha t\{x'_1 \mapsto u'_1, \dots, x'_1 \mapsto x'_n\}}$$

The simultaneous substitution operation $t[u_i/x_i]$ is defined by

$$\begin{aligned} x_k[u_i/x_i] &:= u_k \\ c(x_1, \dots, x_n)[u_i/x_i] &:= c\{u_1, \dots, u_n\} \\ (t\{z_j \mapsto u_j\})[v_i/x_i] &:= t\{z_j \mapsto u_j[v_i/x_i]\} \end{aligned}$$

where in the final rule we assume that each z_j does not occur among the x_i or freely in any of the v_i. ◂

Raw rewrites. Following the pattern set for terms, we define the class of *raw rewrites* between terms by the following grammar, where $t, u_\bullet$ and $v_\bullet$ are (families of) terms, $x_1, \dots, x_n$ are variables and $1 \leqslant i \leqslant n$:

$$\tau, \sigma, \sigma_1, \dots, \sigma_n ::= \mathsf{assoc}_{t;u_\bullet;v_\bullet} \mid \iota_t \mid \varrho^{(i)}_{u_\bullet} \mid \mathrm{id}_t \mid \kappa(x_1, \dots, x_n) \mid \tau \bullet \sigma \mid \tau\{x_1 \mapsto \sigma_n, \dots, x_n \mapsto \sigma_n\}$$

with a family of inverses (for $i = 1, \dots, n$), as follows:

$$\mathsf{assoc}^{-1}_{t;u_\bullet;v_\bullet} \mid \iota^{-1}_t \mid \varrho^{(-i)}_{u_\bullet}$$

Taking the rewrites in turn, we have invertible *structural rewrites* assoc, ι and $\varrho^{(i)}$ $(i = 1, \dots, n)$ and an identity rewrite id_t for every term t. Next, for every constant $\kappa \in \mathcal{G}(A_1, \dots, A_n; B)$ we have a constant rewrite $\kappa(x_1, \dots, x_n)$. Vertical composition is captured by a binary operation on rewrites (*c.f.* [Hil96, Hir13, LSR17]), while the explicit substitution operation mirrors that for terms. (Note that vertical composition follows function composition order, not diagrammatic order.) We adopt the standard category-theoretic convention of writing t for id_t where no ambiguity may arise, as well as adapting the conventions of Notation 3.2.2 to rewrites. In particular, one obtains *whiskering* operations $t\{\sigma\}$ and $\tau\{u\}$ for terms t, u and rewrites $\tau : t \Rightarrow t', \sigma : u \Rightarrow u'$.

α-equivalence on rewrites. The α-equivalence relation extends to rewrites in the way one would expect: as for terms, the substitution operation binds the variables being explicitly substituted for. The definition of the meta-operation of substitution on rewrites is analogous to that employed by Hilken [Hil96] and Hirschowitz [Hir13].

Definition 3.2.5. For any 2-multigraph $\mathcal{G}$ we define the *α-equivalence relation* $=_\alpha$ on rewrites by the rules

$$\frac{}{\tau =_\alpha \tau}\ \text{refl} \qquad \frac{\tau =_\alpha \tau'}{\tau' =_\alpha \tau}\ \text{symm} \qquad \frac{\tau =_\alpha \tau' \qquad \tau' =_\alpha \tau''}{\tau =_\alpha \tau''}\ \text{trans}$$

$$\frac{t =_\alpha t'}{\iota_t =_\alpha \iota_{t'}} \qquad \frac{u_1 =_\alpha u'_1 \quad \dots \quad u_n =_\alpha u'_n}{\varrho^{(k)}_{u_1, \dots, u_n} =_\alpha \varrho^{(k)}_{u'_1, \dots, u'_n}}\ 1 \leqslant k \leqslant n$$

$$\frac{(u_j =_\alpha u'_j)_{j=1, \dots, m} \qquad (v_i =_\alpha v'_i)_{i=1, \dots, n} \qquad t =_\alpha t'}{\mathsf{assoc}_{t, v_\bullet, u_\bullet} =_\alpha \mathsf{assoc}_{t', v'_\bullet, u'_\bullet}}$$

$$\frac{\tau =_\alpha \tau' \qquad \sigma =_\alpha \sigma'}{\tau \bullet \sigma =_\alpha \tau' \bullet \sigma'}$$

$$\frac{\tau[y_i/x_i] =_\alpha \tau'[y_i/x'_i] \qquad (\sigma_i =_\alpha \sigma'_i)_{i=1, \dots, n} \qquad y_1, \dots, y_n \text{ fresh}}{\tau\{x_1 \mapsto \sigma_1, \dots, x_n \mapsto \sigma_n\} =_\alpha \tau\{x'_1 \mapsto \sigma'_1, \dots, x'_1 \mapsto \sigma'_n\}}$$

The meta-operation of capture-avoiding substitution is extended to rewrites as follows:

$$\begin{aligned}
\iota_u[u_i/x_i] &:= \iota_{u[u_i/x_i]} \\
\varrho^{(k)}_{t_1, \dots, t_n}[u_i/x_i] &:= \varrho^{(k)}_{t_\bullet[u_i/x_i]} \\
\mathsf{assoc}_{t, u_\bullet, v_\bullet}[u_i/x_i] &:= \mathsf{assoc}_{t[u_i/x_i], u_\bullet[u_i/x_i], v_\bullet[u_i/x_i]} \\
\kappa(x_1, \dots, x_n)[u_i/x_i] &:= \kappa\{u_1, \dots, u_n\} \\
(\tau' \bullet \tau)[u_i/x_i] &:= \tau'[u_i/x_i] \bullet \tau[u_i/x_i] \\
\mathrm{id}_t[u_i/x_i] &:= \mathrm{id}_{t[u_i/x_i]} \\
(\tau\{z_j \mapsto \sigma_j\})[u_i/x_i] &:= \tau\{z_j \mapsto \sigma_j[u_i/x_i]\}
\end{aligned}$$

where in the final rule we assume that each z_j does not occur among the x_i or freely in any of the u_i. These rules extend to the inverses of rewrites in the obvious fashion. ◂

A structural induction shows the typing judgement respects α-equivalence.

Lemma 3.2.6. Let $\mathcal{G}$ be a 2-multigraph. Then in $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}(\mathcal{G})$:

1. If $\Gamma \vdash t : B$ and $t =_\alpha t'$ then $\Gamma \vdash t' : B$,
2. If $\Gamma \vdash \tau : t \Rightarrow t' : B$ and $\tau =_\alpha \tau'$ then $\Gamma \vdash \tau : t \Rightarrow t' : B$. □

In an explicit substitution calculus the structural operations manifest themselves in a correspondingly explicit manner. Indeed, the fact that $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$ admits arbitrary context renamings follows immediately from the `horiz-comp` rule.

Definition 3.2.7. Let $\Gamma := (x_i : A_i)_{i=1,\dots,n}$ and $\Delta := (y_j : B_j)_{j=1,\dots,m}$ be contexts. A *context renaming* $r : \Gamma \to \Delta$ is a mapping $r : \{x_1, \dots, x_n\} \to \{y_1, \dots, y_m\}$ on variables which respects typing in the sense that whenever $r(x_i) = y_j$ then $A_i = B_j$. ◂

The following rules are then derivable for any context renaming r.

$$\frac{\Gamma \vdash t : A \qquad r : \Gamma \to \Delta}{\Delta \vdash t\{x_1 \mapsto r(x_1), \dots, x_n \mapsto r(x_n)\} : A}$$

$$\frac{\Gamma \vdash \tau : t \Rightarrow t' : A \qquad r : \Gamma \to \Delta}{\Delta \vdash \tau\{x_i \mapsto r(x_i)\} : t\{x_i \mapsto r(x_i)\} \Rightarrow t'\{x_i \mapsto r(x_i)\} : A}$$

Figure 3.2: Context renaming as a derived rule (for $\Gamma = (x_i : A_i)_{i=1,\dots,n}$)

Weakening arises as a special case: for a fresh variable $x \notin \mathrm{dom}(\Gamma)$, one takes the inclusion $\mathrm{inc}_x : \Gamma \hookrightarrow \Gamma, x : A$.

Notation 3.2.8. For a context renaming r we write $t\{r\}$ and $\tau\{r\}$ for the terms and rewrites formed using the admissible rules of Figure 3.2. ◂

$$\frac{}{x_1 : A_1, \dots, x_n : A_n \vdash x_k : A_k}\ \text{var } (1 \leqslant k \leqslant n) \qquad \frac{c \in \mathcal{G}(A_1, \dots, A_n; B)}{x_1 : A_1, \dots, x_n : A_n \vdash c(x_1, \dots, x_n) : B}\ \text{const}$$

$$\frac{x_1 : A_1, \dots, x_n : A_n \vdash t : B \qquad (\Delta \vdash u_i : A_i)_{i=1,\dots,n}}{\Delta \vdash t\{x_1 \mapsto u_1, \dots, x_n \mapsto u_n\} : B}\ \text{horiz-comp}$$

Figure 3.3: Introduction rules on basic terms

$$\frac{x_1 : A_1, \dots, x_n : A_n \vdash t : B}{\begin{array}{c} x_1 : A_1, \dots, x_n : A_n \vdash \iota_t : t \Rightarrow t\{x_i \mapsto x_i\} : B \\ x_1 : A_1, \dots, x_n : A_n \vdash \iota_t^{-1} : t\{x_i \mapsto x_i\} \Rightarrow t : B \end{array}}\ \iota\text{-intro}$$

$$\frac{x_1 : A_1, \dots, x_n : A_n \vdash x_k : A_k \qquad (\Delta \vdash u_i : A_i)_{i=1,\dots,n}}{\begin{array}{c} \Delta \vdash \varrho^{(k)}_{u_1,\dots,u_n} : x_k\{x_i \mapsto u_i\} \Rightarrow u_k : A_k \\ \Delta \vdash \varrho^{(-k)}_{u_1,\dots,u_n} : u_k \Rightarrow x_k\{x_i \mapsto u_i\} : A_k \end{array}}\ \varrho^{(k)}\text{-intro } (1 \leqslant k \leqslant n)$$

$$\frac{\begin{array}{c} (\Delta \vdash u_j : A_j)_{j=1,\dots m} \\ (x_1 : A_1, \dots, x_m : A_m \vdash v_i : B_i)_{i=1,\dots,n} \\ y_1 : B_1, \dots, y_n : B_n \vdash t : C \end{array}}{\begin{array}{c} \Delta \vdash \mathsf{assoc}_{t,v_\bullet,u_\bullet} : t\{y_i \mapsto v_i\}\{x_j \mapsto u_j\} \Rightarrow t\{y_i \mapsto v_i\{x_j \mapsto u_j\}\} : C \\ \Delta \vdash \mathsf{assoc}^{-1}_{t,v_\bullet,u_\bullet} : t\{y_i \mapsto v_i\{x_j \mapsto u_j\}\} \Rightarrow t\{y_i \mapsto v_i\}\{x_j \mapsto u_j\} : C \end{array}}\ \text{assoc-intro}$$

Figure 3.4: Introduction rules on structural rewrites

$$\frac{\Gamma \vdash t : A}{\Gamma \vdash \mathrm{id}_t : t \Rightarrow t : A}\ \text{id-intro}$$

$$\frac{\kappa \in \mathcal{G}(A_1, \dots, A_n; B)(c, c')}{x_1 : A_1, \dots, x_n : A_n \vdash \kappa(x_1, \dots, x_n) : c(x_1, \dots, x_n) \Rightarrow c'(x_1, \dots, x_n) : B}\ \text{2-const}$$

$$\frac{\Gamma \vdash \tau : t \Rightarrow t' : A \qquad \Gamma \vdash \tau' : t' \Rightarrow t'' : A}{\Gamma \vdash \tau' \bullet \tau : t \Rightarrow t'' : A}\ \text{vert-comp}$$

$$\frac{x_1 : A_1, \dots, x_n : A_n \vdash \tau : t \Rightarrow t' : B \qquad (\Delta \vdash \sigma_i : u_i \Rightarrow u_i' : A_i)_{i=1,\dots,n}}{\Delta \vdash \tau\{x_i \mapsto \sigma_i\} : t\{x_i \mapsto u_i\} \Rightarrow t'\{x_i \mapsto u_i'\} : B}\ \text{horiz-comp}$$

Figure 3.5: Introduction rules on basic rewrites

Introduction rules for terms, structural rewrites and basic rewrites in $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}(\mathcal{G})$.

$$\frac{\Gamma \vdash \tau : t \Rightarrow t' : A}{\Gamma \vdash \tau \bullet \mathrm{id}_t \equiv \tau : t \Rightarrow t' : A}\ \bullet\text{-right-unit} \qquad \frac{\Gamma \vdash \tau : t \Rightarrow t' : A}{\Gamma \vdash \tau \equiv \mathrm{id}_{t'} \bullet \tau : t \Rightarrow t' : A}\ \bullet\text{-left-unit}$$

$$\frac{\Gamma \vdash \tau'' : t'' \Rightarrow t''' : A \qquad \Gamma \vdash \tau' : t' \Rightarrow t'' : A \qquad \Gamma \vdash \tau : t \Rightarrow t' : A}{\Gamma \vdash (\tau'' \bullet \tau') \bullet \tau \equiv \tau'' \bullet (\tau' \bullet \tau) : t \Rightarrow t''' : A}\ \bullet\text{-assoc}$$

Figure 3.6: Categorical structure of vertical composition

$$\frac{x_1 : A_1, \dots, x_n : A_n \vdash t : B \qquad (\Delta \vdash u_i : A_i)_{i=1,\dots,n}}{\Delta \vdash \mathrm{id}_t\{x_i \mapsto u_i\} \equiv \mathrm{id}_{t\{x_i \mapsto u_i\}} : t\{x_i \mapsto u_i\} \Rightarrow t\{x_i \mapsto u_i\} : B}\ \text{id-preservation}$$

$$\frac{\begin{array}{cc} x_1 : A_1, \dots, x_n : A_n \vdash \tau : t \Rightarrow t' : B & (\Delta \vdash \sigma_i : u_i \Rightarrow u_i' : A_i)_{i=1,\dots,n} \\ x_1 : A_1, \dots, x_n : A_n \vdash \tau' : t' \Rightarrow t'' : B & (\Delta \vdash \sigma_i' : u_i' \Rightarrow u_i'' : A_i)_{i=1,\dots,n} \end{array}}{\Delta \vdash \tau'\{x_i \mapsto \sigma_i'\} \bullet \tau\{x_i \mapsto \sigma_i\} \equiv (\tau' \bullet \tau)\{x_i \mapsto \sigma_i' \bullet \sigma_i\} : t\{x_i \mapsto u_i\} \Rightarrow t''\{x_i \mapsto u_i''\} : B}\ \text{interchange}$$

Figure 3.7: Preservation rules

$$\frac{(\Delta \vdash \sigma_i : u_i \Rightarrow u_i' : A_i)_{i=1,\dots,n}}{\Delta \vdash \varrho^{(k)}_{u_1',\dots,u_n'} \bullet x_k\{x_i \mapsto \sigma_i\} \equiv \sigma_k \bullet \varrho^{(k)}_{u_1,\dots,u_n} : x_k\{x_i \mapsto u_i\} \Rightarrow u_k' : A_k}\ (1 \leqslant k \leqslant n)$$

$$\frac{x_1 : A_1, \dots, x_n : A_n \vdash \tau : t \Rightarrow t' : B}{x_1 : A_1, \dots, x_n : A_n \vdash \iota_{t'} \bullet \tau \equiv \tau\{x_i \mapsto x_i\} \bullet \iota_t : t \Rightarrow t'\{x_i \mapsto x_i\} : B}$$

$$\frac{\begin{array}{c} (\Delta \vdash \mu_j : u_j \Rightarrow u_j' : A_j)_{j=1,\dots m} \\ (x_1 : A_1, \dots, x_m : A_m \vdash \sigma_i : v_i \Rightarrow v_i' : B_i)_{i=1,\dots,n} \\ y_1 : B_1, \dots, y_n : B_n \vdash \tau : t \Rightarrow t' : C \end{array}}{\begin{array}{l} \Delta \vdash \mathsf{assoc}_{t',v_\bullet,u_\bullet} \bullet \tau\{y_i \mapsto \sigma_i\}\{x_j \mapsto \mu_j\} \equiv \tau\{y_i \mapsto \sigma_i\{x_j \mapsto \mu_j\}\} \bullet \mathsf{assoc}_{t,v_\bullet,u_\bullet} \\ \qquad\qquad : t\{y_i \mapsto v_i\}\{x_j \mapsto u_j\} \Rightarrow t'\{y_i \mapsto v_i'\{x_j \mapsto u_j'\}\} : C \end{array}}$$

Figure 3.8: Naturality rules on structural rewrites

$$\frac{x_1 : A_1, \dots, x_n : A_n \vdash t : B \qquad (\Delta \vdash u_i : A_i)_{i=1,\dots,n}}{\Delta \vdash t\{x_i \mapsto \varrho^{(i)}_{u_\bullet}\} \bullet \mathsf{assoc}_{t,x_\bullet,u_\bullet} \bullet \iota_t\{x_i \mapsto u_i\} \equiv \mathrm{id}_{t\{x_i \mapsto u_i\}} : t\{x_i \mapsto u_i\} \Rightarrow t\{x_i \mapsto u_i\} : B}$$

$$\frac{\begin{array}{cc} (\Delta \vdash u_j : A_j)_{j=1,\dots m} & (y_1 : B_1, \dots, y_n : B_n \vdash w_j : C_k)_{k=1,\dots,l} \\ (x_1 : A_1, \dots, x_m : A_m \vdash v_i : B_i)_{i=1,\dots,n} & z_1 : C_1, \dots, z_l : C_l \vdash t : D \end{array}}{\begin{array}{l} \Delta \vdash t\{z_k \mapsto \mathsf{assoc}_{w_k,v_\bullet,u_\bullet}\} \bullet \mathsf{assoc}_{t,w_\bullet\{y_j \mapsto v_j\},u_\bullet} \bullet \mathsf{assoc}_{t,w_\bullet,v_\bullet}\{x_j \mapsto u_j\} \\ \qquad \equiv \mathsf{assoc}_{t,w_\bullet,v_\bullet\{x_j \mapsto u_i\}} \bullet \mathsf{assoc}_{t\{z_k \mapsto w_k\},v_\bullet,u_\bullet} \\ \qquad\quad : t\{z_k \mapsto w_k\}\{y_i \mapsto v_i\}\{x_j \mapsto u_j\} \Rightarrow t\{z_k \mapsto w_k\{y_i \mapsto v_i\{x_j \mapsto u_j\}\}\} : D \end{array}}$$

Figure 3.9: Biclone laws

Equational theory for structural rewrites in $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}(\mathcal{G})$.

$$\frac{\Gamma \vdash t : B}{\Gamma \vdash \iota_t^{-1} \bullet \iota_t \equiv \mathrm{id}_t : t \Rightarrow t : B}$$

$$\frac{x_1 : A_1, \ldots, x_n : A_n \vdash t : B}{x_1 : A_1, \ldots, x_n : A_n \vdash \iota_t \bullet \iota_t^{-1} \equiv \mathrm{id}_t : t\{x_i \mapsto x_i\} \Rightarrow t\{x_i \mapsto x_i\} : B}$$

$$\frac{x_1 : A_1, \ldots, x_n : A_n \vdash u_1 : A_1 \quad \ldots \quad x_1 : A_1, \ldots, x_n : A_n \vdash u_n : A_n}{x_1 : A_1, \ldots, x_n : A_n \vdash \varrho_{u_\bullet}^{(-k)} \bullet \varrho_{u_\bullet}^{(k)} \equiv \mathrm{id}_{x_k\{x_i \mapsto u_i\}} : x_k\{x_i \mapsto u_i\} \Rightarrow x_k\{x_i \mapsto u_i\} : A_k} \; (1 \leqslant k \leqslant n)$$

$$\frac{x_1 : A_1, \ldots, x_n : A_n \vdash u : B}{x_1 : A_1, \ldots, x_n : A_n \vdash \varrho_{u_\bullet}^{(k)} \bullet \varrho_{u_\bullet}^{(-k)} \equiv \mathrm{id}_u : u \Rightarrow u : A} \; (1 \leqslant k \leqslant n)$$

$$\frac{\begin{array}{c}(\Delta \vdash u_j : A_j)_{j=1,\ldots m} \\ (x_1 : A_1, \ldots, x_m : A_m \vdash v_i : B_i)_{i=1,\ldots,n} \qquad y_1 : B_1, \ldots, y_n : B_n \vdash t : C\end{array}}{\Delta \vdash \mathsf{assoc}_{t,v_\bullet,u_\bullet}^{-1} \bullet \mathsf{assoc}_{t,v_\bullet,u_\bullet} \equiv \mathrm{id}_{t\{v_i\}\{u_j\}} : t\{y_i \mapsto v_i\}\{x_j \mapsto u_j\} \Rightarrow t\{y_i \mapsto v_i\}\{x_j \mapsto u_j\} : C}$$

$$\frac{\begin{array}{c}(\Delta \vdash u_j : A_j)_{j=1,\ldots m} \\ (x_1 : A_1, \ldots, x_m : A_m \vdash v_i : B_i)_{i=1,\ldots,n} \qquad y_1 : B_1, \ldots, y_n : B_n \vdash t : C\end{array}}{\Delta \vdash \mathsf{assoc}_{t,v_\bullet,u_\bullet} \bullet \mathsf{assoc}_{t,v_\bullet,u_\bullet}^{-1} \equiv \mathrm{id}_{t\{v_i\{u_j\}\}} : t\{y_i \mapsto v_i\{x_j \mapsto u_j\}\} \Rightarrow t\{y_i \mapsto v_i\{x_j \mapsto u_j\}\} : C}$$

Figure 3.10: Invertibility of the structural rewrites

$$\frac{\Gamma \vdash \tau : t \Rightarrow t' : A}{\Gamma \vdash \tau \equiv \tau : t \Rightarrow t' : A} \,\text{refl} \qquad \frac{\Gamma \vdash \tau \equiv \tau' : t \Rightarrow t' : A}{\Gamma \vdash \tau' \equiv \tau : t \Rightarrow t' : A} \,\text{symm}$$

$$\frac{\Gamma \vdash \tau' \equiv \tau'' : t \Rightarrow t' : A \qquad \Gamma \vdash \tau \equiv \tau' : t \Rightarrow t' : A}{\Gamma \vdash \tau \equiv \tau'' : t \Rightarrow t' : A} \,\text{trans}$$

$$\frac{\Gamma \vdash \tau' \equiv \sigma' : t' \Rightarrow t'' : A \qquad \Gamma \vdash \tau \equiv \sigma : t \Rightarrow t' : A}{\Gamma \vdash (\tau' \bullet \tau) \equiv (\sigma' \bullet \sigma) : t \Rightarrow t'' : A}$$

$$\frac{x_1 : A_1, \ldots, x_n : A_n \vdash \tau \equiv \tau' : t \Rightarrow t' : B \qquad (\Delta \vdash \sigma_i \equiv \sigma_i' : u_i \Rightarrow u_i' : A_i)_{i=1,\ldots,n}}{\Delta \vdash \tau\{x_i \mapsto \sigma_i\} \equiv \tau'\{x_i \mapsto \sigma_i'\} : t\{x_i \mapsto u_i\} \Rightarrow t'\{x_i \mapsto u_i'\} : B}$$

Figure 3.11: Congruence laws

Equational theory for structural rewrites in $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}(\mathcal{G})$.

Well-formedness properties of $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$. We finish this introduction to $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$ by showing that it satisfies versions of the standard syntactic properties of, for example, the simply-typed lambda calculus (*c.f.* [Cro94, Chapter 4]). The intention is to justify the claim that the properties one would expect by analogy with the simply-typed lambda calculus do in fact hold. The proofs are all straightforward structural inductions.

Definition 3.2.9. Fix a 2-multigraph $\mathcal{G}$. We define the *free variables in a term t* in $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}(\mathcal{G})$ as follows:

$$\begin{aligned}
\mathrm{fv}(x_i) &:= \{x_i\} \qquad \text{for } x_i \text{ a variable,} \\
\mathrm{fv}\big(c(x_1, \dots, x_n)\big) &:= \{x_1, \dots, x_n\} \qquad \text{for } c \in \mathcal{G}(A_1, \dots, A_n; B), \\
\mathrm{fv}(t\{x_1 \mapsto u_1, \dots, x_n \mapsto u_n\}) &:= (\mathrm{fv}(t) - \{x_1, \dots, x_n\}) \cup \textstyle\bigcup_{i=1}^{n} \mathrm{fv}(u_i)
\end{aligned}$$

Similarly, define the *free variables in a rewrite τ* in $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}(\mathcal{G})$ as follows:

$$\begin{aligned}
\mathrm{fv}\big(\iota_t\big) &:= \mathrm{fv}(t) \\
\mathrm{fv}\big(\varrho^{(k)}_{u_1, \dots, u_n}\big) &:= \mathrm{fv}(u_k) \\
\mathrm{fv}\big(\mathsf{assoc}_{t, v_\bullet, u_\bullet}\big) &:= \textstyle\bigcup_{i=1}^{n} \mathrm{fv}(u_i) \\
\mathrm{fv}(\mathrm{id}_t) &:= \mathrm{fv}(t) \\
\mathrm{fv}(\tau' \bullet \tau) &:= \mathrm{fv}(\tau') \cup \mathrm{fv}(\tau) \\
\mathrm{fv}\big(\sigma(x_1, \dots, x_n)\big) &:= \{x_1, \dots, x_n\} \text{ for } \sigma \in \mathcal{G}(A_1, \dots, A_n; B)(c, c') \\
\mathrm{fv}(\tau\{x_1 \mapsto \sigma_1, \dots, x_n \mapsto \sigma_n\}) &:= (\mathrm{fv}(\tau) - \{x_1, \dots, x_n\}) \cup \textstyle\bigcup_{i=1}^{n} \mathrm{fv}(\sigma_i)
\end{aligned}$$

We define the free variables of a specified inverse σ^{-1} to be exactly the free variables of σ. An occurrence of a variable in a term (rewrite) is *bound* if it is not free. ◂

Lemma 3.2.10. Let $\mathcal{G}$ be a 2-multigraph. For any derivable judgements $\Gamma \vdash u : B$ and $\Gamma \vdash \tau : t \Rightarrow t' : B$ in $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}(\mathcal{G})$,

1. $\mathrm{fv}(u) \subseteq \mathrm{dom}(\Gamma)$,
2. $\mathrm{fv}(\tau) \subseteq \mathrm{dom}(\Gamma)$,
3. The judgements $\Gamma \vdash t : B$ and $\Gamma \vdash t' : B$ are both derivable.

Moreover, for any context $\Gamma := (x_i : A_i)_{i=1, \dots, n}$ and derivable terms $(\Delta \vdash u_i : A_i)_{i=1, \dots, n}$,

1. If $\Gamma \vdash t : B$, then $\Delta \vdash t[u_i/x_i] : B$,
2. If $\Gamma \vdash \tau : t \Rightarrow t' : B$, then $\Delta \vdash \tau[u_i/x_i] : t[u_i/x_i] \Rightarrow t[u_i/x_i] : B$. □

3.2.1 The syntactic model

The rules of $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$ are synthesised directly from the construction of the free biclone on a 2-multigraph. It is not surprising, therefore, that its syntactic model satisfies the same free property, justifying our description of $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$ as a type theory for biclones. In this section we spell out the construction and show that it restricts to bicategories.

Constructing the syntactic model is a matter of reversing the correspondence between the rules of $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$ and Construction 3.1.16.

Construction 3.2.11. For any 2-multigraph $\mathcal{G}$ define the *syntactic model* $\mathrm{Syn}(\mathcal{G})$ of $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}(\mathcal{G})$ as follows. The sorts are nodes $A, B, \dots$ of $\mathcal{G}$. For $A_1, \dots, A_n, B \in \mathcal{G}_0$ the hom-category $\mathrm{Syn}(\mathcal{G})(A_1, \dots, A_n; B)$ has objects α-equivalence classes of terms $(x_1 : A_1, \dots, x_n : A_n \vdash t : B)$ derivable in $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}(\mathcal{G})$. We assume a fixed enumeration $x_1, x_2, \dots$ of variables, and that the variable name in the ith position is determined by this enumeration. Morphisms in $\mathrm{Syn}(\mathcal{G})(A_1, \dots, A_n; B)$ are $\alpha{\equiv}$-equivalence classes of rewrites $(x_1 : A_1, \dots, x_n : A_n \vdash \tau : t \Rightarrow t' : B)$. Composition is vertical composition and the identity is id_t.

The substitution operation $\big(t, (u_1, \dots, u_n)\big) \mapsto t[u_1, \dots, u_n]$ is explicit substitution

$$t, (u_1, \dots, u_m) \mapsto t\{x_1 \mapsto u_1, \dots, x_n \mapsto u_n\}$$
$$\tau, (\sigma_1, \dots, \sigma_m) \mapsto \tau\{x_1 \mapsto \sigma_1, \dots, x_n \mapsto \sigma_n\}$$

and the projections $(A_1, \dots, A_n) \to A_k$ are instances of the `var` rule $x_1 : A_1, \dots, x_n : A_n \vdash x_k : A_k$ for $k = 1, \dots, n$. The 2-cells assoc, ι and $\varrho^{(k)}$ are the corresponding structural rewrites. ◂

Notation 3.2.12. We shall generally play fast and loose with the requirement that the variables in a context $(x_1 : A_1, \dots, x_n : A_n)$ are labelled in turn by the enumeration $x_1, \dots, x_n, \dots$. We will allow ourselves to pick more meaningful variable names as a simple form of syntactic sugar, and rely on the fact that the proper variable names can always be recovered when required. ◂

The equational theory guarantees that $\mathrm{Syn}(\mathcal{G})$ is a biclone. The proof of the free property mirrors Lemma 3.1.17.

Lemma 3.2.13. For any 2-multigraph $\mathcal{G}$, biclone $(S, \mathcal{C})$ and 2-multigraph homomorphism $h : \mathcal{G} \to \mathcal{C}$ there exists a unique strict pseudofunctor $h[\![-]\!] : \mathrm{Syn}(\mathcal{G}) \to \mathcal{C}$ such that $h[\![-]\!] \circ \iota = h$, for $\iota : \mathcal{G} \hookrightarrow \mathrm{Syn}(\mathcal{G})$ the inclusion.

Proof. Fix a context $\Gamma := (x_i : A_i)_{i=1,\dots,n}$. We define $h[\![-]\!]$ by induction on the derivation of judgements in $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$:

$$\begin{aligned}
h\llbracket B \rrbracket &:= h(B) \qquad \text{on types} \\
h\llbracket \Gamma \vdash c(x_1, \dots, x_n) : B \rrbracket &:= h(c) \qquad \text{for } c \in \mathcal{G}(A_\bullet; B) \\
h\llbracket \Delta \vdash t\{x_i \mapsto u_i\} : B \rrbracket &:= \big(h\llbracket \Gamma \vdash t : B \rrbracket\big)[h\llbracket \Delta \vdash u_\bullet : A_\bullet \rrbracket] \\
h\llbracket \Gamma \vdash \mathrm{id}_t : t \Rightarrow t : B \rrbracket &:= \mathrm{id}_{h\llbracket \Gamma \vdash t : B \rrbracket} \\
h\llbracket \Gamma \vdash \kappa(x_\bullet) : c(x_\bullet) \Rightarrow c'(x_\bullet) : B \rrbracket &:= h(\kappa) \qquad \text{for } \kappa \in \mathcal{G}(A_\bullet, B)(c, c') \\
h\llbracket \Gamma \vdash \tau' \bullet \tau : t \Rightarrow t'' : B \rrbracket &:= h\llbracket \Gamma \vdash \tau' : t' \Rightarrow t'' : B \rrbracket \bullet h\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket \\
h\llbracket \tau\{x_i \mapsto \sigma_i\} \rrbracket &:= \big(h\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket\big)[h\llbracket \Delta \vdash \sigma_\bullet : u_\bullet \Rightarrow u'_\bullet : A_\bullet \rrbracket]
\end{aligned}$$

where we omit the full typing derivation $\Delta \vdash \tau\{x_i \mapsto \sigma_i\} : t\{x_i \mapsto u_i\} \Rightarrow t'\{x_i \mapsto u'_i\} : B$ in the final case for reasons of space. In order for $h\llbracket - \rrbracket$ to be strict we must require that it strictly preserves the assoc, ι and $\varrho^{(k)}$ 2-cells. Uniqueness holds just as in Lemma 3.1.17. □

Theorem 3.2.14. For any 2-multigraph $\mathcal{G}$, the syntactic model $\mathrm{Syn}(\mathcal{G})$ of $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}(\mathcal{G})$ is the free biclone on $\mathcal{G}$. □

A type theory satisfying a property of this form, and which is therefore sound and complete for reasoning in the freely constructed structure, is often referred to as the *internal language* or *internal logic* (*e.g.* [MR77, LS86, Cro94, GK13]). This terminology is used with varying degrees of precision, and generally not in the precise sense of Lambek [Lam89, Definition 5.3]; nonetheless, we may now justifiably state that $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}$ is the internal language of biclones.

By the theorem, we may identify $\mathrm{Syn}(\mathcal{G})$ with the free biclone $\mathcal{FCl}(\mathcal{G})$ on $\mathcal{G}$. The diagram of adjunctions (3.6) (p. 48) then entails that for a 2-graph $\mathcal{G}$ the nucleus of $\mathrm{Syn}(\mathcal{G})$—obtained by restricting the syntactic model of $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}$ to unary multimaps—is the free bicategory on $\mathcal{G}$. Equivalently, one may restrict the type theory $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}$ to unary contexts and construct its syntactic model as in Construction 3.2.11. Let $\Lambda^{\mathrm{bicat}}_{\mathrm{ps}}$ denote the type theory obtained by replacing the context-formation rules of Figure 3.1 with the single rule of Figure 3.12.

$$\frac{}{x : A \ \mathrm{ctx}}\ (A \in \mathcal{G}_0)$$

Figure 3.12: Context-formation rule for $\Lambda^{\mathrm{bicat}}_{\mathrm{ps}}(\mathcal{G})$.

Construction 3.2.15. For any 2-graph $\mathcal{G}$, define a bicategory $\mathrm{Syn}(\mathcal{G})\big|_1$ as follows. Objects are unary contexts $(x : A)$ for x a *fixed* variable name. The hom-category $\mathrm{Syn}(\mathcal{G})\big|_1\big((x : A), (x : B)\big)$ has objects α-equivalence classes of derivable terms $(x : A \vdash t : B)$ in $\Lambda_{\mathrm{ps}}^{\mathrm{bicat}}$ and morphisms $\alpha\equiv$-equivalence classes of rewrites $(x : A \vdash \tau : t \Rightarrow t' : B)$ in $\Lambda_{\mathrm{ps}}^{\mathrm{bicat}}$. Vertical composition is the $\bullet$ operation. Horizontal composition is given by explicit substitution and the identity on $(x : A)$ by the $\mathtt{var}$ rule $(x : A \vdash x : A)$. The structural isomorphisms l, r and a are ϱ, ι^{-1} and assoc, respectively. ◂

Remark 3.2.16. The structural isomorphism r is given by ι^{-1} because we have directed the structural isomorphisms in a biclone to match that of a skew monoidal category, but followed Bénabou's convention [Bén67] directing the unitors in a bicategory to remove compositions with the identity. ◂

The required theorem follows immediately from Theorem 3.2.14 and the chain of isomorphisms (3.7) (p. 48).

Theorem 3.2.17. *For any 2-graph $\mathcal{G}$, the syntactic model $\mathrm{Syn}(\mathcal{G})\big|_1$ of $\Lambda_{\mathrm{ps}}^{\mathrm{bicat}}(\mathcal{G})$ is the free bicategory on $\mathcal{G}$.* □

The restriction to a fixed variable name is necessary for the free property to be strict. Without such a restriction there are countably many equivalent objects $(x_1 : A), (x_2 : A), \dots$ in $\mathrm{Syn}(\mathcal{G})\big|_1$, and the action of the pseudofunctor defined in Lemma 3.2.13 is unique only up to its action on each variable name. The next lemma shows that—up to biequivalence—this restriction is immaterial.

Lemma 3.2.18. *Let $\mathcal{B}$ be a bicategory and $\mathcal{S}$ a sub-bicategory. Suppose that for every $X \in \mathcal{B}$ there exists a chosen $[X] \in \mathcal{S}$ with a specified adjoint equivalence $f_X : X \leftrightarrows [X] : g_X$ in $\mathcal{B}$ such that*

1. *For $X \in \mathcal{S}$ the equivalence $X \simeq [X]$ is the identity, and*
2. *If $h : X \to Y$ is a 1-cell in $\mathcal{S}$, then so is the composite $(g_Y \circ h) \circ f_X : [X] \to [Y]$.*

Then $\mathcal{B}$ and $\mathcal{S}$ are biequivalent.

Proof. Let us denote the 2-cells witnessing the equivalence $X \simeq [X]$ by

$$\mathsf{v}_X : \mathrm{Id}_{[X]} \Rightarrow g_X \circ f_X$$
$$\mathsf{w}_X : f_X \circ g_X \Rightarrow \mathrm{Id}_X$$

There exists an evident pseudofunctor $\iota : \mathcal{S} \hookrightarrow \mathcal{B}$ given by the inclusion. In the other direction, we define $E : \mathcal{B} \to \mathcal{S}$ by setting

$$E(X) := [X] \quad \text{and} \quad E(\tau : t \Rightarrow t' : X \to Y) := (g_Y \circ \tau) \circ f_X$$

We then define $\psi_X := \mathrm{Id}_{[X]} \xRightarrow{\mathsf{v}_X} g_X \circ f_X \xRightarrow{\cong} (g_X \circ \mathrm{Id}_X) \circ h_X = E(\mathrm{Id}_X)$. For a composable pair $X \xrightarrow{u} Y \xrightarrow{t} Z$ we define $\phi_{t,u}$ by commutativity of the following diagram:

$$\begin{tikzcd}
(g_Z \circ (t \circ f_Y))) \circ (g_Y \circ (u \circ f_X)) \arrow[rr, "\phi_{t,u}"] \arrow[d, "\cong"'] & & g_Z \circ ((t \circ u) \circ f_X) \\
(g_Z \circ t) \circ ((f_Y \circ g_Y) \circ (u \circ f_X)) \arrow[rr, "(g_Z \circ t) \circ (\mathsf{w}_Y \circ (u \circ f_X))"'] & & (g_Z \circ t) \circ (\mathrm{Id}_Y \circ (u \circ f_X)) \arrow[u, "\cong"']
\end{tikzcd}$$

The unit and associativity laws for a pseudofunctor follow from coherence and the triangle laws of an adjoint equivalence. We then need to construct pseudonatural transformations $(\alpha, \overline{\alpha}) : \mathrm{id}_{\mathcal{B}} \leftrightarrows \iota \circ E : (\beta, \overline{\beta})$ and $(\gamma, \overline{\gamma}) : \mathrm{id}_{\mathcal{S}} \leftrightarrows E \circ \iota : (\delta, \overline{\delta})$.

For α, we take $\alpha_X := g_X$ and $\overline{\alpha}_t$ to be the composite

$$\begin{tikzcd}
g_Y \circ t \arrow[r, "\overline{\alpha}_t"] \arrow[d, "\cong"'] & (g_Y \circ (t \circ f_X)) \circ g_X \\
(g_Y \circ t) \circ \mathrm{Id}_X \arrow[r, "g_Y \circ t \circ \mathsf{w}_X^{-1}"'] & (g_Y \circ t) \circ (f_X \circ g_X) \arrow[u, "\cong"']
\end{tikzcd}$$

for $t : X \to Y$. For β and $\overline{\beta}$ the idea is the same. We define $\beta_X := f_X$ and for $t : X \to Y$ we set

$$\begin{tikzcd}
f_Y \circ (g_Y \circ (t \circ f_Y)) \arrow[r, "\overline{\beta}_t"] \arrow[d, "\cong"'] & t \circ f_X \\
(f_Y \circ g_Y) \circ (t \circ f_X) \arrow[r, "\mathsf{w}_Y \circ t \circ f_X"'] & \mathrm{Id}_Y \circ (t \circ f_X) \arrow[u, "\cong"']
\end{tikzcd}$$

The definitions of $(\gamma, \overline{\gamma})$ and $(\delta, \overline{\delta})$ are identical. One then obtains modifications $\Xi : \mathrm{id} \xrightarrow{\cong} \alpha \circ \beta$ and $\Psi : \beta \circ \alpha \xrightarrow{\cong} \mathrm{id}$ by taking $\Xi_X := \mathrm{Id}_X \xRightarrow{\mathsf{v}_X} g_X \circ f_X$ and $\Psi_X := f_X \circ g_X \xRightarrow{\mathsf{w}_X} X$; similarly $\gamma \circ \delta \cong \mathrm{id}$ and $\delta \circ \gamma \cong \mathrm{id}$. □

Hence, $\Lambda_{\mathrm{ps}}^{\mathrm{bicat}}$ is the internal language for bicategories. If one restricts to a single variable name the universal property is strict, else it is up to biequivalence. In the next section we show that the syntactic model of $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$ is biequivalent as a biclone to the syntactic model of a strict type theory. From this we deduce a coherence result for biclones, which amounts to a form of normalisation for the rewrites of $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$. All of this will restrict to unary contexts, and hence to $\Lambda_{\mathrm{ps}}^{\mathrm{bicat}}$, recovering a version of the coherence theorem of Mac Lane & Paré [MP85].

3.3 Coherence for biclones

In practice, the coherence theorem for bicategories [MP85] entails that one may treat any bicategory as though it were a 2-category: roughly, one may assume that the structural isomorphisms a, l and r behave as though they were the identity (see *e.g.* [Lei04, Chapter 1] for a detailed exposition). In terms of $\Lambda_{\mathrm{ps}}^{\mathrm{bicat}}$, this amounts to treating $\mathsf{assoc}, \varrho^{(i)}$ and ι as though they were all identities. Our aim in this section is to extend this result to $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$.

The motivation is three-fold. First, the coherence theorem will simplify the calculations we shall require in future chapters. Second, the proof involves some of the calculations we shall need to extend when it comes to defining a pseudofunctorial interpretation of the full type theory $\Lambda_{\mathrm{ps}}^{\times,\to}$ (see Section 5.3.3). Finally, the proof strategy is of interest in itself. The strategy may be regarded as a version of Mac Lane's classical strategy for monoidal categories [Mac98, Chapter VII], in which the syntax of the respective type theories provide *structural induction* principles. It is reasonable to imagine that one may prove similar results for monoidal bicategories (via a linear calculus), tricategories (via a 3-dimensional calculus) or even higher-dimensional structures, by an analogous strategy.

To foreshadow the coherence result we shall prove in later chapters, let us make precise the notion of normalisation we are interested in. We wish to lift the standard notion of normalisation for systems such as the (untyped) λ-calculus (*e.g.* [GTL89]) to a normalisation property on rewrites. More precisely, we wish to consider versions of *abstract reduction systems* [Hue80] in which one also tracks how a reduction might happen; that is, the possible *witnesses* of a reduction. Our notion of normalisation then becomes: there is at most one witness to any possible reduction. This suggests the following definitions. We use the term *constructive* by analogy with constructive proofs, in which one requires an explicit witness to the truth of a statement, to emphasise that we are requiring an explicit witnesses to the existence of a reduction.

Definition 3.3.1.

1. An *abstract reduction system (ARS)* $(A, \to)$ is a set A equipped with a binary *reduction relation* $\to \subseteq A \times A$.

2. A *constructive abstract reduction system (CARS)* consists of a set A together with a family of sets $W_A(a, b)$ of *reduction witnesses* indexed by $a, b \in A$. A CARS is *coherent* if for every $a, b \in A$ and $u, v \in W_A(a, b)$, one has $u = v$. ◀

In a CARS we are not merely interested in the existence of a reduction: we are also interested in the equality relation on reductions. In particular, an ARS in the usual sense is a CARS in which every $W(a, a')$ is either empty or a singleton: either a reduces to a, or it does not.

The term 'coherent' is motivated by the following example.

Example 3.3.2.

1. Every graph $\mathcal{G}$ defines a CARS $A(\mathcal{G})$ with underlying set $\mathcal{G}_0$ and reduction witnesses $W_{A(\mathcal{G})}(t, t') := \mathcal{G}(t, t')$.

2. Every category $\mathbb{C}$ defines a CARS $\overline{\mathbb{C}}$ on $ob(\mathbb{C})$ by taking $W_{\overline{\mathbb{C}}}(A, B) := \mathbb{C}(A, B)$. The coherence theorem for monoidal categories of [Mac98, Chapter VII] then states that the CARS corresponding to the free monoidal category on one generator is coherent. ◂

In the bicategorical setting, we are interested in coherence in each hom-category.

Definition 3.3.3.

1. A 2-multigraph $\mathcal{G}$ is *locally coherent* if for every $A_1, \ldots, A_n, B \in \mathcal{G}_0$ the associated CARS $A\big(\mathcal{G}(A_1, \ldots, A_n; B)\big)$ is coherent.
2. A biclone (bicategory) is *locally coherent* if its underlying 2-multigraph is locally coherent. ◂

Spelling out the definitions, a 2-multigraph $\mathcal{G}$ is locally coherent if for all edges e and e' in $\mathcal{G}(A_1, \ldots, A_n; B)$ there exists at most one surface $\kappa : e \Rightarrow e'$, and a biclone is locally coherent if there is at most one 2-cell between any parallel pair of terms. The coherence theorem for bicategories [MP85] can therefore be rephrased as stating that the free bicategory on a 2-multigraph is locally coherent.

Now, every type theory consisting of types, terms and rewrites has an underlying 2-multigraph with nodes given by the types, edges $A_1, \ldots, A_n \to B$ by the α-equivalence classes of derivable terms $x_1 : A_1, \ldots, x_n : A_n \vdash t : B$ and surfaces by the derivable rewrites modulo α-equivalence and the equational theory. We call the type theory *locally coherent* if this 2-multigraph is locally coherent. We spend the rest of this chapter proving that $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}$ is locally coherent.

Our strategy is the following. We shall adapt the calculi of Hilken [Hil96] and Hirschowitz [Hir13] to construct a type theory that matches $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}$ but has a strict substitution operation; the syntactic model will be the free 2-clone (*c.f.* Construction 3.1.16). We shall then construct an equivalence between the two syntactic models by induction on the respective type theories. We finish by briefly commenting how the result restricts to bicategories.

3.3.1 A strict type theory

The first step is the construction of a strict type theory. Since we draw heavily on previous work, our presentation will be brief. Fix some 2-multigraph $\mathcal{G}$. The type theory $\mathrm{H}^{\mathrm{cl}}(\mathcal{G})$ (where H stands for both *Hilken* and *Hirschowitz*) is constructed as follows. Contexts are as in $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}$. The *raw terms* are either variables or constants, given by the following grammar:

$$u_1, \ldots, u_n ::= x \mid c(u_1, \ldots, u_n)$$

As for $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}$, we think of constants $c(x_1, \ldots, x_n)$ as n-ary *operators*. The *raw rewrites* are vertical composites of identity maps and constant rewrites:

$$\sigma_1, \ldots, \sigma_n, \tau, \sigma ::= \mathrm{id}_t \mid \kappa(u_1, \ldots, u_n) \mid c(\sigma_1, \ldots, \sigma_n) \mid \tau \bullet \sigma \qquad (u_1, \ldots, u_n \text{ terms})$$

Note that we require two forms of constant rewrite, corresponding to substitution of terms into rewrites and substitution of rewrites into terms: these form the right and left whiskering operations in the syntactic model.

The typing rules for $\mathrm{H}^{\mathrm{cl}}(\mathcal{G})$ are collected in Figures 3.13–3.16.

$$\frac{}{x_1 : A_1, \dots, x_n : A_n \vdash x_k : A_k}\ \text{var}$$

$$\frac{c \in \mathcal{G}(A_1, \dots, A_n; B) \qquad (\Delta \vdash u_i : A_i)_{i=1,\dots,n}}{x_1 : A_1, \dots, x_n : A_n \vdash c(u_1, \dots, u_n) : B}\ \text{const} \qquad \frac{\Gamma \vdash t : B}{\Gamma \vdash \mathrm{id}_t : t \Rightarrow t : B}\ \text{id}$$

$$\frac{\Gamma \vdash \tau' : t' \Rightarrow t'' : B \qquad \Gamma \vdash \tau : t \Rightarrow t' : B}{\Gamma \vdash \tau' \bullet \tau : t \Rightarrow t'' : B}\ \text{vert-comp}$$

$$\frac{\kappa \in \mathcal{G}(A_1, \dots, A_n; B)(c, c') \qquad (\Delta \vdash u_i : A_i)_{i=1,\dots,n}}{\Delta \vdash \kappa(u_1, \dots, u_n) : c(u_1, \dots, u_n) \Rightarrow c'(u_1, \dots, u_n) : B}\ \text{right-whisker}$$

$$\frac{c \in \mathcal{G}(A_1, \dots, A_n; B) \qquad (\Delta \vdash \sigma_i : u_i \Rightarrow u_i' : A_i)_{i=1,\dots,n}}{\Delta \vdash c(\sigma_1, \dots, \sigma_n) : c(u_1, \dots, u_n) \Rightarrow c'(u_1, \dots, u_n) : B}\ \text{left-whisker}$$

Figure 3.13: Introduction rules for $\mathrm{H}^{\mathrm{cl}}(\mathcal{G})$.

$$\frac{\Gamma \vdash \tau : t \Rightarrow t' : A}{\Gamma \vdash \tau \bullet \mathrm{id}_t \equiv \tau : t \Rightarrow t' : A}\ \bullet\text{-right-unit} \qquad \frac{\Gamma \vdash \tau : t \Rightarrow t' : A}{\Gamma \vdash \tau \equiv \mathrm{id}_{t'} \bullet \tau : t \Rightarrow t' : A}\ \bullet\text{-left-unit}$$

$$\frac{\Gamma \vdash \tau'' : t'' \Rightarrow t''' : A \qquad \Gamma \vdash \tau' : t' \Rightarrow t'' : A \qquad \Gamma \vdash \tau : t \Rightarrow t' : A}{\Gamma \vdash (\tau'' \bullet \tau') \bullet \tau \equiv \tau'' \bullet (\tau' \bullet \tau) : t \Rightarrow t''' : A}\ \bullet\text{-assoc}$$

Figure 3.14: Categorical rules for vertical composition

$$\frac{c \in \mathcal{G}(A_1, \dots, A_n; B) \qquad (\Delta \vdash \sigma'_i : u'_i \Rightarrow u''_i : A_i)_{i=1,\dots,n} \qquad (\Delta \vdash \sigma_i : u_i \Rightarrow u'_i : A_i)_{i=1,\dots,n}}{\Delta \vdash c(\tau'_1, \dots, \tau'_n) \bullet c(\tau_1, \dots, \tau_n) \equiv c(\tau'_1 \bullet \tau_1, \dots, \tau'_n \bullet \tau_n) : c(u_1, \dots, u_n) \Rightarrow c(u''_1, \dots, u''_n) : B}$$

$$\frac{c \in \mathcal{G}(A_1, \dots, A_n; B \qquad (\Delta \vdash u_i : A_i)_{i=1,\dots,n}}{\Delta \vdash c(\mathrm{id}_{u_1}, \dots, \mathrm{id}_{u_n}) \equiv \mathrm{id}_{c(u_1,\dots,u_n)} : c(u_1, \dots, u_n) \Rightarrow c(u_1, \dots, u_n) : B}$$

$$\frac{\kappa \in \mathcal{G}(A_1, \dots, A_n; B)(c, c') \qquad (\Delta \vdash \sigma_i : u_i \Rightarrow u'_i : A_i)_{i=1,\dots,n}}{\Delta \vdash \kappa(u'_1, \dots, u'_n) \bullet c(\sigma_1, \dots, \sigma_n) \equiv c'(\sigma_1, \dots, \sigma_n) \bullet \kappa(u_1, \dots, u_n) : c(u_\bullet) \Rightarrow c'(u'_\bullet) : B}$$

Figure 3.15: Compatibility laws for constants

$$\frac{\Gamma \vdash \tau : t \Rightarrow t' : A}{\Gamma \vdash \tau \equiv \tau : t \Rightarrow t' : A}\,\text{refl} \qquad \frac{\Gamma \vdash \tau \equiv \tau' : t \Rightarrow t' : A}{\Gamma \vdash \tau' \equiv \tau : t \Rightarrow t' : A}\,\text{symm}$$

$$\frac{\Gamma \vdash \tau' \equiv \tau'' : t \Rightarrow t' : A \qquad \Gamma \vdash \tau \equiv \tau' : t \Rightarrow t' : A}{\Gamma \vdash \tau \equiv \tau'' : t \Rightarrow t' : A}\,\text{trans}$$

$$\frac{\Gamma \vdash \tau' \equiv \sigma' : t' \Rightarrow t'' : A \qquad \Gamma \vdash \tau \equiv \sigma : t \Rightarrow t' : A}{\Gamma \vdash \tau' \bullet \tau \equiv \sigma' \bullet \sigma : t \Rightarrow t'' : A}$$

$$\frac{c \in \mathcal{G}(A_1, \dots, A_n; B) \qquad (\Delta \vdash \sigma_i \equiv \sigma' : u_i \Rightarrow u'_i : A_i)_{i=1,\dots,n}}{\Delta \vdash c(\sigma_1, \dots, \sigma_n) \equiv c(\sigma'_1, \dots, \sigma'_n) : c(u_1, \dots, u_n) \Rightarrow c(u'_1, \dots, u'_n)}$$

Figure 3.16: Congruence rules

For H^{cl} to be a strict biclone we require a strictly associative and unital substitution operation. Accordingly, we define substitution of terms into terms, of terms into rewrites, and of rewrites into terms as follows.

$$\begin{aligned}
x_k[u_i/x_i] &:= u_k \\
c(u_1, \dots, u_n)[v_j/y_j] &:= c\big(u_1[v_j/y_j], \dots, u_n[v_j/y_j]\big) \\[1ex]
\mathrm{id}_t[u_i/x_i] &:= \mathrm{id}_{t[u_i/x_i]} \\
(\tau' \bullet \tau)[u_i/x_i] &:= \tau'[u_i/x_i] \bullet \tau[u_i/x_i] \\
c(\sigma_1, \dots, \sigma_n)[u_i/x_i] &:= c\big(\sigma_1[u_i/x_i] \dots, \sigma_n[u_i/x_i]\big) \\
\sigma(u_1, \dots, u_n)[v_j/y_j] &:= \sigma\big(u_1[v_j/y_j], \dots, u_n[v_j/y_j]\big) \\[1ex]
x_k[\sigma_i/x_i] &:= \sigma_k \\
c(u_1, \dots, u_n)[\sigma_j/y_j] &:= c\big(u_1[\sigma_j/y_j], \dots, u_n[\sigma_j/y_j]\big)
\end{aligned}$$

The Substitution Lemma holds for all three forms of substitution.

Lemma 3.3.4. For any 2-multigraph $\mathcal{G}$, the following rules are admissible in $\mathrm{H}^{\mathrm{cl}}(\mathcal{G})$:

$$\frac{x_1 : A_1, \dots , x_n : A_n \vdash t : B \qquad (\Delta \vdash u_i : A_i)_{i=1,\dots,n}}{\Delta \vdash t[u_i/x_i] : B}$$

$$\frac{x_1 : A_1, \dots , x_n : A_n \vdash \tau : t \Rightarrow t' : B \qquad (\Delta \vdash u_i : A_i)_{i=1,\dots,n}}{\Delta \vdash \tau[u_i/x_i] : t[u_i/x_i] \Rightarrow t'[u_i/x_i] : B}$$

$$\frac{x_1 : A_1, \dots , x_n : A_n \vdash t : B \qquad (\Delta \vdash \sigma_i : u_i \Rightarrow u_i' : A_i)_{i=1,\dots,n}}{\Delta \vdash t[\sigma_i/x_i] : t[u_i/x_i] \Rightarrow t[u_i'/x_i] : B}$$

□

As there are no operations that bind variables, the definition of α-equivalence is trivial. The equational theory $\equiv$ is defined in Figures 3.14–3.16. The rules diverge from $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$ most importantly in Figure 3.15, which ensures the meta-operation of substitution is functorial, and that the two different ways of composing with constant rewrites are equal. This guarantees that the composites $\tau[u_i'/x_i] \bullet t[\sigma_i/x_i]$ and $t'[\sigma_i/x_i] \bullet \tau[u_i/x_i]$ coincide (*c.f.* the *permutation equivalence* of [Hir13]).

Following the pattern of [Hil96, Hir13], we define a substitution operation making the following rule admissible, where $\tau[\sigma_i/x_i] := t'[\sigma_i/x_i] \bullet \tau[u_i/x_i]$:

$$\frac{x_1 : A_1, \dots , x_n : A_n \vdash \tau : t \Rightarrow t' : B \qquad (\Delta \vdash \sigma_i : u_i \Rightarrow u_i' : A_i)_{i=1,\dots,n}}{\Delta \vdash \tau[\sigma_i/x_i] : t[u_i/x_i] \Rightarrow t'[u_i'/x_i] : B}\ \text{subst}$$

We could have defined vertical composition by whiskering in the opposite order, thus: $\tau[\sigma_i/x_i] := \tau[u_i'/x_i] \bullet t[\sigma_i/x_i]$. The next lemma guarantees that these two coincide. The proof is by structural induction, using Figure 3.15 for the constant cases.

Lemma 3.3.5. For any 2-multigraph $\mathcal{G}$, the following rule is admissible in $\mathrm{H}^{\mathrm{cl}}(\mathcal{G})$:

$$\frac{x_1 : A_1, \dots , x_n : A_n \vdash \tau : t \Rightarrow t' : B \qquad (\Delta \vdash \sigma_i : u_i \Rightarrow u_i' : A_i)_{i=1,\dots,n}}{\Delta \vdash t'[\sigma_i/x_i] \bullet \tau[u_i/x_i] \equiv \tau[u_i'/x_i] \bullet t[\sigma_i/x_i] : t[u_i/x_i] \Rightarrow t'[u_i'/x_i] : B}$$

□

Further structural inductions establish the key properties we shall be relying on.

Lemma 3.3.6. For any 2-multigraph $\mathcal{G}$ and terms $t, u_1, \dots , u_n$ in $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}(\mathcal{G})$:

1. $x_k[u_i/x_i] = u_k$,
2. $t[x_i/x_i] = t$,
3. $t[u_i/x_i][v_j/y_j] = t\big[u_i[v_j/y_j]/x_i\big]$.

Moreover, for any rewrites $\tau, \sigma_1, \dots , \sigma_n$,

1. $\mathrm{id}_{x_k}[\sigma_i/x_i] \equiv \sigma_k$,

2. $\tau[\mathrm{id}_{x_i}/x_i] \equiv \tau$,
3. $\tau[\sigma_i/x_i][\mu_j/y_j] \equiv \tau\big[\sigma_i[\mu_j/y_j]/x_i\big]$. □

Hence the three laws of an abstract clone hold on both terms and rewrites. It is similarly straightforward to establish that $t[\sigma_i' \bullet \sigma_i/x_i] \equiv t[\sigma_i'/x_i] \bullet t[\sigma_i/x_i]$ and hence deduce the *interchange law* $(\tau' \bullet \tau)[\sigma_i' \bullet \sigma_i/x_i] \equiv \tau'[\sigma_i'/x_i] \bullet \tau[\sigma_i/x_i]$. Finally we observe that $\mathrm{id}_t[\mathrm{id}_{u_i}/x_i] \equiv \mathrm{id}_{t[u_i/x_i]}$. Together these considerations establish the following does indeed define a strict biclone.

Construction 3.3.7. For any 2-multigraph $\mathcal{G}$, define a strict biclone $\mathcal{H}(\mathcal{G})$ as follows. The sorts are nodes in $\mathcal{G}$. The 1-cells are terms $(x_1 : A_1, \ldots, x_n : A_n \vdash t : B)$ derivable in $\mathrm{H}^{\mathrm{cl}}(\mathcal{G})$, for $x_1, x_2, \ldots$ a chosen enumeration of variables, and the 2-cells are $\equiv$-classes of rewrites $(x_1 : A_1, \ldots, x_n : A_n \vdash \tau : t \Rightarrow t' : B)$. Composition is the $\bullet$ operation and the identity on a term-in-context t is id_t.

Substitution is the meta-operation of substitution in $\mathrm{H}^{\mathrm{cl}}(\mathcal{G})$:

$$t, (u_1, \ldots, u_n) \mapsto t[u_1/x_1, \ldots, u_n/x_n]$$
$$\tau, (\sigma_1, \ldots, \sigma_n) \mapsto \tau[\sigma_1/x_1, \ldots, \sigma_n/x_n]$$

The projections $\mathsf{p}^{(i)}_{A_\bullet} : A_1, \ldots, A_n \to A_i$ are given by the `var` rule. ◂

It is not hard to see that $\mathcal{H}(\mathcal{G})$ is the free 2-clone on $\mathcal{G}$.

Lemma 3.3.8. For any 2-multigraph $\mathcal{G}$, strict biclone $(T, \mathcal{D})$ and 2-multigraph homomorphism $h : \mathcal{G} \to \mathcal{D}$, there exists a unique strict pseudofunctor $h[\![-]\!] : \mathcal{H}(\mathcal{G}) \to \mathcal{D}$ such that $h[\![-]\!] \circ \iota = h$, for $\iota : \mathcal{G} \hookrightarrow \mathcal{H}(\mathcal{G})$ the inclusion.

Proof. A straightforward adaptation of the proof of Lemma 3.2.13. The most significant work is showing that the pseudofunctor $h[\![-]\!]$ respects substitution, in the sense that

$$\begin{aligned} &h[\![\Delta \vdash \tau[\sigma_i/x_i] : t[u_i/x_i] \Rightarrow t'[u_i'/x_i] : B]\!] \\ &\qquad\qquad = \big(h[\![x_1 : A_1, \ldots, x_n : A_n \vdash \tau : t \Rightarrow t' : B]\!]\big)[\Delta \vdash \sigma_\bullet : u_\bullet \Rightarrow u'_\bullet : A_\bullet] \end{aligned}$$

for all judgements $x_1 : A_1, \ldots, x_n : A_n \vdash \tau : t \Rightarrow t' : B$ and $(\Delta \vdash \sigma_i : u_i \Rightarrow u_i' : A_i)_{i=1,\ldots,n}$. This is proven by two structural inductions, one for each of the whiskering operations. □

3.3.2 Proving biequivalence

The next stage of the proof is to construct a biequivalence of biclones $\mathcal{H}(\mathcal{G}) \simeq \mathrm{Syn}(\mathcal{G})$ over a fixed 2-multigraph $\mathcal{G}$. We shall then see how this restricts to a biequivalence of bicategories when $\mathcal{G}$ is a 2-graph and H^{cl} and $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}$ are restricted to unary contexts.

Fix a 2-multigraph $\mathcal{G}$. We begin by constructing pseudofunctors $(\!| - |\!) : \mathcal{H}(\mathcal{G}) \leftrightarrows \mathrm{Syn}(\mathcal{G}) : \overline{(-)}$. The definition of $\overline{(-)}$ is simpler, so we do this first. Intuitively, this mapping is a *strictification* evaluating away explicit substitutions; for constants we exploit the fact the underlying signatures are the same.

Construction 3.3.9. For any 2-multigraph $\mathcal{G}$, we define a mapping from raw terms in $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}(\mathcal{G})$ to raw terms in $\mathrm{H}^{\mathrm{cl}}(\mathcal{G})$ as follows:

$$\begin{aligned} \overline{x_k} &:= x_k \\ \overline{c(x_1, \dots, x_n)} &:= c(x_1, \dots, x_n) \\ \overline{t\{x_i \mapsto u_i\}} &:= \overline{t}[\overline{u_i}/x_i] \end{aligned}$$

This extends to a map on raw rewrites:

$$\begin{aligned} \overline{\mathsf{assoc}_{t,u_\bullet,v_\bullet}} &:= \mathrm{id}_{\overline{t}[\overline{u_i}/x_i][\overline{v_j}/y_j]} & \overline{\mathrm{id}_t} &:= \mathrm{id}_{\overline{t}} \\ \overline{\iota_t} &:= \mathrm{id}_{\overline{t}} & \overline{\kappa(x_1, \dots, x_n)} &:= \kappa(x_1, \dots, x_n) \\ \overline{\varrho^{(k)}_{u_\bullet}} &:= \mathrm{id}_{\overline{u_k}} & \overline{\tau \bullet \sigma} &:= \overline{\tau} \bullet \overline{\sigma} \\ & & \overline{\tau\{x_i \mapsto \sigma_i\}} &:= \overline{\tau}[\overline{\sigma_i}/x_i] \end{aligned}$$

◀

This mapping respects typing and the equational theory.

Lemma 3.3.10. For any 2-multigraph $\mathcal{G}$,

1. For all derivable terms t, t' in $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}(\mathcal{G})$, if $t =_\alpha t'$ then $\overline{t} = \overline{t'}$,
2. For all derivable rewrites τ, τ' in $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}(\mathcal{G})$, if $\tau =_\alpha \tau'$ then $\overline{\tau} = \overline{\tau'}$,
3. If $\Gamma \vdash t : B$ in $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}(\mathcal{G})$ then $\Gamma \vdash \overline{t} : B$ in $\mathrm{H}^{\mathrm{cl}}(\mathcal{G})$,
4. If $\Gamma \vdash \tau : t \Rightarrow t' : B$ in $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}(\mathcal{G})$ then $\Gamma \vdash \overline{\tau} : \overline{t} \Rightarrow \overline{t'} : B$ in $\mathrm{H}^{\mathrm{cl}}(\mathcal{G})$,
5. If $\Gamma \vdash \tau \equiv \tau' : t \Rightarrow t' : B$ in $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}(\mathcal{G})$ then $\Gamma \vdash \overline{\tau} \equiv \overline{\tau'} : \overline{t} \Rightarrow \overline{t'} : B$ in $\mathrm{H}^{\mathrm{cl}}(\mathcal{G})$.

Proof. By structural induction. □

Proposition 3.3.11. For any 2-multigraph $\mathcal{G}$ the mapping $\overline{(-)}$ extends to a pseudofunctor $\mathrm{Syn}(\mathcal{G}) \to \mathcal{H}(\mathcal{G})$.

Proof. By Lemma 3.3.10 and the definition of $\overline{(-)}$ on identities and vertical compositions, the mapping $\overline{(-)}$ defines a functor $\mathrm{Syn}(\mathcal{G})(A_\bullet; B) \to \mathcal{H}(A_\bullet; B)$ on each hom-category by

$$\overline{(\Gamma \vdash \tau : t \Rightarrow t' : B)} := (\Gamma \vdash \overline{\tau} : \overline{t} \Rightarrow \overline{t'} : B)$$

For preservation of projections and substitution, one notes that

$$\overline{x_1 : A_1, \dots, x_n : A_n \vdash x_k : A_k} = (x_1 : A_1, \dots, x_n : A_n \vdash x_k : A_k)$$

and that, for $\Gamma = (x_i : A_i)_{i=1,\dots,n}$,

$$
\begin{aligned}
\overline{(\Gamma \vdash t : B)}\big[\overline{\Delta \vdash u_1 : A_1}, \dots, \overline{\Delta \vdash u_n : A_n}\big] &= (\Gamma \vdash \bar{t} : B)[\Delta \vdash \overline{u_\bullet} : A_\bullet] \\
&= (\Delta \vdash \bar{t}[\overline{u_i}/x_i] : B) \\
&= \overline{\Delta \vdash t\{x_i \mapsto u_i\} : B}
\end{aligned}
$$

so $\overline{(-)}$ is indeed a strict pseudofunctor. □

Now we turn to defining the pseudofunctor $(\!| - |\!) : \mathcal{H}(\mathcal{G}) \to \mathrm{Syn}(\mathcal{G})$. The mapping we choose makes precise the sense in which H^{cl} is a fragment of $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$.

Construction 3.3.12. For any 2-multigraph $\mathcal{G}$, define a mapping from raw terms in $\mathrm{H}^{\mathrm{cl}}(\mathcal{G})$ to raw terms in $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}(\mathcal{G})$ as follows:

$$
\begin{aligned}
(\!| x_k |\!) &:= x_k \\
(\!| c(u_1, \dots, u_n) |\!) &:= c\{(\!| u_1 |\!), \dots, (\!| u_n |\!)\}
\end{aligned}
$$

Extend this to a map on raw rewrites as follows:

$$
\begin{aligned}
(\!| \mathrm{id}_t |\!) &:= \mathrm{id}_{(\!| t |\!)} & \qquad (\!| c(\sigma_1, \dots, \sigma_n) |\!) &:= c\{x_i \mapsto (\!| \sigma_i |\!)\} \\
(\!| \tau \bullet \sigma |\!) &:= (\!| \tau |\!) \bullet (\!| \sigma |\!) & \qquad (\!| \kappa(u_1, \dots, u_n) |\!) &:= \kappa\{x_i \mapsto (\!| u_i |\!)\}
\end{aligned}
$$

◂

Once again, the mapping respects typings and the equational theory.

Lemma 3.3.13. For any 2-multigraph $\mathcal{G}$,

1. For all derivable terms t, t' in $\mathrm{H}^{\mathrm{cl}}(\mathcal{G})$, if $t = t'$ then $(\!| t |\!) =_\alpha (\!| t' |\!)$,
2. For all derivable rewrites τ, τ' in $\mathrm{H}^{\mathrm{cl}}(\mathcal{G})$, if $\tau = \tau'$ then $(\!| \tau |\!) =_\alpha (\!| \tau' |\!)$,
3. If $\Gamma \vdash t : B$ in $\mathrm{H}^{\mathrm{cl}}(\mathcal{G})$ then $\Gamma \vdash (\!| t |\!) : B$ in $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}(\mathcal{G})$,
4. If $\Gamma \vdash \tau : t \Rightarrow t' : B$ in $\mathrm{H}^{\mathrm{cl}}(\mathcal{G})$ then $\Gamma \vdash (\!| \tau |\!) : (\!| t |\!) \Rightarrow (\!| t' |\!) : B$ in $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}(\mathcal{G})$,
5. If $\Gamma \vdash \tau \equiv \tau' : t \Rightarrow t' : B$ in $\mathrm{H}^{\mathrm{cl}}(\mathcal{G})$ then $\Gamma \vdash (\!| \tau |\!) \equiv (\!| \tau' |\!) : (\!| t |\!) \Rightarrow (\!| t' |\!) : B$ in $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}(\mathcal{G})$. □

It is immediate from the preceding lemma that $(\!| - |\!)$ defines a functor $\mathcal{H}(\mathcal{G})(A_\bullet; B) \to \mathrm{Syn}(\mathcal{G})(A_\bullet; B)$ on each hom-category, and that $(\!| - |\!)$ strictly preserves identities. For preservation of substitution, however, we are required to construct a family of 2-cells $(\!| t |\!)\{x_i \mapsto (\!| u_i |\!)\} \Rightarrow (\!| t[u_i/x_i] |\!)$. This should be compared to [RdP97], where a similar translation is constructed at the meta-level.

Construction 3.3.14. For any 2-multigraph $\mathcal{G}$, define a family of rewrites sub in $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}(\mathcal{G})$ so that the rule

$$\frac{x_1 : A_1, \dots, x_n : A_n \vdash \llparenthesis\, t \,\rrparenthesis : B \qquad (\Delta \vdash \llparenthesis\, u_i \,\rrparenthesis : A_i)_{i=1,\dots,n}}{\Delta \vdash \mathsf{sub}(t; u_\bullet) : \llparenthesis\, t \,\rrparenthesis\{x_i \mapsto \llparenthesis\, u_i \,\rrparenthesis\} \Rightarrow \llparenthesis\, t[u_i/x_i] \,\rrparenthesis : B}$$

is admissible by setting

$$\begin{aligned}
\mathsf{sub}(x_k; u_\bullet) &:= x_k\{x_i \mapsto \llparenthesis\, u_i \,\rrparenthesis\} \xRightarrow{\varrho^{(k)}_{\llparenthesis u_\bullet \rrparenthesis}} \llparenthesis\, u_k \,\rrparenthesis \\
\mathsf{sub}(c(u_\bullet); v_\bullet) &:= c\{u_i\}\{v_j\} \xRightarrow{\mathsf{assoc}_{c(x_\bullet), u_\bullet, v_\bullet}} c\{u_i\{v_j\}\} \xRightarrow{c\{\mathsf{sub}(u_i; v_\bullet)\}} c\{\llparenthesis\, u_i[v_j/y_j] \,\rrparenthesis\}
\end{aligned}$$

◂

We establish the various properties required of sub by induction. The naturality of structural rewrites implies the following.

Lemma 3.3.15. For any 2-multigraph $\mathcal{G}$, the following judgements are derivable in $\mathrm{Syn}(\mathcal{G})$:

$$\frac{\Gamma \vdash \llparenthesis\, t \,\rrparenthesis : B \qquad (\Delta \vdash \llparenthesis\, \sigma_i \,\rrparenthesis : \llparenthesis\, u_i \,\rrparenthesis \Rightarrow \llparenthesis\, u'_i \,\rrparenthesis : A_i)_{i=1,\dots,n}}{\Delta \vdash \mathsf{sub}(t; u'_\bullet) \bullet \llparenthesis\, t \,\rrparenthesis\{\llparenthesis\, \sigma_i \,\rrparenthesis\} \equiv \llparenthesis\, t[\sigma_i/x_i] \,\rrparenthesis \bullet \mathsf{sub}(t; u_\bullet) : \llparenthesis\, t \,\rrparenthesis\{\llparenthesis\, u_i \,\rrparenthesis\} \Rightarrow \llparenthesis\, t' \,\rrparenthesis\{\llparenthesis\, u_i \,\rrparenthesis\} : B}$$

$$\frac{\Gamma \vdash \llparenthesis\, \tau \,\rrparenthesis : \llparenthesis\, t \,\rrparenthesis \Rightarrow \llparenthesis\, t' \,\rrparenthesis : B \qquad (\Delta \vdash \llparenthesis\, u_i \,\rrparenthesis : A_i)_{i=1,\dots,n}}{\Delta \vdash \mathsf{sub}(t'; u_\bullet) \bullet \llparenthesis\, \tau \,\rrparenthesis\{\llparenthesis\, u_i \,\rrparenthesis\} \equiv \llparenthesis\, \tau[u_i/x_i] \,\rrparenthesis \bullet \mathsf{sub}(t; u_\bullet) : \llparenthesis\, t \,\rrparenthesis\{\llparenthesis\, u_i \,\rrparenthesis\} \Rightarrow \llparenthesis\, t \,\rrparenthesis\{\llparenthesis\, u'_i \,\rrparenthesis\} : B}$$

Hence the following judgement is derivable:

$$\frac{\Gamma \vdash \llparenthesis\, \tau \,\rrparenthesis : \llparenthesis\, t \,\rrparenthesis \Rightarrow \llparenthesis\, t' \,\rrparenthesis : B \qquad (\Delta \vdash \llparenthesis\, \sigma_i \,\rrparenthesis : \llparenthesis\, u_i \,\rrparenthesis \Rightarrow \llparenthesis\, u'_i \,\rrparenthesis : A_i)_{i=1,\dots,n}}{\Delta \vdash \mathsf{sub}(t'; u'_\bullet) \bullet \llparenthesis\, \tau \,\rrparenthesis\{\llparenthesis\, \sigma_i \,\rrparenthesis\} \equiv \llparenthesis\, \tau[\sigma_i/x_i] \,\rrparenthesis \bullet \mathsf{sub}(t; u_\bullet) : \llparenthesis\, t \,\rrparenthesis\{\llparenthesis\, u_i \,\rrparenthesis\} \Rightarrow \llparenthesis\, t' \,\rrparenthesis\{\llparenthesis\, u'_i \,\rrparenthesis\} : B}$$

and the sub rewrites are natural. □

Next we want to prove the three coherence laws for a pseudofunctor. The law for $\varrho^{(i)}$ (3.3) holds by definition. We prove the other two laws using correlates of Mac Lane's original five axioms of a monoidal category [Mac63].

Lemma 3.3.16. For any biclone $(S, \mathcal{C})$ the following diagrams commute:

$$\begin{tikzcd}
\mathsf{p}^{(k)} \arrow[r, "\iota"] & \mathsf{p}^{(k)}[\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)}] \\
\mathsf{p}^{(k)}[\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)}] \arrow[u, "\varrho^{(k)}"] \arrow[ur, equal] &
\end{tikzcd}
\qquad
\begin{tikzcd}
\mathsf{p}^{(k)}[\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)}] \arrow[r, "\varrho^{(k)}"] & \mathsf{p}^{(k)} \\
\mathsf{p}^{(k)} \arrow[u, "\iota"] \arrow[ur, equal] &
\end{tikzcd}$$

$$\begin{tikzcd}
t[u_\bullet]\,[\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)}] \arrow[r, "\mathsf{assoc}"] & t[u_\bullet[\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)}]] \\
t[u_\bullet] \arrow[u, "\iota"] \arrow[ur, swap, "{t[\iota, \dots, \iota]}"] &
\end{tikzcd}
\qquad
\begin{tikzcd}
\mathsf{p}^{(k)}[u_\bullet][v_\bullet] \arrow[r, "\varrho^{(k)}"] & u_k[v_\bullet] \\
\mathsf{p}^{(k)}[u_\bullet]\,[v_\bullet] \arrow[u, "\mathsf{assoc}"] \arrow[ur, swap, "{\varrho^{(k)}[v_\bullet]}"] &
\end{tikzcd}$$

Proof. By adapting Kelly's arguments for monoidal categories [Kel64]. □

Lemma 3.3.17. For any 2-multigraph $\mathcal{G}$ and derivable terms $(x_1 : A_1, \dots, x_n : A_n \vdash \llparenthesis t \rrparenthesis : C)$, $(y_1 : B_1, \dots, y_m : B_m \vdash u_i : A_i)_{i=1,\dots,m}$ and $(\Delta \vdash v_j : B_j)_{j=1,\dots,m}$ in $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}(\mathcal{G})$, the following diagrams commute in $\mathrm{Syn}(\mathcal{G})$:

$$\begin{tikzcd}
\llparenthesis t \rrparenthesis\{x_i \mapsto x_i\} \arrow[r, "\mathsf{sub}(t;x_\bullet)"] & \llparenthesis t \rrparenthesis \\
\llparenthesis t \rrparenthesis \arrow[u, "\iota"] \arrow[ur, equal] &
\end{tikzcd}$$

$$\begin{tikzcd}
\llparenthesis t \rrparenthesis\{\llparenthesis u_i \rrparenthesis\}\{\llparenthesis v_j \rrparenthesis\} \arrow[r, "\mathsf{sub}(t;u_\bullet)\{v_j\}"] \arrow[d, "\mathsf{assoc}"'] & \llparenthesis t[u_i/x_i] \rrparenthesis\{\llparenthesis v_j \rrparenthesis\} \arrow[dd, "\mathsf{sub}(t[u_i/x_i];v_\bullet)"] \\
\llparenthesis t \rrparenthesis\{\llparenthesis u_i \rrparenthesis\{\llparenthesis v_j \rrparenthesis\}\} \arrow[d, "\llparenthesis t \rrparenthesis\{\mathsf{sub}(u_i;v_\bullet)\}"'] & \\
\llparenthesis t \rrparenthesis\{\llparenthesis u_i[v_j/y_j] \rrparenthesis\} \arrow[r, "\mathsf{sub}(t;u_\bullet[v_j/y_j])"'] & \llparenthesis t\big[u_i[v_j/y_j]/x_i\big] \rrparenthesis
\end{tikzcd}$$

Proof. Both claims are proven by induction using the laws of Lemma 3.3.16. For the unit law one uses the two laws on ι; for the associativity law one uses naturality and the law relating $\varrho^{(i)}$ and assoc. □

We have shown that sub is natural and satisfies the three laws of a pseudofunctor.

Corollary 3.3.18. For any 2-multigraph $\mathcal{G}$ the mapping $\llparenthesis - \rrparenthesis$ extends to a pseudofunctor $\mathcal{H}(\mathcal{G}) \to \mathrm{Syn}(\mathcal{G})$. □

Relating the two composites. With the two pseudofunctors in hand, we next examine the composites $\llparenthesis - \rrparenthesis \circ \overline{(-)}$ and $\overline{(-)} \circ \llparenthesis - \rrparenthesis$. Our first observation is that the strictification of an already-strict term $\llparenthesis t \rrparenthesis$ is simply t.

Lemma 3.3.19. For any 2-multigraph $\mathcal{G}$, the composite $\overline{(-)} \circ \llparenthesis - \rrparenthesis$ is the identity on $\mathcal{H}(\mathcal{G})$.

Proof. On objects the claim is trivial. On multimaps one proceeds inductively:

$$x_k \mapsto \llparenthesis x_k \rrparenthesis = x_k \mapsto \overline{x_k} = x_k$$
$$c(u_1, \dots, u_n) \mapsto c\{\llparenthesis u_1 \rrparenthesis, \dots, \llparenthesis u_n \rrparenthesis\} \mapsto c(x_1, \dots, x_n)\left[\overline{\llparenthesis u_i \rrparenthesis}/x_i\right] = c(u_1, \dots, u_n)$$

The induction for 2-cells is similar:

$$\mathrm{id}_t \mapsto \mathrm{id}_{\llparenthesis t \rrparenthesis} \mapsto \mathrm{id}_{\overline{\llparenthesis t \rrparenthesis}} = \mathrm{id}_t \qquad \text{by the preceding}$$
$$\tau' \bullet \tau \mapsto \llparenthesis \tau' \rrparenthesis \bullet \llparenthesis \tau \rrparenthesis \mapsto \overline{\llparenthesis \tau' \rrparenthesis} \bullet \overline{\llparenthesis \tau' \rrparenthesis} = \tau' \bullet \tau \qquad \text{by inductive hypothesis}$$

$$\kappa(u_1, \dots, u_n) \mapsto \kappa\{\llparenthesis u_1 \rrparenthesis, \dots, \llparenthesis u_n \rrparenthesis\} \mapsto \kappa(x_1, \dots, x_n)[\overline{\llparenthesis u_i \rrparenthesis}/x_i] = \kappa(u_1, \dots, u_n)$$
$$c(\sigma_1, \dots, \sigma_n) \mapsto c\{\llparenthesis \sigma_1 \rrparenthesis, \dots, \llparenthesis \sigma_n \rrparenthesis\} \mapsto c(x_1, \dots, x_n)[\overline{\llparenthesis \sigma_i \rrparenthesis}/x_i] = c(\sigma_1, \dots, \sigma_n)$$

□

We finish our construction of the biequivalence $\mathcal{H}(\mathcal{G}) \simeq \mathrm{Syn}(\mathcal{G})$ by defining an invertible pseudonatural transformation $\llparenthesis - \rrparenthesis \circ \overline{(-)} \cong \mathrm{id}_{\mathrm{Syn}(\mathcal{G})}$. This amounts to defining a reduction procedure within $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}(\mathcal{G})$ taking a term to one in which explicit substitutions occur as far to the left as possible. The sub rewrites of Construction 3.3.14 will play a crucial role.

Construction 3.3.20. For any 2-multigraph $\mathcal{G}$, define a rewrite reduce typed by the rule

$$\frac{\Gamma \vdash t : B}{\Gamma \vdash \mathsf{reduce}(t) : t \Rightarrow \llparenthesis \overline{t} \rrparenthesis : B}$$

inductively as follows:

$$\begin{aligned}
\mathsf{reduce}(x_k) &:= x_k \xRightarrow{\mathrm{id}_{x_k}} x_k \\
\mathsf{reduce}(c(x_1, \dots, x_n)) &:= c(x_1, \dots, x_n) \overset{\iota}{\Rightarrow} c\{x_1, \dots, x_n\} = \overline{c(x_1, \dots, x_n)} \\
\mathsf{reduce}(t\{x_i \mapsto u_i\}) &:= t\{x_i \mapsto u_i\} \xRightarrow{\mathsf{reduce}(t)\{\mathsf{reduce}(u_i)\}} \llparenthesis \overline{t} \rrparenthesis \{x_i \mapsto \llparenthesis \overline{u_i} \rrparenthesis\} \xRightarrow{\mathsf{sub}(\overline{t};\overline{u}_\bullet)} \llparenthesis \overline{t}[\overline{u_i}/x_i] \rrparenthesis
\end{aligned}$$

◂

We think of reduce as a *normalisation* procedure on terms. When such a procedure is defined as a meta-operation, it passes through the term constructors; in $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}$, it is natural.

Lemma 3.3.21. For any 2-multigraph $\mathcal{G}$, the following rule is admissible in $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}(\mathcal{G})$:

$$\frac{\Gamma \vdash \tau : t \Rightarrow t' : B}{\Gamma \vdash \llparenthesis \overline{\tau} \rrparenthesis \bullet \mathsf{reduce}(t) \equiv \mathsf{reduce}(t') \bullet \tau : t \Rightarrow \llparenthesis \overline{t'} \rrparenthesis : B}$$

Proof. By induction on the derivation of τ. For the structural maps one uses the fact the structural maps are all natural; for ι and assoc one also makes use of the unit and associativity laws of Lemma 3.3.17, respectively. The other cases are straightforward. □

Terms in which no substitutions occur do not reduce any further.

Lemma 3.3.22. For any 2-multigraph $\mathcal{G}$ and judgement $\Gamma \vdash t : B$ derivable in $\mathrm{H}^{\mathrm{cl}}(\mathcal{G})$, the rule

$$\frac{\Gamma \vdash \llparenthesis t \rrparenthesis : B}{\Gamma \vdash \mathsf{reduce}(\llparenthesis t \rrparenthesis) \equiv \mathrm{id}_{\llparenthesis t \rrparenthesis} : \llparenthesis t \rrparenthesis \Rightarrow \llparenthesis t \rrparenthesis : B}$$

is admissible in $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}(\mathcal{G})$.

Proof. The claim is well-typed because $\llparenthesis \overline{\llparenthesis t \rrparenthesis} \rrparenthesis = \llparenthesis t \rrparenthesis$ by Lemma 3.3.19. The result then follows by structural induction: the `var` case holds by definition, while the `const` case is just the triangle law of a biclone. □

The reduce rewrite is central to our definition of the invertible transformation $\mathrm{id}_{\mathsf{Syn}(\mathcal{G})} \Rightarrow \llparenthesis \overline{(-)} \rrparenthesis$; the rest of the work is book-keeping. We define a transformation of pseudofunctors (Definition 3.1.20) as follows. Take the identity $\varrho^{(1)}_B : B \to B$ on multimaps; as a term this is $(x_1 : B \vdash x_1 : B)$. For each $\Gamma := (x_i : A_i)_{i=1,\dots,n}$ and derivable term $(\Gamma \vdash t : B)$ we are now required to give a 2-cell

$$(\Gamma \vdash x_1\{x_1 \mapsto t\} : B) \Rightarrow (\Gamma \vdash \llparenthesis \bar{t} \rrparenthesis \{x_i \mapsto x_i\{x_i \mapsto x_i\}\} : B)$$

For this, take the composite $\widetilde{\mathsf{r}}(t)$ defined by

$$\begin{tikzcd}
x_1\{x_1 \mapsto t\} \arrow[rr, Rightarrow, dashed, "\widetilde{\mathsf{r}}(t)"] \arrow[d, Rightarrow, "\varrho^{(1)}"'] & & \llparenthesis \bar{t} \rrparenthesis \{x_i \mapsto x_i\{x_i \mapsto x_i\}\} \\
t \arrow[r, Rightarrow, "\mathsf{reduce}(t)"'] & \llparenthesis \bar{t} \rrparenthesis \arrow[r, Rightarrow, "\iota"'] & \llparenthesis \bar{t} \rrparenthesis \{x_i \mapsto x_i\} \arrow[u, Rightarrow, "\llparenthesis \bar{t} \rrparenthesis\{x_i \mapsto \iota\}"']
\end{tikzcd} \tag{3.9}$$

in context Γ. The composite is natural because reduce is.

Corollary 3.3.23. For any 2-multigraph $\mathcal{G}$, the multimaps $\varrho^{(1)}_B : B \to B$ together with the 2-cells $\widetilde{\mathsf{r}}(t)$ defined in (3.8) form an invertible transformation $\mathrm{id}_{\mathsf{Syn}(\mathcal{G})} \overset{\cong}{\Rightarrow} \llparenthesis \overline{(-)} \rrparenthesis$.

Proof. By induction, the 2-cell reduce is invertible, so $\widetilde{\mathsf{r}}(t)$ is invertible for every derivable term t. It remains to check the two axioms, for which one uses naturality and the laws of Lemma 3.3.16. □

Let us summarise what we have seen in this section. We have a pair of pseudofunctors $\llparenthesis - \rrparenthesis : \mathcal{H}(\mathcal{G}) \leftrightarrows \mathrm{Syn}(\mathcal{G}) : \overline{(-)}$ related by invertible transformations $\llparenthesis - \rrparenthesis \circ \overline{(-)} \cong \mathrm{id}_{\mathsf{Syn}(\mathcal{G})}$ and $\overline{(-)} \circ \llparenthesis - \rrparenthesis = \mathrm{id}_{\mathcal{H}(\mathcal{G})}$. Together these form the claimed biequivalence.

Theorem 3.3.24. For any 2-multigraph $\mathcal{G}$, the pseudofunctors $\llparenthesis - \rrparenthesis : \mathcal{H}(\mathcal{G}) \leftrightarrows \mathrm{Syn}(\mathcal{G}) : \overline{(-)}$ form a biequivalence of biclones. □

We restate the result as a statement of coherence in the style of [JS93].

Corollary 3.3.25. For any 2-multigraph $\mathcal{G}$, the free biclone on $\mathcal{G}$ is biequivalent to the free strict biclone on $\mathcal{G}$. □

We can use Lemma 3.1.23 to parlay the preceding corollary into a normalisation result for $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}$. Since we have no control over the behaviour of constant rewrites, we restrict to 2-multigraphs with no surfaces.

Theorem 3.3.26. Let $\mathcal{G}$ be a 2-multigraph such that for any nodes $A_1, \dots, A_n, B \in \mathcal{G}_0$ and edges $f, g : A_1, \dots, A_n \to B$ the set $\mathcal{G}(A_\bullet; B)(f, g)$ of surfaces $f \Rightarrow g$ is empty. Then $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}(\mathcal{G})$ is locally coherent.

Proof. The approach is standard (*c.f.* [Lei04, p. 16]). Suppose given a pair of rewrites in $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}(\mathcal{G})$ typed by $\Gamma \vdash \tau : t \Rightarrow t' : B$ and $\Gamma \vdash \sigma : t \Rightarrow t' : B$. Since there are no constant rewrites, the definition of $\overline{(-)}$ entails that $\overline{\tau} = \mathrm{id}_{\overline{t}} = \overline{\sigma}$ in $\mathrm{H}^{\mathrm{cl}}(\mathcal{G})$. By Lemma 3.1.23 the pseudofunctor $\overline{(-)}$ is locally faithful, so $\tau \equiv \sigma$, as required. □

Loosely speaking, any diagram of rewrites in $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}$ formed from $\mathsf{assoc}, \iota, \varrho^{(i)}$ and id using the operations of vertical composition and explicit substitution must commute. We shall freely make use of this property from now on.

Adapting the preceding argument to apply to bicategories—and hence recover a version of the classic result of [MP85]—is a minor adjustment. Fix a 2-graph $\mathcal{G}$. Restricting the construction of $\mathcal{H}(-)$ to unary contexts and a fixed variable name (*c.f.* Construction 3.2.15) yields a 2-category; this is free on $\mathcal{G}$ by Lemma 3.1.18. Similarly, the biequivalence of biclones $(\!| - |\!) : \mathcal{H}(\mathcal{G}) \leftrightarrows \mathrm{Syn}(\mathcal{G}) : \overline{(-)}$ restricts to a biequivalence of bicategories. One therefore obtains the following.

Corollary 3.3.27. For any 2-graph $\mathcal{G}$, the free bicategory on $\mathcal{G}$ is biequivalent to the free 2-category on $\mathcal{G}$. □

Alternatively, one may observe that since the internal language for bicategories $\Lambda^{\mathrm{bicat}}_{\mathrm{ps}}$ is constructed by restricting the internal language $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}$ for biclones to unary contexts, any composite of the rewrites assoc, ι and $\varrho^{(i)}$ in $\Lambda^{\mathrm{bicat}}_{\mathrm{ps}}$ must exist in $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}$. Hence the local coherence of $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}$ entails the local coherence of $\Lambda^{\mathrm{bicat}}_{\mathrm{ps}}$.

Corollary 3.3.28. Let $\mathcal{G}$ be a 2-graph such that for any nodes $A, B \in \mathcal{G}_0$ and edges $f, g : A \to B$ the set $\mathcal{G}(A, B)(f, g)$ of surfaces $f \Rightarrow g$ is empty. Then $\Lambda^{\mathrm{bicat}}_{\mathrm{ps}}(\mathcal{G})$ is locally coherent. □

Chapter 4

A type theory for fp-bicategories

In this chapter we extend the type theory $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$ with finite products. We develop a theory of product structures in biclones, and use this to synthesise our type theory $\Lambda_{\mathrm{ps}}^{\times}$. Along the way we pursue a connection with the *representable multicategories* of Hermida [Her00]. Hermida's work can be seen as bridging multicategories and monoidal categories; we show that similar connections hold between clones and cartesian categories, and also between biclones and bicategories with finite products. The resulting translation mediates between products presented by biuniversal arrows (in the style of Hermida's representability) and the presentation in terms of natural isomorphisms or pseudonatural equivalences.

With this abstract framework in place, we examine its implications for the construction of an internal language for biclones with finite products and—by extension—for bicategories with finite products. The resulting type theory provides a calculus for the kind of universal-property reasoning commonly employed when dealing with (bi)limits, and contrasts with previous work on type-theoretic descriptions of 2-dimensional cartesian (closed) structure, in which products are defined by an invertible unit and counit satisfying the triangle laws of an adjunction (*e.g.* [See87, Hil96, Hir13]).

4.1 fp-Bicategories

Let us begin by recalling the notions of bicategory with finite products and product-preserving pseudofunctor. It will be convenient to directly consider all finite products, so that the bicategory is equipped with n-ary products for each $n \in \mathbb{N}$. This reduces the need to deal with the equivalent objects given by re-bracketing binary products. To avoid confusion with the 'cartesian bicategories' of Carboni and Walters [CW87, CKWW08], we call a bicategory with all finite products an *fp-bicategory*. (We will, however, freely make use of the term 'cartesian' when defining finite products in (bi)clones and (bi)multicategories.)

We define n-ary products in a bicategory as a bilimit over a discrete bicategory (set) with n

objects. As we saw in Remark 2.4.2, this can be expressed equivalently as a right biadjoint. For bicategories $\mathcal{B}_1, \dots, \mathcal{B}_n$ the *product bicategory* $\prod_{i=1}^{n} \mathcal{B}_i$ has objects $(B_1, \dots, B_n) \in \prod_{i=1}^{n} ob(\mathcal{B}_i)$ and structure given pointwise. An fp-bicategory is a bicategory $\mathcal{B}$ equipped with a right biadjoint to the diagonal pseudofunctor $\Delta^n : \mathcal{B} \to \mathcal{B}^{\times n} : B \mapsto (B, \dots, B)$ for every $n \in \mathbb{N}$. Applying Definition 2.4.1 in this context, one may equivalently ask for a biuniversal arrow $(\pi_1, \dots, \pi_n) : \Delta^n\big(\prod_n(A_1, \dots, A_n)\big) \to (A_1, \dots, A_n)$ for every $A_1, \dots, A_n \in \mathcal{B}$ $(n \in \mathbb{N})$.

Definition 4.1.1. An *fp-bicategory* $(\mathcal{B}, \Pi_n(-))$ is a bicategory $\mathcal{B}$ equipped with the following data for every $A_1, \dots, A_n \in \mathcal{B}$ $(n \in \mathbb{N})$:

1. A chosen object $\prod_n(A_1, \dots, A_n)$,
2. Chosen arrows $\pi_k : \prod_n(A_1, \dots, A_n) \to A_k$ $(k = 1, \dots, n)$, referred to as *projections*,
3. For every $X \in \mathcal{B}$ an adjoint equivalence

$$\begin{tikzcd} \mathcal{B}(X, \prod_n(A_1, \dots, A_n)) \arrow[r, bend left, "(\pi_1 \circ -, \dots, \pi_n \circ -)"] \arrow[r, phantom, "\perp\simeq"] & \prod_{i=1}^{n} \mathcal{B}(X, A_i) \arrow[l, bend left, "\langle -, \dots, = \rangle"] \end{tikzcd} \tag{4.1}$$

defined by choosing a family of universal arrows we denote $\varpi = (\varpi^{(1)}, \dots, \varpi^{(n)})$.

We call the right adjoint $\langle -, \dots, = \rangle$ the *n-ary tupling.* ◂

Remark 4.1.2. The preceding definition admits two degrees of strictness. Requiring the equivalence (4.1) to be an isomorphism, and $\mathcal{B}$ to be a 2-category, yields the definition of *2-categorical* (Cat-enriched) products. These products are not strict in the 1-categorical sense, however: as the example of $(\mathbf{Cat}, \times, \mathbb{1})$ shows, it may not be the case that $(A \times B) \times C = A \times (B \times C)$. In this thesis, we shall generally write *strict* to mean only that (4.1) is an isomorphism, and specify explicitly when we mean the stronger sense. ◂

Explicitly, the universal arrows of (4.1) may be specified as follows. For any finite family of 1-cells $(t_i : X \to A_i)_{i=1,\dots,n}$, one requires a 1-cell $\langle t_1, \dots, t_n \rangle : X \to \prod_n(A_1, \dots, A_n)$ and a family of invertible 2-cells $(\varpi^{(k)}_{t_1,\dots,t_n} : \pi_k \circ \langle t_\bullet \rangle \Rightarrow t_k)_{k=1,\dots,n}$. These 2-cells are universal in the sense that, for any family of 2-cells $(\alpha_i : \pi_i \circ u \Rightarrow t_i : \Gamma \to A_i)_{i=1,\dots,n}$, there exists a 2-cell $\mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n) : u \Rightarrow \langle t_1, \dots, t_n \rangle : \Gamma \to \prod_{i=1}^{n} A_i$, unique such that

$$\varpi^{(k)}_{t_1,\dots,t_n} \bullet \big(\pi_k \circ \mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n)\big) = \alpha_k : \pi_k \circ u \Rightarrow t_k \tag{4.2}$$

for $k = 1, \dots, n$. One thereby obtains a functor $\langle -, \dots, = \rangle$ and an adjoint equivalence as in (4.1) with counit $\varpi = (\varpi^{(1)}, \dots, \varpi^{(n)})$ and unit $\mathsf{p}^\dagger(\mathrm{id}_{\pi_1 \circ t}, \dots, \mathrm{id}_{\pi_n \circ t}) : t \Rightarrow \langle \pi_1 \circ t, \dots, \pi_n \circ t \rangle$. This defines a *lax* n-ary product structure: one merely obtains an adjunction in (4.1). One turns this into a bicategorical (*pseudo*) product by further requiring the unit and counit to be invertible. The *terminal object* $\mathbf{1}$ arises as $\prod_0()$.

Remark 4.1.3. Throughout we shall assume that the chosen unary product structure on an fp-bicategory is trivial, in the sense that $\prod_1(A) = A$, $\langle t \rangle = t$ and $\varpi_A^{(1)} = \mathfrak{l}_A : \mathrm{Id} \circ t \Rightarrow t$. ◂

Notation 4.1.4.

1. We denote the unit $\mathsf{p}^\dagger(\mathrm{Id}_{\pi_1 \circ t}, \dots, \mathrm{Id}_{\pi_n \circ t}) : t \Rightarrow \langle \pi_1 \circ t, \dots, \pi_n \circ t \rangle$ by ς_t. (We reserve η and ε for the unit and counit of exponential structure.)
2. We write $A_1 \times \cdots \times A_n$ or $\prod_{i=1}^n A_i$ for $\prod_n(A_1, \dots, A_n)$,
3. We write $\langle f_i \rangle_{i=1,\dots,n}$ or simply $\langle f_\bullet \rangle$ for the n-ary tupling $\langle f_1, \dots, f_n \rangle$,
4. Following the 1-categorical notation, for any family of 1-cells $f_i : A_i \to A_i'$ $(i = 1, \dots, n)$ we write $\prod_n(f_1, \dots, f_n)$ or $\prod_{i=1}^n f_i$ for the n-ary tupling $\langle f_1 \circ \pi_1, \dots, f_n \circ \pi_n \rangle : \prod_{i=1}^n A_i \to \prod_{i=1}^n A_i'$, and likewise on 2-cells. ◂

One must take treat the $\prod_i f_i$ notation with some care. In a 1-category, the morphism $f \times A = f \times \mathrm{id}_A$ is equal to the pairing $\langle f \circ \pi_1, \pi_2 \rangle$. In an fp-bicategory, this may not be the case: $f \times A = f \times \mathrm{Id}_A = \langle f \circ \pi_1, \mathrm{Id}_A \circ \pi_2 \rangle$.

Remark 4.1.5. Like any biuniversal arrow, products are unique up to equivalence (*c.f.* Lemma 2.2.7). Explicitly, given adjoint equivalences $(g : C \leftrightarrows \prod_{i=1}^n B_i : h)$ and $(e_i : B_i \leftrightarrows A_i : f_i)_{i=1,\dots,n}$ in a bicategory $\mathcal{B}$, the composite

$$\begin{tikzcd}
& \mathcal{B}(X, \prod_{i=1}^n B_i) \arrow[r, bend left, "(\pi_1\circ-, \dots, \pi_n\circ-)"] \arrow[ld, bend left, "h\circ-"] \arrow[r, phantom, "\perp\simeq"] & \prod_{i=1}^n \mathcal{B}(X, B_i) \arrow[l, bend left, "\langle -, \dots, = \rangle"] \arrow[rd, bend left, "\Pi_{i=1}^n(e_i\circ-)"] & \\
\mathcal{B}(X, C) \arrow[ru, bend left, "g\circ-"] \arrow[ru, phantom, "\perp\simeq"] & & & \prod_{i=1}^n \mathcal{B}(X, A_i) \arrow[lu, bend left, "\Pi_{i=1}^n(f_i\circ-)"] \arrow[lu, phantom, "\perp\simeq"]
\end{tikzcd}$$

yields an adjoint equivalence

$$\begin{tikzcd}
\mathcal{B}(X, C) \arrow[r, bend left, "(((e_1\circ\pi_1)\circ g)\circ-, \dots, ((e_n\circ\pi_n)\circ g)\circ-)"] \arrow[r, phantom, "\perp\simeq"] & \prod_{i=1}^n \mathcal{B}(X, A_i) \arrow[l, bend left, "h\circ\langle f_1\circ-, \dots, f_n\circ= \rangle"]
\end{tikzcd}$$

presenting C as the product of $A_1, \dots, A_n$. ◂

One may generally think of bicategorical product structure as an intensional version of the familiar categorical structure, except the usual equations (*e.g.* [Gib97]) are now witnessed by natural families of invertible 2-cells. It will be useful to have explicit names for these 2-cells.

Construction 4.1.6. Let $(\mathcal{B}, \Pi_n(-))$ be an fp-bicategory. We define the following families of invertible 2-cells:

1. For $(h_i : Y \to A_i)_{i=1,\dots,n}$ and $g : X \to Y$, we define

$$\mathsf{post}(h_\bullet; g) : \langle h_1, \dots, h_n\rangle \circ g \Rightarrow \langle h_1 \circ g, \dots, h_n \circ g\rangle$$

   as $\mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n)$, where α_k is the composite

$$\pi_k \circ (\langle h_1, \dots, h_n\rangle \circ g) \overset{\cong}{\Rightarrow} (\pi_k \circ \langle h_1, \dots, h_n\rangle) \circ g \xRightarrow{\varpi^{(k)} \circ g} h_k \circ g$$

   for $k = 1, \dots, n$.

2. For $(h_i : A_i \to B_i)_{i=1,\dots,n}$ and $(g_i : X \to A_i)_{i=1,\dots,n}$, we define

$$\mathsf{fuse}(h_\bullet; g_\bullet) : (\textstyle\prod_{i=1}^n h_i) \circ \langle g_1, \dots, g_n\rangle \Rightarrow \langle h_1 \circ g_1, \dots, h_n \circ g_n\rangle$$

   as $\mathsf{p}^\dagger(\beta_1, \dots, \beta_n)$, where β_k is defined by the diagram

$$\begin{tikzcd}
\pi_k \circ ((\prod_{i=1}^n h_i) \circ \langle g_1, \dots, g_n\rangle) \arrow[rr, "\beta_k"] \arrow[d, "\cong"'] & & h_k \circ g_k \\
(\pi_k \circ \prod_{i=1}^n h_i) \circ \langle g_1, \dots, g_n\rangle \arrow[r, "\varpi^{(k)} \circ \langle g_1, \dots, g_n\rangle"'] & (h_k \circ \pi_k) \circ \langle g_1, \dots, g_n\rangle \arrow[r, "\cong"'] & h_k \circ (\pi_k \circ \langle g_1, \dots, g_n\rangle) \arrow[u, "h_k \circ \varpi^{(k)}"']
\end{tikzcd}$$

   for $k = 1, \dots, n$.

3. For $(h_i : A_i \to B_i)_{i=1,\dots,n}$ and $(g_j : X_j \to A_j)_{j=1,\dots,n}$ we define

$$\Phi_{h_\bullet, g_\bullet} : (\textstyle\prod_{i=1}^n h_i) \circ (\prod_{i=1}^n g_i) \Rightarrow \prod_{i=1}^n (h_i g_i)$$

   to be the composite $\langle \mathsf{a}^{-1}_{h_1, g_1, \pi_1}, \dots, \mathsf{a}^{-1}_{h_n, g_n, \pi_n}\rangle \bullet \mathsf{fuse}(h_\bullet; g_1 \circ \pi_1, \dots, g_n \circ \pi_n)$. This 2-cell witnesses the pseudofunctoriality of $\prod_n(-, \dots, =)$. ◂

Informally, one can use the preceding construction to translate a sequence of equalities relating the product structure of a cartesian category into a composite of invertible 2-cells—the difficulty, as outlined in the introduction to this thesis, is verifying such a composite satisfies the required coherence laws. As a further step to simplifying this effort, we observe that each of the 2-cells just constructed is natural and satisfies the expected equations. The many isomorphisms required to state these lemmas in their full bicategorical generality tend to obscure the 'self-evident' nature of these results, so we state them for 2-categories with pseudo (bicategorical) products.

Lemma 4.1.7. Let $\mathcal{B}$ be a 2-category with finite pseudo-products. Then for all families of suitable 1-cells f, g, h, f_i, g_i, h_i $(i = 1, \ldots, n)$, the following diagrams commute whenever they are well-typed:

$$\begin{tikzcd}
\langle f_1, \ldots, f_n \rangle \arrow[r, equal] \arrow[dr, equal] & \langle f_1, \ldots, f_n \rangle \circ \mathrm{Id} \arrow[d, "\mathsf{post}"] \\
& \langle f_1 \circ \mathrm{Id}, \ldots, f_n \circ \mathrm{Id} \rangle
\end{tikzcd} \tag{4.3}$$

$$\begin{tikzcd}
\prod_{i=1}^{n} f_i \arrow[r, "(\prod_i f_i) \circ \varsigma_{\mathrm{Id}}"] \arrow[dr, equal] & (\prod_{i=1}^{n} f_i) \circ \langle \pi_1, \ldots, \pi_n \rangle \arrow[d, "\mathsf{fuse}"] \\
& \langle f \circ \pi_1, \ldots, f_n \circ \pi_n \rangle
\end{tikzcd} \tag{4.4}$$

$$\begin{tikzcd}
f \circ g \arrow[r, "\varsigma_f \circ g"] \arrow[dr, "\varsigma_{fg}"'] & \langle \pi_1 \circ f, \ldots, \pi_n \circ f \rangle \circ g \arrow[d, "\mathsf{post}"] \\
& \langle \pi_1 \circ f \circ g, \ldots, \pi_n \circ f \circ g \rangle
\end{tikzcd} \tag{4.5}$$

$$\begin{tikzcd}
\langle f_\bullet \rangle \circ g \circ h \arrow[r, "\mathsf{post} \circ h"] \arrow[dr, "\mathsf{post}"'] & \langle f_\bullet \circ g \rangle \circ h \arrow[d, "\mathsf{post}"] \\
& \langle f_\bullet \circ g \circ h \rangle
\end{tikzcd} \tag{4.6}$$

$$\begin{tikzcd}
\left(\prod_{i=1}^{n} f_i \right) \circ \left(\prod_{i=1}^{n} g_i \right) \circ \langle h_1, \ldots, h_n \rangle \arrow[r, "\Phi_{f_\bullet, g_\bullet} \circ \langle h_1, \ldots, h_n \rangle"] \arrow[d, "(\prod_i f_i) \circ \mathsf{fuse}"'] & \prod_{i=1}^{n} (f_i \circ g_i) \circ \langle h_1, \ldots, h_n \rangle \arrow[d, "\mathsf{fuse}"] \\
\left(\prod_{i=1}^{n} f_i \right) \circ \langle g_1 \circ h_1, \ldots, g_n \circ h_n \rangle \arrow[r, "\mathsf{fuse}"'] & \langle f_1 \circ g_1 \circ h_1, \ldots, f_n \circ g_n \circ h_n \rangle
\end{tikzcd} \tag{4.7}$$

$$\begin{tikzcd}
\left(\prod_{i=1}^{n} f_i \right) \circ \langle g_1, \ldots, g_n \rangle \circ h \arrow[r, "(\prod_i f_i) \circ \mathsf{post}"] \arrow[d, "\mathsf{fuse} \circ h"'] & \left(\prod_{i=1}^{n} f_i \right) \circ \langle g_1 \circ h, \ldots, g_n \circ h \rangle \arrow[d, "\mathsf{fuse}"] \\
\langle f_1 \circ g_1, \ldots, f_n \circ g_n \rangle \circ h \arrow[r, "\mathsf{post}"'] & \langle f_1 \circ g_1 \circ h, \ldots, f_n \circ g_n \circ h \rangle
\end{tikzcd} \tag{4.8}$$

□

In Lemma 4.3.14 we shall see that these laws hold equally within the syntax of the type theory $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$ for fp-bicategories.

The restriction to a base 2-category, rather than a bicategory, turns out to be of no great consequence. Indeed, Power's coherence result restricts as follows to fp-bicategories.

Proposition 4.1.8 ([Pow89b, Theorem 4.1])**.** Every fp-bicategory is biequivalent to a 2-category with strict (2-categorical) products.

Proof. We present Power's proof, adapted to the special case of products. Let $(\mathcal{B}, \Pi_n(-))$ be an fp-bicategory. By the Mac Lane-Paré coherence theorem, $\mathcal{B}$ is biequivalent to a 2-category; by Lemma 2.2.13, this is a 2-category with bicategorical products. We may therefore assume without loss of generality that $(\mathcal{B}, \Pi_n(-))$ is a 2-category with bicategorical products. Now let $\mathrm{Y} : \mathcal{B} \to \mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$ be the Yoneda embedding and $\overline{\mathcal{B}}$ be the closure of $ob(\mathrm{Y}\mathcal{B})$ in $\mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$ under equivalences. The Yoneda embedding factors as a composite $\mathcal{B} \xrightarrow{i} \overline{\mathcal{B}} \xrightarrow{j} \mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$. Since Y is locally an equivalence, the inclusion $i : \mathcal{B} \to \overline{\mathcal{B}}$ is a biequivalence. Choose a pseudoinverse $k : \overline{\mathcal{B}} \to \mathcal{B}$.

Now, for any $P_1, \dots, P_n \in \overline{\mathcal{B}}$ $(n \in \mathbb{N})$ a 2-categorical product $\prod_n(jP_1, \dots, jP_n)$ exists (pointwise) in the 2-category $\mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$: one can show this by a direct calculation or by applying general theory as in [Pow89b, Proposition 3.6] (see also Chapter 6). We show this product also lies in $\overline{\mathcal{B}}$. Since an isomorphism of hom-categories is certainly an equivalence of hom-categories, $\prod_n(jP_1, \dots, jP_n)$ is (up to equivalence) the bicategorical product of $jP_1, \dots, jP_n$ in $\mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$. Moreover, since i and k form a biequivalence, $\mathrm{Y} \circ k = (j \circ i) \circ k \simeq j \circ \mathrm{id}_{\overline{\mathcal{B}}} = j$. So, applying the uniqueness of products up to equivalence and the fact that Y preserves products (Lemma 2.3.4):

$$\prod_n(jP_1, \dots, jP_n) \simeq \prod_n((\mathrm{Y}k)P_1, \dots, (\mathrm{Y}k)P_n) \simeq \mathrm{Y}(\prod_n(kP_1, \dots, kP_n))$$

Since $\mathrm{Y}(\prod_n(kP_1, \dots, kP_n))$ certainly lies in $\overline{\mathcal{B}}$, it follows that $\prod_n(jP_1, \dots, jP_n)$ also lies in $\overline{\mathcal{B}}$, as claimed. □

This result obviates the need to deal with the various 2-cells of Construction 4.1.6. The reader may therefore simplify some of the longer 2-cells we shall construct (for example, in Chapter 7). However, we shall *not* rely on it in what follows.

4.1.1 Preservation of products

fp-Pseudofunctors. Defining preservation of products is straightforward: it is just an instance of preservation of bilimits. We ask that for each $n \in \mathbb{N}$ the biuniversal arrow defining the n-ary product is preserved. Strict preservation of these biuniversal arrows amounts to requiring that the chosen product structure in the domain is taken to exactly the chosen product structure in the target.

Definition 4.1.9. An *fp-pseudofunctor* $(F, \mathrm{q}^{\times})$ between fp-bicategories $(\mathcal{B}, \Pi_n(-))$ and $(\mathcal{C}, \Pi_n(-))$ is a pseudofunctor $F : \mathcal{B} \to \mathcal{C}$ equipped with specified adjoint equivalences

$$\langle F\pi_1, \dots, F\pi_n \rangle : F(\prod_{i=1}^n A_i) \leftrightarrows \prod_{i=1}^n (FA_i) : \mathrm{q}^{\times}_{A_\bullet}$$

for every $A_1, \ldots, A_n \in \mathcal{B}$ $(n \in \mathbb{N})$. We denote the 2-cells witnessing these equivalences as follows:

$$\begin{aligned} \mathsf{u}^{\times}_{A_\bullet} &: \mathrm{Id}_{(\prod_i FA_i)} \Rightarrow \langle F\pi_1, \ldots, F\pi_n \rangle \circ \mathrm{q}^{\times}_{A_\bullet} \\ \mathsf{c}^{\times}_{A_\bullet} &: \mathrm{q}^{\times}_{A_\bullet} \circ \langle F\pi_1, \ldots, F\pi_n \rangle \Rightarrow \mathrm{Id}_{(F\Pi_i A_i)} \end{aligned}$$

We call $(F, \mathrm{q}^{\times})$ *strict* if F is strict and satisfies

$$\begin{aligned} F(\textstyle\prod_n(A_1, \ldots, A_n)) &= \textstyle\prod_n(FA_1, \ldots, FA_n) \\ F(\pi_i^{A_1, \ldots, A_n}) &= \pi_i^{FA_1, \ldots, FA_n} \\ F\langle t_1, \ldots, t_n \rangle &= \langle Ft_1, \ldots, Ft_n \rangle \\ F\varpi^{(i)}_{t_1, \ldots, t_n} &= \varpi^{(i)}_{Ft_1, \ldots, Ft_n} \\ \mathrm{q}^{\times}_{A_1, \ldots, A_n} &= \mathrm{Id}_{\Pi_n(FA_1, \ldots, FA_n)} \end{aligned}$$

with adjoint equivalences canonically induced by the 2-cells $\mathsf{p}^{\dagger}(\mathsf{r}_{\pi_1}, \ldots, \mathsf{r}_{\pi_n}) : \mathrm{Id} \overset{\cong}{\Rightarrow} \langle \pi_1, \ldots, \pi_n \rangle$. ◂

By Lemma 2.2.17, a strict fp-pseudofunctor commutes with the $\mathsf{p}^{\dagger}(-, \ldots, =)$ operation on 2-cells: $F\big(\mathsf{p}^{\dagger}(\alpha_1, \ldots, \alpha_n)\big) = \mathsf{p}^{\dagger}(F\alpha_1, \ldots, F\alpha_n)$.

Remark 4.1.10. The fact that products are unique up to equivalence has the following consequence for fp-pseudofunctors. If $\mathcal{B}$ is a bicategory equipped with two product structures, say $(\mathcal{B}, \Pi_n(-))$ and $\big(\mathcal{B}, \mathrm{Prod}_n(-)\big)$, then for any fp-pseudofunctor $(F, \mathrm{q}^{\times}) : (\mathcal{B}, \Pi_n(-)) \to (\mathcal{C}, \Pi_n(-))$ there exists an (equivalent) fp-pseudofunctor $\big(\mathcal{B}, \mathrm{Prod}_n(-)\big) \to (\mathcal{C}, \Pi_n(-))$ with witnessing equivalence

$$F(\mathrm{Prod}_n(A_1, \ldots, A_n)) \simeq F(\textstyle\prod_n(A_1, \ldots, A_n)) \xrightarrow{\mathrm{q}^{\times}_{A_\bullet}} \textstyle\prod_n(FA_1, \ldots, FA_n)$$

arising from the tupling map $\langle \pi_1, \ldots, \pi_n \rangle : \mathrm{Prod}_n(A_1, \ldots, A_n) \to \prod_n(A_1, \ldots, A_n)$. ◂

We saw in Lemma 2.4.4 that, when a biadjunction is preserved, one obtains an equivalence of pseudofunctors relating the two biadjunctions. We shall make use of the following concrete instance of this fact.

Lemma 4.1.11. For any fp-pseudofunctor $(F, \mathrm{q}^{\times}) : (\mathcal{B}, \Pi_n(-)) \to (\mathcal{C}, \Pi_n(-))$ the family of 1-cells $\mathrm{q}^{\times}_{A_\bullet} : \prod_{i=1}^n FA_i \to F(\prod_{i=1}^n A_i)$ are the components of a pseudonatural transformation $\prod_{i=1}^n (F(-), \ldots, F(=)) \Rightarrow (F \circ \prod_{i=1}^n)(-, \ldots, =)$, and hence an equivalence in $\mathrm{Hom}(\prod_{i=1}^n \mathcal{B}, \mathcal{C})$.

Proof. The witnessing 2-cells $\mathsf{nat}_{f_\bullet}$ filling

$$\begin{tikzcd} \prod_{i=1}^n FA_i \arrow[r, "\prod_i Ff_i"] \arrow[d, "\mathrm{q}^{\times}_{A_\bullet}"'] \arrow[dr, phantom, "\overset{\mathsf{nat}_{f_\bullet}}{\Leftarrow}"] & \prod_{i=1}^n FA_i' \arrow[d, "\mathrm{q}^{\times}_{A'_\bullet}"] \\ F(\prod_{i=1}^n A_i) \arrow[r, "F(\prod_i f_i)"'] & F(\prod_{i=1}^n A_i') \end{tikzcd}$$

are defined as the following composite:

$$\begin{tikzcd}
\mathrm{q}^{\times}_{A'_\bullet} \circ \prod_{i=1}^{n} Ff_i \arrow[r, "\mathsf{nat}_{f_\bullet}"] \arrow[d, "\cong"'] & F(\prod_{i=1}^{n} f_i) \circ \mathrm{q}^{\times}_{A_\bullet} \\
\big(\mathrm{q}^{\times}_{A'_\bullet} \circ (\prod_{i=1}^{n} Ff_i)\big) \circ \mathrm{Id}_{(\prod_n FA_\bullet)} \arrow[d, "\mathrm{q}^{\times}_{A'_\bullet} \circ (\prod_{i=1}^{n} Ff_i) \circ \mathsf{u}^{\times}_{A_\bullet}"'] & \mathrm{Id}_{F(\prod_n A'_\bullet)} \circ \Big(F(\prod_{i=1}^{n} f_i) \circ \mathrm{q}^{\times}_{A_\bullet}\Big) \arrow[u, "\cong"'] \\
\big(\mathrm{q}^{\times}_{A'_\bullet} \circ \prod_{i=1}^{n} F(f_i)\big) \circ \Big(\langle F(\pi_\bullet)\rangle \circ \mathrm{q}^{\times}_{A_\bullet}\Big) \arrow[d, "\cong"'] & \Big(\mathrm{q}^{\times}_{A'_\bullet} \circ \langle F\pi_\bullet\rangle\Big) \circ \big(F(\prod_{i=1}^{n} f_i) \circ \mathrm{q}^{\times}_{A_\bullet}\big) \arrow[u, "\mathsf{c}^{\times}_{A'_\bullet} \circ F(\prod_i f_i) \circ \mathrm{q}^{\times}_{A_\bullet}"'] \\
\mathrm{q}^{\times}_{A'_\bullet} \circ \big(\big(\prod_{i=1}^{n} F(f_i) \circ \langle F(\pi_\bullet)\rangle\big) \circ \mathrm{q}^{\times}_{A_\bullet}\big) \arrow[d, "\mathrm{q}^{\times}_{A'_\bullet} \circ \mathsf{fuse} \circ \mathrm{q}^{\times}_{A_\bullet}"'] & \mathrm{q}^{\times}_{A'_\bullet} \circ \big(\big(\langle F\pi_\bullet\rangle \circ F(\prod_{i=1}^{n} f_i)\big) \circ \mathrm{q}^{\times}_{A_\bullet}\big) \arrow[u, "\cong"'] \\
\mathrm{q}^{\times}_{A'_\bullet} \circ \Big(\langle F(f_\bullet) \circ F(\pi_\bullet)\rangle \circ \mathrm{q}^{\times}_{A_\bullet}\Big) \arrow[d, "\mathrm{q}^{\times}_{A'_\bullet} \circ \big\langle \phi^{F}_{f_\bullet;\pi_\bullet} \big\rangle \circ \mathrm{q}^{\times}_{A_\bullet}"'] & \mathrm{q}^{\times}_{A'_\bullet} \circ \Big(\langle F(\pi_\bullet) \circ F(\prod_{i=1}^{n} f_i)\rangle \circ \mathrm{q}^{\times}_{A_\bullet}\Big) \arrow[u, "\mathrm{q}^{\times}_{A'_\bullet} \circ \mathsf{fuse}^{-1} \circ \mathrm{q}^{\times}_{A_\bullet}"'] \\
\mathrm{q}^{\times}_{A'_\bullet} \circ \Big(\langle F(f_\bullet \circ \pi_\bullet)\rangle \circ \mathrm{q}^{\times}_{A_\bullet}\Big) \arrow[r, "\mathrm{q}^{\times}_{A'_\bullet} \circ \big\langle F(\varpi^{(-1)}), \dots, F(\varpi^{(-n)}) \big\rangle \circ \mathrm{q}^{\times}_{A_\bullet}"'] & \mathrm{q}^{\times}_{A'_\bullet} \circ \Big(\langle F(\pi_\bullet \circ \prod_{i=1}^{n} f_i)\rangle \circ \mathrm{q}^{\times}_{A_\bullet}\Big) \arrow[u, "\mathrm{q}^{\times}_{A'_\bullet} \circ \big\langle (\phi^{F}_{\pi_\bullet;\prod_i f_i})^{-1} \big\rangle \circ \mathrm{q}^{\times}_{A_\bullet}"']
\end{tikzcd}$$

□

In a cartesian category it is is often useful to 'unpack' an n-ary tupling from inside a cartesian functor in the following manner:

$$\begin{aligned}
\langle F\pi_1, \dots, F\pi_n\rangle \circ F\langle f_1, \dots, f_n\rangle &= \langle F(\pi_\bullet) \circ F\langle f_1, \dots, f_n\rangle\rangle \\
&= \langle F(\pi_\bullet \circ \langle f_1, \dots, f_n\rangle)\rangle \\
&= \langle Ff_1, \dots, Ff_n\rangle
\end{aligned}$$

In an fp-bicategory, one obtains a natural family of 2-cells we call unpack.

Construction 4.1.12. For any fp-pseudofunctor $F : (\mathcal{B}, \Pi_n(-)) \to (\mathcal{C}, \Pi_n(-))$ the invertible 2-cell $\mathsf{unpack}_{f_\bullet} : \langle F\pi_1, \dots, F\pi_n\rangle \circ F\langle f_1, \dots, f_n\rangle \Rightarrow \langle Ff_1, \dots, Ff_n\rangle : FX \to \prod_{i=1}^{n} FB_i$ is defined to be $\mathsf{p}^{\dagger}(\tau_1, \dots, \tau_n)$, where τ_k $(k = 1, \dots, n)$ is given by the following diagram:

$$\begin{tikzcd}
\pi_k \circ (\langle F\pi_1, \dots, F\pi_n\rangle \circ F\langle f_1, \dots, f_n\rangle) \arrow[r, "\tau_k"] \arrow[d, "\cong"'] & Ff_k \\
(\pi_k \circ \langle F\pi_1, \dots, F\pi_n\rangle) \circ F\langle f_1, \dots, f_n\rangle \arrow[d, "\varpi^{(k)} \circ F\langle f_1, \dots, f_n\rangle"'] & \\
F(\pi_k) \circ F\langle f_1, \dots, f_n\rangle \arrow[r, "\phi^{F}_{\pi_k,\langle f_\bullet\rangle}"'] & F\left(\pi_i \circ \langle f_1, \dots, f_n\rangle\right) \arrow[uu, "F\varpi^{(k)}"']
\end{tikzcd}$$

◂

As with the 2-cells of Construction 4.1.6, it is useful to have certain coherence properties ready-made. For unpack one has the following.

Lemma 4.1.13. For any fp-pseudofunctor $(F, \mathrm{q}^{\times}) : (\mathcal{B}, \Pi_n(-)) \to (\mathcal{C}, \Pi_n(-))$ and family of 1-cells $(f_i : X_i \to Y_i)_{i=1,\dots,n}$ in $\mathcal{B}$, the following diagram commutes:

$$\begin{tikzcd}
(\langle F\pi_1, \dots, F\pi_n \rangle \circ F(\prod_{i=1}^n f_i)) \circ \mathrm{q}^{\times}_{X_\bullet} \arrow[r, "\mathsf{unpack} \circ \mathrm{q}^{\times}_{X_\bullet}"] \arrow[d, "\cong"'] & \langle F(f_1 \circ \pi_1), \dots, F(f_n \circ \pi_n) \rangle \circ \mathrm{q}^{\times}_{X_\bullet} \\
\langle F\pi_1, \dots, F\pi_n \rangle \circ \Big(F(\prod_{i=1}^n f_i) \circ \mathrm{q}^{\times}_{X_\bullet} \Big) \arrow[d, "\langle F\pi_1, \dots, F\pi_n \rangle \circ \mathsf{nat}_{f_\bullet}"'] & \langle Ff_1 \circ F\pi_1, \dots, Ff_n \circ F\pi_n \rangle \circ \mathrm{q}^{\times}_{X_\bullet} \arrow[u, "\langle \phi^F_{f_1,\pi_1}, \dots, \phi^F_{f_n,\pi_n} \rangle \circ \mathrm{q}^{\times}_{X_\bullet}"'] \\
\langle F\pi_1, \dots, F\pi_n \rangle \circ \Big(\mathrm{q}^{\times}_{Y_\bullet} \circ (\prod_{i=1}^n Ff_i) \Big) \arrow[d, "\cong"'] & ((\prod_{i=1}^n Ff_i) \circ \langle F\pi_1, \dots, F\pi_n \rangle) \circ \mathrm{q}^{\times}_{X_\bullet} \arrow[u, "\mathsf{fuse} \circ \mathrm{q}^{\times}_{X_\bullet}"'] \\
\Big(\langle F\pi_1, \dots, F\pi_n \rangle \circ \mathrm{q}^{\times}_{Y_\bullet} \Big) \circ (\prod_{i=1}^n Ff_i) \arrow[d, "(\mathrm{u}^{\times}_{Y_\bullet})^{-1} \circ (\prod_i Ff_i)"'] & (\prod_{i=1}^n Ff_i) \circ \Big(\langle F\pi_1, \dots, F\pi_n \rangle \circ \mathrm{q}^{\times}_{X_\bullet} \Big) \arrow[u, "\cong"'] \\
\mathrm{Id}_{(\prod_i FY_i)} \circ (\prod_{i=1}^n Ff_i) \arrow[r, "\cong"'] & (\prod_{i=1}^n Ff_i) \circ \mathrm{Id}_{(\prod_i FX_i)} \arrow[u, "(\prod_i Ff_i) \circ \mathrm{u}^{\times}_{X_\bullet}"']
\end{tikzcd}$$

□

Morphisms of fp-pseudofunctors. The tricategorical nature of **Bicat** leads naturally to a consideration of 2- and 3-cells relating fp-pseudofunctors. Experience from the 1-categorical setting, however, suggests that new definitions are not needed. For cartesian functors $F, G : (\mathbb{C}, \Pi_n(-)) \to (\mathbb{D}, \Pi_n(-))$ it is elementary to check that every natural transformation $\alpha : F \Rightarrow G$ satisfies

$$\begin{tikzcd}
F\big(\prod_{i=1}^n A_i \big) \arrow[r, "\langle F\pi_1, \dots, F\pi_n \rangle"] \arrow[d, "\alpha_{(\prod_n A_\bullet)}"'] & \prod_{i=1}^n F(A_i) \arrow[d, "\prod_{i=1}^n \alpha_{A_i}"] \\
G\big(\prod_{i=1}^n A_i \big) \arrow[r, "\langle G\pi_1, \dots, G\pi_n \rangle"'] & \prod_{i=1}^n G(A_i)
\end{tikzcd} \tag{4.9}$$

The corresponding bicategorical fact is the following: every pseudonatural transformation extends canonically to an *fp-transformation* (*c.f.* the *monoidal pseudonatural transformations* of [Hou07, Chapter 3]).

Definition 4.1.14. Let $(F, \mathrm{q}^{\times})$ and $(G, \mathrm{u}^{\times})$ be fp-pseudofunctors $(\mathcal{B}, \Pi_n(-)) \to (\mathcal{C}, \Pi_n(-))$. An *fp-transformation* $(\alpha, \overline{\alpha}, \alpha^{\times})$ is a pseudonatural transformation $(\alpha, \overline{\alpha}) : F \Rightarrow G$ equipped with a 2-cell $\alpha^{\times}_{A_1, \dots, A_n}$ as in the following diagram for every $A_1, \dots, A_n \in \mathcal{B}$ $(n \in \mathbb{N})$:

$$\begin{tikzcd}
F\big(\prod_{i=1}^n A_i\big) \arrow[r, "\langle F\pi_1, \dots, F\pi_n\rangle"] \arrow[d, swap, "\alpha_{(\prod_n A_\bullet)}"] \arrow[dr, phantom, "\overset{\alpha^{\times}_{A_1, \dots, A_n}}{\Leftarrow}"] & \prod_{i=1}^n F(A_i) \arrow[d, "\prod_{i=1}^n \alpha_{A_i}"] \\
G\big(\prod_{i=1}^n A_i\big) \arrow[r, swap, "\langle G\pi_1, \dots, G\pi_n\rangle"] & \prod_{i=1}^n G(A_i)
\end{tikzcd}$$

These 2-cells are required to satisfy

$$\begin{tikzcd}
\pi_k \circ ((\prod_{i=1}^n \alpha_{A_i}) \circ \langle F\pi_1, \dots, F\pi_n\rangle) \arrow[rr, "\pi_k \circ \alpha^{\times}_{A_1, \dots, A_n}"] \arrow[d, swap, "\cong"] & & \pi_k \circ \big(\langle G\pi_1, \dots, G\pi_n\rangle \circ \alpha_{(\prod_n A_\bullet)}\big) \arrow[d, "\cong"] \\
(\pi_k \circ \prod_{i=1}^n \alpha_{A_i}) \circ \langle F\pi_1, \dots, F\pi_n\rangle \arrow[d, swap, "\varpi^{(k)} \circ \langle F\pi_\bullet\rangle"] & & (\pi_k \circ \langle G\pi_1, \dots, G\pi_n\rangle) \circ \alpha_{(\prod_n A_\bullet)} \arrow[dd, "\varpi^{(k)} \circ \alpha_{(\prod_n A_\bullet)}"] \\
(\alpha_{A_k} \circ \pi_k) \circ \langle F\pi_1, \dots, F\pi_n\rangle \arrow[d, swap, "\cong"] & & \\
\alpha_{A_k} \circ (\pi_k \circ \langle F\pi_1, \dots, F\pi_n\rangle) \arrow[r, swap, "\alpha_{A_k} \circ \varpi^{(k)}"] & \alpha_{A_k} \circ F\pi_k \arrow[r, swap, "\overline{\alpha}_{\pi_k}"] & G\pi_k \circ \alpha_{(\prod_n A_\bullet)}
\end{tikzcd}$$

◂

Lemma 4.1.15. Let $(F, \mathrm{q}^{\times})$ and $(G, \mathrm{u}^{\times})$ be fp-pseudofunctors $(\mathcal{B}, \Pi_n(-)) \to (\mathcal{C}, \Pi_n(-))$ and $(\alpha, \overline{\alpha}) : F \Rightarrow G$ a pseudonatural transformation. Then, where $\alpha^{\times}_{A_1, \dots, A_n}$ is defined to be the composite

$$\begin{tikzcd}
(\prod_{i=1}^n \alpha_{A_i}) \circ \langle F\pi_1, \dots, F\pi_n\rangle \arrow[r, "\alpha^{\times}_{A_1, \dots, A_n}"] \arrow[d, swap, "\mathsf{fuse}"] & \langle G\pi_1, \dots, G\pi_n\rangle \circ \alpha_{A_1 \times \dots \times A_n} \\
\langle \alpha_{A_1} \circ F\pi_1, \dots, \alpha_{A_n} \circ F\pi_n\rangle \arrow[r, swap, "\langle \overline{\alpha}_{\pi_1}, \dots, \overline{\alpha}_{\pi_n}\rangle"] & \big\langle G\pi_1 \circ \alpha_{(\prod_n A_\bullet)}, \dots, G\pi_n \circ \alpha_{(\prod_n A_\bullet)}\big\rangle \arrow[u, swap, "\mathsf{post}^{-1}"]
\end{tikzcd}$$

the triple $(\alpha, \overline{\alpha}, \alpha^{\times})$ is an fp-transformation.

Proof. A straightforward diagram chase unwinding the definitions of fuse and post. □

In a similar vein, one might define an *fp-biequivalence* of fp-bicategories to consist of a pair of fp-pseudofunctors $(F, \mathrm{q}^{\times})$ and $(G, \mathrm{u}^{\times})$, with fp-transformations $FG \leftrightarrows \mathrm{id}$ and $GF \leftrightarrows \mathrm{id}$ and invertible modifications forming equivalences $FG \simeq \mathrm{id}$ and $GF \simeq \mathrm{id}$. The composition of fp-transformations is the usual composition of pseudonatural transformations, with the composite witnessing 2-cell for (4.9) given by the evident pasting diagram. However, this apparently more-structured notion of biequivalence may always be constructed from a biequivalence of the underlying bicategories.

Lemma 4.1.16. For any fp-bicategories $(\mathcal{B}, \Pi_n(-))$ and $(\mathcal{C}, \Pi_n(-))$, there exists an fp-biequivalence $(\mathcal{B}, \Pi_n(-)) \simeq (\mathcal{C}, \Pi_n(-))$ if and only if there exists a biequivalence of the underlying bicategories.

Proof. The reverse direction is immediate. The forward direction follows from Lemma 2.2.13 and Lemma 4.1.15. □

In this thesis we will only ever be concerned with the existence of a biequivalence between fp-bicategories, not its particular structure. It will therefore suffice to work with biequivalences throughout.

4.2 Product structure from representability

In Chapter 3 we saw that a type theory for biclones—and, by restriction to unary contexts, bicategories—could be extracted directly from the construction of the free biclone on a signature. In order to take a similar approach in the case of fp-bicategories, we develop the theory of product structures in biclones.

What does it mean to define products in a biclone? As usual, the categorical case is informative. Thinking of (sorted) clones as cartesian versions of multicategories suggests that products in a clone ought to arise in a way paralleling tensor products in a multicategory. Translating the work of Hermida [Her00] to clones in the most naïve way possible, one might require a family of arrows $\rho_{X_\bullet} : X_1, \ldots, X_n \to \prod_n(X_1, \ldots, X_n)$ in a clone $\mathbb{C}$ inducing isomorphisms $\mathbb{C}(X_1, \ldots, X_n; A) \cong \mathbb{C}(\prod_n(X_1, \ldots, X_n); A)$ by precomposition. On the other hand, Lambek [Lam89] defines products in a multicategory $\mathbb{L}$ by requiring isomorphisms of the form $\mathbb{L}(\Gamma; \prod_n(X_1, \ldots, X_n)) \cong \prod_{i=1}^n \mathbb{L}(\Gamma; A_i)$. Connecting these two approaches to product structure will be the focus of the next section.

Taking multicategories as our starting point, we shall study two forms of universal property, corresponding to Hermida's and Lambek's definitions, respectively. We shall show how these notions may be applied to clones and, moreover, demonstrate that for clones they actually coincide (Theorem 4.2.20).

Thereafter, in Section 4.2.2, we shall see how one can extract the usual product structure of the simply-typed lambda calculus from the theory of such *cartesian clones*. This will provide the template for lifting this work to the bicategorical setting, and hence for the product structure of the type theory $\Lambda_{\text{ps}}^{\times}$.

4.2.1 Cartesian clones and representability

We start by recalling a little of the theory of (representable) multicategories and their relationship to monoidal categories. Extensive overviews are available in [Lei04, Yau16].

Representable multicategories. The notion of *multicategory* is a crucial part of Lambek's extended study of deductive systems [Lam69, Lam80, Lam86, Lam89]. The motivating example takes objects to be types in some sequent calculus and multimaps $X_1, \dots, X_n \vdash Y$ to be derivable sequents; composition is given by a cut rule. Lambek defines tensor products and (left and right) internal homs in a multicategory by the existence of certain natural isomorphisms. More recent work by Hermida [Her00] connects these ideas to the categorical setting by making precise the correspondence between monoidal categories and so-called *representable* multicategories.

Definition 4.2.1 ([Lam69, Lam89])**.** A *multicategory* $\mathbb{L}$ consists of the following data:

- A set $ob(\mathbb{L})$ of *objects*,
- For every sequence $X_1, \dots, X_n$ $(n \in \mathbb{N})$ of objects and object Y a *hom-set* $\mathbb{L}(X_1, \dots, X_n; Y)$ consisting of *multimaps* or *arrows* $f : X_1, \dots, X_n \to Y$ (here n may be zero). As with (bi)clones, we sometimes denote sequences $X_1, \dots, X_n$ by Greek letters $\Gamma, \Delta, \dots$ to emphasise the connection with contexts,
- For every $X \in ob(\mathbb{L})$ an *identity multimap* $\mathrm{id}_X : X \to X$,
- For every set of sequences $\Gamma_1, \dots, \Gamma_n$ and objects $Y_1, \dots, Y_n, Z$, a *composition* operation

  $$\circ_{\Gamma_\bullet; Y_\bullet; Z} : \mathbb{L}(Y_1, \dots, Y_n; Z) \times \textstyle\prod_{i=1}^{n} \mathbb{L}(\Gamma_i; Y_i) \to \mathbb{L}(\Gamma_1, \dots, \Gamma_n; Z)$$

  we denote by $\circ_{\Gamma_\bullet; Y_\bullet; Z}\big(f, (g_1, \dots, g_n)\big) := f \circ \langle g_1, \dots, g_n \rangle$.

This is subject to three axioms requiring that composition is associative and unital. We call multimaps of the form $X \to Y$ *linear.* ◂

Notation 4.2.2. Note that we write composition in a multicategory as $f \circ \langle g_1, \dots, g_n \rangle$ and substitution in a clone as $f[g_1, \dots, g_n]$. ◂

Multicategories are also known as *coloured (planar) operads* (*e.g.* [Yau16]). Multicategories form a category MultiCat of multicategories and their functors, and also a 2-category of multicategories, multicategory functors, and transformations (*e.g.* [Lei04, Chapter 2]).

Definition 4.2.3.

1. A *functor* $F : \mathbb{L} \to \mathbb{M}$ between multicategories $\mathbb{L}$ and $\mathbb{M}$ consists of:

   - A mapping $F : ob(\mathbb{L}) \to ob(\mathbb{M})$ on objects,
   - For every $X_1, \dots, X_n, Y \in \mathbb{L}$ $(n \in \mathbb{N})$ a mapping on hom-sets $F_{X_\bullet; Y} : \mathbb{L}(X_1, \dots, X_n; Y) \to \mathbb{M}(FX_1, \dots, FX_n; FY)$,

   such that composition and the identity are preserved.

2. A *transformation* $\alpha : F \Rightarrow G$ between multicategory functors $F, G : \mathbb{L} \to \mathbb{M}$ is a family of multimaps $(\alpha_X : FX \to GX)_{X \in ob(\mathbb{L})}$ such that for every $f : X_1, \ldots, X_n \to Y$ the equation $Ff \circ (\alpha_{X_1}, \ldots, \alpha_{X_n}) = \alpha_Y \circ (Gf)$ holds. ◂

From the perspective of deductive systems, moving from multicategories to clones amounts to changing the composition operation from a cut rule to a substitution operation. The composition operation of a multicategory is *linear*: given maps $(h_i : \Gamma \to Y_i)_{i=1,\ldots,m}$ and $f : Y_1, \ldots, Y_m \to Z$ in a multicategory, the composite $f \circ \langle h_1, \ldots, h_m \rangle$ has type $\Gamma, \ldots, \Gamma \to Z$. By contrast, the substitution operation in a clone is *cartesian*: given maps h_i and f as above, the substitution $f[h_1, \ldots, h_m]$ has type $\Gamma \to Z$.

Every multicategory $\mathbb{L}$ defines a category $\overline{\mathbb{L}}$ by restricting to linear morphisms. Conversely, every monoidal category $(\mathbb{C}, \otimes, I)$ canonically defines a multicategory with objects those of $\mathbb{C}$ and multimaps $X_1, \ldots, X_n \to Y$ given by morphisms $X_1 \otimes \cdots \otimes X_n \to Y$ (for a specified bracketing of the n-ary tensor product). A natural question is therefore the following: under what conditions is the category $\overline{\mathbb{L}}$ corresponding to a multicategory monoidal? Hermida answers this by showing that there exists a 2-equivalence between the 2-category **MonCat** of monoidal categories and the 2-category of representable multicategories.

Definition 4.2.4. A *representable multicategory* $\mathbb{L}$ is a multicategory equipped with a chosen object $\mathrm{T}_n(X_1, \ldots, X_n) \in \mathbb{L}$ and a chosen multimap $\rho_{X_1,\ldots,X_n} : X_1, \ldots, X_n \to \mathrm{T}_n(X_1, \ldots, X_n)$ for every $X_1, \ldots, X_n \in \mathbb{L}$ $(n \in \mathbb{N})$ such that

1. Each chosen $\rho_{X_1,\ldots,X_n}$ is *representable*: for every $Y \in \mathbb{L}$, precomposition with $\rho_{X_1,\ldots,X_n}$ induces an isomorphism $\mathbb{L}(X_1, \ldots, X_n; Y) \cong \mathbb{L}(\mathrm{T}_n(X_1, \ldots, X_n), Y)$ of hom-sets, and
2. The representable arrows are closed under composition. ◂

Thus, a multimap $\rho_{X_\bullet}$ is representable if and only if for every $h : X_1, \ldots, X_n \to Y$ there exists a unique multimap $h^\sharp : \prod_n(X_1, \ldots, X_n) \to Y$ such that $h^\sharp \circ \rho_{X_1,\ldots,X_n} = h$.

Remark 4.2.5. It is common to refer to the arrows $\rho_{X_\bullet}$ of the preceding definition as *universal*; we change the terminology slightly because we will imminently define a multicategorical version of universal arrows in the sense of Chapter 2. The two concepts are related: the representability condition (1) above is equivalent to requiring that each $\mathbb{L}(X_1, \ldots, X_n; -) : \mathbb{L} \to \mathrm{Set}$ is representable, which is in turn equivalent to specifying a universal arrow from the terminal set to this functor (*c.f.* [Mac98, Chapter III]). ◂

We briefly recapitulate Hermida's construction.

Lemma 4.2.6 ([Her00, Definition 9.6])**.** For every representable multicategory $\mathbb{L}$, the associated category $\overline{\mathbb{L}}$ is monoidal.

Proof. The tensor product $X \otimes Y$ is $\mathrm{T}_2(X, Y)$ and the unit I arises from the empty sequence, as $\mathrm{T}_0()$. The map $f \otimes g$ is defined by the universal property, as the unique linear map filling the following diagram:

$$\begin{tikzcd}
\mathrm{T}_2(X,Y) \arrow[r, dashed, "f\otimes g"] & \mathrm{T}_2(X',Y') \\
X,Y \arrow[u, "\rho_{X,Y}"] \arrow[r, "(f,g)"'] & X',Y' \arrow[u, "\rho_{X',Y'}"']
\end{tikzcd}$$

□

The second condition (2) is necessary: it allows one to use the universal property to check the axioms of a monoidal category involving iterated tensors $(A \otimes B) \otimes C$ (*c.f.* the preservation conditions for lifting monoidal structure to a category of algebras [Sea13], in particular the *left-linear classifiers* of [FS18]).

Cartesian multicategories. Representability is a universal property that allows us to construct *monoidal* structure. To construct *cartesian* structure, however, one requires more. In particular, one ought to obtain Lambek's definition of *cartesian multicategory* [Lam89, §4], requiring multimaps $\pi_i : \prod_n(A_1, \dots, A_n) \to A_i$ $(i = 1, \dots, n)$ inducing natural isomorphisms $\mathbb{L}(\Gamma; \prod_n(X_1, \dots, X_n)) \cong \prod_{i=1}^n \mathbb{L}(\Gamma; A_i)$. Next we shall see how to obtain a definition equivalent to Lambek's, but phrased in terms of universal arrows. This will be the starting point for our comparison between product structure and representability.

Definition 4.2.7. Let $F : \mathbb{L} \to \mathbb{M}$ be a functor of multicategories and $X \in \mathbb{M}$. A *universal arrow from F to X* is a pair $(R, u : FR \to X)$ such that for every $h : FA_1, \dots, FA_n \to X$ there exists a unique multimap $h^\dagger : A_1, \dots, A_n \to R$ such that $u \circ (Fh^\dagger) = h$. ◂

Remark 4.2.8. One could define universal arrows slightly more generally, by taking a universal arrow from F to X to be a *sequence* of objects $R_1, \dots, R_n$ with a universal multimap $FR_1, \dots, FR_n \to X$. The definition given seems sufficient for our purposes, so we do not seek this extra generality. ◂

As in the categorical case, we can rephrase the definition of universal arrow as a natural isomorphism.

Lemma 4.2.9. Let $F : \mathbb{L} \to \mathbb{M}$ be a functor of multicategories and $X \in \mathbb{M}$. The following are equivalent:

1. A specified universal arrow (R, u) from F to X,
2. A choice of object $R \in \mathbb{L}$ and an isomorphism $\mathbb{L}(A_1, \ldots, A_n; R) \cong \mathbb{M}(FA_1, \ldots, FA_n; X)$, multinatural in the sense that for any $f : A_1, \ldots, A_n \to B$ the following diagram commutes:

$$\begin{tikzcd}
\mathbb{L}(B; R) \arrow[r, "\cong"] \arrow[d, "(-)\circ\langle f\rangle"'] & \mathbb{M}(FB; X) \arrow[d, "(-)\circ\langle Ff\rangle"] \\
\mathbb{L}(A_1, \ldots, A_n; R) \arrow[r, "\cong"'] & \mathbb{M}(FA_1, \ldots, FA_n; X)
\end{tikzcd}$$

Proof. The direction (1)⇒(2) is clear. For the reverse, denote the isomorphism by $\phi_{A_\bullet} : \mathbb{L}(A_1, \ldots, A_n; R) \to \mathbb{M}(FA_1, \ldots, FA_n; X)$ and its inverse by $\psi_{A_\bullet}$. We show that $u := \phi_R(\mathrm{id}_R) : FR \to X$ is a universal arrow by showing that that $\psi_{A_\bullet}(-)$ is inverse to $\phi_R(\mathrm{id}_R) \circ \langle F(-)\rangle$.

First, for any $h : FA_1, \ldots, FA_n \to X$, naturality of ϕ with respect to the multimap $\psi_{A_\bullet}(h) : A_1, \ldots, A_n \to R$ gives the equation $\phi_R(\mathrm{id}_R) \circ \langle F\psi_{A_\bullet}(h)\rangle = \phi_{A_\bullet}\psi_{A_\bullet}(h) = h$. Second, let $g : A_1, \ldots, A_n \to R$. The naturality of ψ with respect to g entails that $\psi_{A_\bullet}(\phi_R(\mathrm{id}_R) \circ \langle Fg\rangle) = \psi_R\phi_R(\mathrm{id}_R) \circ \langle g\rangle = g$, as required. □

The category of multicategories MultiCat has products given as follows. For multicategories $\mathbb{L}$ and $\mathbb{M}$ the product $\mathbb{L} \times \mathbb{M}$ has objects pairs $(M, N) \in ob(\mathbb{L}) \times ob(\mathbb{M})$ and hom-sets

$$(\mathbb{L} \times \mathbb{M})((A_1, B_1), \ldots, (A_n, B_n); (X, Y)) := \mathbb{L}(A_1, \ldots, A_n; X) \times \mathbb{M}(B_1, \ldots, B_n; Y)$$

Composition is defined pointwise:

$$\begin{tikzcd}
\mathbb{L}(A_\bullet; X) \times \mathbb{M}(B_\bullet; Y) \times \prod_{i=1}^n (\mathbb{L}(\Gamma_i, A_i) \times \mathbb{M}(\Delta_i, B_i)) \arrow[rr, "\circ_{\mathbb{L}\times\mathbb{M}}"] \arrow[dr, "\cong"'] & & \mathbb{L}(\Gamma_\bullet; X) \times \mathbb{M}(\Delta_\bullet; Y) \\
& (\mathbb{L}(A_\bullet; X) \times \prod_{i=1}^n(\mathbb{L}(\Gamma_i, A_i)) \times (\mathbb{M}(B_\bullet; Y) \times \prod_{i=1}^n \mathbb{M}(\Delta_i, B_i)) \arrow[ur, "\circ_{\mathbb{L}} \times \circ_{\mathbb{M}}"'] &
\end{tikzcd} \tag{4.10}$$

The product structure is then almost identical to that in Cat. Then for every multicategory $\mathbb{L}$ and $n \in \mathbb{N}$ there exists a diagonal functor $\Delta^n : \mathbb{L} \to \mathbb{L}^{\times n} : X \mapsto (X, \ldots, X)$, and Definition 4.2.7 provides a natural notion of multicategory with finite products.

Definition 4.2.10. A *cartesian multicategory* is a multicategory $\mathbb{L}$ equipped with a choice of universal arrow $\Delta^n \prod_n(X_1, \ldots, X_n) \to (X_1, \ldots, X_n)$ from Δ^n to $(X_1, \ldots, X_n)$ for every $X_1, \ldots, X_n \in \mathbb{L}$ $(n \in \mathbb{N})$. ◂

Applying Lemma 4.2.9, asking for a multicategory to have finite products is equivalent to asking for a chosen sequence of linear multimaps $(\pi_i : \prod_n(X_1, \ldots, X_n) \to X_i)_{i=1,\ldots,n}$, inducing a multinatural family of isomorphisms

$$\mathbb{L}(\Gamma; \textstyle\prod_n(X_1, \dots, X_n)) \cong \mathbb{L}^{\times n}\big((\Gamma, \dots, \Gamma); (X_1, \dots, X_n)\big) = \textstyle\prod_{i=1}^{n} \mathbb{L}(\Gamma; X_i) \tag{4.11}$$

for every $X_1, \dots, X_n \in \mathbb{L}$ $(n \in \mathbb{N})$. One thereby recovers Lambek's definition of cartesian products in a multicategory [Lam89, §4].

Cartesian clones. We wish to extend the two definitions we have just seen from multicategories to clones. Thinking of (sorted) clones as cartesian versions of multicategories suggests the following construction, in which we re-use the notation of Notation 3.1.19 (p. 49).

Construction 4.2.11. Every clone $(S, \mathbb{C})$ canonically defines a multicategory $\mathrm{M}\mathbb{C}$ with

- $ob(\mathrm{M}C) := S$,
- $(\mathrm{M}\mathbb{C})(X_1, \dots, X_n; Y) := \mathbb{C}(X_1, \dots, X_n; Y)$

Composition is defined as follows. For every family of multimaps $g_i : \Gamma_i \to Y_i$ $(i = 1, \dots, n)$ and multimap $f : Y_1, \dots, Y_n \to Z$ we define the composite $f \circ \langle g_1, \dots, g_n \rangle$ in $\mathrm{M}\mathbb{C}$ to be the substitution $f[g_1 \boxtimes \cdots \boxtimes g_n]$ in $\mathbb{C}$. The identity $\mathrm{id}_{X,X} \in (\mathrm{M}C)(X; X)$ is the unary projection $\mathsf{p}^{(1)} \in \mathbb{C}(X, X)$, and the axioms follow directly from the three laws of a clone. ◂

Notation 4.2.12. Motivated by the preceding construction, we shall sometimes write id_A for the projection $\mathsf{p}_1^{(1)} : A \to A$ in a clone, and refer to it as the *identity* on A. ◂

It is clear that this construction extends to a faithful functor $\mathrm{M}(-) : \mathrm{Clone} \to \mathrm{MultiCat}$, yielding a commutative diagram

$$\begin{tikzcd}
\mathrm{Clone} \arrow[rr, "\mathrm{M}(-)"] \arrow[dr, "\overline{(-)}"'] & & \mathrm{MultiCat} \arrow[dl, "\overline{(-)}"] \\
& \mathrm{Cat} &
\end{tikzcd} \tag{4.12}$$

in which the downward arrows restrict to unary/linear arrows. We define representability and products in Clone by applying the definition to the image of $\mathrm{M}(-)$.

Definition 4.2.13.

1. A *representable clone* is a clone $(S, \mathbb{C})$ equipped with a choice of representable structure on $\mathrm{M}\mathbb{C}$.
2. A *cartesian clone* is a clone $(S, \mathbb{C})$ equipped with a choice of cartesian structure on $\mathrm{M}\mathbb{C}$. ◂

Example 4.2.14. Every category with finite products $(\mathbb{C}, \Pi_n(-))$ defines a clone $\mathrm{Cl}(\mathbb{C})$ (recall Example 3.1.7(2) on page 39). This clone is cartesian, with product structure exactly as in $\mathbb{C}$. ◂

A clone may therefore be equipped with two kinds of tensor. In the representability case, one asks for representable arrows $X_1, \dots, X_n \to \mathrm{T}_n(X_1, \dots, X_n)$. In the cartesian case, one asks for universal arrows $\prod_n(X_1, \dots, X_n) \to X_i$ for $i = 1, \dots, n$. In terms of the internal language, these may be thought of as *tupling* and *projection* operations, respectively. Identifying representable arrows with a tupling operation (an identification we shall make precise in Corollary 4.2.21), the question then becomes: how does one construct a tupling operation given only projections, and how does one construct projections given only a tupling operation?

In the light of Lemma 4.2.9, we can already construct a tupling operation from projections, and so from cartesian structure. If $\mathrm{M}\mathbb{C}$ has finite products witnessed by a universal arrow $\pi = (\pi_1, \dots, \pi_n) : \prod_n(X_1, \dots, X_n) \to (X_1, \dots, X_n)$ for each $X_1, \dots, X_n \in S$ $(n \in \mathbb{N})$, then for every sequence of objects Γ one obtains a mapping $\psi_\Gamma : \prod_{i=1}^n(\mathrm{M}\mathbb{C})(\Gamma; X_i) \to (\mathrm{M}\mathbb{C})\big(\Gamma; \prod_n(X_1, \dots, X_n)\big)$ such that the following equations hold for every multimap $h : \Gamma \to \prod_n(X_1, \dots, X_n)$ and sequence of multimaps $(f_i : \Gamma \to X_i)_{i=1,\dots,n}$:

$$\psi_\Gamma(\pi_1[h], \dots, \pi_n[h]) = h \quad \text{and} \quad \pi_i[\psi_\Gamma(f_1, \dots, f_n)] = f_i \ \ (i = 1, \dots, n) \tag{4.13}$$

Thus, $\psi_\Gamma(-, \dots, =)$ provides a ‘tupling’ operation. This is substantiated by the next lemma.

Definition 4.2.15. Let $(S, \mathbb{C})$ be a clone. A multimap $f : X_1, \dots, X_n \to Y$ in $\mathbb{C}$ is *invertible* or an *iso* if there exists a family of unary multimaps $(g_i : Y \to X_i)_{i=1,\dots,n}$ in $\mathbb{C}$ such that $f[g_1, \dots, g_n] = \mathrm{id}_Y$ and $g_i[f] = \mathsf{p}^{(i)}_{X_\bullet}$ for $i = 1, \dots, n$. If there exists an invertible multimap $f : X_1, \dots, X_n \to Y$ we say $X_1, \dots, X_n$ and Y are *isomorphic*, and write $X_1, \dots, X_n \cong Y$. ◂

A small adaptation of the usual categorical proof shows that inverses in a clone are unique, in the sense that if f has inverses $(g_1, \dots, g_n)$ and $(g'_1, \dots, g'_n)$ then $g_i = g'_i$ for $i = 1, \dots, n$.

Lemma 4.2.16. Let $(S, \mathbb{C})$ be a cartesian clone. Then, where the n-ary product of $X_1, \dots, X_n \in S$ $(n \in \mathbb{N})$ is witnessed by the universal arrow $(\pi_1, \dots, \pi_n) : \prod_n(X_1, \dots, X_n) \to (X_1, \dots, X_n)$,

$$\psi_{X_\bullet}(\mathsf{p}^{(1)}_{X_\bullet}, \dots, \mathsf{p}^{(n)}_{X_\bullet})[\pi_1, \dots, \pi_n] = \mathrm{id}_{\prod_n(X_1,\dots,X_n)}$$

Hence $X_1, \dots, X_n \cong \prod_n(X_1, \dots, X_n)$.

Proof. For the first part one uses the two equations of (4.13):

$$\begin{aligned}
\psi_{X_\bullet}(\mathsf{p}^{(1)}_{X_\bullet}, \dots, \mathsf{p}^{(n)}_{X_\bullet})[\pi_1, \dots, \pi_n] &= \psi_{(\prod_n X_\bullet)}\left(\pi_\bullet\left[\psi_{X_\bullet}(\mathsf{p}^{(1)}_{X_\bullet}, \dots, \mathsf{p}^{(n)}_{X_\bullet})[\pi_1, \dots, \pi_n]\right]\right) && \text{by (4.13)} \\
&= \psi_{(\prod_n X_\bullet)}\left(\pi_\bullet\left[\psi_{X_\bullet}(\mathsf{p}^{(1)}_{X_\bullet}, \dots, \mathsf{p}^{(n)}_{X_\bullet})\right][\pi_1, \dots, \pi_n]\right) \\
&= \psi_{(\prod_n X_\bullet)}\left(\mathsf{p}^{(\bullet)}_{X_\bullet}[\pi_1, \dots, \pi_n]\right) && \text{by (4.13)} \\
&= \psi_{(\prod_n X_\bullet)}(\pi_1, \dots, \pi_n)
\end{aligned}$$

$$\begin{aligned} &= \psi_{(\prod_n X_\bullet)}\big(\pi_1\big[\mathrm{id}_{(\prod_n X_\bullet)}\big], \ldots, \pi_n\big[\mathrm{id}_{(\prod_n X_\bullet)}\big]\big) \\ &= \mathrm{id}_{(\prod_n X_\bullet)} && \text{by (4.13)} \end{aligned}$$

Then $(\pi_i : \prod_n(X_1, \ldots, X_n) \to X_i)_{i=1,\ldots,n}$ and $\psi_{X_\bullet}(\mathsf{p}^{(1)}_{X_\bullet}, \ldots, \mathsf{p}^{(n)}_{X_\bullet})$ form the claimed isomorphism. □

We now turn to examinining how representability (thought of as 'tupling') gives rise to 'projections'. The next lemma is the key construction.

Lemma 4.2.17. For any representable clone $(S, \mathbb{C})$ and $X_1, \ldots, X_n \in S$ $(n \in \mathbb{N})$ there exist multimaps $\pi_i : \mathrm{T}_n(X_1, \ldots, X_n) \to X_i$ $(i = 1, \ldots, n)$ such that

$$\pi_i \circ \rho_{X_\bullet} = \mathsf{p}^{(i)}_{X_\bullet} \quad \text{and} \quad \rho_{X_\bullet}[\pi_1, \ldots, \pi_n] = \mathrm{id}_{\prod X_\bullet}$$

where $\rho_{X_\bullet}$ is the representable arrow.

Proof. By representability, we may define $\pi_i := \big(\mathsf{p}^{(i)}_{X_\bullet}\big)^\sharp$. The first claim then holds by assumption. For the second, observing that $(\rho_{X_\bullet})^\sharp = \mathrm{id}_{\prod X_\bullet}$, it suffices to show that $\rho_{X_\bullet}[\pi_1, \ldots, \pi_n]\,[\rho_{X_\bullet}] = \rho_{X_\bullet}$. But this is straightforward:

$$\rho_{X_\bullet}[\pi_1, \ldots, \pi_n]\,[\rho_{X_\bullet}] = \rho_{X_\bullet}[\pi_\bullet[\rho_{X_\bullet}]] = \rho_{X_\bullet}\big[\mathsf{p}^{(1)}, \ldots, \mathsf{p}^{(n)}\big] = \rho_{X_\bullet}$$

□

Another important consequence of Lemma 4.2.17 is that, in the case of clones, representable arrows are always closed under composition.

Lemma 4.2.18. For any clone $(S, \mathbb{C})$, the multicategory $\mathrm{M}\mathbb{C}$ is representable if and only if for every $X_1, \ldots, X_n \in S$ $(n \in \mathbb{N})$ there exists a chosen object $\mathrm{T}_n(X_1, \ldots, X_n)$ and a representable multimap $\rho_{X_\bullet} : X_1, \ldots, X_n \to \mathrm{T}_n(X_1, \ldots, X_n)$.

Proof. It suffices to show that, for any clone $(S, \mathbb{C})$, the representable multimaps in $\mathrm{M}\mathbb{C}$ are closed under composition. Suppose given representable multimaps

$$\begin{gathered} \rho_{X_\bullet} : X_1, \ldots, X_n \to \mathrm{T}_n(X_1, \ldots, X_n) \\ \rho_{Y_\bullet} : Y_1, \ldots, Y_m \to \mathrm{T}_m(Y_1, \ldots, Y_m) \\ \rho_{(\mathrm{T}X_\bullet, \mathrm{T}Y_\bullet)} : \mathrm{T}_n X_\bullet, \mathrm{T}_m Y_\bullet \to \mathrm{T}_2(\mathrm{T}_n X_\bullet, \mathrm{T}_m Y_\bullet) \end{gathered}$$

We want to show that the composite $\rho_{(\mathrm{T}X_\bullet, \mathrm{T}Y_\bullet)} \circ \langle \rho_{X_\bullet}, \rho_{Y_\bullet} \rangle$ in $\mathrm{M}\mathbb{C}$, which is the composite $\rho_{(\mathrm{T}X_\bullet, \mathrm{T}Y_\bullet)}\big[\rho_{X_\bullet} \boxtimes \rho_{Y_\bullet}\big] = \rho_{(\mathrm{T}X_\bullet, \mathrm{T}Y_\bullet)}\big[\rho_{X_\bullet}\big[\mathsf{p}^{(1)}, \ldots, \mathsf{p}^{(n)}\big], \rho_{Y_\bullet}\big[\mathsf{p}^{(n+1)}, \ldots, \mathsf{p}^{(n+m)}\big]\big]$ in $\mathbb{C}$, is representable.

By Lemma 4.2.17, we may define multimaps

$$\begin{aligned}
\pi_i^X &: \mathrm{T}_n(X_1, \dots, X_n) \to X_i && \text{for } i = 1, \dots, n \\
\pi_j^Y &: \mathrm{T}_m(Y_1, \dots, Y_m) \to Y_j && \text{for } j = 1, \dots, m \\
\pi_1^{X,Y} &: \mathrm{T}_2(\mathrm{T}_n X_\bullet, \mathrm{T}_m Y_\bullet) \to \mathrm{T}_n X_\bullet \\
\pi_2^{X,Y} &: \mathrm{T}_2(\mathrm{T}_n X_\bullet, \mathrm{T}_m Y_\bullet) \to \mathrm{T}_m Y_\bullet
\end{aligned}$$

Then, setting

$$Z_i := \begin{cases} X_i & \text{for } i = 1, \dots, n \\ Y_{i-n} & \text{for } i = n+1, \dots, n+m \end{cases}$$

we define $\overline{\pi}_i : \mathrm{T}_2(\mathrm{T}_n X_\bullet, \mathrm{T}_m Y_\bullet) \to Z_i$ by iterated applications of π_i:

$$\overline{\pi}_i := \begin{cases} \pi_i^X\left[\pi_1^{X,Y}\right] & \text{for } 1 \leqslant i \leqslant n \\ \pi_{i-n}^Y\left[\pi_2^{X,Y}\right] & \text{for } n+1 \leqslant i \leqslant n+m \end{cases} \tag{4.14}$$

The rest of the proof revolves around proving the following two equalities in $\mathbb{C}$:

$$\begin{tikzcd}
X_1, \dots, X_n, Y_1, \dots, Y_m \arrow[r, "\mathsf{p}^{(i)}"] \arrow[d, "{[\rho_{X_\bullet} \boxtimes \rho_{Y_\bullet}]}"'] & Z_i \\
\mathrm{T}_n X_\bullet, \mathrm{T}_m Y_\bullet \arrow[r, "\rho_{(\mathrm{T}X_\bullet, \mathrm{T}Y_\bullet)}"'] & \mathrm{T}_2(\mathrm{T}_n X_\bullet, \mathrm{T}_m Y_\bullet) \arrow[u, "\overline{\pi}_i"']
\end{tikzcd} \tag{4.15}$$

$$\begin{tikzcd}
\mathrm{T}_2(\mathrm{T}_n X_\bullet, \mathrm{T}_m Y_\bullet) \arrow[r, equal] \arrow[d, "{[\overline{\pi}_1, \dots, \overline{\pi}_{n+m}]}"'] & \mathrm{T}_2(\mathrm{T}_n X_\bullet, \mathrm{T}_m Y_\bullet) \\
X_1, \dots, X_n, Y_1, \dots, Y_m \arrow[r, "{[\rho_{X_\bullet} \boxtimes \rho_{Y_\bullet}]}"'] & \mathrm{T}_n X_\bullet, \mathrm{T}_m Y_\bullet \arrow[u, "\rho_{(\mathrm{T}X_\bullet, \mathrm{T}Y_\bullet)}"']
\end{tikzcd} \tag{4.16}$$

Indeed, if these two diagrams commute, then for any $g : X_1, \dots, X_n, Y_1, \dots, Y_m \to A$ one may define $g^\sharp : \mathrm{T}_2(\mathrm{T}_n X_\bullet, \mathrm{T}_m Y_\bullet) \to A$ to be the composite $g[\overline{\pi}_1, \dots, \overline{\pi}_{n+m}]$. It then follows that that $(-)^\sharp$ is the inverse to precomposing with $\overline{\rho} := \rho_{(\mathrm{T}X_\bullet, \mathrm{T}Y_\bullet)}[\rho_{X_\bullet} \boxtimes \rho_{Y_\bullet}]$:

$$g[\overline{\pi}_1, \dots, \overline{\pi}_{n+m}]\,[\overline{\rho}] = g[\overline{\pi}_1[\overline{\rho}], \dots, \overline{\pi}_{n+m}[\overline{\rho}]] \overset{(4.15)}{=} g\big[\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n+m)}\big] = g$$

while, for any $h : \mathrm{T}_2(\mathrm{T}_n X_\bullet, \mathrm{T}_m Y_\bullet) \to A$,

$$h[\overline{\rho}]\,[\overline{\pi}_1, \dots, \overline{\pi}_{n+m}] \overset{(4.16)}{=} h\Big[\mathsf{p}^{(1)}_{\mathrm{T}(\mathrm{T}X_\bullet, \mathrm{T}Y_\bullet)}\Big] = h$$

as required.

It therefore remains to establish the commutativity of the two diagrams above. We compute (4.15) directly. For example, for $1 \leqslant i \leqslant n$, unfolding the universal property of each of the projections gives

$$\begin{aligned}
\overline{\pi}_i\big[\rho_{(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}\big]\,[\rho_{X_\bullet} \boxtimes \rho_{Y_\bullet}] &= \pi_i^X\left[\pi_1^{X,Y}\right]\big[\rho_{(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}\big]\,[\rho_{X_\bullet} \boxtimes \rho_{Y_\bullet}] \\
&= \pi_i^X\left[\pi_1^{X,Y}\big[\rho_{(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}\big]\right][\rho_{X_\bullet} \boxtimes \rho_{Y_\bullet}] \\
&= \pi_i^X\left[\mathsf{p}^{(1)}_{(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}\right][\rho_{X_\bullet} \boxtimes \rho_{Y_\bullet}] \\
&= \pi_i^X\left[\mathsf{p}^{(1)}_{(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}[\rho_{X_\bullet} \boxtimes \rho_{Y_\bullet}]\right] \\
&= \pi_i^X\big[\rho_{X_\bullet}[\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)}]\big] \\
&= \pi_i^X[\rho_{X_\bullet}][\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)}] \\
&= \mathsf{p}^{(i)}[\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)}] \\
&= \mathsf{p}^{(i)}
\end{aligned}$$

as required. For (4.16), Lemma 4.2.17 entails that

$$\rho_{X_\bullet}[\overline{\pi}_1, \dots, \overline{\pi}_n] = \rho_{X_\bullet}\left[\pi_1^X\left[\pi_1^{X,Y}\right], \dots, \pi_n^X\left[\pi_1^{X,Y}\right]\right] = \rho_{X_\bullet}\left[\pi_\bullet^X\right]\left[\pi_1^{X,Y}\right] = \pi_1^{X,Y}$$

and hence that

$$\begin{aligned}
\rho_{(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}\big[\rho_{X_\bullet}[\mathsf{p}^{(\bullet)}], \rho_{Y_\bullet}[\mathsf{p}^{(\bullet)}]\big]\,[\overline{\pi}_\bullet] &= \rho_{(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}\big[\rho_{X_\bullet}[\overline{\pi}_\bullet], \rho_{Y_\bullet}[\overline{\pi}_\bullet]\big] \\
&= \rho_{(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}\left[\pi_1^{X,Y}, \pi_2^{X,Y}\right] \\
&= \mathrm{id}_{\mathrm{T}(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}
\end{aligned}$$

as required. □

We now make precise the sense in which the inverse to precomposing with a representable arrow provides a tupling operation. The product structure on a representable clone is, as expected, given by the 1-cells constructed in Lemma 4.2.17.

Lemma 4.2.19. For any clone $(S, \mathbb{C})$, the following are equivalent:

1. $(S, \mathbb{C})$ is representable,
2. $(S, \mathbb{C})$ is cartesian.

Proof. $\boxed{\Rightarrow}$ We prove the forward direction first. Suppose $\rho_{X_\bullet} : X_1, \dots, X_n \to \mathrm{T}_n(X_1, \dots, X_n)$ is representable; we claim the required universal arrow is given by the sequence of multimaps $(\pi_1, \dots, \pi_n) : \Delta\mathrm{T}_n(X_1, \dots, X_n) \to (X_1, \dots, X_n)$ defined in Lemma 4.2.17. To this end, let

$(f_i : \Gamma \to X_i)_{i=1,\dots,n}$ in $\mathbb{C}$. We set $\psi_\Gamma(f_1, \dots, f_n) : \Gamma \to \mathrm{T}_n(X_1, \dots, X_n)$ to be the composite $\rho_{X_\bullet}[f_1, \dots, f_n]$. By Lemma 4.2.17,

$$\pi_i \circ \big(\psi_\Gamma(f_1, \dots, f_n)\big) = \pi_i[\rho_{X_\bullet}[f_1, \dots, f_n]] = \mathsf{p}^{(i)}_{X_\bullet}[f_1, \dots, f_n] = f_i$$

for $i = 1, \dots, n$, so it remains to show that $\psi_\Gamma(\pi_1[h], \dots, \pi_n[h]) = h$ for every $h : \Gamma \to \mathrm{T}_n(X_1, \dots, X_n)$. Applying the lemma again,

$$\psi_\Gamma(\pi_1[h], \dots, \pi_n[h]) = \rho_{X_\bullet}[\pi_1[h], \dots, \pi_n[h]] = \rho_{X_\bullet}[\pi_1, \dots, \pi_n]\,[h] = h$$

as required.

$\boxed{\Leftarrow}$ We claim that $\rho_{X_\bullet} := \psi_{X_\bullet}(\mathsf{p}^{(1)}_{X_\bullet}, \dots, \mathsf{p}^{(n)}_{X_\bullet}) : X_1, \dots, X_n \to \prod_n(X_1, \dots, X_n)$ is representable.

To this end, suppose $h : X_1, \dots, X_n \to A$. We define $h^\dagger : \prod_n(X_1, \dots, X_n) \to A$ to be the composite $h[\pi_1, \dots, \pi_n]$. Then

$$\begin{aligned}
h^\dagger[\rho_{X_\bullet}] &= h[\pi_1, \dots, \pi_n]\Big[\psi_\Gamma(\mathsf{p}^{(1)}_{X_\bullet}, \dots, \mathsf{p}^{(n)}_{X_\bullet})\Big] \\
&= h\Big[\pi_\bullet\Big[\psi_\Gamma(\mathsf{p}^{(1)}_{X_\bullet}, \dots, \mathsf{p}^{(n)}_{X_\bullet})\Big]\Big] \\
&= h\Big[\mathsf{p}^{(1)}_{X_\bullet}, \dots, \mathsf{p}^{(n)}_{X_\bullet}\Big] \\
&= h
\end{aligned}$$

so the existence part of the claim holds. It remains to check the equality $(f[\rho_{X_\bullet}])^\dagger = f$ for an arbitrary $f : \prod_n(X_1, \dots, X_n) \to A$. Examining the equality

$$(f[\rho_{X_\bullet}])^\dagger = f[\rho_{X_\bullet}]\,[\pi_1, \dots, \pi_n] = f\Big[\psi_{X_\bullet}(\mathsf{p}^{(1)}_{X_\bullet}, \dots, \mathsf{p}^{(n)}_{X_\bullet})[\pi_1, \dots, \pi_n]\Big]$$

it suffices to show that $\psi_{X_\bullet}(\mathsf{p}^{(1)}_{X_\bullet}, \dots, \mathsf{p}^{(n)}_{X_\bullet})[\pi_1, \dots, \pi_n]$ is the identity. This is Lemma 4.2.16. □

We summarise the last two results in the following theorem. The final case is Lemma 4.2.9.

Theorem 4.2.20. For any clone $(S, \mathbb{C})$, the following are equivalent:

1. $(S, \mathbb{C})$ is representable,
2. For every $X_1, \dots, X_n \in S$ $(n \in \mathbb{N})$ there exists a choice of object $\prod_n(X_1, \dots, X_n) \in S$ together with a representable multimap $\rho_{X_\bullet} : X_1, \dots, X_n \to \prod_n(X_1, \dots, X_n)$,
3. $(S, \mathbb{C})$ is cartesian,
4. For any $X_1, \dots, X_n \in S$ $(n \in \mathbb{N})$ there exists a chosen object $\prod_n(X_1, \dots, X_n) \in S$ and an isomorphism $(\mathrm{M}\mathbb{C})\big(\Gamma; \prod_n(X_1, \dots, X_n)\big) \cong \prod_{i=1}^n (\mathrm{M}\mathbb{C})(\Gamma; X_i)$, multinatural in the sense that for any $f : \Gamma \to A$ the following diagram commutes:

$$\begin{tikzcd}
(\mathrm{M}\mathbb{C})\big(\Gamma; \prod_n(X_1, \dots, X_n)\big) \arrow[r, "\cong"] & \prod_{i=1}^n (\mathrm{M}\mathbb{C})(\Gamma; X_i) \\
(\mathrm{M}\mathbb{C})\big(A; \prod_n(X_1, \dots, X_n)\big) \arrow[u, "(-)\circ\langle f\rangle"] \arrow[r, "\cong"'] & \prod_{i=1}^n (\mathrm{M}\mathbb{C})(A; X_i) \arrow[u, "(-)\circ\langle f\rangle"']
\end{tikzcd}$$

□

In the case of clones, therefore, the two approaches to defining product structure—Hermida's representability or Lambek's natural isomorphisms—actually coincide. We tie this back to Hermida's equivalence between monoidal categories and representable multicategories with the following observation.

Corollary 4.2.21. For any representable clone $(S, \mathbb{C})$, the monoidal structure on the category $\overline{\mathrm{M}\mathbb{C}}$ associated to $\mathrm{M}\mathbb{C}$ is cartesian.

Proof. The required natural isomorphism follows by restricting the isomorphism (4.11) to linear multimaps. Explicitly, the n-ary product of $X_1, \ldots, X_n$ is $\prod_n(X_1, \ldots, X_n)$, and the projections are $\pi_i : \prod_n(X_1, \ldots, X_n) \to X_i$. The n-ary tupling of maps $(f_i : A \to X_i)_{i=1,\ldots,n}$ is given via the representable arrow $\rho_{X_\bullet}$ for $X_1, \ldots, X_n$, as $\rho_{X_\bullet}[f_1, \ldots, f_n]$. □

It is reasonable to suggest that one could refine Hermida's 2-equivalence between monoidal categories and representable multicategories to a 2-equivalence between cartesian categories and representable clones; the calculations required would take us beyond the theory we shall actually need, so we do not pursue the point here. Instead we turn to the syntactic implications of the theory just developed.

4.2.2 From cartesian clones to type theory

From cartesian clones to cartesian categories. In Chapter 3 we saw that the free category on a graph could be constructed by restricting the free clone on that graph to its unary operations. This fact extends to cartesian clones and cartesian categories. To show this, we need to enrich our notion of signature to include product structure. The definition was already hinted at in Example 3.1.8.

Definition 4.2.22. A $\Lambda^{\times}$-*signature* $\mathcal{S} = (\mathfrak{B}, \mathcal{G})$ consists of

1. A set of base types $\mathfrak{B}$,
2. A multigraph $\mathcal{G}$ with nodes generated by the grammar

$$A_1, \ldots, A_n ::= B \mid \textstyle\prod_n(A_1, \ldots, A_n) \qquad (B \in \mathfrak{B}, n \in \mathbb{N}) \tag{4.17}$$

If the graph $\mathcal{G}$ is a 2-graph we call the signature *unary.* A *homomorphism* of $\Lambda^{\times}$-signatures $h : \mathcal{S} \to \mathcal{S}'$ is a multigraph homomorphism $h : \mathcal{G} \to \mathcal{G}'$ which respects the product structure in the sense that $h(\prod_n(A_1, \ldots, A_n)) = \prod_n(hA_1, \ldots, hA_n)$. We denote the category of $\Lambda^{\times}$-signatures and their homomorphisms by $\Lambda^{\times}$-sig, and the full subcategory of unary $\Lambda^{\times}$-signatures by $\Lambda^{\times}\text{-sig}\big|_1$. ◂

Notation 4.2.23. For any $\Lambda^{\times}$-signature $\mathcal{S} = (\mathfrak{B}, \mathcal{G})$ we write $\widetilde{\mathfrak{B}}$ for the set generated from $\mathfrak{B}$ by the grammar (4.17) (equivalently, the set $\mathcal{G}_0$ of nodes in $\mathcal{G}$). In particular, when the signature is just a set (*i.e.* the graph $\mathcal{G}$ has no edges) we denote the signature $\mathcal{S} = (\mathfrak{B}, \mathcal{S})$ simply by $\widetilde{\mathfrak{B}}$. ◂

The following lemma mirrors the situation for graphs and 2-multigraphs.

Lemma 4.2.24. The embedding $\iota : \Lambda^{\times}\text{-sig}\big|_1 \hookrightarrow \Lambda^{\times}\text{-sig}$ has a right adjoint.

Proof. Define the functor $\widetilde{\mathcal{L}} : \Lambda^{\times}\text{-sig} \to \Lambda^{\times}\text{-sig}\big|_1$ to be the restriction of the corresponding functor $\mathcal{L} : \mathrm{MGrph} \to \mathrm{Grph}$. Thus, $\widetilde{\mathcal{L}}$ restricts a signature $(\mathfrak{B}, \mathcal{G})$ to the signature with base types $\mathfrak{B}$ and multigraph $\mathcal{LG}$ containing only edges of the form $X \to Y$. This is a right adjoint to the given inclusion because $\mathcal{L}$ is right adjoint to the inclusion $\mathrm{Grph} \hookrightarrow \mathrm{MGrph}$. □

Every cartesian category $(\mathbb{C}, \Pi_n(-))$ has an underlying unary $\Lambda^{\times}$-signature with edges $X \to Y$ given by morphisms $X \to Y$ in $\mathbb{C}$ (*c.f.* [Cro94, Theorem 4.9.2]). Similarly, every cartesian clone $(S, \mathbb{C}, \Pi_n(-))$ has an underlying $\Lambda^{\times}$-signature with the edges given by multimaps. We wish to construct the free cartesian clone over such a signature. Theorem 4.2.20 guarantees that it is sufficient to add a representable arrow $A_1, \ldots, A_n \to \prod_n(A_1, \ldots, A_n)$ for every sequence of types $A_1, \ldots, A_n$ $(n \in \mathbb{N})$. For the construction we follow the forward direction of the proof of Lemma 4.2.19.

Construction 4.2.25. For any $\Lambda^{\times}$-signature $\mathcal{S} = (\mathfrak{B}, \mathcal{G})$, define a clone $(\mathcal{G}_0, \mathbb{FCl}^{\times}(\mathcal{S}))$ with sorts generated from $\mathfrak{B}$ by the rules

$$A_1, \ldots, A_n ::= B \mid \textstyle\prod_n(A_1, \ldots, A_n) \qquad (B \in \mathfrak{B}, n \in \mathbb{N})$$

as the following deductive system:

$$\frac{c \in \mathcal{G}(A_1, \ldots, A_n; B)}{c \in \mathbb{FCl}^{\times}(\mathcal{S})(A_1, \ldots, A_n; B)}$$

$$\frac{}{\mathsf{p}^{(i)}_{A_1, \ldots, A_n} \in \mathbb{FCl}^{\times}(\mathcal{S})(A_1, \ldots, A_n; A_i)}\ (1 \leqslant i \leqslant n)$$

$$\frac{f \in \mathbb{FCl}^{\times}(\mathcal{S})(A_1, \ldots, A_n; B) \qquad \big(g_i \in \mathbb{FCl}^{\times}(\mathcal{S})(X_\bullet; A_i)\big)_{i=1,\ldots,n}}{f[g_1, \ldots, g_n] \in \mathbb{FCl}^{\times}(\mathcal{S})(X_\bullet; B)}$$

$$\frac{}{\mathsf{tup}_{A_\bullet} \in \mathbb{FCl}^{\times}(\mathcal{S})\left(A_1, \ldots, A_n; \textstyle\prod_n(A_1, \ldots, A_n)\right)}$$

$$\frac{}{\mathsf{proj}^{(i)}_{A_\bullet} \in \mathbb{FCl}^{\times}(\mathcal{S})\left(\textstyle\prod_n(A_1, \ldots, A_n); A_i\right)}\ (1 \leqslant i \leqslant n)$$

subject to an equational theory requiring

- The clone laws hold with projection $\mathsf{p}^{(i)}_{A_\bullet}$ and substitution $f[g_1, \dots, g_n]$,
- $\mathsf{proj}^{(i)}_{A_\bullet}[\mathsf{tup}_{A_\bullet}] \equiv \mathsf{p}^{(i)}_{A_\bullet}$ for $i = 1, \dots, n$,
- $\mathsf{tup}_{A_\bullet}\Big[\mathsf{proj}^{(n)}_{A_\bullet}, \dots, \mathsf{proj}^{(n)}_{A_\bullet}\Big] \equiv \mathsf{p}^{(1)}_{(\prod_n A_\bullet)}$. ◂

The clone $\mathbb{F}\mathbb{C}\mathrm{l}^{\times}(\mathcal{S})$ is cartesian because it is representable. Indeed, for any $A_1, \dots, A_n, B \in \mathcal{G}_0$, the equational laws ensure that the map $(-) \circ \mathsf{tup}_{A_\bullet}$ has inverse $(-)\Big[\mathsf{proj}^{(n)}_{A_\bullet}, \dots, \mathsf{proj}^{(n)}_{A_\bullet}\Big]$, giving rise to the required natural isomorphism $\mathbb{F}\mathbb{C}\mathrm{l}^{\times}(\mathcal{S})(\prod_n(A_1, \dots, A_n); B) \cong \mathbb{F}\mathbb{C}\mathrm{l}^{\times}(\mathcal{S})(A_1, \dots, A_n; B)$.

In order to state that this construction yields the free cartesian clone, we need to define a notion of product-preserving clone homomorphism. This is the clone-theoretic translation of Definition 2.2.11, requiring that the universal arrow is preserved.

Definition 4.2.26. A *cartesian clone homomorphism* $h : (S, \mathbb{C}, \Pi_n(-)) \to (T, \mathbb{D}, \Pi_n(-))$ is a clone homomorphism $h : (S, \mathbb{C}) \to (T, \mathbb{D})$ such that the canonical map $\psi_{\prod A_\bullet}(h\pi_1, \dots, h\pi_n) : h(\prod_n(A_1, \dots, A_n)) \to \prod_n (hA_1, \dots, A_n)$ is invertible for every $A_1, \dots, A_n \in S$ $(n \in \mathbb{N})$.

We call h *strict* if

$$h(\textstyle\prod_n(A_1, \dots, A_n)) = \textstyle\prod_n (hA_1, \dots, hA_n)$$
$$h(\pi_i^{A_\bullet}) = \Big(\textstyle\prod_n(hA_1, \dots, hA_n) \xrightarrow{\pi_i} h(A_i)\Big) \qquad (i = 1, \dots, n)$$

for every $A_1, \dots, A_n \in S$ $(n \in \mathbb{N})$. ◂

Lemma 4.2.27. For any cartesian clone $(T, \mathbb{D}, \Pi_n(-))$, $\Lambda^{\times}$-signature $\mathcal{S}$ and $\Lambda^{\times}$-signature homomorphism $h : \mathcal{S} \to \mathbb{D}$, there exists a unique strict cartesian clone homomorphism $h^{\#} : \mathbb{F}\mathbb{C}\mathrm{l}^{\times}(\mathcal{S}) \to \mathbb{D}$ such that $h^{\#} \circ \iota = h$, for $\iota : \mathcal{S} \hookrightarrow \mathbb{F}\mathbb{C}\mathrm{l}^{\times}(\mathcal{S})$ the inclusion.

Proof. We define $h^{\#}$ by induction. The requirement that $h^{\#} \circ \iota = h$ completely determines the action of $h^{\#}$ on objects, and also entails that $h^{\#}(c) = h(c)$ on constants. On multimaps, the clone homomorphism axioms require that we set

$$h^{\#}(\mathsf{p}^{(i)}_{A_\bullet}) := \mathsf{p}^{(i)}_{h^{\#}A_\bullet}$$
$$h^{\#}(f[g_1, \dots, g_n]) := h^{\#}(f)\big[h^{\#}(g_1), \dots, h^{\#}(g_n)\big]$$

The definition on $\mathsf{proj}^{(i)}$ is determined by the hypothesis. Finally, on tup we set $h^{\#}\left(\mathsf{tup}_{A_\bullet}\right) := \rho_{h^{\#}(A_\bullet)}$, so that $h^{\#}$ sends $\mathsf{tup}_{A_\bullet}$ to the representable arrow on $A_1, \dots, A_n$ (which exists by Lemma 4.2.19). For uniqueness, it remains to show that the action of $h^{\#}$ on tup is determined by the hypotheses. For this, consider

$$\begin{aligned}
\rho_{(h^{\#}A_\bullet)} &= \rho_{(h^{\#}A_\bullet)}\Big[\mathsf{p}^{(1)}_{h^{\#}(A_\bullet)}, \ldots, \mathsf{p}^{(n)}_{h^{\#}(A_\bullet)}\Big] \\
&= \rho_{(h^{\#}A_\bullet)}\Big[h^{\#}(\mathsf{p}^{(1)}_{A_\bullet}), \ldots, h^{\#}(\mathsf{p}^{(n)}_{A_\bullet})\Big] \\
&= \rho_{(h^{\#}A_\bullet)}\Big[h^{\#}(\mathsf{proj}^{(1)}[\rho_{A_\bullet}]), \ldots, h^{\#}(\mathsf{proj}^{(n)}[\rho_{A_\bullet}])\Big] && \text{by Lemma 4.2.17} \\
&= \rho_{(h^{\#}A_\bullet)}\Big[h^{\#}(\mathsf{proj}^{(1)})\big[h^{\#}(\rho_{A_\bullet})\big], \ldots, h^{\#}(\mathsf{proj}^{(n)})\big[h^{\#}(\rho_{A_\bullet})\big]\Big] \\
&= \rho_{(h^{\#}A_\bullet)}\big[\pi_1\big[h^{\#}(\rho_{A_\bullet})\big], \ldots, \pi_n\big[h^{\#}(\rho_{A_\bullet})\big]\big] && \text{by cartesian} \\
&= \rho_{(h^{\#}A_\bullet)}[\pi_1, \ldots, \pi_n]\,\big[h^{\#}(\rho_{A_\bullet})\big] && \text{by Lemma 4.2.17} \\
&= \mathsf{p}^{(1)}_{(\prod_n A_\bullet)}\big[h^{\#}(\rho_{A_\bullet})\big] \\
&= h^{\#}(\rho_{A_\bullet})
\end{aligned}$$

Hence, the action of any clone homomorphism satisfying the two hypotheses is completely determined, and $h^{\#}$ is unique. □

The term calculus corresponding to the deductive system of Construction 4.2.25 is specified by the following rules:

1. For every sequence of types $A_1, \ldots, A_n$ $(n \in \mathbb{N})$, there exists a type $\prod_n(A_1, \ldots, A_n)$,
2. For every context $x_1 : A_1, \ldots, x_n : A_n$ there exists a multimap with components $A_1, \ldots, A_n \to \prod_n(A_1, \ldots, A_n)$; that is, a rule

$$\frac{}{x_1 : A_1, \ldots, x_n : A_n \vdash \langle x_1, \ldots, x_n\rangle : \prod_n(A_1, \ldots, A_n)} \tag{4.18}$$

3. An inverse to precomposing with $\langle x_1, \ldots, x_n\rangle$; following the proof of the forward direction of Lemma 4.2.19, we require multimaps

$$\frac{}{p : \prod_n(A_1, \ldots, A_n) \vdash \pi_i(p) : A_i}\ (1 \leqslant i \leqslant n)$$

such that the equations of Lemma 4.2.17 hold, *i.e.* that the equations

$$\pi_i(\langle x_1, \ldots, x_n\rangle) \equiv x_i\ \ (i = 1, \ldots, n) \qquad \text{and} \qquad p \equiv \langle \pi_1(p), \ldots, \pi_n(p)\rangle$$

obtained by substitution both hold for any $x_1 : A_1, \ldots, x_n : A_n$ and $p : \prod_n(A_1, \ldots, A_n)$.

Thus, we recover the laws for products in the simply-typed lambda calculus, restricted to variables, from purely clone-theoretic reasoning. The usual rules, defined on all terms, also arise from our abstract considerations. Inspecting the proof of Lemma 4.2.19, one sees that for every $(t_i : \Gamma \to X_i)_{i=1,\ldots,n}$ the corresponding multimap $\Gamma \to \prod_n(X_1, \ldots, X_n)$ is given by the composite $\rho_{X_\bullet}[t_1, \ldots, t_n]$. Translating this into the syntax and using the standard equality $\langle x_1, \ldots, x_n \rangle [t_i/x_i] = \langle t_1, \ldots, t_n \rangle$ defining the meta-operation of substitution, one arrives at the rule

$$\frac{(\Gamma \vdash t_i : A_i)_{i=1,\ldots,n}}{\Gamma \vdash \langle t_1, \ldots, t_n \rangle : \prod_n (A_1, \ldots, A_n)}$$

which, in the presence of substitution, is equivalent modulo admissibility to (4.18). This is subject to the two equations $\pi_i\left(\langle t_1, \ldots, t_n \rangle\right) \equiv t_i \ (i = 1, \ldots, n)$ and $t \equiv \langle \pi_1(t), \ldots, \pi_n(t) \rangle$.

We therefore recover a presentation of products—modulo $\beta\eta$—in the simply-typed lambda calculus. More precisely, it is straightforward to see that for any $\Lambda^{\times}$-signature $\mathcal{S}$ the clone $\mathbb{FCl}^{\times}(\mathcal{S})$ of Construction 4.2.25 is canonically isomorphic to the syntactic clone $\mathbb{C}_{\Lambda^{\times}(\mathcal{S})}$ of the simply-typed lambda calculus with products but not exponentials (recall Example 3.1.8 on page 39). Lemma 4.2.27 then implies that $\Lambda^{\times}(\mathcal{S})$ is the internal language of the free cartesian clone on $\mathcal{S}$.

We are ultimately interested in the internal language of the free cartesian *category* on a (unary) signature. For this we need to show that the cartesian category $\overline{\mathbb{C}_{\Lambda^{\times}(\mathcal{S})}}$, obtained by restricting $\mathbb{C}_{\Lambda^{\times}(\mathcal{S})}$ to unary morphisms, is the free cartesian category on $\mathcal{S}$. This is the content of the next lemma, in which we call a cartesian functor *strict* if it strictly preserves the product-forming operation and each projection. We write CartClone and CartCat for the categories of cartesian clones and cartesian categories with their strict morphisms.

As a technical convenience—in order to obtain a strict universal property—we shall assume that all the cartesian categories (resp. cartesian clones) under consideration have unary products given in the canonical way: for every object A the unary product $\prod_1(A)$ is exactly A (recall from Remark 4.1.3 that this is a standing assumption for fp-bicategories).

Lemma 4.2.28. The functor $\overline{(-)}$: CartClone → CartCat restricting a cartesian clone to its nucleus has a left adjoint.

Proof. We show that for any cartesian category $(\mathbb{C}, \Pi_n(-))$, cartesian clone $(T, \mathbb{D}, \Pi_n(-))$ and strict cartesian functor $F : \mathbb{C} \to \overline{\mathbb{D}}$ there exists a cartesian clone $\mathcal{P}\mathbb{C}$ and a strict cartesian clone homomorphism $F^{\#} : \mathcal{P}\mathbb{C} \to \mathbb{D}$, unique such that $\overline{F^{\#}} = F$.

Define $\mathcal{P}\mathbb{C}$ as follows. The sorts are the objects of $\mathbb{C}$ and for hom-sets we take

$$(\mathcal{P}\mathbb{C})(X_1, \ldots, X_n; Y) := \mathbb{C}(X_1 \times \cdots \times X_n; Y)$$

The substitution $t[u_1, \ldots, u_n]$ is defined to be the composite $t \circ \langle u_1, \ldots, u_n \rangle$ and the projections $\mathsf{p}^{(i)}_{X_\bullet}$ are the projections $\pi_i : \prod_n(X_1, \ldots, X_n) \to X_i$ for $i = 1, \ldots, n$. Since we assume the unary product structure on $\mathbb{C}$ is the identity, its cartesian structure immediately defines a cartesian structure on $\mathcal{P}\mathbb{C}$. Note in particular that $\mathcal{P}\mathbb{C}$ has the property that $(\mathcal{P}\mathbb{C})(X_1, \ldots, X_n; Y) = (\mathcal{P}\mathbb{C})(\prod_n(X_1, \ldots, X_n); Y)$.

Now, $\overline{\mathcal{P}\mathbb{C}}$ is the cartesian category with objects those of $\mathbb{C}$ and hom-sets of form $\mathbb{C}(\prod_1(X), Y)$. So $\overline{\mathcal{P}\mathbb{C}} = \mathbb{C}$. We therefore take the unit to be $\eta_{\mathbb{C}} := \mathrm{id}_{\mathbb{C}}$.

Next suppose that $F : \mathbb{C} \to \overline{\mathbb{D}}$ is a strict cartesian functor. The functor $F^{\#}$ is exactly F on objects, while for a multimap $t : X_1, \dots, X_n \to Y$ in $\mathcal{P}\mathbb{C}$ we define

$$F^{\#}(t) := \big(FX_1, \dots, FX_n \xrightarrow{\psi_{FX_\bullet}(\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)})} \textstyle\prod_{i=1}^{n} FX_i = F(\prod_{i=1}^{n} X_i) \xrightarrow{Ft} FY\big)$$

By the assumption that unary products are the identity, $F^{\#}(u) = F(u)$ for every unary morphism $u : X \to Y$. In particular, this holds for the projections π_i, so $F^{\#}$ is a strict cartesian clone homomorphism.

Finally, suppose that $G : \mathcal{P}\mathbb{C} \to \mathbb{D}$ is any strict cartesian clone homomorphism satisfying $\overline{G} = F$. Since $ob\mathcal{P}\mathbb{C} = ob\mathbb{C}$ we must have $FX = GX$ on objects. On arrows, note first that G preserves the tupling operation:

$$\begin{aligned}
&G(\psi_{X_\bullet}(\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)})) \\
&\qquad = \mathrm{Id}_{\prod_n GX_\bullet}[G(\psi_{X_\bullet}(\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)}))] \\
&\qquad = \psi_{GX_\bullet}(\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)})[\pi_1, \dots, \pi_n]\big[G(\psi_{X_\bullet}(\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)}))\big] && \text{by Lemma (4.2.16)} \\
&\qquad = \psi_{GX_\bullet}(\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)})[G\pi_1, \dots, G\pi_n]\big[G(\psi_{X_\bullet}(\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)}))\big] && \text{by strict preservation} \\
&\qquad = \psi_{GX_\bullet}(\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)})\big[G(\pi_\bullet[\psi_{X_\bullet}(\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)})])\big] \\
&\qquad = \psi_{GX_\bullet}(\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)})\big[G(\mathsf{p}^{(1)}), \dots, G(\mathsf{p}^{(n)})\big] && \text{by equation (4.13)} \\
&\qquad = \psi_{GX_\bullet}(\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)})
\end{aligned}$$

It follows that, for any $t : X_1, \dots, X_n \to Y$ in $\mathcal{P}\mathbb{C}$,

$$\begin{aligned}
F^{\#}(t) &= (Ft)[\psi_{FX_\bullet}(\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)})] \\
&= (\overline{G}t)[\psi_{GX_\bullet}(\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)})] \\
&= (Gt)[\psi_{GX_\bullet}(\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)})] \\
&= G(t[\psi_{X_\bullet}(\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)}_{)}]) \\
&= G(t \circ \langle \pi_1, \dots, \pi_n \rangle) \\
&= Gt
\end{aligned}$$

where the penultimate equality uses the fact that the cartesian structure of the clone $\mathcal{P}\mathbb{C}$ is inherited from that of the category $\mathbb{C}$. Hence $G = F^{\#}$, as required. □

With this lemma in hand, one obtains a diagram restricting (3.1) (p. 41) to the cartesian setting; the construction of the free cartesian category $\mathbb{FC}\mathrm{at}^{\times}(\mathcal{S})$ on a unary $\Lambda^{\times}_{\mathrm{ps}}$-signature $\mathcal{S}$ is standard (*c.f.* the construction of the free cartesian closed category in [Cro94, Chapter 4]):

$$\begin{tikzcd}
& \mathrm{CartClone} \arrow[dl, bend right, "\mathrm{forget}"'] \arrow[dr, bend left, "\overline{(-)}"] & \\
\Lambda^{\times}\text{-sig} \arrow[ur, bend right, "\mathbb{FCl}^{\times}(-)"'] \arrow[ur, phantom, "\top"] \arrow[dr, bend right, "\tilde{\mathcal{L}}"'] \arrow[dr, phantom, "\perp"] & & \mathrm{CartCat} \arrow[ul, bend left, "\mathcal{P}"'] \arrow[ul, phantom, "\top"] \arrow[dl, bend left, "\mathrm{forget}"] \arrow[dl, phantom, "\perp"] \\
& \Lambda^{\times}\text{-sig}\big|_1 \arrow[ul, hook, bend right] \arrow[ur, bend left, "\mathbb{FC}\mathrm{at}^{\times}(-)"] &
\end{tikzcd} \tag{4.19}$$

Moreover, the outer diagram commutes and, as we observed in the proof of the preceding lemma, $\overline{(-)} \circ \mathcal{P} = \mathrm{id}_{\mathrm{CartCat}}$. One thereby obtains the following chain of natural isomorphisms (*c.f.* equation (3.2)):

$$\mathrm{CartCat}(\mathbb{FC}\mathrm{at}^{\times}(\mathcal{S}), \mathbb{C}) = \mathrm{CartCat}\Big(\overline{\mathcal{P}(\mathbb{FC}\mathrm{at}^{\times}(\mathcal{S}))}, \mathbb{C}\Big) \cong \mathrm{CartCat}\Big(\overline{\mathbb{FCl}^{\times}(\iota\mathcal{S})}, \mathbb{C}\Big) \tag{4.20}$$

Hence, just as it was sufficient to construct an internal language for (bi)clones to describe (bi)categories, so it is sufficient to construct an internal language for cartesian clones—namely the simply-typed lambda calculus with just products—to describe cartesian categories.

Our aim in the next section is to reverse this process: we shall lift the theory just presented to the bicategorical setting, and use it to extract a principled construction of the type theory $\Lambda^{\times}_{\mathrm{ps}}$ with finite products.

4.2.3 Cartesian biclones and representability

Representable bi-multicategories. Our first step is to bicategorify the definition of multicategory. Multicategories can be defined in any monoidal category (*e.g.* [Yau16, Definition 11.2.1]); taking the definition in **Cat** with the product monoidal structure and weakening the equalities to isomorphisms suggests the following definition (*c.f.* also the definition of *cartesian 2-multicategory* [LSR17]).

Definition 4.2.29. A *bi-multicategory* $\mathcal{M}$ consists of the following data:

- A class $ob(\mathcal{M})$ of objects,
- For every $X_1, \ldots, X_n, Y \in ob(\mathcal{M})$ $(n \in \mathbb{N})$ a *hom-category* $(\mathcal{M}(X_1, \ldots, X_n; Y), \bullet, \mathrm{id})$ consisting of *multimaps* or *1-cells* $f : X_1, \ldots, X_n \to Y$ and *2-cells* $\tau : f \Rightarrow f'$, subject to a *vertical composition* operation,
- For every $X \in ob(\mathcal{M})$ an *identity* functor $\mathrm{Id}_X : \mathbb{1} \to \mathcal{M}(X; X)$,

- For every family of sequences $\Gamma_1, \dots, \Gamma_n$ and objects $Y_1, \dots, Y_n, Z$ $(n \in \mathbb{N})$ a *horizontal composition functor*:

  $$\circ_{\Gamma_\bullet; Y_\bullet; Z} : \mathcal{M}(Y_1, \dots, Y_n; Z) \times \textstyle\prod_{i=1}^{n} \mathcal{M}(\Gamma_i; Y_i) \to \mathcal{M}(\Gamma_1, \dots, \Gamma_n; Z)$$

  We denote the composition $\circ_{\Gamma_\bullet; Y_\bullet; Z}\big(f, (g_1, \dots, g_n)\big)$ by $f \circ \langle g_1, \dots, g_n \rangle$,

- Natural families of invertible 2-cells

  $$\mathsf{a}_{f; g_\bullet; h_\bullet} : (f \circ \langle g_\bullet \rangle) \circ \langle h_1^{(1)}, \dots, h_{m_1}^{(1)}, \dots, h_1^{(n)}, \dots, h_{m_n}^{(n)} \rangle \Rightarrow f \circ \langle g_1 \circ \langle h_\bullet^{(1)} \rangle, \dots, g_n \circ \langle h_\bullet^{(n)} \rangle \rangle$$
  $$\mathsf{r}_f : f \Rightarrow f \circ \langle \mathrm{Id}_{Y_1}, \dots, \mathrm{Id}_{Y_n} \rangle$$
  $$\mathsf{l}_f : \mathrm{Id}_Z \circ \langle f \rangle \Rightarrow f$$

  for all $f : Y_1, \dots, Y_n \to Z$, $(g_i : X_1^{(i)}, \dots, X_{m_n}^{(i)} \to Y_i)_{i=1,\dots,n}$ and $(h_j^{(i)} : \Delta_j^{(i)} \to X_j^{(i)})_{\substack{j=1,\dots,m_i \\ i=1,\dots,n}}$.

This data is subject to a triangle law and a pentagon law:

$$\begin{tikzcd}
f \circ \langle g_1, \dots, g_n \rangle \arrow[r, "\mathsf{r}_f \circ \langle g_1, \dots, g_n \rangle"] \arrow[d, equal] & (f \circ \langle \mathrm{Id}, \dots, \mathrm{Id} \rangle) \circ \langle g_1, \dots, g_n \rangle \arrow[d, "\mathsf{a}_{(f; \mathrm{Id}_{Y_\bullet}; g_\bullet)}"] \\
f \circ \langle g_1, \dots, g_n \rangle & f \circ \langle \mathrm{Id} \circ \langle g_1, \dots, g_n \rangle, \dots, \mathrm{Id} \circ \langle g_1, \dots, g_n \rangle \rangle \arrow[l, "f \circ \langle \mathsf{l}_{g_1}, \dots, \mathsf{l}_{g_n} \rangle"']
\end{tikzcd}$$

$$\begin{tikzcd}
\big((f \circ \langle g_\bullet \rangle) \circ \langle h_\bullet \rangle\big) \circ \langle i_\bullet \rangle \arrow[rr, "\mathsf{a}_{(f \circ \langle g_\bullet \rangle; h_\bullet; i_\bullet)}"] \arrow[d, "\mathsf{a}_{(f; g_\bullet; i_\bullet)} \circ \langle i_\bullet \rangle"'] & & (f \circ \langle g_\bullet \rangle) \circ \langle h_\bullet \circ \langle i_\bullet \rangle \rangle \arrow[d, "\mathsf{a}_{(f; g_\bullet; h_\bullet \circ \langle i_\bullet \rangle)}"] \\
\big(f \circ \langle g_\bullet \circ \langle h_\bullet \rangle \rangle\big) \circ \langle i_\bullet \rangle \arrow[r, "\mathsf{a}_{(f; g_\bullet \circ \langle h_\bullet \rangle; i_\bullet)}"'] & f \circ \langle (g_\bullet \circ \langle h_\bullet \rangle) \circ \langle i_\bullet \rangle \rangle \arrow[r, "f \circ \langle \mathsf{a}_{(g_1; h_\bullet; i_\bullet)}, \dots, \mathsf{a}_{(g_n; h_\bullet; i_\bullet)} \rangle"'] & f \circ \langle g_\bullet \circ \langle h_\bullet \circ \langle i_\bullet \rangle \rangle \rangle
\end{tikzcd}$$

A multimap (resp. 2-cell) of form $f : X \to Y$ (resp. $\tau : f \Rightarrow f' : X \to Y$) is called *linear*. ◂

Notation 4.2.30. Note that, just as for clones and multicategories, we use square brackets to denote biclone substitution and angle brackets to denote bi-multicategory composition (*c.f.* Notation 4.2.2). ◂

Remark 4.2.31. It is natural to conjecture that a construction similar to Construction 3.1.16 would enable one to construct the free bi-multicategory on a 2-multigraph and hence a *linear* version of $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$. Then the argument of Section 3.3 should readily extend to a coherence theorem for bi-multicategories. ◂

Examples of bi-multicategories arise naturally, mirroring the 1-categorical situation. Every bi-multicategory $\mathcal{M}$ gives rise to a bicategory $\overline{\mathcal{M}}$ by restricting to the linear multimaps and their 2-cells $\big($*c.f.* Example 3.1.12(3)$\big)$, and—by the following lemma—every monoidal bicategory gives rise to a bi-multicategory (*c.f.* [Her00, Definition 9.2]).

Lemma 4.2.32. Every monoidal bicategory $(\mathcal{B}, \otimes, I)$ induces a bi-multicategory.

Proof. By the coherence theorem for tricategories [GPS95], we may assume without loss of generality that the monoidal bicategory is a *Gray monoid*, *i.e.* a monoid in the monoidal category Gray (see *e.g.* [Gur13, Chapter 3] and [Hou07, Definition 3.8]). Since Gray monoids also satisfy a coherence theorem, we may assume that the underlying bicategory $\mathcal{B}$ is a 2-category, and that any pair of composites of the structural equivalences $a_{A,B,C} : (A \otimes B) \otimes C \to A \otimes (B \otimes C)$, $l_A : I \otimes A \to A$ and $r_A : A \otimes I \to A$ are related by a unique isomorphism (see [Gur06, Theorem 10.4] and [Hou07, Theorem 4.1]).

The bi-multicategory $\int \mathcal{B}$ has objects those of $\mathcal{B}$ and hom-categories $(\int \mathcal{B})(X_1, \ldots, X_n; Y) := \mathcal{B}(X_1 \otimes \cdots \otimes X_n, Y)$, where we specify the left-most bracketing $\big(((X_1 \otimes X_2) \otimes X_3) \otimes \cdots\big) \otimes X_n$.

For sequences of objects $\Gamma_i := (A_j^{(i)})_{j=1,\ldots,m_i}$ $(i = 1, \ldots, n)$ and multimaps $(g_i : \Gamma_i \to X_i)_{i=1,\ldots,n}$ and $f : X_1 \otimes \cdots \otimes X_n \to Y$, the composite $f \circ \langle g_1, \ldots, g_n \rangle$ is defined to be

$$A_1^{(1)} \otimes \cdots \otimes A_1^{(i)} \otimes \cdots \otimes A_{m_i}^{(i)} \otimes \cdots \otimes A_1^{(n)} \otimes \cdots \otimes A_{m_n}^{(n)} \xrightarrow{\simeq} \bigotimes_{i=1}^{n} \Gamma_i \xrightarrow{\otimes_{i=1}^{n} g_i} X_1 \otimes \cdots \otimes X_n \xrightarrow{f} Y$$

where the equivalence is the canonical such. By the coherence theorem for Gray monoids, there is a unique choice of isomorphism for each of the structural 2-cells, and these must satisfy the triangle and pentagon laws. □

For morphisms of bi-multicategories we borrow the terminology from **Bicat**. Thus, bi-multicategories are related by *pseudofunctors*, *transformations* and *modifications*.

Definition 4.2.33.

1. A *pseudofunctor* $F : \mathcal{M} \to \mathcal{M}'$ of bi-multicategories consists of:

   a) A map $F : ob(\mathcal{M}) \to ob(\mathcal{M}')$ on objects,

   b) A functor $F_{X_\bullet;Y} : \mathcal{M}(X_1, \ldots, X_n; Y) \to \mathcal{M}'(FX_1, \ldots, FX_n; FY)$ for every sequence of objects $X_1, \ldots, X_n, Y \in ob(\mathcal{M})$ $(n \in \mathbb{N})$,

   c) An invertible 2-cell $\psi_X : \mathrm{Id}_{FX} \Rightarrow F\mathrm{Id}_X$ for every $X \in ob(\mathcal{M})$,

   d) An invertible 2-cell $\phi_{f;g_\bullet} : F(f) \circ \langle Fg_1, \ldots, Fg_n \rangle \Rightarrow F\left(f \circ \langle g_1, \ldots, g_n \rangle\right)$ for every $f : X_1, \ldots, X_n \to Y$ $(n \in \mathbb{N})$ and $(g_i : \Gamma_i \to X_i)_{i=1,\ldots,n}$ in $\mathcal{M}$, natural in the sense of Definition 4.2.3(2).

   This data is subject to the following three coherence laws:

$$\begin{tikzcd}
\mathrm{Id}_{FZ} \circ \langle Ff \rangle \arrow[r, "\mathsf{l}_{Ff}"] \arrow[d, "\psi_Z \circ \langle Ff \rangle"'] & Ff \\
F(\mathrm{Id}_Z) \circ \langle Ff \rangle \arrow[r, "\phi_{(\mathrm{Id}_Z;f)}"'] & F\big(\mathrm{Id}_Z \circ \langle f \rangle\big) \arrow[u, "F\mathsf{l}_f"']
\end{tikzcd}
\qquad
\begin{tikzcd}
Ff \arrow[r, "F\mathsf{r}_f"] \arrow[d, "\mathsf{r}_{Ff}"'] & F\left(f \circ \langle \mathrm{Id}_{Y_1}, \ldots, \mathrm{Id}_{Y_n} \rangle\right) \\
F(f) \circ \langle \mathrm{Id}_{FY_1}, \ldots, \mathrm{Id}_{FY_n} \rangle \arrow[r, "F(f) \circ \langle \psi_{Y_1}, \ldots, \psi_{Y_n} \rangle"'] & F(f) \circ \langle F\mathrm{Id}_{Y_1}, \ldots, F\mathrm{Id}_{Y_n} \rangle \arrow[u, "\phi_{(f;\mathrm{Id}_{FY_\bullet})}"']
\end{tikzcd}$$

$$\begin{tikzcd}
(Ff \circ \langle Fg_\bullet \rangle) \circ \langle Fh_\bullet \rangle \arrow[r, "\mathsf{a}_{(Ff;Fg_\bullet;Fh_\bullet)}"] \arrow[d, "\phi_{(f;g_\bullet)} \circ \langle Fh_\bullet \rangle"'] & F(f) \circ \left\langle Fg_1 \circ \langle Fh_\bullet^{(1)} \rangle, \ldots, Fg_n \circ \langle Fh_\bullet^{(n)} \rangle \right\rangle \arrow[d, "F(f) \circ \langle \phi_{(g_1;h_\bullet)}, \ldots, \phi_{(g_n;h_\bullet)} \rangle"] \\
F\left(f \circ \langle g_\bullet \rangle\right) \circ \langle Fh_\bullet \rangle \arrow[d, "\phi_{(f \circ \langle g_\bullet \rangle; h_\bullet)}"'] & Ff \circ \left\langle F\big(g_1 \circ \langle h_\bullet^{(1)} \rangle\big), \ldots, F\big(g_n \circ \langle h_\bullet^{(n)} \rangle\big) \right\rangle \arrow[d, "\phi_{(f; g_\bullet \circ \langle h_\bullet^{(\bullet)} \rangle)}"] \\
F\left((f \circ \langle g_\bullet \rangle) \circ \langle h_\bullet \rangle\right) \arrow[r, "F\mathsf{a}_{(f;g_\bullet;h_\bullet)}"'] & F\left(f \circ \left\langle g_1 \circ \langle h_\bullet^{(1)} \rangle, \ldots, g_n \circ \langle h_\bullet^{(n)} \rangle \right\rangle\right)
\end{tikzcd}$$

2. A *transformation* $(\alpha, \overline{\alpha}) : F \Rightarrow F'$ between pseudofunctors $F, F' : \mathcal{M} \to \mathcal{M}$ of bi-multicategories consists of

   a) A linear multimap $\alpha_X : FX \to F'X$ for every $X \in \mathcal{M}$,

   b) A 2-cell $\overline{\alpha}_f : \alpha_Z \circ \langle Ff \rangle \Rightarrow Gf \circ \langle \alpha_{Y_1}, \ldots, \alpha_{Y_n} \rangle$ for every $f : Y_1, \ldots, Y_n \to Z$ in $\mathcal{M}$, natural in f in the sense of Definition 4.2.3(2).

   This data is subject to the following associativity and unit laws for every $f : Y_1, \ldots, Y_n \to Z$ and $(g_i : \Gamma_i \to Y_i)_{i=1,\ldots,n}$ in $\mathcal{M}$:

$$\begin{tikzcd}
\mathrm{Id}_{GY} \circ \langle \alpha_Y \rangle \arrow[rr, "\psi_Y \circ \langle \alpha_Y \rangle"] \arrow[d, "\mathsf{l}_{\alpha_Y}"'] & & G\mathrm{Id}_Y \circ \langle \alpha_Y \rangle \\
\alpha_Y \arrow[r, "\mathsf{r}_{\alpha_Y}"'] & \alpha_Y \circ \langle \mathrm{Id}_{FY} \rangle \arrow[r, "\alpha_Y \circ \langle \psi_Y \rangle"'] & \alpha_Y \circ \langle F\mathrm{Id}_Y \rangle \arrow[u, "\overline{\alpha}_{\mathrm{Id}_Y}"']
\end{tikzcd}$$

$$\begin{tikzcd}
(\alpha_Y \circ \langle Ff \rangle) \circ \langle Fg_\bullet \rangle \arrow[r, "\mathsf{a}_{(\alpha_Y;Ff;Fg_\bullet)}"] \arrow[d, "\overline{\alpha}_f \circ \langle Fg_\bullet \rangle"'] & \alpha_Y \circ \langle (F(f) \circ \langle Fg_\bullet \rangle) \rangle \arrow[r, "\alpha_Y \circ \langle \phi_{(f;g_\bullet)} \rangle"] & \alpha_Y \circ \langle F\left(f \circ \langle g_\bullet \rangle\right) \rangle \arrow[ddddd, "\overline{\alpha}_{f \circ \langle g_\bullet \rangle}"] \\
(G(f) \circ \langle \alpha_{Y_1}, \ldots, \alpha_{Y_n} \rangle) \circ \langle Fg_\bullet \rangle \arrow[d, "\mathsf{a}_{(Gf;\alpha_{Y_\bullet};Fg_\bullet)}"'] & & \\
G(f) \circ \langle \alpha_{Y_1} \circ \langle Fg_1 \rangle, \ldots, \alpha_{Y_n} \circ \langle Fg_n \rangle \rangle \arrow[d, "G(f) \circ \langle \overline{\alpha}_{g_1}, \ldots, \overline{\alpha}_{g_n} \rangle"'] & & \\
G(f) \circ \langle Gg_1 \circ \langle \alpha_{\Gamma_1} \rangle, \ldots, Gg_n \circ \langle \alpha_{\Gamma_n} \rangle \rangle \arrow[d, "\mathsf{a}^{-1}_{(Gf;Gg_\bullet;\alpha_\bullet)}"'] & & \\
& & \\
(G(f) \circ \langle Gg_1, \ldots, Gg_n \rangle) \circ \langle \alpha_\bullet \rangle \arrow[rr, "\phi_{(f;g_\bullet)} \circ \langle \alpha_\bullet \rangle"'] & & G\big(f \circ \langle g_\bullet \rangle\big) \circ \langle \alpha_\bullet \rangle
\end{tikzcd}$$

   Note that, where $\Gamma_i := A_1^{(i)}, \ldots, A_{m_i}^{(i)}$, we write α_{Γ_i} for the sequence $\alpha_{A_1^{(i)}}, \ldots, \alpha_{A_{m_i}^{(i)}}$.

3. A *modification* $\Xi : (\alpha, \overline{\alpha}) \to (\beta, \overline{\beta})$ between transformations $(\alpha, \overline{\alpha}), (\beta, \overline{\beta}) : F \Rightarrow F'$ is a family of 2-cells $\Xi_X : \alpha_X \Rightarrow \beta_X$ such that the following diagram commutes for every $f : Y_1, \ldots, Y_n \to Z$:

$$\begin{tikzcd}
\alpha_Z \circ \langle Ff \rangle \arrow[r, "\Xi_Z \circ \langle Ff \rangle"] \arrow[d, "\overline{\alpha}_f"'] & \beta_Z \circ \langle Ff \rangle \arrow[d, "\overline{\beta}_f"] \\
G(f) \circ \langle \alpha_{Y_1}, \dots, \alpha_{Y_n} \rangle \arrow[r, "G(f) \circ \langle \Xi_{Y_1}, \dots, \Xi_{Y_n} \rangle"'] & G(f) \circ \langle \beta_{Y_1}, \dots, \beta_{Y_n} \rangle
\end{tikzcd}$$

◂

One would expect that bi-multicategories, pseudofunctors, transformations and modifications organise themselves into a tricategory; we do not pursue such considerations here. Instead, we lift Hermida's notion of representability to bi-multicategories. As usual, it is convenient to require as much as possible of the definition to be data.

Definition 4.2.34. A *representable bi-multicategory* $(\mathcal{M}, \mathrm{T}_n)$ consists of the following data:

1. For every $X_1, \dots, X_n \in \mathcal{M}$ $(n \in \mathbb{N})$, a chosen object $\mathrm{T}_n(X_1, \dots, X_n) \in \mathcal{M}$ and chosen *birepresentable* multimap $\rho_{X_\bullet} : X_1, \dots, X_n \to \mathrm{T}_n(X_1, \dots, X_n)$, such that the birepresentable multimaps are closed under composition,
2. For every $A, X_1, \dots, X_n \in \mathcal{M}$ $(n \in \mathbb{N})$, an adjoint equivalence

$$\begin{tikzcd}
\mathcal{M}(\mathrm{T}_n(X_1, \dots, X_n); A) \arrow[r, bend left, "(-)\circ\langle \rho_{X_\bullet} \rangle"] \arrow[r, phantom, "\perp\simeq"] & \mathcal{M}(X_1, \dots, X_n; A) \arrow[l, bend left, "\psi_{X_\bullet}"]
\end{tikzcd}$$

specified by a choice of universal arrow $\varepsilon_{X_\bullet}$. ◂

The birepresentability of $\rho_{X_\bullet}$ entails the following. For every $f : X_1, \dots, X_n \to A$ we require a choice of multimap $\psi_{X_\bullet}(f) : \mathrm{T}_n(X_1, \dots, X_n) \to A$ and 2-cell $\varepsilon_{X_\bullet;f} : \psi_{X_\bullet}(f) \circ \langle \rho_{X_\bullet} \rangle \Rightarrow f$. This 2-cell is universal in the sense that for any $g : \mathrm{T}_n(X_1, \dots, X_n) \to A$ and $\sigma : g \circ \langle \rho_{X_\bullet} \rangle \Rightarrow f$ there exists a unique 2-cell $\sigma^\dagger : g \Rightarrow \psi_{X_\bullet}(f)$ such that

$$\begin{tikzcd}
g \circ \langle \rho_{X_\bullet} \rangle \arrow[rr, "\sigma^\dagger \circ \langle \rho_{X_\bullet} \rangle"] \arrow[dr, "\sigma"'] & & \psi_{X_\bullet}(f) \circ \langle \rho_{X_\bullet} \rangle \arrow[dl, "\varepsilon_{X_\bullet;f}"] \\
& f &
\end{tikzcd} \tag{4.21}$$

Remark 4.2.35. Hermida's construction suggests that every representable bi-multicategory ought to induce a monoidal bicategory, and indeed that there exists a triequivalence between representable bi-multicategories and monoidal bicategories. Here we shall restrict ourselves to proving that every representable biclone induces an fp-bicategory: a considerably easier task, as one only needs to check a universal property, rather than many coherence axioms. ◂

Following the 1-categorical template of Section 4.2.1, we next examine the construction of finite products in a bi-multicategory. To avoid the double prefix in 'fp-bi-multicategories' we refer to such objects as 'cartesian bi-multicategories'.

Cartesian bi-multicategories. Once again, we translate between the categorical and bicategorical settings by replacing universal arrows with biuniversal arrows.

Definition 4.2.36. Let $F : \mathcal{M} \to \mathcal{M}'$ be a pseudofunctor of bi-multicategories and $X \in \mathcal{M}'$. A *biuniversal arrow* (R, u) *from* F *to* X consists of

1. An object $R \in \mathcal{M}$,
2. A linear multimap $u : FR \to X$,
3. For every $A \in \mathcal{M}$, a chosen adjoint equivalence

$$\begin{tikzcd}
\mathcal{M}(A_1, \dots, A_n; R) \arrow[r, bend left, "u\circ\langle F(-)\rangle"] \arrow[r, phantom, "\perp\simeq"] & \mathcal{M}'(FA_1, \dots, FA_n; X) \arrow[l, bend left, "\psi_{A_\bullet}"]
\end{tikzcd}$$

specified by a choice of universal arrow $\varepsilon_h : u \circ \langle F\psi_{A_\bullet}(h)\rangle \Rightarrow h : FA_1, \dots, FA_n \to X$ (*c.f.* Definition 2.2.2). ◂

We translate this into a 'global' definition in the by-now-familiar way.

Lemma 4.2.37. For any pseudofunctor of bi-multicategories $F : \mathcal{M} \to \mathcal{M}'$ and $X \in \mathcal{M}'$, the following are equivalent:

1. A choice of biuniversal arrow from F to X,
2. Chosen adjoint equivalences $\kappa_{A_\bullet} : \mathcal{M}(A_1, \dots, A_n; R) \leftrightarrows \mathcal{M}'(FA_1, \dots, FA_n; X) : \delta_{A_\bullet}$ for $A_1, \dots, A_n \in \mathcal{M}$ $(n \in \mathbb{N})$, specified by a choice of universal arrow and pseudonatural in the sense that for every $f : A_1, \dots, A_n \to R$ and $(g_i : \Gamma_i \to A_i)_{i=1,\dots,n}$ there exists an invertible 2-cell $\nu_{f;g_\bullet} : \kappa_{A_\bullet}(f) \circ \langle Fg_1, \dots, Fg_n\rangle \Rightarrow \kappa_{A_\bullet}(f \circ \langle g_1, \dots, g_n\rangle)$, multinatural in $f, g_1, \dots, g_n$ and satisfying

$$\begin{tikzcd}
\kappa_{A_\bullet}(f) \arrow[r, "\kappa_{A_\bullet}(\mathrm{r}_f)"] \arrow[d, "\mathrm{r}_{\kappa_{A_\bullet}(f)}"'] & \kappa_{A_\bullet}(f \circ \langle \mathrm{Id}_{A_\bullet}\rangle) \\
\kappa_{A_\bullet}(f) \circ \langle \mathrm{Id}_{A_\bullet}\rangle \arrow[r, "\kappa_{A_\bullet}(f)\circ\langle\psi_\bullet\rangle"'] & \kappa_{A_\bullet}(f) \circ \langle F\mathrm{Id}_{A_\bullet}\rangle \arrow[u, "(\nu_{f;\mathrm{Id}_{A_\bullet}})"']
\end{tikzcd} \tag{4.22}$$

$$\begin{tikzcd}
(\kappa_{A_\bullet}(f) \circ \langle Fg_\bullet \rangle) \circ \langle Fh_\bullet \rangle \arrow[r, "\mathsf{a}_{(\kappa_{A_\bullet}(f); Fg_\bullet; Fh_\bullet)}"] \arrow[d, "\nu_{(g;f_\bullet)} \circ \langle Fh_\bullet \rangle"'] & \kappa_{A_\bullet}(f) \circ \left\langle Fg_1 \circ \langle Fh_\bullet^{(1)} \rangle, \dots, Fg_n \circ \langle Fh_\bullet^{(n)} \rangle \right\rangle \arrow[d, "\kappa_{A_\bullet}(f) \circ \langle \phi_{(g_1; h_\bullet)}, \dots, \phi_{(g_n; h_\bullet)} \rangle"] \\
\kappa_{\Gamma_\bullet}(f \circ \langle g_\bullet \rangle) \circ \langle Fh_\bullet \rangle \arrow[d, "\nu_{(f \circ \langle g_\bullet \rangle; h)}"'] & \kappa_{A_\bullet}(f) \circ \left\langle F\big(g_1 \circ \langle h_\bullet^{(1)} \rangle\big), \dots, F\big(g_n \circ \langle h_\bullet^{(n)} \rangle\big) \right\rangle \arrow[d, "\nu_{(f; g_\bullet \circ \langle h_\bullet \rangle)}"] \\
\kappa_{\Delta_\bullet}((f \circ \langle g_\bullet \rangle) \circ \langle h_\bullet \rangle) \arrow[r, "\kappa_{\Delta_\bullet}(\mathsf{a}_{(f; g_\bullet; h_\bullet)})"'] & \kappa_{\Delta_\bullet}\Big(f \circ \left\langle g_1 \circ \langle h_\bullet^{(1)} \rangle, \dots, g_n \circ \langle h_\bullet^{(n)} \rangle \right\rangle\Big)
\end{tikzcd} \tag{4.23}$$

for $\Gamma_i := X_1^{(i)}, \dots, X_{m_i}^{(i)}$ and $(h_j^{(i)} : \Delta_j^{(i)} \to X_j^{(i)})_{\substack{j=1,\dots,m_i \\ i=1,\dots,n}}$.

Proof. $\boxed{(1)\Rightarrow(2)}$ By biuniversality, $u \circ \langle F(-) \rangle$ is part of an adjoint equivalence for every $A_1, \dots, A_n \in \mathcal{M}$ $(n \in \mathbb{N})$, so it remains to check pseudonaturality. Taking $\kappa_{A_\bullet}$ to be $u \circ \langle F(-) \rangle$, we are required to provide 2-cells $\nu_{f;g_\bullet}$ of type $(u \circ \langle Ff \rangle) \circ \langle Fg_1, \dots, Fg_n \rangle \Rightarrow u \circ \langle F\big(f \circ \langle g_1, \dots, g_n \rangle\big) \rangle$, for which we take $\big(u \circ \langle \phi_{f;g_\bullet} \rangle\big) \bullet \mathsf{a}_{u;Ff;Fg_\bullet}$. The naturality condition and two axioms (4.22) and (4.23) then follow directly from the coherence laws of a pseudofunctor.

$\boxed{(2)\Rightarrow(1)}$ This direction is a little more delicate, but we can follow the template provided by Lemma 4.2.9. Let us first make explicit the content of the adjoint equivalence

$$\kappa_{A_\bullet} : \mathcal{M}(A_1, \dots, A_n; R) \leftrightarrows \mathcal{M}'(FA_1, \dots, FA_n; X) : \delta_{A_\bullet}$$

Choosing a universal arrow entails that for every $f : FA_1, \dots, FA_n \to X$ there exists a multimap $\delta_{A_\bullet}(f) : A_1, \dots, A_n \to R$ and a 2-cell $\overline{\delta}_f : \kappa_{A_\bullet}\delta_{A_\bullet}(f) \Rightarrow f$, universal in the sense that for any $g : A_1, \dots, A_n \to R$ and $\sigma : \kappa_{A_\bullet}(g) \Rightarrow f$ there exists a unique 2-cell $\sigma^\sharp : g \Rightarrow \delta_{A_\bullet}(f)$ such that

$$\begin{tikzcd}
\kappa_{A_\bullet}(g) \arrow[rr, "\kappa_{A_\bullet}(\sigma^\sharp)"] \arrow[dr, "\sigma"'] & & \kappa_{A_\bullet}\delta_{A_\bullet}(f) \arrow[dl, "\overline{\delta}_f"] \\
& f &
\end{tikzcd} \tag{4.24}$$

We claim that $u := \kappa_R(\mathrm{Id}_R) : FR \to X$ is biuniversal. Thus, for every $f : FA_1, \dots, FA_n \to X$ we need to provide an arrow $\overline{f} : A_1, \dots A_n \to R$ and a universal 2-cell $\varepsilon_{A_\bullet;f} : u \circ \langle F\overline{f} \rangle \Rightarrow f$.

For the arrow we take $\overline{f} := \delta_{A_\bullet}(f)$. For the 2-cell we make use of the naturality condition to define $\varepsilon_{A_\bullet;f}$ as the invertible composite

$$\begin{tikzcd}
u \circ \langle F\delta_{A_\bullet}(f) \rangle \arrow[rr, "\varepsilon_{A_\bullet;f}"] \arrow[d, equal] & & f \\
\kappa_R(\mathrm{Id}_R) \circ \langle F\delta_{A_\bullet}(f) \rangle \arrow[r, "\nu_{(\mathrm{Id}_R; \delta_{A_\bullet}(f))}"'] & \kappa_{A_\bullet}\left(\mathrm{Id}_R \circ \langle \delta_{A_\bullet}(f) \rangle\right) \arrow[r, "\kappa_{A_\bullet}(\mathsf{l}_{\delta_{A_\bullet}(f)})"'] & \kappa_{A_\bullet}\delta_{A_\bullet}(f) \arrow[u, "\overline{\delta}_f"']
\end{tikzcd}$$

To establish universality, let $g : A_1, \dots, A_n \to R$ be a multimap and $\gamma : u \circ \langle Fg \rangle \Rightarrow f$ be any 2-cell. We need to show there exists a unique 2-cell $\gamma^\dagger : g \Rightarrow \overline{f}$ such that

$$\begin{tikzcd}
u \circ \langle Fg \rangle \arrow[rr, "u \circ \langle F\gamma^\dagger \rangle"] \arrow[dr, "\gamma"'] & & u \circ \langle F\overline{f} \rangle \arrow[dl, "\varepsilon_{A_\bullet ; f}"] \\
& f &
\end{tikzcd} \tag{4.25}$$

By the universal property (4.24), to define $\gamma^\dagger : g \Rightarrow \overline{f} = \delta_{A_\bullet}(f)$ it suffices to define a 2-cell $\kappa_{A_\bullet}(g) \Rightarrow f$, for which we take

$$\alpha_{\gamma,f,g} := \kappa_{A_\bullet}(g) \xRightarrow{\kappa_{A_\bullet}(\mathfrak{l}_g^{-1})} \kappa_{A_\bullet}(\mathrm{Id}_R \circ \langle g \rangle) \xRightarrow{\nu^{-1}_{\mathrm{Id}_R;g}} \kappa_{A_\bullet}(\mathrm{Id}_R) \circ \langle Fg \rangle \overset{\gamma}{\Rightarrow} f$$

We define $\gamma^\dagger := (\alpha_{\gamma,f,g})^\sharp$. That this fills (4.25) is an easy check using the definition and naturality of ν. For uniqueness, suppose $\sigma : g \Rightarrow \overline{f} = \delta_{A_\bullet}(f)$ also fills (4.25). By the universal property defining $\gamma^\dagger$ it suffices to show that σ is the unique 2-cell corresponding to $\alpha_{\gamma,f,g}$ via (4.24). This follows from the naturality of ν and $\mathfrak{l}$ and the definition of $\alpha_{\gamma,f,g}$.

This completes the construction of an *adjunction* $\mathcal{M}(A_1, \dots, A_n; R) \leftrightarrows \mathcal{M}'(FA_1, \dots, FA_n; X)$; to show this is an adjoint equivalence, we need to show the unit is also invertible. But the unit is given by applying the $(-)^\dagger$ operation to the identity, *i.e.* by applying the $(-)^\sharp$ operation to an invertible 2-cell. This is invertible by Lemma 2.2.8. □

The definition of product of multicategories lifts straightforwardly to bi-multicategories. For bi-multicategories $\mathcal{M}$ and $\mathcal{M}'$, the bi-multicategory $\mathcal{M} \times \mathcal{M}'$ has objects pairs $(X, X') \in ob(\mathcal{M}) \times ob(\mathcal{M}')$ and composition as in (4.10) on page 91. The structural isomorphisms are given pointwise. Then there exists a canonical *diagonal pseudofunctor* $\Delta^n : \mathcal{M} \to \mathcal{M}^{\times n}$ for every bi-multicategory $\mathcal{M}$ and $n \in \mathbb{N}$. This suggests the following definition.

Definition 4.2.38. A *cartesian bi-multicategory* $(\mathcal{M}, \Pi_n(-))$ consists of a bi-multicategory $\mathcal{M}$ equipped with the following data for every $X_1, \dots, X_n \in \mathcal{M}$ $(n \in \mathbb{N})$:

1. A chosen object $\prod_n(X_1, \dots, X_n)$,
2. A choice of biuniversal arrow $\pi = (\pi_1, \dots, \pi_n) : \Delta^n(\prod_n(X_1, \dots, X_n)) \to (X_1, \dots, X_n)$ from Δ^n to $(X_1, \dots, X_n) \in \mathcal{M}^{\times n}$. ◂

By the preceding lemma, a bi-multicategory is cartesian if and only if there exists a pseudonatural family of adjoint equivalences

$$\mathcal{M}\big(\Gamma; \textstyle\prod_n(X_1, \dots, X_n)\big) \simeq \mathcal{M}^{\times n}(\Delta^n(\Gamma); (X_1, \dots, X_n)) = \textstyle\prod_{i=1}^n \mathcal{M}(\Gamma; X_i)$$

The universal property therefore manifests itself as follows. For every sequence of multimaps $(t_i : \Gamma \to X_i)_{i=1,\dots,n}$ there exists a multimap $\mathsf{tup}(t_1, \dots, t_n) : \Gamma \to \prod_n(X_1, \dots, X_n)$ and a 2-cell ϖ

with components $\varpi^{(i)}_{t_\bullet} : \pi_i \circ \langle \mathsf{tup}(t_1, \ldots, t_n) \rangle \Rightarrow t_i$ for $i = 1, \ldots, n$. This 2-cell is universal in the sense that, if $u : \Gamma \to \prod_n(X_1, \ldots, X_n)$ and $\alpha_i : \pi_i \circ \langle u \rangle \Rightarrow t_i$ for $i = 1, \ldots, n$, then there exists a unique 2-cell $\mathsf{p}^\dagger(\alpha_1, \ldots, \alpha_n) : u \Rightarrow \mathsf{tup}(t_1, \ldots, t_n)$ filling the following diagram for $i = 1, \ldots, n$:

$$\begin{tikzcd}
\pi_i \circ \langle u \rangle \arrow[rr, "\pi_i \circ \langle \alpha \rangle"] \arrow[dr, "\alpha_i"'] & & \pi_i \circ \langle \mathsf{tup}(t_1, \ldots, t_n) \rangle \arrow[dl, "\varpi^{(i)}_{t_\bullet}"] \\
& t_i &
\end{tikzcd} \tag{4.26}$$

Finally, the unit $\eta_u := \mathsf{p}^\dagger(\mathrm{id}_{\pi_1 \circ \langle u \rangle}, \ldots, \mathrm{id}_{\pi_n \circ \langle u \rangle}) : u \Rightarrow \mathsf{tup}(\pi_1 \circ \langle u \rangle, \ldots, \pi_n \circ \langle u \rangle)$ is required to be invertible for every $u : \Gamma \to \prod_n(X_1, \ldots X_n)$.

Our next task is to extend the theory of representable and cartesian bi-multicategories to biclones.

Cartesian biclones. As we did for clones, we define products in a biclone by first defining a bi-multicategory structure on each biclone (*c.f.* Construction 4.2.11).

Construction 4.2.39. Every biclone $(S, \mathcal{C})$ canonically defines a bi-multicategory $\mathrm{M}\mathcal{C}$ as follows:

- $ob(\mathrm{M}\mathcal{C}) := S$,
- $(\mathrm{M}\mathcal{C})(X_1, \ldots, X_n; Y) := \mathcal{C}(X_1, \ldots, X_n; Y)$,
- $\mathrm{Id}_X := \mathsf{p}^{(1)}_1 : \mathbb{1} \to (\mathrm{M}\mathcal{C})(X; X)$,
- The composition functor $(\mathrm{M}\mathcal{C})(Y_1, \ldots, Y_n; Z) \times \prod_{i=1}^n (\mathrm{M}\mathcal{C})(\Gamma_i; Y_i) \to (\mathrm{M}\mathcal{C})(\Gamma_1, \ldots, \Gamma_n; Z)$ is defined by
$$f \circ \langle g_1, \ldots, g_n \rangle := f[g_1 \boxtimes \cdots \boxtimes g_n]$$
using the notation of Notation 3.1.19,
- The unitor structural isomorphisms are defined as follows, for $f : X_1, \ldots, X_n \to Y$:
$$\mathsf{r}_f := f \overset{\iota}{\Rightarrow} f\Big[\mathsf{p}^{(1)}_{X_\bullet}, \ldots, \mathsf{p}^{(n)}_{X_\bullet}\Big] \xRightarrow{f\left[\varrho^{(-1)}, \ldots, \varrho^{(-1)}\right]} f\Big[\mathsf{p}^{(1)}_{X_1}\Big[\mathsf{p}^{(1)}_{X_\bullet}\Big], \ldots, \mathsf{p}^{(1)}_{X_n}\Big[\mathsf{p}^{(n)}_{X_\bullet}\Big]\Big]$$
$$\mathsf{l}_f := \mathsf{p}^{(1)}_Y\Big[f\Big[\mathsf{p}^{(1)}_{X_\bullet}, \ldots, \mathsf{p}^{(n)}_{X_\bullet}\Big]\Big] \overset{\varrho^{(1)}}{\Longrightarrow} f\Big[\mathsf{p}^{(1)}_{X_\bullet}, \ldots, \mathsf{p}^{(n)}_{X_\bullet}\Big] \overset{\iota^{-1}}{\Longrightarrow} f$$
The associativity structural isomorphism is a little complex. Suppose given sequences of objects $\Gamma_i := B^{(i)}_1, \ldots, B^{(i)}_{m_i} (i = 1, \ldots, n)$ and multimaps $(g_i : \Gamma_i \to Y_i)_{i=1,\ldots,n}$ and $f : Y_1, \ldots, Y_n \to Z$. Moreover suppose that $\Delta^{(i)}_j := A^{(i,j)}_1, \ldots, A^{(i,j)}_{k(i,j)}$, and that $h^{(i)}_j : \Delta^{(i)}_j \to B^{(i)}_j$ for $j = 1, \ldots, m_i$ and $i = 1, \ldots, n$.

  Now, writing $\overline{\mathsf{p}}(R)$ for the projection picking out the element R in the codomain, there exists a map
$$h^{(i)}_j\Big[\overline{\mathsf{p}}(A^{(i,j)}_1), \ldots, \overline{\mathsf{p}}(A^{(i,j)}_{k(i,j)})\Big] : \Delta^{(1)}_1, \ldots, \Delta^{(1)}_{m_1}, \ldots, \Delta^{(n)}_1, \ldots, \Delta^{(n)}_{m_n} \to B^{(i)}_j \tag{4.27}$$

for every $i = 1, \dots, n$ and $j = 1, \dots, m_i$. On the other hand, one may first project out from the full sequence $\Delta^{(1)}_1, \dots, \Delta^{(1)}_{m_1}, \dots, \Delta^{(n)}_1, \dots, \Delta^{(n)}_{m_n}$ to the subsequence $\Delta^{(i)}_1, \dots, \Delta^{(i)}_{m_i}$ and then project again before applying $h^{(i)}_j$. Abusively writing $\left[\overline{\mathsf{p}}(\Delta^{(i)}_1), \dots, \overline{\mathsf{p}}(\Delta^{(i)}_{m_i})\right]$ for the sequence $\left[\overline{\mathsf{p}}(A^{(i,1)}_1), \dots, \overline{\mathsf{p}}(A^{(i,m_i)}_{k(i,m_i)})\right]$, one thereby obtains

$$h^{(i)}_j\left[\overline{\mathsf{p}}(A^{(i,j)}_1), \dots, \overline{\mathsf{p}}(A^{(i,j)}_{k(i,j)})\right]\left[\overline{\mathsf{p}}(\Delta^{(i)}_1), \dots, \overline{\mathsf{p}}(\Delta^{(i)}_{m_i})\right] \tag{4.28}$$

The pair of parallel multimaps (4.27) and (4.28) are related by a canonical composite of structural isomorphisms:

$$\begin{aligned} h^{(i)}_j\left[\overline{\mathsf{p}}(A^{(i,j)}_1), \dots, \overline{\mathsf{p}}(A^{(i,j)}_{k(i,j)})\right]&\left[\overline{\mathsf{p}}(\Delta^{(i)}_1), \dots, \overline{\mathsf{p}}(\Delta^{(i)}_{m_i})\right] \\ &\cong h^{(i)}_j\left[\dots, \overline{\mathsf{p}}(A^{(i,j)}_l)[\overline{\mathsf{p}}(\Delta^{(i)}_1), \dots, \overline{\mathsf{p}}(\Delta^{(i)}_{m_i})], \dots\right] \\ &\cong h^{(i)}_j\left[\overline{\mathsf{p}}(A^{(i,j)}_1), \dots, \overline{\mathsf{p}}(A^{(i,j)}_{k(i,j)})\right] \end{aligned} \tag{4.29}$$

Making use of the same notation, $(f \circ \langle g_1, \dots, g_n\rangle) \circ \langle h^{(1)}_1, \dots, h^{(1)}_{m_1}, \dots, h^{(n)}_1, \dots, h^{(n)}_{m_n}\rangle$ is

$$f\left[\dots, g_i\left[\overline{\mathsf{p}}(B^{(i)}_1), \dots, \overline{\mathsf{p}}(B^{(i)}_{m_i})\right], \dots\right]\left[\dots, h^{(i)}_j\left[\overline{\mathsf{p}}(\Delta^{(i)}_1), \dots, \overline{\mathsf{p}}(\Delta^{(j)}_{m_j})\right], \dots\right]$$

and $f \circ \left\langle g_1 \circ \langle h^{(1)}_1, \dots, h^{(1)}_{m_1}\rangle, \dots, g_n \circ \langle h^{(n)}_1, \dots, h^{(n)}_{m_n}\rangle\right\rangle$ is

$$f\left[\dots, g_i\left[\dots, h^{(i)}_j\left[\overline{\mathsf{p}}(A^{(i,j)}_1), \dots, \overline{\mathsf{p}}(A^{(i,j)}_{k(i,j)})\right], \dots\right]\left[\overline{\mathsf{p}}(\Delta^{(i)}_1), \dots, \overline{\mathsf{p}}(\Delta^{(i)}_{m_i})\right], \dots\right]$$

so $\mathsf{a}_{f;g_\bullet;h_\bullet}$ is the composite

$$\begin{tikzcd}
f[g_1 \boxtimes \dots \boxtimes g_n]\left[h^{(1)}_1 \boxtimes \dots \boxtimes h^{(i)}_j \boxtimes \dots \boxtimes h^{(n)}_{m_n}\right] \arrow[d, "\mathsf{f}_{f;g_\bullet;h_\bullet}"'] \\
f\left[g_1\left[h^{(1)}_1 \boxtimes \dots \boxtimes h^{(1)}_{m_1}\right], \dots, g_n\left[h^{(n)}_1 \boxtimes \dots \boxtimes h^{(n)}_{m_n}\right]\right] \arrow[d, "\cong"', "(4.29)"] \\
f\left[\dots, g_i\left[\dots, h^{(i)}_j\left[\overline{\mathsf{p}}(A^{(i,j)}_1), \dots, \overline{\mathsf{p}}(A^{(i,j)}_{k(i,j)})\right]\left[\overline{\mathsf{p}}(\Delta^{(i)}_1), \dots, \overline{\mathsf{p}}(\Delta^{(i)}_{m_i})\right], \dots\right], \dots\right] \arrow[d, "\cong"'] \\
f\left[\dots, g_i\left[\dots, h^{(i)}_j\left[\overline{\mathsf{p}}(A^{(i,j)}_1), \dots, \overline{\mathsf{p}}(A^{(i,j)}_{k(i,j)})\right], \dots\right]\left[\overline{\mathsf{p}}(\Delta^{(i)}_1), \dots, \overline{\mathsf{p}}(\Delta^{(i)}_{m_i})\right], \dots\right]
\end{tikzcd}$$

where the final isomorphism is the evident composite of structural isomorphisms in $(S, \mathcal{C})$ and $\mathsf{f}_{f;g_\bullet;h_\bullet}$ is defined after Notation 3.1.19 (page 49).

The two coherence laws hold by the coherence of biclones. ◂

We now see where the awkwardness in the definition of pseudofunctors and transformations of biclones arises (Definitions 3.1.14 and 3.1.20): the more natural definitions are for bi-multicategories, and the versions for biclones arise via Construction 4.2.39.

Notation 4.2.40. Following the preceding construction, we sometimes write Id_A for the projection $\mathsf{p}_A^{(1)}$ in a biclone, and refer to it as *the identity on A*. ◂

Remark 4.2.41. For a biclone $(S, \mathcal{C})$, the bicategory $\overline{\mathcal{C}}$ obtained by restricting to unary hom-categories is biequivalent to the restriction $\overline{\mathrm{M}\mathcal{C}}$ of the corresponding bi-multicategory to linear hom-categories $\big($*c.f.* (4.12)$\big)$. Indeed, the objects and hom-categories are equal: the only difference is that for $f : X \to Y$ and $g : Y \to Z$ in $(S, \mathcal{C})$ the corresponding composite in $\overline{\mathcal{C}}$ is $f[g]$ while in $\overline{\mathrm{M}\mathcal{C}}$ it is $f\Big[g\Big[\mathsf{p}_Y^{(1)}\Big]\Big]$. ◂

The definitions of representable and cartesian biclones are now induced from their bi-multicategorical counterparts (*c.f.* Definition 4.2.13).

Definition 4.2.42.

1. A *representable biclone* is a biclone $(S, \mathcal{C})$ equipped with a choice of representable structure $\mathrm{T}_n(-)$ on $\mathrm{M}\mathcal{C}$.
2. A *cartesian biclone* is a biclone $(S, \mathcal{C})$ equipped with a choice of cartesian structure $\prod_n(-)$ on $\mathrm{M}\mathcal{C}$. ◂

Remark 4.2.43. As for fp-bicategories, we stipulate that the unary product structure in a cartesian biclone is the identity (*c.f.* Remark 4.1.3). ◂

For a clone $(S, \mathbb{C})$, the mapping $(-)[h]$ composing with a single multimap $h : X_1, \ldots, X_n \to R$ is equal to the mapping $(-) \circ \langle h \rangle$ performing the same composition in $\mathrm{M}\mathbb{C}$, since for any $g : R \to A$ one has $g \circ \langle h \rangle \overset{\text{def.}}{=} g\Big[h\Big[\mathsf{p}_{X_\bullet}^{(1)}, \ldots, \mathsf{p}_{X_\bullet}^{(n)}\Big]\Big] = g[h]$. In the world of biclones, however, the functors $(-)[h]$ and $(-) \circ \langle h \rangle$ are related by a structural isomorphism (*c.f.* Remark 4.2.41). Since $(\mathrm{M}\mathcal{C})(\Gamma; A) = \mathcal{C}(\Gamma; A)$ for every Γ and A, a choice of adjoint equivalence $\psi_{X_\bullet} : (\mathrm{M}\mathcal{C})(X_1, \ldots, X_n; A) \leftrightarrows (\mathrm{M}\mathcal{C})(R; A) : (-) \circ \langle h \rangle$ is equivalently a choice of adjoint equivalence $\psi'_{X_\bullet} : \mathcal{C}(X_1, \ldots, X_n; A) \leftrightarrows \mathcal{C}(R; A) : (-)[h]$. (To see this, apply the fact that for any morphisms $f : X \to Y$ and $g, g' : Y \to X$ in a 2-category, if $g \cong g'$ then f and g are the 1-cells of an equivalence $X \simeq Y$ if and only if f and g' are the 1-cells of such an equivalence.)

It follows that a representable biclone $(S, \mathcal{C}, \mathrm{T}_n)$ is equivalently a biclone $(S, \mathcal{C})$ equipped with a choice of object $\mathrm{T}_n(X_1, \ldots, X_n)$ and multimap $\rho_{X_\bullet} : X_1, \ldots, X_n \to \mathrm{T}_n(X_1, \ldots, X_n)$ for every $X_1, \ldots, X_n \in S$ $(n \in \mathbb{N})$, together with a choice of adjoint equivalence $\mathcal{C}(X_1, \ldots, X_n; A) \simeq \mathcal{C}\left(\mathrm{T}_n(X_1, \ldots, X_n); A\right)$ induced by pre-composing with $\rho_{X_\bullet}$ for every $A \in S$. Explicitly, this entails that for every $t : X_1, \ldots, X_n \to A$ there exists a chosen multimap $\psi_{X_\bullet}(t) : \mathrm{T}_n(X_1, \ldots, X_n) \to A$

and a 2-cell $\varepsilon_{X_\bullet;f} : \psi_{X_\bullet}(f)[\rho_{X_\bullet}] \Rightarrow f$, universal in the sense that for any $g : \mathrm{T}_n(X_1, \dots, X_n) \to A$ and $\sigma : g[\rho_{X_\bullet}] \Rightarrow f$ there exists a unique 2-cell $\sigma^\dagger : g \Rightarrow \psi_{X_\bullet}(f)$ such that

$$\begin{tikzcd}
g[\rho_{X_\bullet}] \arrow[rr, "\sigma^\dagger[\rho_{X_\bullet}]"] \arrow[dr, swap, "\sigma"] & & \psi_{X_\bullet}(f)[\rho_{X_\bullet}] \arrow[dl, "\varepsilon_{X_\bullet;f}"] \\
& f &
\end{tikzcd} \tag{4.30}$$

A similar story holds for cartesian biclones. For a sequence of multimaps $(\pi_i : R \to X_i)_{i=1,.,n}$ and $u : \Gamma \to A_i$ in the bi-multicategory $\mathrm{M}\mathcal{C}$ associated to a cartesian biclone $(S, \mathcal{C}, \Pi_n(-))$, there exists the following composite of structural isomorphisms:

$$\pi_i \circ \langle u \rangle = \pi_i\Big[u\Big[\mathsf{p}_\Gamma^{(1)}, \dots, \mathsf{p}_\Gamma^{(|\Gamma|)}\Big]\Big] \cong \pi_i[u]\Big[\mathsf{p}_\Gamma^{(1)}, \dots, \mathsf{p}_\Gamma^{(|\Gamma|)}\Big] \cong \pi_i[u]$$

It follows that the functor $(\pi_1 \circ \langle - \rangle, \dots, \pi_n \circ \langle - \rangle) : (\mathrm{M}\mathcal{C})(\Gamma; R) \to \prod_{i=1}^n (\mathrm{M}\mathcal{C})(\Gamma; X_i)$ is naturally isomorphic to the functor $(\pi_1[-], \dots, \pi_n[-]) : \mathcal{C}(\Gamma; R) \to \prod_{i=1}^n \mathcal{C}(\Gamma; X_i)$. A cartesian biclone $(S, \mathcal{C}, \Pi_n(-))$ is therefore equivalently a biclone equipped with a choice of object $\prod_n(X_1, \dots, X_n)$ and multimaps $(\pi_i : \prod_n(X_1, \dots, X_n) \to X_i)_{i=1,\dots,n}$ for every sequence $X_1, \dots, X_n \in S$ $(n \in \mathbb{N})$, together with a choice of adjoint equivalence $\mathcal{C}(\Gamma; \prod_n(X_1, \dots, X_n)) \simeq \prod_{i=1}^n \mathcal{C}(\Gamma; X_i)$. The counit of this adjoint equivalence is then characterised by the following universal property. For every sequence of multimaps $(t_i : \Gamma \to X_i)_{i=1,\dots,n}$ there exists a multimap $\mathsf{tup}(t_1, \dots, t_n) : \Gamma \to \prod_n(X_1, \dots, X_n)$ and a 2-cell ϖ with components $\varpi_{t_\bullet}^{(i)} : \pi_i[\mathsf{tup}(t_1, \dots, t_n)] \Rightarrow t_i$ for $i = 1, \dots, n$. This 2-cell is universal in the sense that, if $u : \Gamma \to \prod_n(X_1, \dots, X_n)$ and $\alpha_i : \pi_i[u] \Rightarrow t_i$ for $i = 1, \dots, n$, then there exists a unique 2-cell $\mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n) : u \Rightarrow \mathsf{tup}(t_1, \dots, t_n)$ filling the following diagram for $i = 1, \dots, n$:

$$\begin{tikzcd}
\pi_i[u] \arrow[rr, "\pi_i[\alpha]"] \arrow[dr, swap, "\alpha_i"] & & \pi_i[\mathsf{tup}(t_1, \dots, t_n)] \arrow[dl, "\varpi_{t_\bullet}^{(i)}"] \\
& t_i &
\end{tikzcd} \tag{4.31}$$

Rather than translating between compositions $f \circ \langle g_\bullet \rangle$ and $f[g_\bullet]$ throughout, in what follows we employ the biclone version of the universal property.

Remark 4.2.44. We have just shown that a *biuniversal arrow in a biclone*—defined exactly as in Definition 4.2.36—exists if and only if there exists a biuniversal arrow in the corresponding bi-multicategory. ◂

Example 4.2.45. Every fp-bicategory $(\mathcal{B}, \Pi_n(-))$ defines a biclone $\mathrm{Bicl}(\mathcal{B})$ with sorts $ob(\mathcal{B})$ and hom-categories $\mathrm{Bicl}(\mathcal{B})(X_1, \dots, X_n; Y) := \mathcal{B}(\prod_n(X_1, \dots, X_n), Y)$ (*c.f.* Example 4.2.14 on page 92). The substitution $f[g_1, \dots, g_n]$ is $f \circ \langle g_1, \dots, g_n \rangle$. This biclone is cartesian: for the adjoint equivalence (4.31) one takes the adjoint equivalence defining finite products in $\mathcal{B}$. ◂

The equivalence between representability and cartesian structure. Our aim now is to prove a version of Theorem 4.2.20 for biclones, establishing that a biclone admits a representable structure (embodied by (4.30)) if and only if it admits a cartesian structure (embodied by (4.31)). In the 1-categorical case the key to this equivalence is the construction of a sequence of multimaps $\pi_i : \mathrm{T}_n(X_1, \dots , X_n) \to X_i$ satisfying two equations for $i = 1, \dots , n$. The corresponding bicategorical construction is up-to-isomorphism.

Lemma 4.2.46. For any representable biclone $(S, \mathcal{C}, \mathrm{T}_n)$ and $X_1, \dots , X_n \in S$ $(n \in \mathbb{N})$ there exist multimaps $\pi_i : \mathrm{T}_n(X_1, \dots , X_n) \to X_i$ and invertible 2-cells $\mu^{(i)}_{X_\bullet} : \pi_i[\rho_{X_\bullet}] \Rightarrow \mathsf{p}^{(i)}_{X_\bullet}$ and $\varsigma_{X_\bullet} : \mathrm{Id}_{\mathrm{T}_n(X_1,\dots,X_n)} \Rightarrow \rho_{X_\bullet}[\pi_1, \dots , \pi_n]$ (for $i = 1, \dots , n$), as in the diagrams below:

$$\begin{tikzcd}
 & \mathrm{T}_n(X_1, \dots , X_n) \arrow[dr, "\pi_i"] \arrow[d, phantom, "\Downarrow \mu^{(i)}_{X_\bullet}"] & \\
X_1, \dots , X_n \arrow[ur, "\rho_{X_\bullet}"] \arrow[rr, "\mathsf{p}^{(i)}_{X_\bullet}"'] & {} & X_i
\end{tikzcd}
\qquad
\begin{tikzcd}
 & X_1, \dots , X_n \arrow[dr, "\rho_{X_\bullet}"] \arrow[d, phantom, "\Uparrow \varsigma_{X_\bullet}"] & \\
\mathrm{T}_n(X_1, \dots , X_n) \arrow[ur, "{[\pi_1, \dots , \pi_n]}"] \arrow[rr, "\mathrm{Id}"'] & {} & \mathrm{T}_n(X_1, \dots , X_n)
\end{tikzcd}$$

Proof. Define $\pi_i := \psi_{X_\bullet}(\mathsf{p}^{(i)}_{X_\bullet})$. For $\mu^{(i)}_{X_\bullet}$, we may immediately take the universal 2-cell $\varepsilon_{X_\bullet;\mathsf{p}^{(i)}}$ of (4.30). For $\varsigma_{X_\bullet}$ we apply the universal property (4.30) to the structural isomorphism $\varrho^{(1)}_{(\mathrm{T}_n X_\bullet)}$ to obtain an invertible 2-cell $\left(\varrho^{(1)}_{X_\bullet}\right)^\dagger : \mathrm{Id}_{(\mathrm{T}_n X_\bullet)} \Rightarrow \psi_{X_\bullet}(\rho_{X_\bullet})$. We complete the construction by defining a 2-cell $\rho_{X_\bullet}[\pi_1, \dots , \pi_n] \Rightarrow \psi_{X_\bullet}(\rho_{X_\bullet})$. Define $\alpha_{X_\bullet}$ to be the composite

$$\rho_{X_\bullet}[\pi_1, \dots , \pi_n]\,[\rho_{X_\bullet}] \overset{\cong}{\Rightarrow} \rho_{X_\bullet}[\pi_\bullet[\rho_{X_\bullet}]] \xRightarrow{\rho_{X_\bullet}\left[\mu^{(\bullet)}_{X_\bullet}\right]} \rho_{X_\bullet}\left[\mathsf{p}^{(1)}_{X_\bullet}, \dots , \mathsf{p}^{(n)}_{X_\bullet}\right] \xRightarrow{\iota^{-1}} \rho_{X_\bullet}$$

Since this composite is invertible, by the universal property (4.30) there exists an invertible 2-cell $(\alpha_{X_\bullet})^\dagger : \rho_{X_\bullet}[\pi_1, \dots , \pi_n] \Rightarrow \psi_{X_\bullet}(\rho_{X_\bullet})$. We therefore define $\varsigma_{X_\bullet}$ to be the composite

$$\mathrm{Id}_{(\mathrm{T}X_\bullet)} \xRightarrow{{\varrho^{(1)}_{X_\bullet}}^\dagger} \psi_{X_\bullet}(\rho_{X_\bullet}) \xRightarrow{(\alpha^\dagger_{X_\bullet})^{-1}} \rho_{X_\bullet}[\pi_1, \dots , \pi_n]$$

□

To bicategorify Lemma 4.2.19 we shall also employ a kind of 'mirror image' of the preceding lemma, capturing the crucial construction available in the presence of cartesian structure; this should be compared to the discussion preceding Definition 4.2.15 (page 93). Just as we had to generalise the notion of isomorphism for the clone case, so we need to generalise the notion of (adjoint) equivalence for the biclone case.

Definition 4.2.47. Let $(S, \mathcal{C})$ be a biclone.

1. An *adjunction* $X_1 \ldots, X_n \leftrightarrows Y$ in $(S, \mathcal{C})$ consists of 1-cells $e : X_1, \ldots, X_n \to Y$ and $f_i : Y \to X_i$ $(i = 1, \ldots, n)$ with 2-cells

$$\eta : \mathsf{p}_Y^{(1)} \Rightarrow e[f_1, \ldots, f_n] : Y \to Y$$
$$\varepsilon_i : f_i[e] \Rightarrow \mathsf{p}^{(i)}_{X_1, \ldots, X_n} : X_1, \ldots, X_n \to X_i \quad (i = 1, \ldots, n)$$

such that the following diagrams commute for $i = 1, \ldots, n$:

$$\begin{tikzcd}
\mathsf{p}_Y^{(1)}[e] \arrow[r, "{\eta[e]}"] \arrow[d, "\varrho_e^{(1)}"'] & e[f_\bullet][e] \arrow[r, "\mathsf{assoc}_{e;f_\bullet;e}"] & e[f_\bullet[e]] \arrow[d, "{e[\varepsilon_1, \ldots, \varepsilon_n]}"] \\
e \arrow[rr, "\iota_e"'] & & e\left[\mathsf{p}^{(1)}_{X_\bullet}, \ldots, \mathsf{p}^{(n)}_{X_\bullet}\right]
\end{tikzcd} \tag{4.32}$$

$$\begin{tikzcd}
f_i \arrow[r, "\iota_{f_i}"] \arrow[d, equal] & f_i\left[\mathsf{p}_Y^{(1)}\right] \arrow[r, "{f_i[\eta]}"] & f_i[e[f_1, \ldots, f_n]] \arrow[d, "\mathsf{assoc}^{-1}_{f_i;e;f_\bullet}"] \\
f_i & \mathsf{p}^{(i)}[f_1, \ldots, f_n] \arrow[l, "\varrho^{(i)}_{f_\bullet}"] & f_i[e][f_1, \ldots, f_n] \arrow[l, "{\varepsilon_i[f_1, \ldots, f_n]}"]
\end{tikzcd} \tag{4.33}$$

2. An *equivalence* in $(S, \mathcal{C})$ consists of 1-cells $e : X_1, \ldots, X_n \to Y$ and $f_i : Y \to X_i (i = 1, \ldots, n)$ with invertible 2-cells

$$\eta : \mathsf{p}_Y^{(1)} \overset{\cong}{\Rightarrow} e[f_1, \ldots, f_n] : Y \to Y$$
$$\varepsilon_i : f_i[e] \overset{\cong}{\Rightarrow} \mathsf{p}^{(i)}_{X_1, \ldots, X_n} : X_1, \ldots, X_n \to X_i \ (i = 1, \ldots, n)$$

3. A *adjoint equivalence* in $(S, \mathcal{C})$ is an adjunction for which η and ε_i are invertible for $i = 1, \ldots, n$. ◀

In particular, a unary (adjoint) equivalence $X \simeq Y$ is just an (adjoint) equivalence in the usual, bicategorical sense.

Lemma 4.2.48. For any sequence of objects $X_1, \ldots, X_n (n \in \mathbb{N})$ in a cartesian biclone $(S, \mathcal{C}, \Pi_n(-))$, there exists an adjoint equivalence between $X_1, \ldots, X_n \simeq \prod_n(X_1, \ldots, X_n)$.

Proof. We employ the notation of (4.31) for cartesian structure. For the 2-cell $\pi_i[\mathsf{tup}(\mathsf{p}^{(1)}_{X_\bullet}, \ldots, \mathsf{p}^{(n)}_{X_\bullet})] \Rightarrow \mathsf{p}^{(i)}_{X_\bullet}$ we can immediately take $\varpi^{(i)}_{X_\bullet}$. The real work is in providing a 2-cell $\gamma : \mathrm{Id}_{(\prod X_\bullet)} \Rightarrow \mathsf{tup}(\mathsf{p}^{(1)}, \ldots, \mathsf{p}^{(n)})[\pi_1, \ldots, \pi_n]$. By the universality of the counit $\varpi = (\varpi^{(1)}, \ldots, \varpi^{(n)})$ it suffices to define a family of invertible 2-cells $\zeta_i : \pi_i\big[\mathsf{tup}(\mathsf{p}^{(1)}, \ldots, \mathsf{p}^{(n)})[\pi_1, \ldots, \pi_n]\big] \Rightarrow \pi_i$ for $i = 1, \ldots, n$. We may then define γ to be the composite

$$\mathrm{Id}_{(\prod X_\bullet)} \xRightarrow{\varsigma_{\mathrm{Id}_{(\prod X_\bullet)}}} \mathsf{tup}(\pi_\bullet\big[\mathrm{Id}_{(\prod X_\bullet)}\big]) \xRightarrow{\mathsf{tup}(\iota^{-1}, \ldots, \iota^{-1})} \mathsf{tup}(\pi_\bullet) \xRightarrow{(\mathsf{p}^\dagger(\zeta_1, \ldots, \zeta_n))^{-1}} \mathsf{tup}(\mathsf{p}^{(\bullet)})[\pi_\bullet]$$

where ς is the unit of the adjoint equivalence witnessing $(\pi_1, \ldots, \pi_n)$ as a biuniversal arrow. The 2-cells ζ_i are defined as follows:

$$\pi_i\big[\mathsf{tup}(\mathsf{p}^{(1)}, \ldots, \mathsf{p}^{(n)})[\pi_\bullet]\big] \xRightarrow{\mathsf{assoc}^{-1}} \pi_i\big[\mathsf{tup}(\mathsf{p}^{(1)}, \ldots, \mathsf{p}^{(n)})\big]\,[\pi_\bullet] \xRightarrow{\varpi^{(i)}_{X_\bullet}[\pi_\bullet]} \mathsf{p}^{(i)}[\pi_\bullet] \xRightarrow{\varrho^{(i)}} \pi_i$$

Since each ζ_i is invertible, $\mathsf{p}^\dagger(\zeta_1, \ldots, \zeta_n)$ is also invertible. Checking that diagram (4.33) commutes is straightforward; for (4.32) one must use the universal property, checking that both routes around the diagram are the unique 2-cell corresponding to the composite

$$\pi_i\big[\mathsf{tup}(\mathsf{p}^{(1)}, \ldots, \mathsf{p}^{(n)})[\pi_\bullet][\mathsf{tup}(\mathsf{p}^{(1)}, \ldots, \mathsf{p}^{(n)})]\big] \xRightarrow{\pi_i[\beta_\bullet]} \pi_i\big[\mathsf{tup}(\mathsf{p}^{(1)}, \ldots, \mathsf{p}^{(n)})\big] \xRightarrow{\varpi_{X_\bullet}^{(i)}} \mathsf{p}^{(i)}$$

where β_i is defined to be

$$\mathsf{tup}(\mathsf{p}^{(\bullet)})[\pi_\bullet]\Big[\mathsf{tup}(\mathsf{p}^{(\bullet)})\Big] \xRightarrow{\mathsf{assoc}} \mathsf{tup}(\mathsf{p}^{(\bullet)})\Big[\pi_\bullet\Big[\mathsf{tup}(\mathsf{p}^{(\bullet)})\Big]\Big] \xRightarrow{\mathsf{tup}(\mathsf{p}^{(\bullet)})\left[\varpi_{X_\bullet}^{(\bullet)}\right]} \mathsf{tup}(\mathsf{p}^{(\bullet)})\Big[\mathsf{p}^{(\bullet)}\Big] \xRightarrow{\iota^{-1}} \mathsf{tup}(\mathsf{p}^{(\bullet)})$$

for $i = 1, \ldots, n$. □

As for clones, the extra structure of a biclone entails that birepresentable arrows are closed under composition. The strategy for the proof is familiar from Lemma 4.2.18.

Lemma 4.2.49. A biclone $(S, \mathcal{C})$ admits a representable structure if and only if for every $X_1, \ldots, X_n \in \mathcal{M}$ $(n \in \mathbb{N})$ there exists a chosen object $\mathrm{T}_n(X_1, \ldots, X_n) \in \mathcal{M}$ and a birepresentable multimap $\rho_{X_\bullet} : X_1, \ldots, X_n \to \mathrm{T}_n(X_1, \ldots, X_n)$.

Proof. It suffices to show that birepresentable multimaps are closed under composition. Mirroring the proof of Lemma 4.2.18, suppose given birepresentable multimaps

$$\rho_{X_\bullet} : X_1, \ldots, X_n \to \mathrm{T}_n(X_1, \ldots, X_n)$$
$$\rho_{Y_\bullet} : Y_1, \ldots, Y_m \to \mathrm{T}_m(Y_1, \ldots, Y_m)$$
$$\rho_{(\prod X_\bullet, \prod Y_\bullet)} : \mathrm{T}_n X_\bullet, \mathrm{T}_m Y_\bullet \to \mathrm{T}_2(\mathrm{T}_n X_\bullet, \mathrm{T}_m Y_\bullet)$$

We want to show that the composite $\rho_{(\prod X_\bullet, \prod Y_\bullet)} \circ (\rho_{X_\bullet}, \rho_{Y_\bullet})$ in $\mathrm{M}\mathcal{C}$, which is the composite $\overline{\rho} := \rho_{(\prod X_\bullet, \prod Y_\bullet)}\big[\rho_{X_\bullet}\big[\mathsf{p}^{(1)}, \ldots, \mathsf{p}^{(n)}\big], \rho_{Y_\bullet}\big[\mathsf{p}^{(n+1)}, \ldots, \mathsf{p}^{(n+m)}\big]\big]$ in $\mathcal{C}$, is birepresentable. Define projections $\pi_i^X : \mathrm{T}_n(X_1, \ldots, X_n) \to X_i$, $\pi_j^Y : \mathrm{T}_m(Y_1, \ldots, Y_m) \to Y_j$ and $\pi^{X,Y}$ as in the proof of Lemma 4.2.18, and likewise define a family of multimaps $\overline{\pi}_i : \mathrm{T}_2(\mathrm{T}_n X_\bullet, \mathrm{T}_m Y_\bullet) \to Z_i$ for $i = 1, \ldots, n+m$ (where Z_i is X_i for $1 \leqslant i \leqslant n$ and Y_{i-n} for $n+1 \leqslant i \leqslant n+m$) as in (4.14). Finally, for $1 \leqslant i \leqslant n$ define an invertible 2-cell $\beta^{(1)} : \rho_{X_\bullet}[\overline{\pi}_1, \ldots, \overline{\pi}_n] \Rightarrow \pi_1^{X,Y} : \mathrm{T}_2(\mathrm{T}_n X_\bullet, \mathrm{T}_m X_\bullet) \to \mathrm{T}_n X_\bullet$ by

$$\begin{tikzcd}
\rho_{X_\bullet}[\overline{\pi}_1, \ldots, \overline{\pi}_n] \arrow[r, "\beta^{(1)}"] \arrow[d, equal] & \pi_1^{X,Y} \\
\rho_{X_\bullet}\left[\pi_1^X\left[\pi_1^{X,Y}\right], \ldots, \pi_n^X\left[\pi_1^{X,Y}\right]\right] \arrow[d, "\mathsf{assoc}^{-1}_{\rho_{X_\bullet};\pi_\bullet;\pi_1}"'] & \\
\rho_{X_\bullet}\left[\pi_1^X, \ldots, \pi_n^X\right]\left[\pi_1^{X,Y}\right] \arrow[r, "\varsigma_{X_\bullet}^{-1}\left[\pi_1^{X,Y}\right]"'] & \mathrm{Id}_{(\mathrm{T}X_\bullet}\left[\pi_1^{X,Y})\right] \arrow[uu, "\varrho_{\pi_1}^{(1)}"']
\end{tikzcd}$$

We define $\beta^{(2)} : \rho_{Y_\bullet}[r_{n+1}, \dots, \overline{\pi}_{n+m}] \Rightarrow \pi_2^{X,Y} : \mathrm{T}_2(\mathrm{T}_n X_\bullet, \mathrm{T}_m X_\bullet) \to \mathrm{T}_m Y_\bullet$ similarly.

We are now in a position to define the pseudo-inverse to $(-) \circ \langle \overline{\rho} \rangle : \mathcal{M}\big(\mathrm{T}_2(\mathrm{T}_n X_\bullet, \mathrm{T}_m Y_\bullet); A\big) \to \mathcal{M}(X_1, \dots, X_n, Y_1, \dots, Y_m; A)$. For $h : X_1, \dots, X_n, Y_1, \dots, Y_m \to A$ we define $\overline{\psi}(h)$ to be the composite

$$\mathrm{T}_2(\mathrm{T}_n X_\bullet, \mathrm{T}_m Y_\bullet) \xrightarrow{[\overline{\pi}_1, \dots, \overline{\pi}_{n+m}]} X_1, \dots, X_n, Y_1, \dots, Y_m \xrightarrow{h} A$$

in $\mathcal{C}$; this mapping is clearly functorial. It therefore suffices to construct natural isomorphisms $\mathrm{id}_{\mathcal{M}(\mathrm{T}(\mathrm{T}X_\bullet, \mathrm{T}Y_\bullet);A)} \cong \overline{\psi}\big((-) \circ \langle \overline{\rho} \rangle\big)$ and $\mathrm{id}_{\mathcal{M}(X_1,\dots,X_n,Y_1,\dots,Y_m;A)} \cong \big(\overline{\psi}(-)\big) \circ \langle \overline{\rho} \rangle$; this lifts to an adjoint equivalence between the same 1-cells by the usual well-known argument (*e.g.* [Mac98, IV.3]).

To this end, let us define invertible 2-cells τ and σ_i $(i = 1, \dots, n+m)$ that will make up the bulk of the required isomorphisms. The 2-cell τ is defined as follows:

$$\begin{tikzcd}
\rho_{(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}\big[\rho_{X_\bullet}\big[\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)}\big], \rho_{Y_\bullet}\big[\mathsf{p}^{(n+1)}, \dots, \mathsf{p}^{(n+m)}\big]\big]\,[\overline{\pi}_1, \dots, \overline{\pi}_{n+m}] \arrow[r, "\tau"] \arrow[d, "\cong"'] & \mathrm{Id}_{\mathrm{T}(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)} \\
\rho_{(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}\big[\rho_{X_\bullet}\big[\mathsf{p}^{(\bullet)}[\overline{\pi}_\bullet]\big], \rho_{Y_\bullet}\big[\mathsf{p}^{(\bullet)}[\overline{\pi}_\bullet]\big]\big] \arrow[d, "\rho_{(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}\left[\rho_{X_\bullet}\left[\varrho^{(\bullet)}\right]{,}\rho_{Y_\bullet}\left[\varrho^{(\bullet)}\right]\right]"'] & \\
\rho_{(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}[\rho_{X_\bullet}[\overline{\pi}_1, \dots, \overline{\pi}_n], \rho_{Y_\bullet}[\overline{\pi}_{n+1}, \dots, \overline{\pi}_{n+m}]] \arrow[r, "\rho_{(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}\left[\beta^{(1)}{,}\beta^{(2)}\right]"'] & \rho_{(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}\Big[\pi_1^{X,Y}, \pi_2^{X,Y}\Big] \arrow[uu, "\varsigma^{-1}_{(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}"']
\end{tikzcd}$$

The 2-cells $\sigma_1, \dots, \sigma_n$, on the other hand, are defined by the following diagram; the definitions of $\sigma_{n+1}, \dots, \sigma_{n+m}$ are the same, modulo the obvious adjustments.

$$\begin{tikzcd}
\overline{\pi}_i\big[\rho_{(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}\big[\rho_{X_\bullet}\big[\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)}\big], \rho_{Y_\bullet}\big[\mathsf{p}^{(n+1)}, \dots, \mathsf{p}^{(n+m)}\big]\big]\big] \arrow[r, "\sigma_i"] \arrow[d, equal] & \mathsf{p}^{(i)}_{X_1, \dots, X_n, Y_1, \dots, Y_m} \\
\pi_i^X\Big[\pi_1^{X,Y}\Big]\big[\rho_{(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}\big[\rho_{X_\bullet}\big[\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)}\big], \rho_{Y_\bullet}\big[\mathsf{p}^{(n+1)}, \dots, \mathsf{p}^{(n+m)}\big]\big]\big] \arrow[d, "\cong"'] & \\
\pi_i^X\big[\pi_1^X\big[\rho_{(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}\big]\big]\,\big[\rho_{X_\bullet}\big[\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)}\big], \rho_{Y_\bullet}\big[\mathsf{p}^{(n+1)}, \dots, \mathsf{p}^{(n+m)}\big]\big] \arrow[d, "\pi_i^X\left[\mu^{(1)}_{\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet}\right]\left[\rho_{X_\bullet}\left[\mathsf{p}^{(\bullet)}\right]{,}\rho_{Y_\bullet}\left[\mathsf{p}^{(\bullet)}\right]\right]"'] & \\
\pi_i^X\Big[\mathsf{p}^{(1)}_{X_\bullet}\Big]\,\big[\rho_{X_\bullet}\big[\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)}\big], \rho_{Y_\bullet}\big[\mathsf{p}^{(n+1)}, \dots, \mathsf{p}^{(n+m)}\big]\big] \arrow[d, "\cong"'] & \\
\pi_i^X[\rho_{X_\bullet}]\,\big[\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)}\big] \arrow[r, "\mu^{(i)}_{X_\bullet}\left[\mathsf{p}^{(1)}{,}\dots{,}\mathsf{p}^{(n)}\right]"'] & \mathsf{p}^{(i)}\big[\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)}\big] \arrow[uuuu, "\varrho^{(i)}_{\mathsf{p}^{(\bullet)}}"']
\end{tikzcd}$$

The required natural isomorphisms are then defined to be the composites

$$\overline{\psi}(g) \circ \langle \overline{\rho} \rangle = g[\overline{\pi}_1, \dots, \overline{\pi}_{n+m}]\,[\overline{\rho}] \overset{\mathsf{assoc}}{\Longrightarrow} g[r_\bullet[\overline{\rho}]] \overset{g[\sigma_\bullet]}{\Longrightarrow} g\big[\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n+m)}\big] \overset{\iota^{-1}}{\Longrightarrow} g$$

$$\overline{\psi}(h \circ \langle \overline{\rho} \rangle) = h[\overline{\rho}]\,[\overline{\pi}_1, \dots, \overline{\pi}_{n+m}] \overset{\mathsf{assoc}}{\Longrightarrow} h[\overline{\rho}[\overline{\pi}_1, \dots, \overline{\pi}_{n+m}]] \overset{h[\tau]}{\Longrightarrow} h\big[\mathrm{Id}_{\mathrm{T}(\mathrm{T}X_\bullet,\mathrm{T}Y_\bullet)}\big] \overset{\iota^{-1}}{\Longrightarrow} h$$

for $g : \mathrm{T}_2(\mathrm{T}_n X_\bullet, \mathrm{T}_m Y_\bullet) \to A$ and $h : X_1, \dots, X_n, Y_1, \dots, Y_m \to A$. □

We now prove the central result of this section.

Lemma 4.2.50. A biclone $(S, \mathcal{C})$ admits a choice of representable structure if and only if it admits a choice of cartesian structure.

Proof. $\boxed{\Rightarrow}$ Let $\rho_{X_\bullet} : X_1, \dots, X_n \to \mathrm{T}_n(X_1, \dots, X_n)$ be a birepresentable multimap. We claim the sequence of multimaps $(\pi_i : \mathrm{T}_n(X_1, \dots, X_n) \to X_i)_{i=1,\dots,n}$ defined in Lemma 4.2.46 form a biuniversal multimap. We are therefore required to provide a mapping $\mathsf{tup} : \prod_{i=1}^{n} \mathcal{M}(\Gamma; X_i) \to \mathcal{M}\big(\Gamma; \mathrm{T}_n(X_1, \dots, X_n)\big)$ and a universal 2-cell with components $\varpi^{(i)}_{X_\bullet} : \pi_i[\mathsf{tup}(f_1, \dots, f_n)] \Rightarrow f_i$ for $i = 1, \dots, n$. We define $\mathsf{tup}(f_1, \dots, f_n) := \rho_{X_\bullet}[f_1, \dots, f_n]$ and set $\varpi^{(i)}_{X_\bullet}$ to be the composite

$$\pi_i[\rho_{X_\bullet}[f_1, \dots, f_n]] \xRightarrow{\mathsf{assoc}^{-1}} \pi_i[\rho_{X_\bullet}]\,[f_1, \dots, f_n] \xRightarrow{\mu^{(i)}_{X_\bullet}[f_\bullet]} \mathsf{p}^{(i)}[f_1, \dots, f_n] \xRightarrow{\varrho^{(i)}} f_i$$

For universality, suppose $g : \Gamma \to \mathrm{T}_n(X_1, \dots, X_n)$ and $\alpha_i : \pi_i[g] \Rightarrow f_i$ for $i = 1, \dots, n$. We define 2-cell $\mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n) : g \Rightarrow \mathsf{tup}(f_1, \dots, f_n)$ by the commutativity of the following diagram:

$$\begin{tikzcd}
g \arrow[rr, "\mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n)"] \arrow[d, "\varrho_g^{(-1)}"'] & & \rho_{X_\bullet}[f_1, \dots, f_n] \\
\mathrm{Id}_{(\mathrm{T}X_\bullet)}[g] \arrow[r, "\varsigma_{X_\bullet}[g]"'] & \rho_{X_\bullet}[\pi_1, \dots, \pi_n]\,[g] \arrow[r, "\mathsf{assoc}_{\rho_{X_\bullet};\pi_\bullet;g}"'] & \rho_{X_\bullet}[\pi_1[g], \dots, \pi_n[g]] \arrow[u, "\rho_{X_\bullet}[\alpha_\bullet]"']
\end{tikzcd} \tag{4.34}$$

where we employ the 2-cell $\varsigma_{X_\bullet}$ defined in Lemma 4.2.46. For the existence part of the claim, we need to check that the composite

$$\pi_i[g] \xRightarrow{\pi_i\left[\mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n)\right]} \pi_i[\mathsf{tup}(f_1, \dots, f_n)] \xRightarrow{\varpi^{(i)}_{X_\bullet}} f_i$$

is equal to α_i for $i = 1, \dots, n$. Most of the calculation is straightforward; the key lemma is that the following diagram commutes for $i = 1, \dots, n$:

$$\begin{tikzcd}
\pi_i \arrow[r, equal] \arrow[d, "\iota_{\pi_i}"'] & \pi_i \\
\pi_i\left[\mathrm{Id}_{(\mathrm{T}X_\bullet)}\right] \arrow[d, "\pi_i[\varsigma_{X_\bullet}]"'] & \\
\pi_i[\rho_{X_\bullet}[\pi_1, \dots, \pi_n]] \arrow[d, "\mathsf{assoc}^{-1}_{\pi_i;\rho_{X_\bullet};\pi_\bullet}"'] & \\
\pi_i[\rho_{X_\bullet}]\,[\pi_1, \dots, \pi_n] \arrow[r, "\mu^{(i)}_{X_\bullet}[\pi_\bullet]"'] & \mathsf{p}^{(i)}[\pi_1, \dots, \pi_n] \arrow[uuu, "\varrho^{(i)}_{\pi_\bullet}"']
\end{tikzcd}$$

For uniqueness, let $g : \Gamma \to \mathrm{T}_n(X_1, \dots, X_n)$ be any multimap and suppose that $\sigma : g \Rightarrow \mathsf{tup}(f_1, \dots, f_n)$ satisfies $\varpi^{(i)}_{X_\bullet} \bullet \pi_i[\sigma] = \alpha_i$ for $i = 1, \dots, n$. Substituting this equation into the definition of $\mathsf{p}^\dagger(\alpha_1, \ldots, \alpha_n)$ and using the above diagram, one sees that $\sigma = \mathsf{p}^\dagger(\alpha_1, \ldots, \alpha_n)$ as required.

Finally, it remains to check that the unit and counit of the adjunction we have just constructed are invertible. The counit is the universal 2-cell, which is certainly invertible. The unit is constructed by applying $\mathsf{p}^\dagger(-, \ldots, =)$ to the identity, which is invertible since it is a composite of invertible 2-cells.

$\boxed{\Leftarrow}$ For the converse, we claim that $\rho_{X_\bullet} := \mathsf{tup}(\mathsf{p}^{(1)}_{X_\bullet}, \ldots, \mathsf{p}^{(n)}_{X_\bullet}) : X_1, \ldots, X_n \to \prod_n(X_1, \ldots, X_n)$ is birepresentable. We therefore need to supply a mapping $\psi_{X_\bullet} : (\mathrm{M}\mathcal{C})(X_1, \ldots, X_n; A) \to (\mathrm{M}\mathcal{C})\big(\prod_n(X_1, \ldots, X_n); A\big)$ and a universal 2-cell $\varepsilon_{A,g} : \psi_{X_\bullet}(g)[\rho_{X_\bullet}] \Rightarrow g$. We define $\psi_{X_\bullet}(g) := g[\pi_1, \ldots, \pi_n]$ and set $\varepsilon_{A,g}$ to be the invertible composite

$$\begin{tikzcd}
g[\pi_1, \ldots, \pi_n]\Big[\mathsf{tup}(\mathsf{p}^{(1)}_{X_\bullet}, \ldots, \mathsf{p}^{(n)}_{X_\bullet})\Big] \arrow[rr, "\varepsilon_{A,g}"] \arrow[d, "\mathsf{assoc}^{-1}_{g;\pi_\bullet;\mathsf{tup}(\mathsf{p}^{(\bullet)})}"'] & & g \\
g\Big[\pi_\bullet\Big[\mathsf{tup}(\mathsf{p}^{(1)}_{X_\bullet}, \mathsf{p}^{(n)}_{X_\bullet})\Big]\Big] \arrow[rr, "{g[\varpi^{(\bullet)}_{X_\bullet}]}"'] & & g\Big[\mathsf{p}^{(1)}_{X_\bullet}, \ldots, \mathsf{p}^{(n)}_{X_\bullet}\Big] \arrow[u, "\iota_g^{-1}"']
\end{tikzcd}$$

For universality, let $f : \prod_n(X_1, \ldots, X_n) \to A$ by any multimap and $\delta : f\big[\mathsf{tup}(\mathsf{p}^{(1)}, \ldots, \mathsf{p}^{(n)})\big] \Rightarrow g$ be any 2-cell. We define $\delta^\dagger$ as the following invertible composite, using the 2-cell γ from the adjoint equivalence of Lemma 4.2.48:

$$f \overset{\iota}{\Rightarrow} f\Big[\mathsf{p}^{(1)}_{(\prod X_\bullet})\Big] \xRightarrow{f[\gamma^{-1}]} f\Big[\mathsf{tup}(\mathsf{p}^{(\bullet)}_{X_\bullet})[\pi_1, \ldots, \pi_n]\Big] \xRightarrow{\mathsf{assoc}^{-1}} f\Big[\mathsf{tup}(\mathsf{p}^{(\bullet)}_{X_\bullet})\Big][\pi_\bullet] \xRightarrow{\delta[\pi_\bullet]} g[\pi_\bullet]$$

The rest of the proof is a diagram chase. To check the existence part of the universal property one uses law (4.32) of an adjoint equivalence; for uniqueness one uses (4.33). Since $\delta^\dagger$ is invertible whenever δ is, the unit is invertible and one obtains the required adjoint equivalence. □

We collect these results together to obtain a bicategorical version of Theorem 4.2.20. The final case is Lemma 4.2.37.

Theorem 4.2.51. Let $(S, \mathcal{C})$ be a biclone. Then the following are equivalent:

1. $(S, \mathcal{C})$ admits a representable structure,
2. For every $X_1, \ldots, X_n \in S$ $(n \in \mathbb{N})$ there exists a choice of object $\prod_n(X_1, \ldots, X_n)$ and a birepresentable multimap $\rho_{X_\bullet} : X_1, \ldots, X_n \to \prod_n(X_1, \ldots, X_n)$,
3. $(S, \mathcal{C})$ admits a cartesian structure,
4. For every $X_1, \ldots, X_n \in S$ $(n \in \mathbb{N})$ there exists a choice of object $\prod_n(X_1, \ldots, X_n)$ together with a chosen family of adjoint equivalences $(\mathrm{M}\mathcal{C})\big(\Gamma; \prod_n(X_1, \ldots, X_n)\big) \simeq \prod_{i=1}^n (\mathrm{M}\mathcal{C})(\Gamma; X_i)$, pseudonatural in the sense of Lemma 4.2.37(2). □

Restricting to unary hom-categories, case (4) of the theorem entails the following.

Corollary 4.2.52. For any representable biclone $(S, \mathcal{C}, \mathrm{T}_n)$, the nucleus $\overline{\mathcal{C}}$ is an fp-bicategory with product structure defined as in $\mathcal{C}$. □

4.2.4 Synthesising a type theory for fp-bicategories

fp-Bicategories from cartesian biclones. On page 104 we used diagram (4.19) and the isomorphisms following to argue that, in order to construct a type theory describing cartesian categories, it is sufficient to construct a type theory for cartesian clones. Moreover, we showed how such a type theory could be synthesised from the construction of the free cartesian clone on a $\Lambda^{\times}$-signature.

We repeat this process to synthesise the type theory $\Lambda^{\times}_{\mathrm{ps}}$. The starting point is an appropriate notion of signature. To extend from clones to biclones we extended from multigraphs to 2-multigraphs; to extend from cartesian clones to cartesian biclones we extend $\Lambda^{\times}$-signatures in the same way.

Definition 4.2.53. A $\Lambda^{\times}_{\mathrm{ps}}$-*signature* $\mathcal{S} = (\mathfrak{B}, \mathcal{G})$ consists of

1. A set of base types $\mathfrak{B}$,
2. A 2-multigraph $\mathcal{G}$ for which the set of nodes $\mathcal{G}_0$ is generated by the grammar

$$A_1, \ldots, A_n ::= B \mid \textstyle\prod_n(A_1, \ldots, A_n) \qquad (B \in \mathfrak{B}, n \in \mathbb{N}) \tag{4.35}$$

A *homomorphism* $h : \mathcal{S} \to \mathcal{S}'$ of $\Lambda^{\times}_{\mathrm{ps}}$-signatures is a 2-multigraph homomorphism $h : \mathcal{G} \to \mathcal{G}'$ that respects products, in the sense that $h_0(\prod_n(A_1, \ldots, A_n)) = \prod_n (h_0A_1, \ldots, h_0A_n)$ for all $A_1, \ldots, A_n \in \mathcal{G}_0$ $(n \in \mathbb{N})$.

We denote the category of $\Lambda^{\times}_{\mathrm{ps}}$-signatures by $\Lambda^{\times}_{\mathrm{ps}}$-sig and the full sub-category of *unary* $\Lambda^{\times}_{\mathrm{ps}}$-signatures—in which the 2-multigraph $\mathcal{G}$ is a 2-graph—by $\Lambda^{\times}_{\mathrm{ps}}\text{-sig}\big|_1$. ◂

Every cartesian bi-multicategory (resp. cartesian biclone) determines an $\Lambda^{\times}_{\mathrm{ps}}$-signature, and every fp-bicategory determines a unary $\Lambda^{\times}_{\mathrm{ps}}$-signature.

Notation 4.2.54 (*c.f.* Notation 4.2.23)**.** For any $\Lambda^{\times}_{\mathrm{ps}}$-signature $\mathcal{S} = (\mathfrak{B}, \mathcal{G})$ we write $\widetilde{\mathfrak{B}}$ for the set generated from $\mathfrak{B}$ by the grammar (4.35). In particular, when the signature is just a set (*i.e.* the graph $\mathcal{G}$ has no edges) we denote the signature $\mathcal{S} = (\mathfrak{B}, \mathcal{S})$ simply by $\widetilde{\mathfrak{B}}$. ◂

The following result is proven in exactly the same way as Lemma 4.2.24.

Lemma 4.2.55. The inclusion $\iota : \Lambda^{\times}_{\mathrm{ps}}\text{-sig}\big|_1 \hookrightarrow \Lambda^{\times}_{\mathrm{ps}}\text{-sig}$ has a right adjoint. □

The construction of the free cartesian clone on a cartesian category (Lemma 4.2.28) relies crucially on the identity $\langle \pi_1, \ldots, \pi_n \rangle = \mathrm{id}_{(\prod_{i=1}^{n} X_i)}$ in a cartesian category so we cannot directly import this into the bicategorical setting. In place of diagram (4.19), therefore, one obtains a slightly restricted result. We will construct the following diagram of adjunctions, in which CartBiclone denotes the category of cartesian biclones and strict pseudofunctors strictly preserving the product structure, and fp-Bicat denotes the category of fp-bicategories and strict fp-pseudofunctors:

$$\begin{tikzcd}
& \mathrm{CartBiclone} \arrow[dl, bend right] & \\
\Lambda^{\times}_{\mathrm{ps}}\text{-sig} \arrow[ur, bend right] \arrow[ur, phantom, "\top"] \arrow[dr, bend right] & & \text{fp-Bicat} \arrow[dl, bend left] \\
& \Lambda^{\times}_{\mathrm{ps}}\text{-sig}\big|_1 \arrow[ul, bend right] \arrow[ul, phantom, "\perp"] \arrow[ur, bend left] \arrow[ur, phantom, "\perp"] &
\end{tikzcd} \tag{4.36}$$

We shall then show that the free fp-bicategory on a unary $\Lambda^{\times}_{\mathrm{ps}}$-signature $\mathcal{S}$ is obtained by restricting the construction of the free cartesian biclone on $\mathcal{S}$ to unary multimaps. Thus, the internal language of the free fp-bicategory on $\mathcal{S}$ is the internal language of the free cartesian biclone on $\mathcal{S}$, in which every rule is restricted to unary multimaps. Here some care is required: as we shall see, this is not the same as taking the nucleus of the free cartesian biclone.

Let us begin by making precise the notion of a (strict) morphism of cartesian biclones. The notion of biuniversal arrow for biclones is defined exactly as for bi-multicategories (Definition 4.2.36); the corresponding notion of preservation extends that for bicategories (Definition 2.2.15).

Definition 4.2.56. Let $F : (S, \mathcal{C}) \to (T, \mathcal{D})$ and $F' : (S', \mathcal{C}') \to (T', \mathcal{D}')$ be pseudofunctors of biclones and suppose (R, u) and (R', u') are biuniversal arrows from F to $C \in T$ and from F' to $C' \in T'$, respectively. A pair of pseudofunctors $(K : \mathcal{D} \to \mathcal{D}', L : \mathcal{C} \to \mathcal{C}')$ is a *strict morphism of biuniversal arrows* from (R, u) to (R', u') if

1. K and L are strict pseudofunctors satisfying $KF = F'L$,
2. $LR = R'$, $KC = C'$ and $Ku = u'$,
3. The mappings $\psi_B : \mathcal{D}(FB, C) \to \mathcal{C}(B, R)$ and $\psi'_{B'} : \mathcal{D}'(F'B', C') \to \mathcal{C}'(B', R')$ are preserved, so that $L\psi_B(f) = \psi'_{LB}K(f)$ for every $f : FB \to C$,
4. For every $B \in S$ and equivalence $u[F(-)] : \mathcal{B}(B, R) \leftrightarrows \mathcal{C}(FB, C) : \psi_B$ the universal arrow $\varepsilon_{B,h} : u[F\psi_B(h)] \Rightarrow h$ is strictly preserved, in the sense that $K_{FB,C}(\varepsilon_{B,h}) = \varepsilon_{LB,Kh}$. ◂

We instantiate this in the case of cartesian biclones using the notation of (4.31) (page 115).

Definition 4.2.57. A *cartesian pseudofunctor* $(F, \mathrm{q}^{\times}) : (S, \mathcal{C}, \Pi_n(-)) \to (S', \mathcal{C}', \Pi_n(-))$ of cartesian biclones is a pseudofunctor $F : \mathcal{C} \to \mathcal{C}'$ equipped with a choice of equivalences $\mathsf{tup}(F\pi_1, \ldots, F\pi_n) : F(\prod_n(A_1, \ldots A_n)) \leftrightarrows \prod_n (FA_1, \ldots, FA_n) : \mathrm{q}^{\times}_{A_\bullet}$ for each $A_1, \ldots, A_n \in S$ $(n \in \mathbb{N})$.

We call $(F, \mathrm{q}^{\times})$ *strict* if F is a strict pseudofunctor and satisfies

$$\begin{aligned}
F(\textstyle\prod_n(A_1, \ldots, A_n)) &= \textstyle\prod_n(FA_1, \ldots, FA_n) \\
F(\pi_i^{A_1,\ldots,A_n}) &= \pi_i^{FA_1,\ldots,FA_n} \\
F(\mathsf{tup}(t_1, \ldots, t_n)) &= \mathsf{tup}(Ft_1, \ldots, Ft_n) \\
F\varpi^{(i)}_{t_1,\ldots,t_n} &= \varpi^{(i)}_{Ft_1,\ldots,Ft_n} \\
\mathrm{q}^{\times}_{A_1,\ldots,A_n} &= \mathrm{Id}_{\Pi_n(FA_1,\ldots,FA_n)}
\end{aligned}$$

and the equivalences are canonically induced by the 2-cells $\mathrm{Id} \xRightarrow{\cong} \mathsf{tup}(\pi_1[\mathrm{Id}], \ldots, \pi_n[\mathrm{Id}]) \xRightarrow{\cong} \mathsf{tup}(\pi_1, \ldots, \pi_n)$. ◂

If $(F, \mathrm{q}^{\times}) : (S, \mathcal{C}, \Pi_n(-)) \to (S', \mathcal{C}', \Pi_n(-))$ is a cartesian pseudofunctor of biclones, one obtains an fp-pseudofunctor between the associated fp-bicategories by restriction. To complete our diagram of adjunctions (4.36) it remains to construct free cartesian biclones and free fp-bicategories. We begin with the former.

Theorem 4.2.20 presents us with a choice. We can encode either representability (via the universal property (4.30)) or cartesian structure (via the universal property (4.31)). In type-theoretic terms, this amounts to defining the universal property with respect to a pairing operation $x_1 : X_1, \ldots, x_n : X_n \vdash \langle x_1, \ldots, x_n \rangle : \prod_n(X_1, \ldots, X_n)$ or, alternatively, to defining the universal property with respect to projections $(p : \prod_n(X_1, \ldots, X_n) \vdash \pi_i(p) : X_i)_{i=1,\ldots,n}$. We choose the latter because it more closely matches our definition of fp-bicategory.

Construction 4.2.58. For any $\Lambda^{\times}_{\mathrm{ps}}$-signature $\mathcal{S}$, define a cartesian biclone $\mathcal{FC}l^{\times}(\mathcal{S})$ with sorts

$$A_1, \ldots, A_n ::= B \mid \textstyle\prod_n(A_1, \ldots, A_n) \qquad (B \in \mathfrak{B}, n \in \mathbb{N})$$

by extending the construction of the free biclone (Construction 3.1.16) with the following rules:

$$\frac{}{\pi_i^{A_\bullet} \in \mathcal{FC}l^{\times}(\mathcal{S})\,(\prod_n(A_1, \ldots, A_n); A_i)}\;(1 \leqslant i \leqslant n)$$

$$\frac{(t_i \in \mathcal{FC}l^{\times}(\mathcal{S})(\Gamma; A_i))_{i=1,\ldots,n}}{\mathsf{tup}(t_1, \ldots, t_n) \in \mathcal{FC}l^{\times}(\mathcal{S})\,(\Gamma; \prod_n(A_1, \ldots, A_n))}$$

$$\frac{(t_i \in \mathcal{FC}l^{\times}(\mathcal{S})(\Gamma; A_i))_{i=1,\ldots,n}}{\varpi^{(i)}_{t_\bullet} \in \mathcal{FC}l^{\times}(\mathcal{S})\,(\Gamma; A_i)\,(\mathsf{tup}(t_1, \ldots, t_n), t_i)}\;(1 \leqslant i \leqslant n)$$

$$\frac{\big(\alpha_i \in \mathcal{FC}l^{\times}(\mathcal{S})(\Gamma; A_i)(\pi_i^{A_\bullet}[u], t_i)\big)_{i=1,\ldots,n}}{\mathsf{p}^{\dagger}(\alpha_1, \ldots, \alpha_n) \in \mathcal{FC}l^{\times}(\mathcal{S})\,(\Gamma; \prod_n(A_1, \ldots, A_n))\,(u, \mathsf{tup}(t_1, \ldots, t_n))}$$

Moreover, extend the equational theory $\equiv$ of Construction 3.1.16 with the following rules encoding the universal property (4.31):

- If $\alpha_i : u \Rightarrow t_i : \Gamma \to A_i$ for $i = 1, \dots, n$, then $\alpha_i \equiv \varpi^{(i)}_{t_\bullet} \bullet \mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n)$ for $i = 1, \dots, n$,
- If $\gamma : u \Rightarrow \mathsf{tup}(t_1, \dots, t_n) : \Gamma \to \prod_n(A_1, \dots, A_n)$, then $\gamma \equiv \mathsf{p}^\dagger(\varpi^{(1)}_{t_\bullet} \bullet \mathrm{Id}_{\pi_1}[\gamma], \dots, \varpi^{(n)}_{t_\bullet} \bullet \mathrm{Id}_{\pi_n}[\gamma])$,
- If $\alpha_i \equiv \alpha'_i$ for α_i, α'_i 2-cells of type $\pi_i^{A_\bullet}[u] \Rightarrow t_i$ for $i = 1, \dots, n$, then $\mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n) \equiv \mathsf{p}^\dagger(\alpha'_1, \dots, \alpha'_n)$.

Finally, we require that every $\varpi^{(i)}_{t_\bullet}$ and $\varsigma_t := \mathsf{p}^\dagger(\mathrm{Id}_{\pi_1[t]}, \dots, \mathrm{Id}_{\pi_n[t]})$ is invertible. ◂

Lemma 4.2.59. For any $\Lambda^{\times}_{\mathrm{ps}}$-signature $\mathcal{S}$ and any finite family of 2-cells $(\alpha_i : \pi_i\{u\} \Rightarrow t_i : \Gamma \to A_i)_{i=1,\dots,n}$ in $\mathcal{FCl}^\times(\mathcal{S})$, then $\mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n)$ is the unique 2-cell γ (modulo $\equiv$) such that $\alpha_i \equiv \varpi^{(i)}_{t_\bullet} \bullet \gamma$ for $i = 1, \dots, n$.

Proof. The existence part of the claim is immediate. For uniqueness, if γ satisfies the given equation then $\gamma \equiv \mathsf{p}^\dagger(\varpi^{(1)}_{t_\bullet} \bullet \mathrm{Id}_{\pi_1}[\gamma], \dots, \varpi^{(n)}_{t_\bullet} \bullet \mathrm{Id}_{\pi_n}[\gamma]) \equiv \mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n)$, as claimed. □

It follows that $\mathcal{FCl}^\times(\mathcal{S})$ is cartesian. The associated free property is then straightforward.

Lemma 4.2.60. For any $\Lambda^{\times}_{\mathrm{ps}}$-signature $\mathcal{S}$, cartesian biclone $(T, \mathcal{D}, \Pi_n(-))$ and $\Lambda^{\times}_{\mathrm{ps}}$-signature homomorphism $h : \mathcal{S} \to \mathcal{D}$ from $\mathcal{S}$ to the $\Lambda^{\times}_{\mathrm{ps}}$-signature underlying $(T, \mathcal{D}, \Pi_n(-))$ there exists a strict cartesian pseudofunctor $h^\# : \mathcal{FCl}^\times(\mathcal{S}) \to \mathcal{D}$, unique such that $h^\# \circ \iota = h$, for $\iota : \mathcal{S} \hookrightarrow \mathcal{FCl}^\times(\mathcal{S})$ the inclusion.

Proof. We extend the pseudofunctor $h^\#$ defined in Lemma 3.1.17 by setting

$$\begin{aligned} h^\#(\textstyle\prod_n(A_1, \dots, A_n)) &:= \textstyle\prod_n\big(h^\#(A_1), \dots, h^\#(A_n)\big) \\ h^\#(\pi_i^{A_\bullet}) &:= \pi_i^{h^\#(A_\bullet)} \\ h^\#(\mathsf{tup}(t_1, \dots, t_n)) &:= \mathsf{tup}(h^\#(t_1), \dots, h^\#(t_n)) \\ h^\#(\varpi^{(i)}_{t_\bullet}) &:= \varpi^{(i)}_{h^\#(t_\bullet)} \\ h^\#\big(\mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n)\big) &:= \mathsf{p}^\dagger(h^\#(\alpha_1), \dots, h^\#(\alpha_n)) \end{aligned}$$

It is clear this defines a strict cartesian pseudofunctor. For uniqueness, all the cases apart from $\mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n)$ are determined by the definition of strict cartesian pseudofunctor. To complete the proof, we adapt the argument of Lemma 2.2.17. For any strict cartesian pseudofunctor $F : \mathcal{FCl}^\times(\mathcal{S}) \to \mathcal{D}$ and 2-cells $(\alpha_i : \pi_i^{A_\bullet}[u] \Rightarrow t_i : \Gamma \to A_i)_{i=1,\dots,n}$,

$$\begin{aligned} \varpi^{(i)}_{Ft_\bullet} \bullet F\big(\mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n)\big) &= F(\varpi^{(i)}_{t_\bullet}) \bullet F\big(\mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n)\big) \\ &= F\Big(\varpi^{(i)}_{Ft_\bullet} \bullet \mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n)\Big) \\ &= F\alpha_i \end{aligned}$$

for $i = 1, \dots, n$. Hence, by the universal property (4.31) of a cartesian biclone, $F\big(\mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n)\big) = \mathsf{p}^\dagger(F\alpha_1, \dots, F\alpha_n)$. □

Remark 4.2.61. The preceding proof should be compared to that for the free cartesian clone on a $\Lambda^{\times}$-signature (Lemma 4.2.28). The argument for uniqueness lifts to 2-cells by virtue of the fact that pseudofunctors strictly preserve vertical composition. ◂

It remains to construct the free fp-bicategory on a unary $\Lambda^{\times}$-signature and relate it to the free cartesian biclone over the same signature. The proof is straightforward: one restricts Lemma 4.2.60 to unary multimaps and observes the same universal property holds. Example 4.2.63 shows that it is important to restrict every rule to unary multimaps—*i.e.* require that $|\Gamma| = 1$ for every rule in Construction 4.2.58—rather than simply taking the nucleus of $\mathcal{FC}l^{\times}(\mathcal{S})$.

Lemma 4.2.62. For any unary $\Lambda^{\times}_{\mathrm{ps}}$-signature $\mathcal{S}$, let $\mathcal{FB}ct^{\times}(\mathcal{S})$ denote the fp-bicategory obtained by restricting every rule of Construction 4.2.58 to unary multimaps and 2-cells between them, and let $h : \mathcal{S} \to \mathcal{C}$ be a $\Lambda^{\times}_{\mathrm{ps}}$-signature homomorphism from $\mathcal{S}$ to the $\Lambda^{\times}_{\mathrm{ps}}$-signature underlying an fp-bicategory $(\mathcal{C}, \Pi_n(-))$. Then there exists a strict fp-pseudofunctor $h^{\#} : \mathcal{FB}ct^{\times}(\mathcal{S}) \to \mathcal{C}$, unique such that $h^{\#} \circ \iota = h$, for $\iota : \mathcal{S} \hookrightarrow \mathcal{FB}ct^{\times}(\mathcal{S})$ the inclusion. □

Example 4.2.63. Fix a $\Lambda^{\times}_{\mathrm{ps}}$-signature $\mathcal{S} = (\mathfrak{B}, \mathcal{G})$. Then the nucleus $\overline{\mathcal{FC}l^{\times}(\mathcal{S})}$ of $\mathcal{FC}l^{\times}(\mathcal{S})$ is not isomorphic to $\mathcal{FB}ct^{\times}(\mathcal{S})$. Roughly speaking, the composite $\mathsf{p}^{(1)}_{A,B}[\pi_1, \pi_2] : A \times B \to A$ exists in the free cartesian biclone on a signature $\mathcal{S}$, but not in the free fp-bicategory on $\mathcal{S}$. Let us make this precise.

Since the freeness universal property of $\mathcal{FB}ct^{\times}(\mathcal{S})$ is strict we may exploit the following principle, which restates the fact that free objects are unique up to *canonical* isomorphism: if $\mathcal{B}$ and $\mathcal{B}'$ are both the free fp-bicategory on $\mathcal{S}$, then the canonical map $\mathcal{B} \to \mathcal{B}'$ extending the unit is an isomorphism. We claim that the canonical map $\iota^{\#} : \mathcal{FB}ct^{\times}(\mathcal{S}) \to \overline{\mathcal{FC}l^{\times}(\mathcal{S})}$ extending the inclusion $\iota : \mathcal{S} \hookrightarrow \overline{\mathcal{FC}l^{\times}(\mathcal{S})}$ is not an isomorphism. Since an isomorphism is necessarily a bijection on hom-sets, it suffices to find a morphism in $\overline{\mathcal{FC}l^{\times}(\mathcal{S})}$ that is not in the image of $\iota^{\#}$. We claim that, where $X, Y \in \widetilde{\mathfrak{B}}$, then $\mathsf{p}^{(1)}_{X,Y}[\pi_1, \pi_2] : X \times Y \to X$ is not in the image of $\iota^{\#}$. To see this is the case, observe that a morphism h is in the image of $\iota^{\#}$ if and only if it falls into one of the following (disjoint) sets:

1. The *basic maps* π_i, eval and Id,
2. Maps in the image of an *operator*: λf or $\langle f_1, \dots, f_n \rangle$ for $f, f_1, \dots, f_n$ in the image of $\iota^{\#}$,
3. The *composites* $f \circ g$ where f and g are both in the image of $\iota^{\#}$.

It is clear that $\mathsf{p}^{(1)}_{X,Y}[\pi_1, \pi_2]$ is not of any of these types, and so is not in the image of $\iota^{\#}$. It follows that $\iota^{\#}$ is not an isomorphism, and hence that $\overline{\mathcal{FC}l^{\times}(\mathcal{S})}$ is not the free fp-bicategory on $\mathcal{S}$. ◂

Lemma 4.2.62 guarantees that the free fp-bicategory on a $\Lambda^{\times}_{\mathrm{ps}}$-signature $\mathcal{S}$ arises by restricting every rule of the type theory for cartesian biclones to unary contexts and constructing the syntactic model. Hence, it suffices to construct a type theory for cartesian biclones. We do this by extending the type theory $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}$ for biclones with rules corresponding to those of Construction 4.2.58.

4.3 The type theory $\Lambda^{\times}_{\mathrm{ps}}$

For a $\Lambda^{\times}_{\mathrm{ps}}$-signature $\mathcal{S} = (\mathfrak{B}, \mathcal{G})$ we denote the associated type theory by $\Lambda^{\times}_{\mathrm{ps}}(\mathcal{S})$. The types of $\Lambda^{\times}_{\mathrm{ps}}(\mathcal{S})$ are the nodes of $\mathcal{G}$. The rules are all those of $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}$ together with those of Figures 4.1–4.4. Note that we specify the invertibility of the unit and counit by introducing explicit inverses for these rewrites (Figure 4.4).

The tupling operation is functorial with respect to vertical composition and the unit of the adjunction is obtained by applying the universal property to the identity (see also Lemma 4.3.12).

Definition 4.3.1.

1. For any family of derivable rewrites $(\Gamma \vdash \tau_i : t_i \Rightarrow t'_i : A_i)_{i=1,\dots,n}$ we define $\mathsf{tup}(\tau_1, \dots, \tau_n) : \mathsf{tup}(t_1, \dots, t_n) \Rightarrow \mathsf{tup}(t'_1, \dots, t'_n)$ to be the rewrite $\mathsf{p}^{\dagger}(\tau_1 \bullet \varpi^{(1)}_{t_1,\dots,t_n}, \dots, \tau_n \bullet \varpi^{(n)}_{t_1,\dots,t_n})$ in context Γ.
2. For any derivable term $\Gamma \vdash t : \prod_n(A_1, \dots, A_n)$ we define the unit $\varsigma_t : t \Rightarrow \mathsf{tup}(\pi_1\{t\}, \dots, \pi_n\{t\})$ to be the rewrite $\mathsf{p}^{\dagger}(\mathrm{id}_{\pi_1\{t\}}, \dots, \mathrm{id}_{\pi_n\{t\}})$ in context Γ. ◂

The rules of $\Lambda^{\times}_{\mathrm{ps}}$ provide a relatively compact way to construct the structure required for cartesian clones. In particular, the focus on (global) biuniversal arrows and (local) universal arrows—and the corresponding fact that one does not need to specify a triangle law relating the unit and counit—contrasts with all previous work on type theories for cartesian closed 2-categories [See87, Hil96, Tab11, Hir13], which encode the pairing and projection operations on rewrites directly. Reproducing the triangle-law approach in the context of fp-bicategories would require:

1. For every sequence of types $A_1, \dots, A_n$ a product type $\prod_n(A_1, \dots, A_n)$,
2. Projection and tupling operations on terms as in the usual simply-typed lambda calculus,
3. Tupling and projection operations on rewrites,
4. An invertible unit $\varsigma_u : u \Rightarrow \langle \pi_1(u), \dots, \pi_n(u) \rangle$ in context Γ for every $\Gamma \vdash u : \prod_n(A_1, \dots, A_n)$ and an invertible counit $\varpi^{(i)}_{t_\bullet} : \pi_i\{\langle t_1, \dots, t_n \rangle\} \Rightarrow t_i$ $(i = 1, \dots, n)$ in context Γ for every $(\Gamma \vdash t_i : A_i)_{i=1,\dots,n}$.

This data must be subject to an equational theory requiring naturality of each ς_u and $\varpi^{(i)}_{t_\bullet}$, the two triangle laws, functorality of the tupling and projection operations on rewrites, and that the equational theory is a congruence with respect to these operations. Such an approach, therefore, requires many more rules. Moreover, the calculus of (bi)universal arrows provided by $\Lambda^{\times}_{\mathrm{ps}}$ captures a categorical style of reasoning, because the syntax allows one to manipulate the universal property through primitives in the type theory.

α-equivalence and free variables. The well-formedness properties of $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$ extend to $\Lambda_{\mathrm{ps}}^{\times}$; we briefly note them here. As we have not introduced any binding constructs, the definition of α-equivalence extends straightforwardly from that for $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$.

Definition 4.3.2. For any $\Lambda_{\mathrm{ps}}^{\times}$-signature $\mathcal{S}$ we extend Definition 3.2.4 to define the *α-equivalence relation* $=_\alpha$ for $\Lambda_{\mathrm{ps}}^{\times}(\mathcal{S})$. For terms we take the same set of rules; the substitution operation $t[u_i/x_i]$ is extended by the rules

$$\pi_k(p)[u/p] := \pi_k\{u\} \quad \text{and} \quad \mathsf{tup}(t_1, \dots, t_n)[u_i/x_i] := \mathsf{tup}(t_1[u_i/x_i], \dots, t_n[u_i/x_i])$$

For rewrites, we add the rules

$$\frac{(t_i =_\alpha t_i')_{i=1,\dots,n}}{\varpi^{(k)}_{t_1,\dots,t_n} =_\alpha \varpi^{(k)}_{t_1',\dots,t_n'}}\ (1 \leqslant k \leqslant n) \qquad \frac{\sigma_1 =_\alpha \sigma_1' \quad \dots \quad \sigma_n =_\alpha \sigma_n'}{\mathsf{p}^\dagger(\sigma_1, \dots, \sigma_n) =_\alpha \mathsf{p}^\dagger(\sigma_1', \dots, \sigma_n')}$$

where the meta-operation of capture-avoiding substitution is extended by the rules

$$\varpi^{(k)}_{t_1,\dots,t_n}[u_i/x_i] := \varpi^{(k)}_{t_1[u_i/x_i],\dots,t_n[u_i/x_i]} \quad \text{and} \quad \mathsf{p}^\dagger(\alpha_\bullet)[u_i/x_i] := \mathsf{p}^\dagger(\alpha_\bullet[u_i/x_i])$$

Finally, we define $\mathrm{fv}(\sigma^{-1}) := \mathrm{fv}(\sigma)$. ◂

As for $\Lambda_{\mathrm{ps}}^{\mathrm{bicat}}$, we work up to α-equivalence of terms and rewrites, silently identifying terms and rewrites with their α-equivalence classes.

Extending the definition of free variables is similarly straightforward.

Definition 4.3.3. Fix a $\Lambda_{\mathrm{ps}}^{\times}$-signature $\mathcal{S}$. We define the *free variables in a term t* in $\Lambda_{\mathrm{ps}}^{\times}(\mathcal{S})$ by extending Definition 3.2.9 as follows:

$$\mathrm{fv}\big(\mathsf{tup}(t_1, \dots, t_n)\big) := \textstyle\bigcup_{i=1}^n \mathrm{fv}(t_i) \quad \text{and} \quad \mathrm{fv}\big(\pi_k(p)\big) := \{p\}$$

Define the *free variables in a rewrite τ* in $\Lambda_{\mathrm{ps}}^{\times}(\mathcal{S})$ by extending Definition 3.2.9 as follows:

$$\mathrm{fv}(\varpi^{(k)}_{t_1,\dots,t_n}) := \mathrm{fv}(t_k) \quad \text{and} \quad \mathrm{fv}\big(\mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n)\big) := \textstyle\bigcup_{i=1}^n \mathrm{fv}(\alpha_i)$$

We define the free variables of a specified inverse σ^{-1} to be exactly the free variables of σ. An occurrence of a variable in a term (resp. rewrite) is *bound* if it is not free. ◂

The next two lemmas—both of which are proven by structural induction—show that the preceding definitions behave in the way one would expect.

Lemma 4.3.4. Let $\mathcal{S}$ be a $\Lambda_{\mathrm{ps}}^{\times}$-signature. Then in $\Lambda_{\mathrm{ps}}^{\times}(\mathcal{S})$:

1. If $\Gamma \vdash t : B$ and $t =_{\alpha} t'$ then $\Gamma \vdash t' : B$,
2. If $\Gamma \vdash \tau : t \Rightarrow t' : B$ and $\tau =_{\alpha} \tau'$ then $\Gamma \vdash \tau' : t \Rightarrow t' : B$,
3. If $\tau_i =_{\alpha} \tau_i'$ for $i = 1, \ldots, n$, then $\mathsf{tup}(\tau_1, \ldots, \tau_n) =_{\alpha} \mathsf{tup}(\tau_1', \ldots, \tau_n')$,
4. If $u =_{\alpha} u'$ then $\varsigma_u =_{\alpha} \varsigma_{u'}$. □

Lemma 4.3.5. Let $\mathcal{S}$ be a $\Lambda_{\mathrm{ps}}^{\times}$-signature. For any derivable judgements $\Gamma \vdash u : B$ and $\Gamma \vdash \tau : t \Rightarrow t' : B$ in $\Lambda_{\mathrm{ps}}^{\times}(\mathcal{S})$,

1. $\mathrm{fv}(u) \subseteq \mathrm{dom}(\Gamma)$,
2. $\mathrm{fv}(\tau) \subseteq \mathrm{dom}(\Gamma)$,
3. The judgements $\Gamma \vdash t : B$ and $\Gamma \vdash t' : B$ are both derivable.

Moreover, whenever $(\Delta \vdash u_i : A_i)_{i=1,\ldots,n}$ and $\Gamma := (x_i : A_i)_{i=1,\ldots,n}$, then

1. If $\Gamma \vdash t : B$, then $\Delta \vdash t[u_i/x_i] : B$,
2. If $\Gamma \vdash \tau : t \Rightarrow t' : B$, then $\Delta \vdash \tau[u_i/x_i] : t[u_i/x_i] \Rightarrow t'[u_i/x_i] : B$. □

4.3.1 The syntactic model for $\Lambda_{\mathrm{ps}}^{\times}$

Lemma 4.2.62 guarantees that, in order to construct a type theory for fp-bicategories, it suffices to construct a type theory for cartesian biclones. To verify that $\Lambda_{\mathrm{ps}}^{\times}$ is such a type theory, furthermore, it suffices to show that its syntactic model is canonically isomorphic to the free cartesian biclone $\mathcal{FC}l^{\times}(\mathcal{S})$ over the same signature in the category CartBiclone.

The syntactic model is constructed by extending Construction 3.2.11.

Construction 4.3.6. For any $\Lambda_{\mathrm{ps}}^{\times}$-signature $\mathcal{S}$ define the *syntactic model* $\mathrm{Syn}^{\times}(\mathcal{S})$ of $\Lambda_{\mathrm{ps}}^{\times}(\mathcal{S})$ as follows. The sorts are nodes $A, B, \ldots$ of $\mathcal{G}$. For $A_1, \ldots, A_n, B \in \mathfrak{B}$ $(n \in \mathbb{N})$ the hom-category $\mathrm{Syn}^{\times}(\mathcal{S})(A_1, \ldots, A_n; B)$ has objects α-equivalence classes of terms $(x_1 : A_1, \ldots, x_n : A_n \vdash t : B)$ derivable in $\Lambda_{\mathrm{ps}}^{\times}(\mathcal{S})$. We assume a fixed enumeration $x_1, x_2, \ldots$ of variables, and that the variable name in the ith position is determined by this enumeration. Morphisms in $\mathrm{Syn}^{\times}(\mathcal{S})(A_1, \ldots, A_n; B)$ are $\alpha\equiv$-equivalence classes of rewrites $(x_1 : A_1, \ldots, x_n : A_n \vdash \tau : t \Rightarrow t' : B)$. Composition is vertical composition with identity id_t; the substitution operation is explicit substitution and the structural rewrites are assoc, ι and $\varrho^{(i)}$. ◂

Inspecting each rule in turn, one sees that $\mathrm{Syn}^{\times}(\mathcal{S})$ is merely $\mathcal{FC}l^{\times}(\mathcal{S})$, presented with the notation $x_1 : X_1, \ldots, x_n : X_n \vdash t : B$ instead of $t : X_1, \ldots, X_n \to B$. We make this statement precise by establishing it satisfies the same universal property.

Lemma 4.2.59, restated in type-theoretic notation, becomes the following.

Lemma 4.3.7. For any $\Lambda^{\times}_{\mathrm{ps}}$-signature $\mathcal{S}$, if the judgements $(\Gamma \vdash \alpha_i : \pi_i\{u\} \Rightarrow t_i : A_i)_{i=1,\dots,n}$ are derivable in $\Lambda^{\times}_{\mathrm{ps}}(\mathcal{S})$ then $\mathsf{p}^{\dagger}(\alpha_1, \dots, \alpha_n)$ is the unique rewrite γ (modulo $\alpha\equiv$) such that the equality

$$\Gamma \vdash \varpi^{(k)}_{t_1,\dots,t_n} \bullet \pi_k\{\gamma\} \equiv \alpha_k : \pi_i\{u\} \Rightarrow t_k : A_k \tag{4.37}$$

is derivable for $k = 1, \dots, n$.

Proof. By `U1` (Figure 4.3) the rewrite $\mathsf{p}^{\dagger}(\alpha_1, \dots, \alpha_n)$ certainly satisfies (4.37). For any other γ satisfying the equation, $\gamma \overset{\texttt{U2}}{\equiv} \mathsf{p}^{\dagger}(\varpi^{(1)}_{t_\bullet} \bullet \pi_1\{\gamma\}, \dots, \varpi^{(n)}_{t_\bullet} \bullet \pi_n\{\gamma\}) \overset{\text{cong}}{\equiv} \mathsf{p}^{\dagger}(\alpha_1, \dots, \alpha_n)$, as claimed. □

Remark 4.3.8. In the light of the preceding lemma, for any $\Lambda^{\times}_{\mathrm{ps}}$-signature $\mathcal{S}$ the mappings

$$\begin{aligned} (\alpha_1, \dots, \alpha_n) &\mapsto \mathsf{p}^{\dagger}(\alpha_1, \dots, \alpha_n) \\ (\varpi^{(1)}_{t_\bullet} \bullet \pi_1\{\tau\}, \dots, \varpi^{(n)}_{t_\bullet} \bullet \pi_n\{\tau\}) &\mapsfrom \tau \end{aligned}$$

define the following bijective correspondence of rewrites, derivable in $\Lambda^{\times}_{\mathrm{ps}}(\mathcal{S})$:

$$\frac{\pi_k\{u\} \Rightarrow t_k \quad (k = 1, \dots, n)}{u \Rightarrow \mathsf{tup}(t_1, \dots, t_n)}$$

It is natural to conjecture that a calculus for fp-*tri*categories (resp. fp-∞-categories) would have three (resp. a countably infinite tower of) such correspondences. Similar considerations will apply to exponentials. ◂

It also follows from the preceding lemma that $\mathrm{Syn}^{\times}(\mathcal{S})$ is cartesian: the adjoint equivalence is exactly

$$\begin{aligned} \mathrm{Syn}^{\times}(\mathcal{S})\big(\Gamma, \textstyle\prod_n(A_1, \dots, A_n)\big) &\xrightarrow{\simeq} \textstyle\prod_{i=1}^{n} \mathrm{Syn}^{\times}(\mathcal{S})(\Gamma; A_i) \\ \big(\Gamma \vdash u : \textstyle\prod_n(A_1, \dots, A_n)\big) &\mapsto (\Gamma \vdash \pi_i\{u\} : A_i)_{i=1,\dots,n} \end{aligned}$$

where the pseudoinverse $\prod_{i=1}^{n} \mathrm{Syn}^{\times}(\mathcal{S})(\Gamma; A_i) \to \mathrm{Syn}^{\times}(\mathcal{S})\big(\Gamma, \prod_n(A_1, \dots, A_n)\big)$ is the tup operation. The universal property of $\mathrm{Syn}^{\times}(\mathcal{S})$ interprets each term as its corresponding construct.

Proposition 4.3.9. For any $\Lambda^{\times}_{\mathrm{ps}}$-signature $\mathcal{S} = (\mathfrak{B}, \mathcal{G})$, cartesian biclone $(T, \mathbb{D}, \Pi_n(-))$ and $\Lambda^{\times}_{\mathrm{ps}}$-signature homomorphism $h : \mathcal{S} \to \mathcal{C}$, there exists a unique strict cartesian pseudofunctor $h[\![-]\!] : \mathrm{Syn}^{\times}(\mathcal{S}) \to \mathcal{C}$ such that $h[\![-]\!] \circ \iota = h$, for $\iota : \mathcal{S} \hookrightarrow \mathrm{Syn}^{\times}(\mathcal{S})$ the inclusion.

Proof. The pseudofunctor is constructed by induction on the syntax of $\Lambda^{\times}_{\mathrm{ps}}(\mathcal{S})$ as follows:

$$\begin{aligned} h[\![B]\!] &:= h(B) \qquad \text{on base types} \\ h[\![\textstyle\prod_m(B_1, \dots, B_m)]\!] &:= \textstyle\prod_m (h[\![B_1]\!], \dots, h[\![B_m]\!]) \\ h[\![\Gamma \vdash x_k : A_i]\!] &:= \mathsf{p}^{(k)}_{h[\![A_1]\!], \dots, h[\![A_n]\!]} \end{aligned}$$

$$\begin{aligned}
h\llbracket \Gamma \vdash c(x_1, \dots, x_n) : B \rrbracket &:= h(c) \qquad \text{for } c \in \mathcal{G}(A_\bullet; B) \\
h\llbracket \Delta \vdash t\{x_i \mapsto u_i\} : B \rrbracket &:= \big(h\llbracket \Gamma \vdash t : B \rrbracket\big)[h\llbracket \Delta \vdash u_\bullet : A_\bullet \rrbracket] \\
h\llbracket \Gamma \vdash \mathsf{tup}(t_1, \dots, t_m) : \textstyle\prod_m(B_1, \dots, B_m) \rrbracket &:= \mathsf{tup}(h\llbracket \Gamma \vdash t_1 : B_1 \rrbracket, \dots, h\llbracket \Gamma \vdash t_m : B_m \rrbracket) \\
h\llbracket p : \textstyle\prod_m(B_1, \dots, B_m) \vdash \pi_k(p) : B_k \rrbracket &:= \pi_k^{h\llbracket B_1 \rrbracket, \dots, h\llbracket B_m \rrbracket}
\end{aligned}$$

$$\begin{aligned}
h\llbracket \Gamma \vdash \mathrm{id}_t : t \Rightarrow t : B \rrbracket &:= \mathrm{id}_{h\llbracket \Gamma \vdash t : B \rrbracket} \\
h\llbracket \Gamma \vdash \kappa(x_\bullet) : c(x_\bullet) \Rightarrow c'(x_\bullet) : B \rrbracket &:= h(\kappa) \qquad \text{for } \kappa \in \mathcal{G}(A_\bullet, B)(c, c') \\
h\llbracket \Gamma \vdash \varpi^{(k)}_{t_1, \dots, t_m} : \pi_k\{\mathsf{tup}(t_1, \dots, t_m)\} \Rightarrow t_k : B_k \rrbracket &:= \varpi^{(k)}_{h\llbracket t_1 \rrbracket, \dots, h\llbracket t_m \rrbracket} \\
h\llbracket \Gamma \vdash \mathsf{p}^\dagger(\alpha_1, \dots, \alpha_m) : u \Rightarrow \mathsf{tup}(t_\bullet) : \textstyle\prod_m B_\bullet \rrbracket &:= \mathsf{p}^\dagger(h\llbracket \Gamma \vdash \alpha_\bullet : \pi_\bullet\{u\} \Rightarrow t_\bullet : B_\bullet \rrbracket) \\
h\llbracket \Gamma \vdash \tau' \bullet \tau : t \Rightarrow t'' : B \rrbracket &:= h\llbracket \Gamma \vdash \tau' : t' \Rightarrow t'' : B \rrbracket \bullet h\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket \\
h\llbracket \Delta \vdash \tau\{\sigma_i\} : t\{u_i\} \Rightarrow t'\{u_i'\} : B \rrbracket &:= \big(h\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket\big)[h\llbracket \sigma_1 \rrbracket, \dots, h\llbracket \sigma_n \rrbracket]
\end{aligned}$$

where $\Gamma := (x_i : A_i)_{i=1,\dots,n}$ and we abbreviate $h\llbracket \Delta \vdash \sigma_i : u_i \Rightarrow u_i' : A_i \rrbracket$ by $h\llbracket \sigma_i \rrbracket$ in the final rule. It is clear that this defines a strict pseudofunctor; the $\mathsf{p}^\dagger(\alpha_1, \dots, \alpha_m)$ case is required by the strict preservation of universal and biuniversal arrows (*c.f.* Lemma 4.2.60). □

Lemma 4.2.62, together with the preceding proposition, entail that the free fp-bicategory on a unary $\Lambda_{\mathrm{ps}}^{\times}$-signature is obtained as follows. First, one restricts $\Lambda_{\mathrm{ps}}^{\times}$ to unary contexts. Then one constructs the syntactic model in the same manner as Construction 4.3.6, except morphisms and 2-cells are equivalence classes of terms and rewrites in this restricted type theory. Thus, define $\Lambda_{\mathrm{ps}}^{\times}\big|_1$ to be the type theory obtained by restricting $\Lambda_{\mathrm{ps}}^{\times}$ to contexts of the form $x : A$ (defined by Figure 3.12 on page 61. The resulting free property is the following.

Theorem 4.3.10. For any unary $\Lambda_{\mathrm{ps}}^{\times}$-signature $\mathcal{S}$, the bicategory $\mathrm{Syn}^{\times}(\mathcal{S})\big|_1$ constructed by restricting Construction 4.3.6 to the type theory $\Lambda_{\mathrm{ps}}^{\times}\big|_1$ is the free fp-bicategory on $\mathcal{S}$, in the sense of Lemma 4.2.62.

Proof. For any fp-bicategory $(\mathcal{C}, \Pi_n(-))$ and $\Lambda_{\mathrm{ps}}^{\times}$-signature homomorphism $h : \mathcal{S} \to \mathcal{C}$ the extension fp-pseudofunctor $h^{\#} : \mathrm{Syn}^{\times}(\mathcal{S})\big|_1 \to \mathcal{C}$ is defined inductively as in Proposition 4.3.9, with the following adjustments:

$$\begin{aligned}
h\llbracket x : A \vdash x : A \rrbracket &:= \mathrm{Id}_{h\llbracket A \rrbracket} \\
h\llbracket z : Z \vdash t\{x \mapsto u\} : B \rrbracket &:= h\llbracket x : A \vdash t : B \rrbracket \circ h\llbracket z : Z \vdash u : A \rrbracket \\
h\llbracket x : A \vdash \mathsf{tup}(t_\bullet) : \textstyle\prod_m(B_1, \dots, B_m) \rrbracket &:= \langle h\llbracket x : A \vdash t_1 : B_1 \rrbracket, \dots, h\llbracket x : A \vdash t_m : B_m \rrbracket \rangle \\
h\llbracket z : Z \vdash \tau\{\sigma\} : t\{u\} \Rightarrow t'\{u'\} : B \rrbracket &:= h\llbracket x : A \vdash \tau : t \Rightarrow t' : B \rrbracket \circ h\llbracket z : Z \vdash \sigma : u \Rightarrow u' : A \rrbracket
\end{aligned}$$

□

Remark 4.3.11. As with the construction of $\mathcal{FB}ct^{\times}(\mathcal{S})$, it is important that we first restrict $\Lambda^{\times}_{\mathrm{ps}}$ to unary contexts, then construct the syntactic model (recall Example 4.2.63). ◂

In the semantics of the simply-typed lambda calculus it is common to restrict the syntactic model to unary contexts in order to achieve the desired universal property (see *e.g.* [Cro94, Chapter 4]). Hence, we are still justified in calling $\Lambda^{\times}_{\mathrm{ps}}$ the internal language of fp-bicategories.

4.3.2 Reasoning within $\Lambda^{\times}_{\mathrm{ps}}$

In later chapters we shall reason within $\Lambda^{\times}_{\mathrm{ps}}$—and its extension $\Lambda^{\times,\to}_{\mathrm{ps}}$ for cartesian closed bicategories—to prove various properties of the syntactic models and their semantic interpretation. We collect together some results to simplify such calculations.

All the rules of the triangle-law approach to defining products are derivable. For example, from Lemma 4.3.7 one recovers the functoriality of the tupling operation and the unit-counit presentation of products (see Figure 4.5). These derived rules should be compared to the primitive rules of [See87, Hil96].

Lemma 4.3.12. For any $\Lambda^{\times}_{\mathrm{ps}}$-signature $\mathcal{S}$, the rules of Figure 4.5 are all admissible.

Proof. The proofs are all similar; we prove naturality of ς as an example of equational reasoning in $\Lambda^{\times}_{\mathrm{ps}}(\mathcal{S})$. One can either use the universal property (Lemma 4.3.7) or reason directly using both the equational rules `U1` and `U2`. We opt for the former. Let $\Gamma \vdash \sigma : u \Rightarrow u' : \prod_n(A_1, \ldots, A_n)$ be any rewrite. Then for $k = 1, \ldots, n$:

$$\begin{aligned} \varpi^{(k)}_{\pi_\bullet u'} \bullet \pi_k\{\varsigma_{u'} \bullet \sigma\} &\equiv \varpi^{(k)}_{\pi_\bullet u'} \bullet \pi_k\{\varsigma_{u'}\} \bullet \pi_k\{\sigma\} \\ &\overset{\texttt{U1}}{\equiv} \mathrm{id}_{\pi_k\{u\}} \bullet \pi_k\{\sigma\} \\ &\equiv \pi_k\{\sigma\} \end{aligned}$$

$$\begin{aligned} \varpi^{(k)}_{\pi_\bullet u'} \bullet \pi_k\{\mathsf{tup}(\pi_1\{\sigma\}, \ldots, \pi_n\{\sigma\}) \bullet \varsigma_u\} &\equiv \varpi^{(k)}_{\pi_\bullet u'} \bullet \pi_k\{\mathsf{tup}(\pi_1\{\sigma\}, \ldots, \pi_n\{\sigma\})\} \bullet \pi_k\{\varsigma_u\} \\ &\overset{\texttt{U1}}{\equiv} \pi_k\{\sigma\} \bullet \varpi^{(k)}_{\pi_\bullet\{u\}} \bullet \pi_k\{\varsigma_u\} \\ &\equiv \pi_k\{\sigma\} \end{aligned}$$

Applying the universal property of $\mathsf{p}^{\dagger}(\pi_1\{\sigma\}, \ldots, \pi_n\{\sigma\})$, one sees that $\varsigma_{u'} \bullet \sigma \equiv \mathsf{tup}(\pi_1\{\sigma\}, \ldots, \pi_n\{\sigma\})$, as required. □

$$\frac{}{p : \prod_n(A_1, \dots, A_n) \vdash \pi_k(p) : A_k}\;k\text{-proj } (1 \leqslant k \leqslant n)$$

$$\frac{\Gamma \vdash t_1 : A_1 \quad \dots \quad \Gamma \vdash t_n : A_n}{\Gamma \vdash \mathsf{tup}(t_1, \dots, t_n) : \prod_n(A_1, \dots, A_n)}\;n\text{-tuple}$$

Figure 4.1: Terms for product structure

$$\frac{\Gamma \vdash t_1 : A_1 \qquad \dots \qquad \Gamma \vdash t_n : A_n}{\Gamma \vdash \varpi^{(k)}_{t_1,\dots,t_n} : \pi_k\{\mathsf{tup}(t_1, \dots, t_n)\} \Rightarrow t_k : A_k}\;\varpi^{(k)}\text{-intro } (1 \leqslant k \leqslant n)$$

$$\frac{\Gamma \vdash u : \prod_n(A_1, \dots, A_n) \qquad (\Gamma \vdash \alpha_i : \pi_i\{u\} \Rightarrow t_i : A_i)_{i=1,\dots,n}}{\Gamma \vdash \mathsf{p}^{\dagger}(\alpha_1, \dots, \alpha_n) : u \Rightarrow \mathsf{tup}(t_1, \dots, t_n) : \prod_n(A_1, \dots, A_n)}\;\mathsf{p}^{\dagger}(\alpha_1, \dots, \alpha_n)\text{-intro}$$

Figure 4.2: Rewrites for product structure

$$\frac{\Gamma \vdash \alpha_1 : \pi_1\{u\} \Rightarrow t_1 : A_1 \quad \dots \quad \Gamma \vdash \alpha_n : \pi_n\{u\} \Rightarrow t_n : A_n}{\Gamma \vdash \alpha_k \equiv \varpi^{(k)}_{t_1,\dots,t_n} \bullet \pi_k\big\{\mathsf{p}^{\dagger}(\alpha_1, \dots, \alpha_n)\big\} : \pi_k\{u\} \Rightarrow t_k : A_k}\;\text{U1 } (1 \leqslant k \leqslant n)$$

$$\frac{\Gamma \vdash \gamma : u \Rightarrow \mathsf{tup}(t_1, \dots, t_n) : \prod_n(A_1, \dots, A_n)}{\Gamma \vdash \gamma \equiv \mathsf{p}^{\dagger}(\varpi^{(1)}_{t_\bullet} \bullet \pi_1\{\gamma\}, \dots, \varpi^{(n)}_{t_\bullet} \bullet \pi_n\{\gamma\}) : u \Rightarrow \mathsf{tup}(t_1, \dots, t_n) : \prod_n(A_1, \dots, A_n)}\;\text{U2}$$

$$\frac{\big(\Gamma \vdash \alpha_i \equiv \alpha'_i : \pi_i\{u\} \Rightarrow t_i : A_i\big)_{i=1,\dots,n}}{\Gamma \vdash \mathsf{p}^{\dagger}(\alpha_1, \dots, \alpha_n) \equiv \mathsf{p}^{\dagger}(\alpha'_1, \dots, \alpha'_n) : u \Rightarrow \mathsf{tup}(t_1, \dots, t_n) : \prod_n(A_1, \dots, A_n)}\;\text{cong}$$

Figure 4.3: Universal property and congruence laws for $\mathsf{p}^{\dagger}(\alpha_1, \dots, \alpha_n)$

$$\frac{\Gamma \vdash t_1 : A_1 \qquad \dots \qquad \Gamma \vdash t_n : A_n}{\Gamma \vdash \varpi^{(-k)}_{t_1,\dots,t_n} : t_k \Rightarrow \pi_k\{\mathsf{tup}(t_1, \dots, t_n)\} : A_k}\;\varpi^{(-k)}\text{-intro } (1 \leqslant k \leqslant n)$$

$$\frac{\Gamma \vdash t : \prod_n(A_1, \dots, A_n)}{\Gamma \vdash \varsigma_t^{-1} : \mathsf{tup}(\pi_1\{t\}, \dots, \pi_n\{t\}) \Rightarrow t : \prod_n(A_1, \dots, A_n)}\;\varsigma^{-1}\text{-intro}$$

$$\frac{\Gamma \vdash t_1 : A_1 \qquad \dots \qquad \Gamma \vdash t_n : A_n}{\Gamma \vdash \varpi^{(-k)}_{t_1,\dots,t_n} \bullet \varpi^{(k)}_{t_1,\dots,t_n} \equiv \mathrm{id}_{\pi_k\{\mathsf{tup}(t_1,\dots,t_n)\}} : \pi_k\{\mathsf{tup}(t_1, \dots, t_n)\} \Rightarrow \pi_k\{\mathsf{tup}(t_1, \dots, t_n)\} : A_k}$$

$$\frac{\Gamma \vdash t_1 : A_1 \qquad \dots \qquad \Gamma \vdash t_n : A_n}{\Gamma \vdash \varpi^{(k)}_{t_1,\dots,t_n} \bullet \varpi^{(-k)}_{t_1,\dots,t_n} \equiv \mathrm{id}_{t_k} : t_k \Rightarrow t_k : A_k}$$

$$\frac{\Gamma \vdash t : \prod_n(A_1, \dots, A_n)}{\Gamma \vdash \varsigma_t^{-1} \bullet \varsigma_t \equiv \mathrm{id}_t : t \Rightarrow t : \prod_n(A_1, \dots, A_n)}$$

$$\frac{\Gamma \vdash t : \prod_n(A_1, \dots, A_n)}{\Gamma \vdash \varsigma_t \bullet \varsigma_t^{-1} \equiv \mathrm{id}_{\mathsf{tup}(\pi_1\{t\},\dots,\pi_n\{t\})} : \mathsf{tup}(\pi_\bullet\{t\}) \Rightarrow \mathsf{tup}(\pi_\bullet\{t\}) : \prod_n(A_1, \dots, A_n)}$$

Figure 4.4: Inverses for the unit and counit

Rules for $\Lambda^{\times}_{\mathrm{ps}}(\mathcal{G})$.

$$\frac{(\Gamma \vdash \mathrm{id}_{t_i} : t_i \Rightarrow t_i : A_i)_{i=1,\dots,n}}{\Gamma \vdash \mathsf{tup}(\mathrm{id}_{t_1}, \dots, \mathrm{id}_{t_n}) \equiv \mathrm{id}_{\mathsf{tup}(t_1,\dots,t_n)} : \mathsf{tup}(t_1, \dots, t_n) \Rightarrow \mathsf{tup}(t_1, \dots, t_n) : \prod_n(A_1, \dots, A_n)}$$

$$\frac{(\Gamma \vdash \tau'_i : t'_i \Rightarrow t''_i : A_i)_{i=1,\dots,n} \qquad (\Gamma \vdash \tau_i : t_i \Rightarrow t'_i : A_i)_{i=1,\dots,n}}{\Gamma \vdash \mathsf{tup}(\tau'_1, \dots, \tau'_n) \bullet \mathsf{tup}(\tau_1, \dots, \tau_n) \equiv \mathsf{tup}(\tau'_1 \bullet \tau_1, \dots, \tau'_n \bullet \tau_n) : \mathsf{tup}(t_\bullet) \Rightarrow \mathsf{tup}(t''_\bullet) : \prod_n(A_\bullet)}$$

$$\frac{\Gamma \vdash \sigma : u \Rightarrow u' : \prod_n(A_1, \dots, A_n)}{\Gamma \vdash \varsigma_{u'} \bullet \sigma \equiv \mathsf{tup}(\pi_1\{\sigma\}, \dots, \pi_n\{\sigma\}) \bullet \varsigma_u : u \Rightarrow \mathsf{tup}(\pi_\bullet\{u'\}) : \prod_n(A_1, \dots, A_n)} \ \varsigma\text{-nat}$$

$$\frac{(\Gamma \vdash \tau_i : t_i \Rightarrow t'_i : A_i)_{i=1,\dots,n}}{\Gamma \vdash \varpi^{(k)}_{t'_1,\dots,t'_n} \bullet \pi_k\{\mathsf{tup}(\tau_1, \dots, \tau_n)\} \equiv \tau_k \bullet \varpi^{(k)}_{t_1,\dots,t_n} : \pi_k\{\mathsf{tup}(t_\bullet)\} \Rightarrow t_k : A_k} \ \varpi^{(k)}\text{-nat } (1 \leqslant k \leqslant n)$$

$$\frac{\Gamma \vdash \mathsf{tup}(t_1, \dots, t_n) : \prod_n(A_1, \dots, A_n)}{\Gamma \vdash \mathsf{tup}(\varpi^{(1)}_{t_\bullet}, \dots, \varpi^{(n)}_{t_\bullet}) \bullet \varsigma_{\mathsf{tup}(t_\bullet)} \equiv \mathrm{id}_{\mathsf{tup}(t_\bullet)} : \mathsf{tup}(t_\bullet) \Rightarrow \mathsf{tup}(t_\bullet) : \prod_n(A_\bullet)} \ \text{triangle-law-1}$$

$$\frac{\Gamma \vdash \pi_k\{u\} : A_k}{\Gamma \vdash \varpi^{(k)}_{t_1,\dots,t_n} \bullet \pi_k\{\varsigma_u\} \equiv \mathrm{id}_{\pi_k\{u\}} : \pi_k\{u\} \Rightarrow \pi_k\{u\} : A_k} \ \text{triangle-law-2 } (1 \leqslant k \leqslant n)$$

Figure 4.5: Admissible rules for $\Lambda^{\times}_{\mathrm{ps}}(\mathcal{G})$

We also give the syntactic constructions of the 2-cells post and fuse (recall Construction 4.1.6 on page 80). Intuitively, post witnesses the identity $\langle t_1, \dots, t_n\rangle[u_i/x_i] = \langle t_1[u_i/x_i], \dots, t_n[u_i/x_i]\rangle$ for capture-avoiding substitution in the simply-typed lambda calculus.

Construction 4.3.13. Let $\mathcal{S}$ be a $\Lambda_{\mathrm{ps}}^{\times}$-signature. Define a 2-cell post in $\Lambda_{\mathrm{ps}}^{\times}(\mathcal{S})$ with typing

$$\frac{x_1 : A_1, \dots, x_n \vdash \mathsf{tup}(t_1, \dots, t_m) : \prod_m(B_1, \dots, B_m) \qquad (\Delta \vdash u_i : A_i)_{i=1,\dots,n}}{\Delta \vdash \mathsf{post}(t_\bullet; u_\bullet) : \mathsf{tup}(t_1, \dots, t_m)\{u_i\} \Rightarrow \mathsf{tup}(t_1\{u_i\}, \dots, t_m\{u_i\}) : \prod_m(B_1, \dots, B_m)}$$

by setting $\mathsf{post}(t_\bullet; u_\bullet) := \mathsf{p}^{\dagger}(\alpha_1, \dots, \alpha_m)$ where

$$\alpha_k := \pi_k\{\mathsf{tup}(t_1, \dots, t_m)\{u_i\}\} \xRightarrow{\mathsf{assoc}^{-1}} \pi_k\{\mathsf{tup}(t_1, \dots, t_m)\}\{u_i\} \xRightarrow{\varpi^{(k)}\{u_i\}} t_k\{u_i\}$$

Also define a 2-cell fuse with signature

$$\frac{(x_i : A_i \vdash t_i : A_i)_{i=1,\dots,n} \qquad (\Delta \vdash u_i : A_i)_{i=1,\dots,n}}{\Delta \vdash \mathsf{fuse}(t_\bullet; u_\bullet) : \mathsf{tup}(t_\bullet\{\pi_\bullet(p)\})\{\mathsf{tup}(u_1, \dots, u_n)\} \Rightarrow \mathsf{tup}(t_1\{u_1\}, \dots, t_n\{u_n\}) : \prod_n(B_1, \dots, B_n)}$$

by setting $\mathsf{fuse}(t_\bullet; u_\bullet) := \mathsf{p}^{\dagger}(\beta_1, \dots, \beta_n)$ for β_k the composite

$$\begin{tikzcd}
\pi_k\{\mathsf{tup}(t_\bullet\{\pi_\bullet(p)\})\{\mathsf{tup}(u_1, \dots, u_n)\}\} \arrow[rr, Rightarrow, "\beta_k"] \arrow[d, Rightarrow, "\mathsf{assoc}^{-1}"'] & & t_k\{u_k\} \\
\pi_k\{\mathsf{tup}(t_\bullet\{\pi_\bullet(p)\})\}\{\mathsf{tup}(u_1, \dots, u_n)\} \arrow[d, Rightarrow, "\varpi^{(k)}\{\mathsf{tup}(u_1,\dots,u_n)\}"'] & & \\
t_k\{\pi_k(p)\}\{\mathsf{tup}(u_1, \dots, u_n)\} \arrow[rr, Rightarrow, "\mathsf{assoc}"'] & & t_k\{\pi_k\{\mathsf{tup}(u_1, \dots, u_n)\}\} \arrow[uu, Rightarrow, "t_k\{\varpi^{(k)}\}"']
\end{tikzcd}$$

◂

Since they are defined by applying the universal property to rewrites that are both natural and invertible, it follows that post and fuse are also invertible, as well as being natural in the sense that the following rules are admissible:

$$\frac{(x_1 : A_1, \dots, x_n : A_n \vdash \tau_j : t_j \Rightarrow t_j' : B_j)_{j=1,\dots,m} \qquad (\Delta \vdash \sigma_i : u_i \Rightarrow u_i' : A_i)_{i=1,\dots,n}}{\Delta \vdash \mathsf{post}(t_\bullet'; u_\bullet') \bullet \mathsf{tup}(\tau_\bullet)\{\sigma_i\} \equiv \mathsf{tup}(\tau_\bullet\{\sigma_i\}) \bullet \mathsf{post}(t_\bullet; u_\bullet) : \mathsf{tup}(t_\bullet)\{u_i\} \Rightarrow \mathsf{tup}(t_\bullet'\{u_i'\}) : \prod B_\bullet}$$

$$\frac{(x_i : A_i \vdash \tau_i : t_i \Rightarrow t_i' : A_i)_{i=1,\dots,n} \qquad (\Delta \vdash \sigma_i : u_i \Rightarrow u_i' : A_i)_{i=1,\dots,n}}{\begin{array}{l}\Delta \vdash \mathsf{fuse}(t_\bullet'; u_\bullet') \bullet \mathsf{tup}(\tau_\bullet\{\pi_\bullet(p)\})\{\mathsf{tup}(\sigma_\bullet)\} \equiv \mathsf{tup}(\tau_\bullet\{\sigma_\bullet\}) \bullet \mathsf{fuse}(t_\bullet; u_\bullet) : \\ \qquad\qquad : \mathsf{tup}(t_\bullet\{\pi_\bullet(p)\})\{\mathsf{tup}(u_1, \dots, u_n)\} \Rightarrow \mathsf{tup}(t_1'\{u_1'\}, \dots, t_n'\{u_n'\}) : \prod_n B_\bullet\end{array}}$$

Moreover, the proofs of Lemma 4.1.7 translate readily to the type theory.

Lemma 4.3.14. Let $\Gamma := (x_i : A_i)_{i=1,\dots,n}$ and $\Delta := (y_l : B_l)_{l=1,\dots,k}$ be contexts and suppose $(\Delta \vdash \sigma_i : u_i \Rightarrow u'_i : A_i)_{i=1,\dots,n}$. Then

1. (Naturality). If $(\Gamma \vdash \tau_j : t_j \Rightarrow t'_j : B_j)_{j=1,\dots,m}$, then

$$\begin{tikzcd}
\mathsf{tup}(t_1, \dots, t_m)\{u_\bullet\} \arrow[r, Rightarrow, "\mathsf{post}"] \arrow[d, Rightarrow, "\mathsf{tup}(\tau_1, \dots, \tau_m)\{\sigma_\bullet\}"'] & \mathsf{tup}(t_1\{u_\bullet\}, \dots, t_m\{u_\bullet\}) \arrow[d, Rightarrow, "\mathsf{tup}(\tau_1\{\sigma_\bullet\}, \dots, \tau_m\{\sigma_\bullet\})"] \\
\mathsf{tup}(t'_1, \dots, t'_m)\{u'_\bullet\} \arrow[r, Rightarrow, "\mathsf{post}"'] & \mathsf{tup}(t'_1\{u'_\bullet\}, \dots, t'_m\{u'_\bullet\})
\end{tikzcd}$$

2. (Compatibility with ι). If $(\Gamma \vdash t_m : B_m)_{j=1,\dots,m}$ then

$$\begin{tikzcd}
\mathsf{tup}(t_1, \dots, t_m) \arrow[r, Rightarrow, "\iota"] \arrow[dr, Rightarrow, "\mathsf{tup}(\iota, \dots, \iota)"'] & \mathsf{tup}(t_1, \dots, t_m)\{x_\bullet\} \arrow[d, Rightarrow, "\mathsf{post}"] \\
 & \mathsf{tup}(t_1\{x_\bullet\}, \dots, t_m\{x_\bullet\})
\end{tikzcd}$$

3. (Compatibility with assoc). For terms $(\Gamma \vdash t_m : C_m)_{j=1,\dots,m}$ and $(\Sigma \vdash v_l : B_l)_{l=1,\dots,k}$ then

$$\begin{tikzcd}
\mathsf{tup}(t_1, \dots, t_m)\{u_\bullet\}\{v_\bullet\} \arrow[r, Rightarrow, "\mathsf{post}\{v_\bullet\}"] \arrow[dd, Rightarrow, "\mathsf{assoc}"'] & \mathsf{tup}(t_1\{u_\bullet\}, \dots, t_m\{u_\bullet\})\{v_\bullet\} \arrow[d, Rightarrow, "\mathsf{post}"] \\
 & \mathsf{tup}(t_1\{u_\bullet\}\{v_\bullet\}, \dots, t_m\{u_\bullet\}\{v_\bullet\}) \arrow[d, Rightarrow, "\mathsf{tup}(\mathsf{assoc}, \dots, \mathsf{assoc})"] \\
\mathsf{tup}(t_1, \dots, t_m)\{u_\bullet\{v_\bullet\}\} \arrow[r, Rightarrow, "\mathsf{post}"'] & \mathsf{tup}(t_1\{u_\bullet\{v_\bullet\}\}, \dots, t_m\{u_\bullet\{v_\bullet\}\})
\end{tikzcd}$$

4. (Compatibility with ς). If $\Gamma \vdash t : \prod_m(B_1, \dots, B_m)$ then

$$\begin{tikzcd}
t\{u_\bullet\} \arrow[r, Rightarrow, "\varsigma\{u_\bullet\}"] \arrow[d, Rightarrow, "\varsigma"'] & \mathsf{tup}(\pi_1\{t\}, \dots, \pi_m\{t\})\{u_\bullet\} \arrow[d, Rightarrow, "\mathsf{post}"] \\
\mathsf{tup}(\pi_1\{t\{u_\bullet\}\}, \dots, \pi_m\{t\{u_\bullet\}\}) & \mathsf{tup}(\pi_1\{t\}\{u_\bullet\}, \dots, \pi_m\{t\}\{u_\bullet\}) \arrow[l, Rightarrow, "\mathsf{tup}(\mathsf{assoc}, \dots, \mathsf{assoc})"]
\end{tikzcd}$$

Proof. The proofs are straightforward calculations using the universal property of Lemma 4.3.7. For example, for naturality we simply observe that

$$\begin{aligned}
&\varpi^{(k)}_{t'_1\{u'_\bullet\}, \dots, t'_m\{u'_\bullet\}} \bullet \pi_k\{\mathsf{tup}(\tau_1\{\sigma_\bullet\}, \dots, \tau_m\{\sigma_\bullet\}) \bullet \mathsf{post}(t_\bullet; u_\bullet)\} \\
&\qquad = \varpi^{(k)}_{t'_1\{u'_\bullet\}, \dots, t'_m\{u'_\bullet\}} \bullet \pi_k\{\mathsf{tup}(\tau_1\{\sigma_\bullet\}, \dots, \tau_m\{\sigma_\bullet\})\} \bullet \pi_k\{\mathsf{post}(t_\bullet; u_\bullet)\} \\
&\qquad = \tau_k\{\sigma_\bullet\} \bullet \varpi^{(k)}_{t_1, \dots, t_m} \bullet \pi_k\{\mathsf{post}(t_\bullet; u_\bullet)\} \\
&\qquad = \tau_k\{\sigma_\bullet\} \bullet \varpi^{(k)}_{t_1,\dots,t_m}\{u_\bullet\} \bullet \mathsf{assoc}^{-1}_{\pi_k(p);\mathsf{tup}(t_1, \dots, t_m);u_\bullet}
\end{aligned}$$

and that

$$\begin{aligned}
\varpi^{(k)}_{t_1'\{u_\bullet'\}, \dots, t_m'\{u_\bullet'\}} & \bullet \pi_k\{\mathsf{post}(t_\bullet'; u_\bullet') \bullet \mathsf{tup}(\tau_1, \dots, \tau_m)\{\sigma_\bullet\}\} \\
&= \varpi^{(k)}_{t_1'\{u_\bullet'\}, \dots, t_m'\{u_\bullet'\}} \bullet \pi_k\{\mathsf{post}(t_\bullet'; u_\bullet')\} \bullet \pi_k\{\mathsf{tup}(\tau_1, \dots, \tau_m)\{\sigma_\bullet\}\} \\
&= \varpi^{(k)}_{t_1', \dots, t_m'}\{u_\bullet'\} \bullet \mathsf{assoc}^{-1}_{\pi_k(p); \mathsf{tup}(t_1', \dots, t_m'); u_\bullet'} \bullet \pi_k\{\mathsf{tup}(\tau_1, \dots, \tau_m)\{\sigma_\bullet\}\} \\
&= \varpi^{(k)}_{t_1', \dots, t_m'}\{u_\bullet'\} \bullet \pi_k\{\mathsf{tup}(\tau_1, \dots, \tau_m)\}\{\sigma_\bullet\} \bullet \mathsf{assoc}^{-1}_{\pi_k(p); \mathsf{tup}(t_1, \dots, t_m); u_\bullet} \\
&= \tau_k\{\sigma_\bullet\} \bullet \varpi^{(k)}_{t_1, \dots, t_m}\{u_\bullet\} \bullet \mathsf{assoc}^{-1}_{\pi_k(p); \mathsf{tup}(t_1, \dots, t_m); u_\bullet}
\end{aligned}$$

Hence, by the universal property of Lemma 4.3.7, the required equality holds. The other cases are similar. □

4.3.3 Products from context extension

We end this chapter by noting a 'degenerate' or 'implicit' way for a deductive system to exhibit product structure. The construction gives rise to a syntactic model that is an fp-bicategory, but does not arise via a cartesian biclone or provide a type-theoretic description of bicategorical products. While this structure is not in the vein of those we have discussed above, it will play an important role: exponentials in the simply-typed lambda calculus are defined with respect to these products. The product structure is given by context concatenation.

Construction 4.3.15. For any $\Lambda_{\mathrm{ps}}^{\times}$-signature $\mathcal{S}$, define a bicategory $\mathcal{T}_{\mathrm{ps}}^{@,\times}(\mathcal{S})$ as follows. Fix an enumeration of variables $x_1, \dots, x_n, \dots$. The objects are then contexts $\Gamma, \Delta, \dots$ in which the ith entry has variable name x_i. The 1-cells $\Gamma \to (y_j : B_j)_{j=1,\dots,m}$ are m-tuples of α-equivalence classes of terms $(\Gamma \vdash t_j : B_j)_{j=1,\dots,m}$ derivable in $\Lambda_{\mathrm{ps}}^{\times}(\mathcal{S})$; the 2-cells are m-tuples of $\alpha{\equiv}$-equivalence classes of rewrites $(\Gamma \vdash \tau : t_j \Rightarrow t_j' : B_j)_{j=1,\dots,m}$.

Vertical composition is given pointwise by the $\bullet$ operation, and horizontal composition by explicit substitution:

$$\begin{gathered}
(t_1, \dots, t_l), (u_1, \dots, u_m) \mapsto (t_1\{x_i \mapsto u_i\}, \dots, t_m\{x_i \mapsto u_i\}) \\
(\tau_1, \dots, \tau_l), (\sigma_1, \dots, \sigma_m) \mapsto (\tau_1\{x_i \mapsto \sigma_i\}, \dots, \tau_m\{x_i \mapsto \sigma_i\})
\end{gathered}$$

The identity on $\Delta = (y_j : B_j)_{j=1,\dots,m}$ is the `var` rule $(\Delta \vdash y_j : B_j)_{j=1,\dots,m}$, and the structural isomorphisms l, r and a are given pointwise by ϱ, ι^{-1} and assoc, respectively. ◀

Since $\Lambda_{\mathrm{ps}}^{\times}$ comes equipped with a product structure, this bicategory has two product structures: one given by the product structure in the type theory, and the other by context extension. We emphasise this with the notation.

The type-theoretic product structure is induced from that on the full sub-bicategory of unary contexts via the following lemma, which can be seen as the type-theoretic translation of Lemma 4.2.48 on page 117.

Lemma 4.3.16. For any $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature $\mathcal{S}$ and context $\Gamma = (x_i : A_i)_{i=1,\dots,n}$, there exists an adjoint equivalence $\Gamma \leftrightarrows \big(p : \prod_n(A_1, \dots, A_n)\big)$ in $\mathcal{T}_{\mathrm{ps}}^{@,\times}(\mathcal{S})$.

Proof. Take the 1-cells

$$(\Gamma \vdash \mathsf{tup}(x_1, \dots, x_n) : \textstyle\prod_n(A_1, \dots, A_n)) : \Gamma \to (p : \prod_n(A_1, \dots, A_n))$$

$$(p : \textstyle\prod_n(A_1, \dots, A_n) \vdash \pi_i(p) : A_i)_{i=1,\dots,n} : (p : \prod_n(A_1, \dots, A_n)) \to \Gamma$$

For the unit and counit of the required adjoint equivalence we take

$$\big(\Gamma \vdash \varpi^{(i)}_{x_\bullet} \pi_i\{\mathsf{tup}(x_1, \dots, x_n)\} \Rightarrow x_i : A_i\big)_{i=1,\dots,n}$$

and the composite

$$\begin{tikzcd}
p \arrow[rr] \arrow[d, "\varsigma_p"'] & & \mathsf{tup}(x_1, \dots, x_n)\{\pi_i(p)\} \\
\mathsf{tup}(\pi_1\{p\}, \dots, \pi_n\{p\}) \arrow[r, "\mathsf{tup}(\varrho^{(-1)}, \dots, \varrho^{(-n)})"'] & \mathsf{tup}(x_1\{\pi_\bullet\{p\}\}, \dots, x_n\{\pi_\bullet\{p\}\}) \arrow[r, "\mathsf{post}(x_\bullet;\pi_\bullet\{p\})^{-1}"'] & \mathsf{tup}(x_1, \dots, x_n)\{\pi_\bullet\{p\}\} \arrow[u, "\mathsf{tup}(x_1, \dots, x_n)\{\iota^{-1}_{\pi_\bullet(p)}\}"']
\end{tikzcd}$$

The proof then amounts to making use of naturality to the point where one can apply the triangle laws of Figure 4.5. □

Remark 4.3.17. The preceding lemma, together with Lemma 3.2.18 on page 62, in fact entails that $\mathcal{T}_{\mathrm{ps}}^{@,\times}(\mathcal{S}) \simeq \mathrm{Syn}^{\times}(\mathcal{S})\big|_1$ for every unary $\Lambda_{\mathrm{ps}}^{\times}$-signature $\mathcal{S}$. ◂

We define the product $(x_i^{(1)} : A_i^{(1)})_{i=1,\dots,m_1} \times \cdots \times (x_i^{(n)} : A_i^{(n)})_{i=1,\dots,m_n}$ of arbitrary contexts to be the product $(p_1 : \prod_{i=1}^{m_1} A_i^{(1)}) \times \cdots \times (p_n : \prod_{i=1}^{m_n} A_i^{(n)})$ of the corresponding unary contexts. The ith projection is the $|\Gamma^{(i)}|$-tuple

$$\Big(p : \textstyle\prod_n\big(\prod_{|\Gamma^{(1)}|} A_\bullet^{(1)}, \dots, \prod_{|\Gamma^{(n)}|} A_\bullet^{(n)}\big) \vdash \pi_j\{\pi_i(p)\} : A_j^{(i)}\Big)_{j=1,\dots,|\Gamma^{(i)}|} \tag{4.38}$$

and the tupling of n maps $(\Delta \to \Gamma^{(i)})_{i=1,\dots,n}$, that is, of $|\Gamma^{(i)}|$-tuples $(\Delta \vdash t_j^{(i)} : A_j^{(i)})_{\substack{j=1,\dots,|\Gamma^{(i)}| \\ i=1,\dots,n}}$, is

$$\Delta \vdash \mathsf{tup}\big(\mathsf{tup}(t_\bullet^{(1)}), \dots, \mathsf{tup}(t_\bullet^{(n)})\big) : \textstyle\prod_n\big(\prod_{|\Gamma^{(1)}|} A_\bullet^{(1)}, \dots, \prod_{|\Gamma^{(n)}|} A_\bullet^{(n)}\big)$$

The counit $\varpi^{(i)}$ is the composite indicated by the pasting diagram

$$\begin{tikzcd}
\prod_n\big(\prod_{|\Gamma^{(1)}|} A_\bullet^{(1)}, \dots, \prod_{|\Gamma^{(n)}|} A_\bullet^{(n)}\big) \arrow[r, "\pi_i(p)"] & \prod_{|\Gamma^{(i)}|} A_\bullet^{(i)} \arrow[r, "(\pi_1(p), \dots, \pi_{|\Gamma^{(i)}|}(p))"] & \Gamma^{(i)} \\
\Delta \arrow[u, "\mathsf{tup}\big(\mathsf{tup}(t_\bullet^{(1)}), \dots, \mathsf{tup}(t_\bullet^{(n)})\big)"] \arrow[ur, bend right=15, "\mathsf{tup}(t_\bullet^{(i)})"'] \arrow[urr, bend right=40, "t_1^{(i)}, \dots, t_{|\Gamma^{(i)}|}^{(i)}"'] \arrow[u, phantom, shift left=12ex, "\overset{\varpi^{(i)}}{\cong}"] \arrow[ur, phantom, shift right=6ex, "\cong"] & &
\end{tikzcd}$$

That is, the $|\Gamma^{(i)}|$-tuple with jth component the composite rewrite

$$\begin{tikzcd}
\pi_j\{\pi_i(p)\}\Big\{\mathsf{tup}\Big(\mathsf{tup}(t_\bullet^{(1)}), \dots, \mathsf{tup}(t_\bullet^{(n)})\Big)\Big\} \arrow[r] \arrow[d, "\cong"'] & t_j^{(i)} \\
\pi_j\Big\{\pi_i\Big\{\mathsf{tup}\Big(\mathsf{tup}(t_\bullet^{(1)}), \dots, \mathsf{tup}(t_\bullet^{(n)})\Big)\Big\}\Big\} \arrow[r, "\pi_j\{\varpi^{(i)}\}"'] & \pi_j\Big\{\mathsf{tup}(t_1^{(i)}, \dots, t_{|\Gamma^{(i)}|}^{(i)})\Big\} \arrow[u, "\varpi^{(j)}"']
\end{tikzcd}$$

The next lemma encapsulates the required universal property.

Lemma 4.3.18. For any unary $\Lambda_{\mathrm{ps}}^{\times}$-signature $\mathcal{S}$, the 1-cell

$$\Big(p : \textstyle\prod_n\big(\prod_{|\Gamma^{(1)}|}A_\bullet^{(1)}, \dots, \prod_{|\Gamma^{(n)}|}A_\bullet^{(n)}\big) \vdash \pi_j\{\pi_i(p)\} : A_j^{(i)}\Big)_{j=1,\dots,|\Gamma^{(i)}|}$$

of (4.38) is a biuniversal arrow defining an fp-structure on $\mathcal{T}_{\mathrm{ps}}^{@,\times}(\mathcal{S})$.

Proof. Taking the structure described above, it remains to check the universal property of the counit. Suppose that $\Delta \vdash u : \big(\prod_{|\Gamma^{(1)}|}A_\bullet^{(1)}, \dots, \prod_{|\Gamma^{(n)}|}A_\bullet^{(n)}\big)$ and that $(\Delta \vdash t_j^{(i)} : A_j^{(i)})_{j=1,\dots,|\Gamma^{(i)}|}$ for $i = 1, \dots, n$, and consider a family of rewrites

$$\Big(\Delta \vdash \alpha_j^{(i)} : \pi_j\{\pi_i(p)\}\{u\} \Rightarrow t_j^{(i)} : A_j^{(i)}\Big)_{\substack{j=1,\dots,|\Gamma^{(i)}| \\ i=1,\dots,n}}$$

One thereby obtains composites $\widetilde{\alpha}_j^{(i)} := \pi_j\{\pi_i\{u\}\} \overset{\cong}{\Rightarrow} \pi_j\{\pi_i(p)\}\{u\} \overset{\alpha_j^{(i)}}{\Longrightarrow} t_j^{(i)}$ for $j = 1, \dots, |\Gamma^{(i)}|$ and $i = 1, \dots, n$. Applying the universal property of ϖ (Lemma 4.3.7) for each i, one obtains $\mathsf{p}^\dagger(\widetilde{\alpha}_1^{(i)}, \dots, \widetilde{\alpha}_{|\Gamma^{(i)}|}^{(i)}) : \pi_k\{u\} \Rightarrow \mathsf{tup}(t_1^{(i)}, \dots, t_{|\Gamma^{(i)}|}^{(i)})$ for $i = 1, \dots, n$. Finally applying the universal property to this family of rewrites, one obtains

$$\mathsf{p}^\dagger\Big(\mathsf{p}^\dagger(\widetilde{\alpha}_1^{(1)}, \dots, \widetilde{\alpha}_{|\Gamma^{(1)}|}^{(1)}), \dots, \mathsf{p}^\dagger(\widetilde{\alpha}_1^{(n)}, \dots, \widetilde{\alpha}_{|\Gamma^{(n)}|}^{(n)})\Big) : u \Rightarrow \mathsf{tup}\big(\mathsf{tup}(t_\bullet^{(1)}), \dots, \mathsf{tup}(t_\bullet^{(n)})\big)$$

To see that this 2-cell satisfies the required universal property, apply the corresponding property from Lemma 4.3.7 twice. □

We now turn to the second, strict, product structure. This arises from context extension. Constructing products in this way is a standard method in the categorical setting (*e.g.* [Pit00]) and is also employed by Hilken [Hil96] in the 2-categorical case to obtain a strict product. Taken on its own, however, it does not enable one to reason about products within the type theory.

Lemma 4.3.19. For any $\Lambda_{\mathrm{ps}}^{\times}$-signature $\mathcal{S}$ the syntactic model $\mathcal{T}_{\mathrm{ps}}^{@,\times}(\mathcal{S})$ of $\Lambda_{\mathrm{ps}}^{\times}(\mathcal{S})$ is an fp-bicategory with product structure given by context extension.

Proof. We claim first that every context $\Gamma := (x_i : A_i)_{i=1,\dots,n}$ is the n-ary product $\prod_{i=1}^{n}(x_i : A_i)$ of unary contexts $(x_1 : A_1), \dots, (x_n : A_n)$. Define projections $\pi_k : \Gamma \to A_k$ for $k = 1, \dots, n$ by $\Gamma \vdash x_k : A_k$. Then, given 1-cells $\Delta \vdash t_i : A_i$ for $i = 1, \dots, n$, define the n-ary tupling to be the n-tuple $(\Delta \vdash t_i : A_i)_{i=1,\dots,n}$. The unit and counit are the 2-cells with components $\varrho^{(-i)}$ and $\varrho^{(i)}$, respectively.

We extend this to all contexts in the obvious way. For contexts Γ_i $(i = 1, \dots, n)$ such that $\Gamma_i := (x_j : A_j^{(i)})_{j=1,\dots,|\Gamma_i|}$ the product $\prod_{i=1}^{n}\Gamma_i$ is the concatenated context $\Gamma_1, \dots, \Gamma_n$ (the enumeration of variables ensures no variable names are duplicated). The kth projection is the $|\Gamma_k|$-tuple $(\Gamma_1, \dots, \Gamma_n \vdash x_j : A_j^{(k)})_{1+\sum_{l=1}^{k-1}|\Gamma_l| \leqslant j \leqslant |\Gamma_k| + \sum_{l=1}^{k-1}|\Gamma_l|}$ and the n-ary tupling of 1-cells $(\bar{t}_i : \Delta \to \Gamma_i)_{i=1,\dots,n}$ with $\bar{t}_i := (\Delta \vdash t_j^{(i)} : A_j^{(i)})_{j=1,\dots,|\Gamma_i|}$ is just the unfolded $\sum_{i=1}^{n}|\Gamma_i|$-tuple $(\Delta \vdash t_j^{(i)} : A_j^{(i)})_{\substack{i=1,\dots,n \\ j=1,\dots,|\Gamma_i|}}$. The unit and counit are as in the unary case. □

Chapter 5

A type theory for cartesian closed bicategories

We now build on the preceding chapters, and the type theory $\Lambda^{\times}_{\mathrm{ps}}$, to construct a type theory for cartesian closed bicategories. First we extend the theory of clones with finite products to include exponentials via a version of Lambek's *internal hom* of a multicategory [Lam89]. Next we extend this to (cartesian) biclones and use it to extract a type theory $\Lambda^{\times,\rightarrow}_{\mathrm{ps}}$ for which the syntactic model is free among cartesian closed biclones. The proof of the corresponding bicategorical free property, however, throws up a subtlety: exponentials in the Lambek style are defined as a right (bi)adjoint to *context extension* rather than the type-theoretic product. In terms of the syntactic models of the preceding chapter, exponentials appear with respect to the context extension product structure, rather than the type-theoretic product structure (recall Section 4.3.3). As we shall see, it follows that the restriction of $\Lambda^{\times,\rightarrow}_{\mathrm{ps}}$ to unary contexts cannot satisfy a strict free property mirroring that of $\Lambda^{\mathrm{bicat}}_{\mathrm{ps}}$ and $\Lambda^{\times}_{\mathrm{ps}}$. We address this by showing that the syntactic model of $\Lambda^{\times,\rightarrow}_{\mathrm{ps}}$ is biequivalent to the cartesian closed bicategory enjoying such a strict free property. (Table A.1 on page 308 provides an index of the various free constructions and syntactic models we employ.) We end the chapter by making precise the claim that $\Lambda^{\times,\rightarrow}_{\mathrm{ps}}$ is the simply-typed lambda calculus up to isomorphism.

5.1 Cartesian closed bicategories

Let us start by recapitulating the definition of cartesian closed bicategory. To give a cartesian closed structure on an fp-bicategory $(\mathcal{B}, \Pi_n(-))$ is to specify a biadjunction $(-) \times A \dashv (A \Rightarrow -)$ for every $A \in \mathcal{B}$. Following Definition 2.4.1, this amounts to choosing an object $(A \Rightarrow B)$ and a biuniversal arrow $\mathrm{eval}_{A,B} : (A \Rightarrow B) \times A \to B$ for every $A, B \in \mathcal{B}$. We unfold the definition as follows.

Definition 5.1.1. A *cartesian closed bicategory* or *cc-bicategory* is an fp-bicategory $(\mathcal{B}, \Pi_n(-))$ equipped with the following data for every $A, B \in \mathcal{B}$:

1. A chosen object $(A \mathrel{=\!\!\rhd} B)$,
2. A specified 1-cell $\mathrm{eval}_{A,B} : (A \mathrel{=\!\!\rhd} B) \times A \to B$,
3. For every $X \in \mathcal{B}$, an adjoint equivalence

$$\begin{tikzcd} \mathcal{B}(X, A \mathrel{=\!\!\rhd} B) \arrow[r, bend left, "\mathrm{eval}_{A,B}\circ(-\times A)"] \arrow[r, phantom, "\perp\simeq"] & \mathcal{B}(X \times A, B) \arrow[l, bend left, "\lambda"] \end{tikzcd} \qquad (5.1)$$

specified by a family of universal arrows $\varepsilon_f : \mathrm{eval}_{A,B} \circ (\lambda f \times A) \Rightarrow f$.

We call the functor $\lambda(-)$ *currying* and refer to λf as the *currying of f*. ◂

Remark 5.1.2. As for products, we shall call an exponential structure *strict* if the equivalences (5.1) are isomorphisms. When the underlying bicategory $\mathcal{B}$ is a 2-category, this yields the definition of cartesian closure in the Cat-enriched sense (*c.f.* Remark 4.1.2). ◂

Explicitly, the equivalences (5.1) are given by the following universal property. For every 1-cell $t : X \times A \to B$ we require a 1-cell $\lambda t : X \to (A \mathrel{=\!\!\rhd} B)$ and an invertible 2-cell $\varepsilon_t : \mathrm{eval}_{A,B} \circ (\lambda t \times A) \Rightarrow t$, universal in the sense that for any 2-cell $\alpha : \mathrm{eval}_{A,B} \circ (u \times A) \Rightarrow t$ there exists a unique 2-cell $\mathsf{e}^\dagger(\alpha) : u \Rightarrow \lambda t$ such that $\varepsilon_t \bullet \big(\mathrm{eval}_{A,B} \circ (\mathsf{e}^\dagger(\alpha) \times A)\big) = \alpha$. Moreover, we require that the unit $\eta_t := \mathsf{e}^\dagger(\mathrm{id}_{\mathrm{eval}_{A,B} \circ (t \times A)})$ is also invertible.

Notation 5.1.3. Following the categorical notation, for 1-cells $f : A' \to A$ and $g : B \to B'$ we write $(f \mathrel{=\!\!\rhd} g) : (A \mathrel{=\!\!\rhd} B) \to (A' \mathrel{=\!\!\rhd} B')$ for the exponential transpose of the composite $(g \circ \mathrm{eval}_{A,B}) \circ (\mathrm{Id}_{A \mathrel{=\!\!\rhd} B} \times f)$, thus:

$$(f \mathrel{=\!\!\rhd} g) := \lambda\big((A \mathrel{=\!\!\rhd} B) \times A' \xrightarrow{(A \mathrel{=\!\!\rhd} B) \times f} (A \mathrel{=\!\!\rhd} B) \times A \xrightarrow{\mathrm{eval}_{A,B}} B \xrightarrow{g} B'\big)$$

and likewise on 2-cells. ◂

As for products, 1-category theoretic notation can be misleading when the identity is referred to explicitly. Consider the identities

$$\begin{aligned} (f \mathrel{=\!\!\rhd} \mathrm{Id}_B) &= \lambda((\mathrm{Id}_B \circ \mathrm{eval}_{A,B}) \circ (f \times \mathrm{Id}_A)) \\ (\mathrm{Id}_A \mathrel{=\!\!\rhd} g) &= \lambda((g \circ \mathrm{eval}_{A,B}) \circ (\mathrm{Id}_{A \mathrel{=\!\!\rhd} B} \times \mathrm{Id}_A)) \end{aligned}$$

In a *2-category* with pseudo-products and pseudo-exponentials, one may safely write $(f \mathrel{=\!\!\rhd} \mathrm{Id}_B)$ as simply $\lambda(\mathrm{eval}_{A,B} \circ (f \times A))$, but cannot simplify $(\mathrm{Id}_A \mathrel{=\!\!\rhd} g)$ in a similar way to $\lambda(g \circ \mathrm{eval}_{A,B})$. Note, however, that this simplification is possible in the presence of strict products, when the unit is an identity.

Remark 5.1.4. The uniqueness of exponentials up to equivalence manifests itself in the same way as for products. For instance, given an adjoint equivalence $e : E \simeq (A \Rightarrow B) : f$, the object E inherits an exponential structure by composition with e and f (*c.f.* Remark 4.1.5). ◂

In Construction 4.1.6 we saw that standard properties of cartesian categories are witnessed by natural families of 2-cells in an fp-bicategory. The same principle holds for cc-bicategories.

Construction 5.1.5. Let $(\mathcal{B}, \Pi_n(-), \Rightarrow)$ be a cc-bicategory. For $g : X \to Y$ and $f : Y \times A \to B$ we define $\mathsf{push}(f, g) : \lambda(f) \circ g \Rightarrow \lambda\big(f \circ (g \times A)\big)$ as $\mathsf{e}^{\dagger}(\tau)$, for τ the composite

$$\begin{tikzcd}
\mathrm{eval}_{A,B} \circ ((\lambda f \circ g) \times A) \arrow[r, "\tau"] \arrow[d, "\mathrm{eval}\circ(\Phi_{f,g})^{-1}"'] & f \circ (g \times A) \\
\mathrm{eval}_{A,B} \circ ((\lambda f \times A) \circ (g \times A)) \arrow[r, "\cong"'] & (\mathrm{eval}_{A,B} \circ (\lambda f \times A)) \circ (g \times A) \arrow[u, "\varepsilon_f \circ (g \times A)"']
\end{tikzcd}$$

where $\Phi_{f,g} : (f \times A) \circ (g \times A) \Rightarrow (fg \times A)$ witnesses $\prod_2(-, =)$ as a pseudofunctor (recall Construction 4.1.6(3)).

◂

This family of 2-cells is natural in each of its arguments and satisfies the expected equations, some of which are collected in the following lemma. As for Lemma 4.1.7, we assume the underlying bicategory is strict for the sake of clarity.

Lemma 5.1.6. Let $(\mathcal{B}, \Pi_n(-), \Rightarrow)$ be a 2-category with finite pseudo-products and pseudo-exponentials. Then for all 1-cells f, g and h, the following diagrams commute whenever they are well-typed:

$$\begin{tikzcd}
(\lambda f) \circ \mathrm{Id} \arrow[r, "\mathsf{push}"] \arrow[d, equal] & \lambda\big(f \circ (\mathrm{Id} \times A)\big) \arrow[d, equal] \\
\lambda f \arrow[r, "\lambda(f \circ \varsigma_f)"'] & \lambda(f \circ \langle \pi_1, \pi_2 \rangle)
\end{tikzcd} \tag{5.2}$$

$$\begin{tikzcd}
f \circ g \arrow[r, "\eta_{f \circ g}"] \arrow[d, "\eta_f \circ g"'] & \lambda\big(\mathrm{eval} \circ (fg \times A)\big) \\
\lambda\big(\mathrm{eval} \circ (f \times A)\big) \circ g \arrow[r, "\mathsf{push}"'] & \lambda\big(\mathrm{eval} \circ (f \times A) \circ (g \times A)\big) \arrow[u, "\lambda(\mathrm{eval}\circ\Phi_{f,g;\mathrm{Id}})"']
\end{tikzcd} \tag{5.3}$$

$$\begin{tikzcd}
(f \Rightarrow g) \circ \mathrm{Id} \arrow[r, "\mathsf{push}"] \arrow[d, equal] & \lambda\big(g \circ \mathrm{eval} \circ ((A \Rightarrow B) \times f) \circ (\mathrm{Id} \times B)\big) \arrow[d, "\lambda(g\circ\mathrm{eval}\circ\Phi_{\mathrm{Id};f,\mathrm{Id}})"] \\
(f \Rightarrow g) \arrow[r, equal] & \lambda\big(g \circ \mathrm{eval} \circ ((A \Rightarrow B) \times f)\big)
\end{tikzcd} \tag{5.4}$$

$$\begin{tikzcd}
\lambda(f) \circ g \circ h \arrow[r, "\mathsf{push} \circ h"] \arrow[d, "\mathsf{push}"'] & \lambda\big(f \circ (g \times A)\big) \circ h \arrow[r, "\mathsf{push}"] & \lambda\big(f \circ (g \times A) \circ (h \times A)\big) \arrow[d, "\lambda(f \circ \Phi_{g,h;\mathrm{Id}})"] \\
\lambda\big(f \circ ((g \circ h) \times A)\big) \arrow[rr, equal] & & \lambda\big(f \circ (gh \times A)\big)
\end{tikzcd} \tag{5.5}$$

□

A pseudofunctor between cartesian closed bicategories is cartesian closed if it preserves both the biuniversal arrows defining products and the biuniversal arrows defining exponentials.

Definition 5.1.7. A *cartesian closed pseudofunctor* or *cc-pseudofunctor* between cc-bicategories $(\mathcal{B}, \Pi_n(-), \Rightarrow)$ and $(\mathcal{C}, \Pi_n(-), \Rightarrow)$ is an fp-pseudofunctor $(F, \mathrm{q}^{\times})$ equipped with specified adjoint equivalences

$$\mathrm{m}_{A,B} : F(A \Rightarrow B) \leftrightarrows (FA \Rightarrow FB) : \mathrm{q}^{\Rightarrow}_{A,B}$$

for every $A, B \in \mathcal{B}$, where $\mathrm{m}_{A,B} : F(A \Rightarrow B) \to (FA \Rightarrow FB)$ is the exponential transpose of $F(\mathrm{eval}_{A,B}) \circ \mathrm{q}^{\times}_{A \Rightarrow B, A}$. We denote the 2-cells witnessing that $\mathrm{q}^{\Rightarrow}_{A,B}$ and $\mathrm{m}_{A,B}$ form an equivalence by

$$\mathsf{u}^{\Rightarrow}_{A,B} : \mathrm{Id}_{(FA \Rightarrow FB)} \Rightarrow \mathrm{m}_{A,B} \circ \mathrm{q}^{\Rightarrow}_{A,B}$$
$$\mathsf{c}^{\Rightarrow}_{A,B} : \mathrm{q}^{\Rightarrow}_{A,B} \circ \mathrm{m}_{A,B} \Rightarrow \mathrm{Id}_{F(A \Rightarrow B)}$$

A cc-pseudofunctor $(F, \mathrm{q}^{\times}, \mathrm{q}^{\Rightarrow})$ is *strict* if $(F, \mathrm{q}^{\times})$ is a strict fp-pseudofunctor such that

$$\begin{aligned}
F(A \Rightarrow B) &= (FA \Rightarrow FB) \\
F(\mathrm{eval}_{A,B}) &= \mathrm{eval}_{FA,FB} \\
F(\lambda t) &= \lambda(Ft) \\
F(\varepsilon_t) &= \varepsilon_{Ft} \\
\mathrm{q}^{\Rightarrow}_{A,B} &= \mathrm{Id}_{FA \Rightarrow FB}
\end{aligned}$$

with equivalences canonically induced by the 2-cells

$$\mathsf{e}^{\dagger}(\mathrm{eval}_{FA,FB} \circ \kappa) : \mathrm{Id}_{(FA \Rightarrow FB)} \overset{\cong}{\Rightarrow} \lambda(\mathrm{eval}_{FA,FB} \circ \mathrm{Id}_{(FA \Rightarrow FB) \times FA})$$

for κ is the canonical isomorphism $\mathrm{Id}_{FA \Rightarrow FB} \times FA \cong \mathrm{Id}_{(FA \Rightarrow FB) \times FA}$. ◂

Remark 5.1.8 (*c.f.* Remark 4.1.10)**.** If $\mathcal{B}$ is a bicategory equipped with two cartesian closed structures, say $(\mathcal{B}, \Pi_n(-), \Rightarrow)$ and $\big(\mathcal{B}, \mathrm{Prod}_n(-), [-,-]\big)$, then for any cc-pseudofunctor $(F, \mathrm{q}^{\times}, \mathrm{q}^{\Rightarrow}) : (\mathcal{B}, \Pi_n(-), \Rightarrow) \to (\mathcal{C}, \Pi_n(-), \Rightarrow)$ there exists an (equivalent) cc-pseudofunctor

$$\big(\mathcal{B}, \mathrm{Prod}_n(-), [-,-]\big) \to (\mathcal{C}, \Pi_n(-), \Rightarrow)$$

with witnessing equivalences arising from the uniqueness of products and exponentials up to equivalence. ◂

cc-Biequivalences from biequivalences. In the preceding chapter (page 86) we saw that, so far as we are concerned, it is unnecessary to distinguish between pseudonatural transformations and their product-respecting counterparts. A similar situation holds in the cartesian closed case. For cartesian closed pseudofunctors $(F, \mathrm{q}^{\times}, \mathrm{q}^{\Rightarrow}), (G, \mathrm{u}^{\times}, \mathrm{u}^{\Rightarrow}) : (\mathcal{B}, \Pi_n(-), \Rightarrow) \to (\mathcal{C}, \Pi_n(-), \Rightarrow)$, a *cc-transformation* $F \Rightarrow G$ is an fp-transformation $(\overline{\alpha}, \alpha, \alpha^{\times}) : (F, \mathrm{q}^{\times}) \Rightarrow (G, \mathrm{u}^{\times})$ (recall Definition 4.1.14) equipped with a 2-cell $\alpha^{\Rightarrow}_{A,B}$ $(A, B \in \mathcal{B})$ as in the diagram below

$$\begin{tikzcd}
F(A \Rightarrow B) \times FA \arrow[r, "\mathrm{m}^F_{A,B} \times FA"] \arrow[rr, bend left=40, "\mathrm{eval}_{FA,FB} \circ (\mathrm{m}^F_{A,B} \times FA)"] \arrow[d, "\alpha_{A \Rightarrow B} \times \alpha_A"'] & (FA \Rightarrow FB) \times FA \arrow[r, "\mathrm{eval}_{FA,FB}"] \arrow[d, phantom, "\overset{\alpha^{\Rightarrow}_{A,B}}{\Leftarrow}"] & FB \arrow[d, "\alpha_B"] \\
G(A \Rightarrow B) \times GA \arrow[r, "\mathrm{m}^G_{A,B} \times GA"'] \arrow[rr, bend right=40, "\mathrm{eval}_{GA,GB} \circ (\mathrm{m}^G_{A,B} \times GA)"'] & (GA \Rightarrow GB) \times GA \arrow[r, "\mathrm{eval}_{GA,GB}"'] & GB
\end{tikzcd}$$

such that the following pasting diagram is equal to $\overline{\alpha}_{\mathrm{eval}_{A,B}}$:

$$\begin{tikzcd}
 & & & F\big((A \Rightarrow B) \times A\big) \arrow[dr, "F\mathrm{eval}_{A,B}"] \arrow[d, phantom, "\overset{\varepsilon}{\cong}"] & \\
F\big((A \Rightarrow B) \times A\big) \arrow[urrr, bend left=20, "\mathrm{Id}_{F((A \Rightarrow B) \times A)}", "\cong"' near end] \arrow[rrrr, bend left=70, "F\mathrm{eval}_{A,B}"] \arrow[r, "\langle F\pi_1, F\pi_2 \rangle"] \arrow[d, "\alpha_{(A \Rightarrow B) \times A}"'] \arrow[dr, phantom, "\overset{\alpha^{\times}_{A \Rightarrow B,A}}{\Leftarrow}"] & F(A \Rightarrow B) \times FA \arrow[urr, "\mathrm{q}^{\times}_{(A \Rightarrow B,A)}"] \arrow[u, phantom, "\cong"] \arrow[rrr, "\mathrm{eval}_{FA,FB} \circ (\mathrm{m}^F_{A,B} \times FA)"] \arrow[d, "\alpha_{A \Rightarrow B} \times \alpha_A"] \arrow[drrr, phantom, "\overset{\alpha^{\Rightarrow}_{A,B}}{\Leftarrow}"] & & & FB \arrow[d, "\alpha_B"] \\
G\big((A \Rightarrow B) \times A\big) \arrow[drrr, bend right=20, "\mathrm{Id}_{G((A \Rightarrow B) \times A)}"', "\cong" near end] \arrow[rrrr, bend right=70, "G\mathrm{eval}_{A,B}"'] \arrow[r, "\langle G\pi_1, G\pi_2 \rangle"'] & G(A \Rightarrow B) \times GA \arrow[drr, "\mathrm{q}^{\times}_{A \Rightarrow B,B}"'] \arrow[d, phantom, "\cong"] \arrow[rrr, "\mathrm{eval}_{GA,GB} \circ (\mathrm{m}^G_{A,B} \times GA)"'] & & & GB \\
 & & & G\big((A \Rightarrow B) \times A\big) \arrow[ur, "G\mathrm{eval}_{AB}"'] \arrow[u, phantom, "\overset{\varepsilon}{\cong}"] &
\end{tikzcd}$$

We call the transformation *strong* if every $\overline{\alpha}_f$, $\alpha^{\times}_{A_1,\dots,A_n}$ and $\alpha^{\Rightarrow}_{A,B}$ is invertible.

In a cc-bicategory, every fp-transformation—and hence every pseudonatural transformation—lifts canonically to a cc-transformation: one simply inverts the coherence law to obtain a definition of $\alpha^{\Rightarrow}_{A,B}$. Moreover, by Lemma 2.2.13 every biequivalence extends canonically to a cc-pseudofunctor. Thus, in order to construct a *cc-biequivalence* between cc-bicategories—namely a biequivalence of the underlying bicategories in which the pseudofunctors are cc-pseudofunctors and the pseudonatural transformations are cc-transformations—it suffices to construct a biequivalence of the underlying bicategories (*c.f.* Lemma 4.1.16).

Lemma 5.1.9. Let $(\mathcal{B}, \Pi_n(-), \Rightarrow)$ and $(\mathcal{C}, \Pi_n(-), \Rightarrow)$ be cc-bicategories. Then there exists a biequivalence $\mathcal{B} \simeq \mathcal{C}$ if and only if there exists a cc-biequivalence $(\mathcal{B}, \Pi_n(-), \Rightarrow) \simeq (\mathcal{C}, \Pi_n(-), \Rightarrow)$. □

5.1.1 Coherence via the Yoneda embedding.

It turns out that one may refine the Yoneda-style proof of coherence for fp-bicategories given on page 81 (Proposition 4.1.8) to encompass exponentials.[1] The proof does not go through verbatim, because the exponentials in $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$ are not generally strict. The solution is to first strictify the bicategory $\mathcal{B}$ to a 2-category $\mathcal{C}$, then pass to the 2-category $[\mathcal{C}, \mathbf{Cat}]$ of 2-functors, 2-natural transformations, and modifications. This is cartesian closed as a 2-category—and hence as a bicategory—by general enriched category theory [Day70, Example 5.2].

Proposition 5.1.10. For any cc-bicategory $(\mathcal{B}, \Pi_n(-), \Rightarrow)$ there exists a strictly cartesian closed 2-category $(\mathcal{C}, \Pi_n(-), \Rightarrow)$ such that $\mathcal{B} \simeq \mathcal{C}$.

Proof. By Proposition 4.1.8 we may assume without loss of generality that $\mathcal{B}$ is a 2-category with 2-categorical products and pseudo-exponentials. It therefore admits a *2-categorical* Yoneda embedding $\mathrm{Y} : \mathcal{B} \hookrightarrow [\mathcal{B}^{\mathrm{op}}, \mathbf{Cat}]$. Let $\overline{\mathcal{B}}$ denote the closure of $\mathrm{Y}(ob(\mathcal{B}))$ under equivalences and factor the Yoneda embedding as $\mathcal{B} \xrightarrow{i} \overline{\mathcal{B}} \xrightarrow{j} [\mathcal{B}^{\mathrm{op}}, \mathbf{Cat}]$. By the 2-categorical Yoneda lemma, i is a biequivalence.

The rest of the argument runs as for Proposition 4.1.8. For any $P, Q \in \overline{\mathcal{B}}$ the strict exponential $(jP \Rightarrow jQ)$ exists in $[\mathcal{B}^{\mathrm{op}}, \mathbf{Cat}]$. But then

$$(jP \Rightarrow jQ) = \big((\mathrm{Y}i^{-1})P \Rightarrow (\mathrm{Y}i^{-1})Q\big) \simeq \mathrm{Y}\big(i^{-1}P \Rightarrow i^{-1}Q\big)$$

so the exponential $(jP \Rightarrow jQ) \in \overline{\mathcal{B}}$, as required. □

In a sense, of course, this proposition solves the problem we set ourselves in the introduction to this thesis: cc-bicategories are coherent. However, the normalisation-by-evaluation proof is valuable in itself. First, it is a new approach to higher-categorical coherence; second, the speculation that it may be refinable to a normalisation algorithm on 2-cells; and third, it makes use of machinery that will play an important role in other, further developments. We therefore keep this result in mind, but do not let it deter us from our work in the rest of this thesis.

[1] I am grateful to André Joyal for suggesting this is possible, especially so because at the time I thought it was not.

5.2 Cartesian closed (bi)clones

We shall follow the procedure of the previous two chapters, synthesising our type theory from the construction of a free biclone. The 1-categorical setting remains an enlightening starting point: in this setting, the type theory we synthesise ought to be the familiar simply-typed lambda calculus. To show this is indeed the case, we shall extend the diagram of adjunctions (4.19) on page 104 to the cartesian closed setting. The ideas involved are not especially novel; however, to the best of my knowledge they have not been presented in this style elsewhere (although Jacobs' [Jac92] shares many of the same basic insights).

5.2.1 Cartesian closed clones

Lambek [Lam89] defines a *(right) internal hom* in a multicategory $\mathbb{L}$ to be a choice of object $A \Rightarrow B$ for every $A, B \in \mathbb{L}$, together with a family of multimaps $\mathrm{eval}_{A,B} : (A \Rightarrow B), A \to B$ inducing isomorphisms

$$\begin{aligned} \mathbb{L}(\Gamma; A \Rightarrow B) &\xrightarrow{\cong} \mathbb{L}(\Gamma, A; B) \\ (h : \Gamma \to A \Rightarrow B) &\mapsto (\Gamma, A \xrightarrow{\mathrm{eval}_{A,B} \circ \langle h, \mathrm{id}_A \rangle} B) \end{aligned}$$

for every Γ, A and B. This suggests the following definition for clones (*c.f.* Definition 4.2.13).

Definition 5.2.1. A clone $(S, \mathbb{C})$ *has a (right) internal hom* if the corresponding multicategory $\mathrm{M}\mathbb{C}$ has a right internal hom. If $\mathbb{C}$ is also cartesian, we say $\mathbb{C}$ is *cartesian closed.* ◂

Example 5.2.2. The cartesian clone $\mathrm{Cl}(\mathbb{C})$ constructed from a cartesian closed category $(\mathbb{C}, \Pi_n(-), \Rightarrow)$ (recall Example 4.2.14 on page 92) is cartesian closed. The exponential of $A, B \in \mathbb{C}$ is $A \Rightarrow B$, the evaluation multimap is the evaluation map of $\mathbb{C}$, and the currying of $f : \prod_{n+1}(A_1, \dots, A_n, X) \to Y$ is the exponential transpose of

$$\textstyle\prod_2(\prod_n(A_1, \dots, A_n), X) \xrightarrow{\cong} \prod_{n+1}(A_1, \dots, A_n, X) \xrightarrow{f} Y$$

◂

Since every cartesian clone is representable, for any cartesian closed clone $(S, \mathbb{C}, \Pi_n(-), \Rightarrow)$ one obtains the following chain of natural isomorphisms for every $A_1, \dots, A_n, B, C \in S$ $(n \in \mathbb{N})$:

$$\begin{aligned} \mathbb{C}\big(\textstyle\prod_{n+1}(A_1, \dots, A_n, B); C\big) &\cong \mathbb{C}(A_1, \dots, A_n, B; C) && \text{by representability} \\ &\cong \mathbb{C}(A_1, \dots, A_n; B \Rightarrow C) && \text{by cartesian closure} \\ &\cong \mathbb{C}(\textstyle\prod_n(A_1, \dots, A_n); B \Rightarrow C) && \text{by representability} \end{aligned} \tag{5.6}$$

Thus, for any multimap $t : A_1, \dots, A_n, B \to C$ in a cartesian closed clone $(S, \mathbb{C}, \Pi_n(-), \Rightarrow)$ there exists a multimap $\lambda t : A_1, \dots, A_n \to (B \Rightarrow C)$ (called the *currying* of t), which is the unique $g : A_1, \dots, A_n \to (B \Rightarrow C)$ satisfying

$$t = \mathrm{eval}_{A,B}\Big[g[\mathsf{p}^{(1)}_{A_\bullet;B}, \dots, \mathsf{p}^{(n)}_{A_\bullet;B}], \mathsf{p}^{(n+1)}_{A_\bullet;B}\Big]$$

Observe in particular how the requirement that the isomorphisms are defined on $\mathrm{M}\mathbb{C}$—rather than on $\mathbb{C}$—abstractly enforces the use of the *weakening* operation taking $h : X_1, \dots, X_n \to Z$ to the multimap $h\Big[\mathsf{p}^{(1)}_{X_\bullet,Y}, \dots, \mathsf{p}^{(n)}_{X_\bullet,Y}\Big] : X_1, \dots, X_n, Y \to Z$.

Remark 5.2.3. For any cartesian closed clone $(S, \mathbb{C}, \Pi_n(-), \Rightarrow)$ the isomorphisms (5.6) entail that the nucleus $\overline{\mathbb{C}}$ is also cartesian closed. Thus products are given as in $(S, \mathbb{C})$, and exponentials are given by the composite natural isomorphism

$$\overline{\mathbb{C}}(X \times A, B) = \mathbb{C}(X \times A, B) \cong \mathbb{C}(X, A; B) \cong \mathbb{C}(X, A \Rightarrow B) = \overline{\mathbb{C}}(X, A \Rightarrow B) \tag{5.7}$$

However, the evaluation map $\mathrm{eval}_{A,B} : (A \Rightarrow B), A \to B$ witnessing exponentials in $\mathbb{C}$ is not a morphism in $\overline{\mathbb{C}}$. Chasing through the isomorphism (5.7), one sees that the evaluation map $(A \Rightarrow B) \times A \to B$ in $\overline{\mathbb{C}}$ is $\mathrm{eval}_{A,B}[\pi_1, \pi_2]$ and the currying of $f : X \times A \to B$ is the 1-cell $\lambda\big(X, A \xrightarrow{\mathsf{tup}(\mathsf{p}^{(1)}_{X,A}, \mathsf{p}^{(2)}_{X,A})} X \times A \xrightarrow{f} B\big)$. To see this is the case, observe first that for any $u : X \to (A \Rightarrow B)$ one has:

$$\begin{aligned}
\mathrm{eval}_{A,B}\Big[u[\mathsf{p}^{(1)}_{X,A}], \mathsf{p}^{(2)}_{X,A}\Big][\pi_1, \pi_2] &= \mathrm{eval}_{A,B}\Big[u[\mathsf{p}^{(1)}_{X,A}][\pi_1, \pi_2], \mathsf{p}^{(2)}_{X,A}[\pi_1, \pi_2]\Big] \\
&= \mathrm{eval}_{A,B}[u[\pi_1], \pi_2]
\end{aligned}$$

Next recall that for any $u : X \to Y$ in $\overline{\mathbb{C}}$ the corresponding morphism $u \times A : X \times A \to Y \times A$ is $\mathsf{tup}(u[\pi_1], \pi_2)$. Putting these components together, one sees that for any $f : X \times A \to B$,

$$\begin{aligned}
&\mathrm{eval}_{A,B}[\pi_1, \pi_2]\Big[\mathsf{tup}\Big(\lambda\big(f[\mathsf{tup}(\mathsf{p}^{(1)}_{X,A}, \mathsf{p}^{(2)}_{X,A})]\big)[\pi_1], \pi_2\Big)\Big] \\
&\qquad = \mathrm{eval}_{A,B}\Big[\lambda\big(f[\mathsf{tup}(\mathsf{p}^{(1)}_{X,A}, \mathsf{p}^{(2)}_{X,A})]\big)[\pi_1], \pi_2\Big] && \text{cartesian structure of } \mathbb{C} \\
&\qquad = \mathrm{eval}_{A,B}\Big[\lambda\big(f[\mathsf{tup}(\mathsf{p}^{(1)}_{X,A}, \mathsf{p}^{(2)}_{X,A})]\big)[\mathsf{p}^{(1)}_{X,A}], \mathsf{p}^{(2)}_{X,A}\Big][\pi_1, \pi_2] \\
&\qquad = f[\mathsf{tup}(\mathsf{p}^{(1)}_{X,A}, \mathsf{p}^{(2)}_{X,A})][\pi_1, \pi_2] && \text{exponentials in } \mathbb{C} \\
&\qquad = f
\end{aligned}$$

The final line follows by Lemma 4.2.17. On the other hand, for any $u : X \to (A \Rightarrow B)$,

$$\begin{aligned}
\lambda\big(\mathrm{eval}_{A,B}[\pi_1, \pi_2][\mathsf{tup}(u[\pi_1], \pi_2)][\mathsf{tup}(\mathsf{p}^{(1)}_{X,A}, \mathsf{p}^{(2)}_{X,A})]\big) &= \lambda\Big(\mathrm{eval}_{A,B}[u[\pi_1], \pi_2]\Big[\mathsf{tup}(\mathsf{p}^{(1)}_{X,A}, \mathsf{p}^{(2)}_{X,A})\Big]\Big) \\
&= \lambda\Big(\mathrm{eval}_{A,B}\Big[u[\mathsf{p}^{(1)}_{X,A}], \mathsf{p}^{(2)}_{X,A}\Big]\Big) \\
&= u
\end{aligned}$$

where the final line follows again from the cartesian closed structure in $(S, \mathbb{C})$. It follows that $\mathrm{eval}_{A,B}[\pi_1, \pi_2]$ is the universal arrow defining exponentials, as claimed.

This structure is not surprising: it corresponds to the cartesian closed structure on the syntactic model of the simply-typed lambda calculus, restricted to unary contexts (*e.g.* [Cro94, Theorem 4.8.4]). ◂

The following two definitions follow the schema of Chapters 3 and 4.

Definition 5.2.4. A $\Lambda^{\times,\rightarrow}$*-signature* $\mathcal{S} = (\mathfrak{B}, \mathcal{G})$ consists of

1. A set of base types $\mathfrak{B}$,
2. A multigraph $\mathcal{G}$ with nodes generated by the grammar

$$A_1, \ldots, A_n, C, D ::= B \mid \textstyle\prod_n(A_1, \ldots, A_n) \mid C \Rightarrow D \qquad (B \in \mathfrak{B}, n \in \mathbb{N}) \tag{5.8}$$

If the multigraph $\mathcal{G}$ is a graph we call the signature *unary.* A *homomorphism* of $\Lambda^{\times,\rightarrow}$-signatures $h : \mathcal{S} \to \mathcal{S}'$ is a morphism $h : \mathcal{G} \to \mathcal{G}'$ of the underlying multigraphs such that, additionally,

$$h(\textstyle\prod_n(A_1, \ldots, A_n)) = \textstyle\prod_n(hA_1, \ldots, hA_n)$$
$$h(C \Rightarrow D) = (hC \Rightarrow hD)$$

We denote the category of $\Lambda^{\times,\rightarrow}$-signatures and their homomorphisms by $\Lambda^{\times,\rightarrow}$-sig, and the full subcategory of unary $\Lambda^{\times,\rightarrow}$-signatures by $\Lambda^{\times,\rightarrow}\text{-sig}\big|_1$. ◂

Notation 5.2.5 (*c.f.* Notation 4.2.23)**.** For any $\Lambda^{\times,\rightarrow}$-signature $\mathcal{S} = (\mathfrak{B}, \mathcal{G})$ we write $\widetilde{\mathfrak{B}}$ for the set generated from $\mathfrak{B}$ by the grammar (5.8). In particular, when the signature is just a set (*i.e.* the graph $\mathcal{G}$ has no edges) we denote the signature $\mathcal{S} = (\mathfrak{B}, \mathcal{S})$ simply by $\widetilde{\mathfrak{B}}$. ◂

Definition 5.2.6. A *cartesian closed clone homomorphism* $h : (S, \mathbb{C}, \Pi_n(-), \Rightarrow) \to (T, \mathbb{D}, \Pi_n(-), \Rightarrow)$ is a cartesian clone homomorphism $(S, \mathbb{C}, \Pi_n(-)) \to (T, \mathbb{D}, \Pi_n(-))$ such that the canonical map $\lambda(h(\mathrm{eval}_{A,B})) : h(A \Rightarrow B) \to (hA \Rightarrow hB)$ is invertible. We call h *strict* if

$$h(A \Rightarrow B) = (hA \Rightarrow hB)$$
$$h(\mathrm{eval}_{A,B}) = \mathrm{eval}_{hA,hB}$$

for every $A, B \in S$. ◂

In a similar fashion, we call a cartesian closed functor *strict* if it strictly preserves exponentials and the evaluation map.

We now construct the following diagram of adjunctions, in which CCCat denotes the category of cartesian closed categories and strict cartesian closed functors and CCClone denotes the category of cartesian closed clones and strict homomorphisms. As in the preceding chapter, we implicitly restrict to cartesian structure in which $\prod_1(-)$ is the identity functor.

$$\begin{tikzcd}
 & \mathrm{CCClone} \arrow[dl, bend right, "\mathrm{forget}"'] \arrow[dr, bend left, "\overline{(-)}"] & \\
\Lambda^{\times,\rightarrow}\text{-sig} \arrow[ur, bend right=15, "\mathbb{F}\mathbb{C}\mathrm{l}^{\times,\rightarrow}(-)"'] \arrow[ur, phantom, "\top"] \arrow[dr, bend right, "\widetilde{\mathcal{L}}"'] & & \mathrm{CCCat} \arrow[ul, bend right=15, "\mathcal{P}"'] \arrow[ul, phantom, "\top"] \arrow[dl, bend left, "\mathrm{forget}"] \\
 & \Lambda^{\times,\rightarrow}\text{-sig}\big|_1 \arrow[ul, hook, bend right=15] \arrow[ul, phantom, "\perp"] \arrow[ur, bend left=15, "\mathrm{free}"] \arrow[ur, phantom, "\perp"] &
\end{tikzcd} \tag{5.9}$$

The right adjoint to the inclusion $\iota : \Lambda^{\times,\rightarrow}\text{-sig}\big|_1 \hookrightarrow \Lambda^{\times,\rightarrow}\text{-sig}$ is defined by $\widetilde{\mathcal{L}}(\mathfrak{B}, \mathcal{G}) = (\mathfrak{B}, \mathcal{L}\mathcal{G})$ for $\mathcal{L} : \mathrm{MGrph} \to \mathrm{Grph}$ the right adjoint to the inclusion $\mathrm{Grph} \hookrightarrow \mathrm{MGrph}$ (*c.f.* Lemma 4.2.24). The free-forgetful adjunction between cartesian closed categories and $\Lambda^{\times,\rightarrow}$-signatures is the classical construction of the syntactic model of the simply-typed lambda calculus over a signature [Lam80]. There are two adjunctions left to construct.

Lemma 5.2.7. The forgetful functor CCClone → $\Lambda^{\times,\rightarrow}$-sig has a left adjoint.

Proof. Define a clone $\mathbb{F}\mathbb{C}\mathrm{l}^{\times,\rightarrow}(\mathcal{S})$ over a signature $(\mathfrak{B}, \mathcal{G})$ as follows. The sorts are generated by the grammar

$$A_1, \ldots, A_n, C, D ::= B \mid \textstyle\prod_n(A_1, \ldots, A_n) \mid C \Rightarrow D \qquad (B \in \mathfrak{B}, n \in \mathbb{N})$$

The operations are those of Construction 4.2.25 (page 99) together with two additional rules:

$$\frac{}{\mathsf{eval}_{B,C} \in \mathbb{F}\mathbb{C}\mathrm{l}^{\times,\rightarrow}(\mathcal{S})(B \Rightarrow C, B; C)} \qquad \frac{t \in \mathbb{F}\mathbb{C}\mathrm{l}^{\times,\rightarrow}(\mathcal{S})(A_1, \ldots, A_n, B; C)}{\lambda t \in \mathbb{F}\mathbb{C}\mathrm{l}^{\times,\rightarrow}(\mathcal{S})(A_1, \ldots, A_n; B \Rightarrow C)}\ (n \in \mathbb{N})$$

Similarly, one extends the equational theory $\equiv$ by requiring that

- $\mathsf{eval}_{B,C}\Big[(\lambda t)[\mathsf{p}^{(1)}_{A_\bullet,B}, \ldots, \mathsf{p}^{(n)}_{A_\bullet,B}], \mathsf{p}^{(n+1)}_{A_\bullet,B}\Big] \equiv t$ for any $t : A_1, \ldots, A_n, B \to C$,
- $\lambda\Big(\mathsf{eval}_{B,C}\Big[u[\mathsf{p}^{(1)}_{A_\bullet,B}, \ldots, \mathsf{p}^{(n)}_{A_\bullet,B}], \mathsf{p}^{(n+1)}_{A_\bullet,B}\Big]\Big) \equiv u$ for any $u : A_1, \ldots, A_n \to (B \Rightarrow C)$.

It is clear $\mathbb{F}\mathbb{C}\mathrm{l}^{\times,\rightarrow}(\mathcal{S})$ is cartesian closed. To see that it is also free, let $h : \mathcal{S} \to \mathbb{D}$ be any $\Lambda^{\times,\rightarrow}$-signature homomorphism from $\mathcal{S}$ to the underlying $\Lambda^{\times,\rightarrow}$-signature of a cartesian closed clone $(T, \mathbb{D}, \Pi_n(-), \Rightarrow)$. Define a cartesian closed clone homomorphism $h^{\#} : \mathbb{F}\mathbb{C}\mathrm{l}^{\times,\rightarrow}(\mathcal{S}) \to \mathbb{D}$ by extending the definition of Lemma 4.2.27 (page 100) as follows:

$$h^{\#}(A \Rightarrow B) := (h^{\#}A \Rightarrow h^{\#}B)$$

$$h^{\#}(\mathsf{eval}_{A,B}) := \mathrm{eval}_{(h^{\#}A, h^{\#}B)}$$
$$h^{\#}(\lambda t) := \lambda(h^{\#}t)$$

For uniqueness, we already know from Lemma 4.2.27 and the definition of a cartesian closed clone homomorphism that any cartesian clone homomorphism strictly preserves all the structure, except for currying. So it suffices to show that any cartesian clone homomorphism preserves the $\lambda(-)$ mapping. Since λt is the unique multimap $g : A_1, \dots, A_n \to (B \Rightarrow C)$ such that $t = \mathrm{eval}_{B,C}\Big[g[\mathsf{p}^{(1)}_{A_\bullet,B}, \dots, \mathsf{p}^{(n)}_{A_\bullet,B}], \mathsf{p}^{(n+1)}_{A_\bullet,B}\Big]$, for any cartesian clone homomorphism $f : \mathbb{F}\mathrm{Cl}^{\times,\to}(\mathcal{S}) \to \mathbb{D}$ one has

$$\begin{aligned} f(t) &= f\Big(\mathsf{eval}_{B,C}\Big[(\lambda t)[\mathsf{p}^{(1)}_{A_\bullet,B}, \dots, \mathsf{p}^{(n)}_{A_\bullet,B}], \mathsf{p}^{(n+1)}_{A_\bullet,B}\Big]\Big) \\ &= \mathrm{eval}_{fB,fC}\Big[f(\lambda t)\Big[\mathsf{p}^{(1)}_{fA_\bullet,fB}, \dots, \mathsf{p}^{(n)}_{fA_\bullet,fB}\Big], \mathsf{p}^{(n+1)}_{fA_\bullet,fB}\Big] \end{aligned}$$

it follows that $f(\lambda t) = \lambda f(t)$ for every $t : A_1, \dots, A_n, B \to (B \Rightarrow C)$, as required. □

It remains to construct the adjunction CCClone $\leftrightarrows$ CCCat.

Lemma 5.2.8. The functor $\overline{(-)}$: CCClone → CCCat restricting a cartesian closed clone to its nucleus has a left adjoint.

Proof. Consider the functor $\mathcal{P}$: CartCat → CartClone defined in Lemma 4.2.28. This restricts to a functor CCCat → CCClone. Explicitly, the evaluation map in $\mathcal{P}\mathbb{C}$ is the evaluation map $\mathrm{eval}_{A,B}$ in $\mathbb{C}$ and for any $f : X_1, \dots, X_n \to (A \Rightarrow B)$ the composite $\mathrm{eval}_{A,B}\Big[f[\mathsf{p}^{(1)}_{X_\bullet,A}, \dots, \mathsf{p}^{(n)}_{X_\bullet,A}], \mathsf{p}^{(n+1)}_{X_\bullet,A}\Big]$ in $\mathcal{P}\mathbb{C}$ is the composite $\mathrm{eval}_{A,B} \circ \langle f \circ \langle \pi_1, \dots, \pi_n \rangle, \pi_{n+1} \rangle = \mathrm{eval}_{A,B} \circ (f \times A) \circ \langle \langle \pi_1, \dots, \pi_n \rangle, \pi_{n+1} \rangle$ in $\mathbb{C}$. The currying of $g : X_1, \dots, X_n, A \to B$ is the currying (in $\mathbb{C}$) of the morphism

$$\lambda\big(\textstyle\prod_{i=1}^{n} X_i \times A \xrightarrow{\cong} X_1 \times \dots \times X_n \times A \xrightarrow{g} B\big)$$

Now suppose that $F : \mathbb{C} \to \overline{\mathbb{D}}$ is a strict cartesian closed functor. Define $F^{\#}$ as the free cartesian extension of F from Lemma 4.2.28:

$$F^{\#}(X_1, \dots, X_n \xrightarrow{t} Y) := \big(FX_1, \dots, FX_n \xrightarrow{\psi_{FX_\bullet}(\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)})} \textstyle\prod_{i=1}^{n} FX_i = F(\prod_{i=1}^{n} X_i) \xrightarrow{Ft} FY\big)$$

To see that $F^{\#}$ preserves the evaluation map, note that—since F is a strict cartesian closed functor—the equation $F(\mathrm{eval}_{A,B}) = \mathrm{eval}_{FA,FB}[\pi_1, \pi_2]$ must hold by Remark 5.2.3. It follows that

$$\begin{aligned} F^{\#}(\mathrm{eval}_{A,B}) &= \mathrm{eval}_{FA,FB}[\pi_1, \pi_2]\,\big[\psi_{FX_\bullet}(\mathsf{p}^{(1)}, \dots, \mathsf{p}^{(n)})\big] \\ &= \mathrm{eval}_{FA,FB}\Big[\mathsf{p}^{(1)}_{FA \Rightarrow FB, FA}, \mathsf{p}^{(2)}_{FA \Rightarrow FB, FA}\Big] && \text{by equation (4.13) on page 93} \\ &= \mathrm{eval}_{FA,FB} \end{aligned}$$

as required. The proof of uniqueness is exactly as in the cartesian case. □

This completes the construction of the diagram of adjunctions (5.9). As for the diagram of adjunctions (4.19) for cartesian strucure, it is easy to see that the outer edges of (5.9) commute and that $\overline{(-)} \circ \mathcal{P} = \mathrm{id}_{\mathrm{CCCat}}$. One thereby obtains the following chain of natural isomorphisms (*c.f.* equation (4.20)), in which we write $\mathbb{F}\mathbb{C}\mathrm{at}^{\times,\to}(\mathcal{S})$ for the free cartesian closed category on a unary signature $\mathcal{S}$:

$$\mathrm{CCCat}(\mathbb{F}\mathbb{C}\mathrm{at}^{\times,\to}(\mathcal{S}), \mathbb{C}) = \mathrm{CCCat}\Big(\overline{\mathcal{P}(\mathbb{F}\mathbb{C}\mathrm{at}^{\times,\to}(\mathcal{S}))}, \mathbb{C}\Big) \cong \mathrm{CCCat}\Big(\overline{\mathbb{F}\mathbb{C}\mathrm{l}^{\times,\to}(\iota\mathcal{S})}, \mathbb{C}\Big) \tag{5.10}$$

It follows that the free cartesian closed category on a $\Lambda^{\times,\to}$-signature is described by restricting the deductive system of Lemma 5.2.7 to unary contexts.

Remark 5.2.9. In the preceding lemma we rely on the equation $\mathrm{eval}_{FA,FB}[\mathsf{p}^{(1)}_{(A \Rightarrow B,A)}, \mathsf{p}^{(2)}_{(A \Rightarrow B,A)}] = \mathrm{eval}_{FA,FB}$ to show that $F^{\#}$ is strictly cartesian closed. In the bicategorical setting, where this equality is generally only an isomorphism, the argument fails. As we shall see, the free cc-bicategory on a signature (in the strict sense of *free* we have been using throughout) is not obtained by restricting the free cartesian biclone on the same signature. ◀

Cartesian closed clones and the simply-typed lambda calculus. Let us examine how one extracts the simply-typed lambda calculus from the internal language of $\mathbb{F}\mathbb{C}\mathrm{l}^{\times,\to}(\mathcal{S})$ (defined in Lemma 5.2.8). The $\mathsf{eval}_{B,C}$ multimap becomes an application operation on variables:

$$\frac{}{f : B \Rightarrow C, x : B \vdash \mathsf{app}(f, x) : C}$$

The weakening operation $t \mapsto t\Big[\mathsf{p}^{(1)}_{A_\bullet,B}, \dots, \mathsf{p}^{(n)}_{A_\bullet,B}\Big]$ is the following form of the usual substitution lemma:

$$\frac{x_1 : A_1, \dots, x_n : A_n \vdash t : C \qquad x_1 : A_1, \dots, x_n : A_n, y : B \vdash t : C}{x_1 : A_1, \dots, x_n : A_n, y : B \vdash t[x_1/x_1, \dots, x_n/x_n] : C}$$

This mirrors the construction in $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}$ and its extensions, where weakening arises from explicit substitutions corresponding to inclusions of contexts.

The $\lambda(-)$ mapping is the usual lambda abstraction operation, and the two equations become the following rules for every $x_1 : A_1, \dots, x_n : A_n, x : A \vdash t : B$ and $x_1 : A_1, \dots, x_n : A_n \vdash u : A \Rightarrow B$:

$$\mathsf{app}((\lambda x.t)[x_1/x_1, \dots, x_n/x_n], x) \quad \text{and} \quad \lambda x.\mathsf{app}(u[x_1/x_1, \dots, x_n/x_n], x) = u$$

As we saw in Section 4.2.2, these rules extend to rules on all terms in the presence of the meta-operation of capture avoiding substitution. Thus, we recover the usual $\beta\eta$-laws of the simply-typed lambda calculus. The diagram of adjunctions (5.9), together with the isomorphism (5.10), then expresses the usual free property of the unary-context syntactic model [Cro94, Chapter 4].

Our aim in what follows is to define cartesian closed biclones, construct the free instance to obtain a diagram matching (5.9), and use this to extract a type theory in the same way as we have just sketched for the simply-typed lambda calculus. As for products, our insistence on strict universal properties makes the full diagram impossible to replicate (recall Example 4.2.63 on page 126). Nonetheless, we shall see that a version of it exists up to biequivalence.

5.2.2 Cartesian closed biclones

The definitions of the previous section bicategorify in the way one would expect.

Definition 5.2.10.

1. A *(right) closed* bi-multicategory is a bi-multicategory $\mathcal{M}$ equipped with the following data for every $A, B \in \mathcal{M}$:

   a) A chosen object $A \Rightarrow B$,

   b) A chosen multimap $\mathrm{eval}_{A,B} : (A \Rightarrow B), A \to B$,

   c) For every sequence of objects Γ in $\mathcal{M}$, an adjoint equivalence

$$\begin{tikzcd}
\mathcal{M}(\Gamma; A \Rightarrow B) \arrow[r, bend left, "\mathrm{eval}_{A,B} \circ \langle (-), \mathrm{Id}_A \rangle"] \arrow[r, phantom, "\perp\simeq"] & \mathcal{M}(\Gamma, A; B) \arrow[l, bend left, "\lambda"]
\end{tikzcd}$$

   specified by choosing a universal arrow with components $\varepsilon_t : \mathrm{eval}_{A,B} \circ \langle \lambda t, \mathrm{Id}_A \rangle \Rightarrow t$.

2. A *(right) closed biclone* is a biclone $(S, \mathcal{C})$ equipped with a choice of right-closed structure on the corresponding bi-multicategory $\mathrm{M}\mathcal{C}$.

3. A *cartesian closed* biclone is a biclone equipped with a choice of both cartesian structure and right-closed structure. ◂

Explicitly, a cartesian closed biclone is defined by the following universal property. For every sequence of objects $\Gamma := (A_1, \ldots, A_n)$ and multimap $t : \Gamma, A \to B$ there exists a multimap $\lambda t : \Gamma \to (A \Rightarrow B)$ and a 2-cell $\varepsilon_t : \mathrm{eval}_{A,B}\big[(\lambda t)[\mathsf{p}^{(1)}_{A_\bullet,B}, \ldots, \mathsf{p}^{(n)}_{A_\bullet,B}], \mathsf{p}^{(n+1)}_{A_\bullet,B}\big] \Rightarrow t$. This 2-cell is universal in the sense that for every $u : \Gamma \to (A \Rightarrow B)$ and $\alpha : \mathrm{eval}_{A,B}\big[u[\mathsf{p}^{(1)}_{A_\bullet,B}, \ldots, \mathsf{p}^{(n)}_{A_\bullet,B}], \mathsf{p}^{(n+1)}_{A_\bullet,B}\big] \Rightarrow t$ there exists a 2-cell $\mathsf{e}^\dagger(\alpha) : u \Rightarrow \lambda t$, unique such that

$$\begin{tikzcd}
\mathrm{eval}_{A,B}\Big[u[\mathsf{p}^{(1)}_{A_\bullet,B}, \ldots, \mathsf{p}^{(n)}_{A_\bullet,B}], \mathsf{p}^{(n+1)}_{A_\bullet,B}\Big] \arrow[rr, "\mathrm{eval}_{A,B}\left[\mathsf{e}^\dagger(\alpha)[\mathsf{p}^{(1)}_{A_\bullet,B}, \cdots, \mathsf{p}^{(n)}_{A_\bullet,B}], \mathsf{p}^{(n+1)}_{A_\bullet,B}\right]"] \arrow[dr, "\alpha"'] & & \mathrm{eval}_{A,B}\Big[(\lambda t)[\mathsf{p}^{(1)}_{A_\bullet,B}, \ldots, \mathsf{p}^{(n)}_{A_\bullet,B}], \mathsf{p}^{(n+1)}_{A_\bullet,B}\Big] \arrow[dl, "\varepsilon_t"] \\
& t &
\end{tikzcd} \tag{5.11}$$

Moreover, since every cartesian biclone is representable (Theorem 4.2.51), one also obtains a sequence of pseudonatural adjoint equivalences lifting (5.6) to biclones:

$$\begin{aligned} \mathcal{C}\big(\textstyle\prod_{n+1}(A_1, \dots, A_n, B); C\big) &\simeq \mathcal{C}(A_1, \dots, A_n, B; C) \\ &\simeq \mathcal{C}(A_1, \dots, A_n; B \Rightarrow C) \\ &\simeq \mathcal{C}\big(\textstyle\prod_n(A_1, \dots, A_n); B \Rightarrow C\big) \end{aligned} \tag{5.12}$$

It follows that, if $(S, \mathcal{C})$ is cartesian closed, then so is its nucleus $\overline{\mathcal{C}}$.

Remark 5.2.11. We saw in Remark 5.2.3 that the evaluation map witnessing cartesian closed structure in the nucleus $\overline{\mathbb{C}}$ of a cartesian closed clone $(S, \mathbb{C}, \Pi_n(-), \Rightarrow)$ is not the evaluation multimap in $\mathbb{C}$. Similarly, chasing through the equivalences (5.12) one sees that the biuniversal arrow witnessing exponentials in the nucleus $\overline{\mathcal{C}}$ of a cartesian closed biclone $(S, \mathcal{C}, \Pi_n(-), \Rightarrow)$ is $\mathsf{eval}_{A,B}[\pi_1, \pi_2] : A \times (A \Rightarrow B) \to B$ and the currying of $f : X \times A \to B$ is $\lambda\Big(f[\mathsf{tup}(\mathsf{p}^{(1)}_{X,A}, \mathsf{p}^{(2)}_{X,A})]\Big)$. To see this defines an exponential, one can replace each of the equalities in the proof of Remark 5.2.3 to construct natural isomorphisms

$$\begin{aligned} \mathrm{eval}_{A,B}\Big[(-)[\mathsf{p}^{(1)}_{X,A}], \mathsf{p}^{(2)}_{X,A}\Big]\,[\pi_1, \pi_2] &\cong \mathrm{id}_{\mathcal{C}(X \times A, B)} \\ \lambda\big(\mathrm{eval}_{A,B}[\pi_1, \pi_2][\mathsf{tup}((-)[\pi_1], \pi_2)][\mathsf{tup}(\mathsf{p}^{(1)}_{X,A}, \mathsf{p}^{(2)}_{X,A})]\big) &\cong \mathrm{id}_{\mathcal{C}(X, A \Rightarrow B)} \end{aligned}$$

witnessing an equivalence, which may be promoted to the required adjoint equivalence without changing the functors (see *e.g.* [Mac98, § IV.4]). ◂

Example 5.2.12 (*c.f.* Example 5.2.2)**.** The cartesian biclone $\mathrm{Bicl}(\mathcal{B})$ constructed from a cc-bicategory $(\mathcal{B}, \Pi_n(-), \Rightarrow)$ (recall Example 4.2.45 on page 115) is cartesian closed. The precise statement requires some juggling of products, for which we introduce the following notation. For any $A_1, \dots, A_n, B \in \mathcal{B}$ $(n \in \mathbb{N})$ there exists a canonical equivalence

$$e_{A_\bullet, B} : \textstyle\prod_{n+1}(A_1, \dots, A_n, B) \leftrightarrows \prod_2\left(\prod_n(A_1, \dots, A_n), B\right) : e^{\star}_{A_\bullet, B} \tag{5.13}$$

where $e_{A_\bullet, B} := \langle\langle \pi_1, \dots, \pi_n \rangle, \pi_{n+1}\rangle$ and $e^{\star}_{A_\bullet, B} := \langle \pi_1 \circ \pi_1, \dots, \pi_n \circ \pi_1, \pi_2 \rangle$. The witnessing 2-cells

$$\begin{aligned} \mathsf{w}_{A_\bullet, B} &: e^{\star}_{A_\bullet, B} \circ e_{A_\bullet, B} \Rightarrow \mathrm{Id}_{\prod_{n+1}(A_1, \dots, A_n, B)} \\ \mathsf{v}_{A_\bullet, B} &: \mathrm{Id}_{\prod_n(A_1, \dots, A_n) \times B} \Rightarrow e_{A_\bullet, B} \circ e^{\star}_{A_\bullet, B} \end{aligned} \tag{5.14}$$

are defined by the two diagrams below:

$$\begin{tikzcd}
\left\langle \pi_1 \circ \pi_1, \ \ldots \ , \pi_n \circ \pi_1, \pi_2 \right\rangle \circ \left\langle \left\langle \pi_1, \ \ldots \ , \pi_n \right\rangle, \pi_{n+1} \right\rangle \arrow[r, "\mathsf{w}_{A_\bullet, B}"] \arrow[d, "\mathsf{post}"'] & \mathrm{Id}_{\prod_{n+1}(A_1, \ldots, A_n, B)} \\
\left\langle (\pi_1 \circ \pi_1) \circ e_{A_\bullet, B}, \ \ldots \ , (\pi_n \circ \pi_1) \circ e_{A_\bullet, B}, \pi_2 \circ e_{A_\bullet, B} \right\rangle \arrow[d, "\cong"'] & \left\langle \pi_1, \ \ldots \ , \pi_n, \pi_{n+1} \right\rangle \arrow[u, "\widehat{\varsigma\mathrm{Id}}^{-1}"'] \\
\left\langle \pi_1 \circ \left(\pi_1 \circ e_{A_\bullet, B}\right), \ \ldots \ , \pi_n \circ \left(\pi_1 \circ e_{A_\bullet, B}\right), \pi_2 \circ e_{A_\bullet, B} \right\rangle \arrow[r, "\langle \pi_1 \circ \varpi^{(1)}, \ldots \pi_n \circ \varpi^{(1)}, \varpi^{(2)} \rangle"'] & \left\langle \pi_1 \circ \langle \pi_\bullet \rangle, \ \ldots \ , \pi_n \circ \langle \pi_\bullet \rangle, \pi_{n+1} \right\rangle \arrow[u, "\langle \varpi^{(1)}, \ldots, \varpi^{(n)}, \pi_{n+1} \rangle"']
\end{tikzcd}$$

$$\begin{tikzcd}
\mathrm{Id}_{\prod_n(A_1, \ldots, A_n) \times B} \arrow[r, "\mathsf{v}_{A_\bullet, B}"] \arrow[d, "\widehat{\varsigma}_{\mathrm{Id}}"'] & \left\langle \left\langle \pi_1, \ \ldots \ , \pi_n \right\rangle, \pi_{n+1} \right\rangle \circ e^{\star}_{A_\bullet, B} \\
\left\langle \pi_1, \pi_2 \right\rangle \arrow[d, "\cong"'] & \left\langle \left\langle \pi_1, \ \ldots \ , \pi_n \right\rangle \circ e^{\star}_{A_\bullet, B}, \pi_{n+1} \circ e^{\star}_{A_\bullet, B} \right\rangle \arrow[u, "\mathsf{post}^{-1}"'] \\
\left\langle \mathrm{Id}_{\prod_n(A_1, \ldots, A_n)} \circ \pi_1, \pi_2 \right\rangle \arrow[d, "\langle \widehat{\varsigma}_{\mathrm{Id}} \circ \pi_1, \pi_2 \rangle"'] & \left\langle \left\langle \pi_1 \circ e^{\star}_{A_\bullet, B}, \ \ldots \ , \pi_n \circ e^{\star}_{A_\bullet, B} \right\rangle, \pi_{n+1} \circ e^{\star}_{A_\bullet, B} \right\rangle \arrow[u, "\langle \mathsf{post}^{-1}, \pi_{n+1} \circ e^{\star} \rangle"'] \\
\left\langle \left\langle \pi_1, \ \ldots \ , \pi_n \right\rangle \circ \pi_1, \pi_2 \right\rangle \arrow[r, "\langle \mathsf{post}, \pi_2 \rangle"'] & \left\langle \left\langle \pi_\bullet \circ \pi_1 \right\rangle, \pi_2 \right\rangle \arrow[u, "\left\langle \langle \varpi^{(-1)}, \ldots, \varpi^{(-n)} \rangle, \varpi^{(-(n+1))} \right\rangle"']
\end{tikzcd}$$

Here $\widehat{\varsigma}_{\mathrm{Id}_X}$ abbreviates the following composite:

$$\widehat{\varsigma}_{\mathrm{Id}_X} := \mathrm{Id}_X \xRightarrow{\varsigma_{\mathrm{Id}_X}} \langle \pi_1 \circ \mathrm{Id}_X, \ \ldots \ , \pi_n \circ \mathrm{Id}_X \rangle \overset{\cong}{\Rightarrow} \langle \pi_1, \ \ldots \ , \pi_n \rangle \tag{5.15}$$

The exponential of $A, B \in \mathcal{B}$ is $A \Rightarrow B$, the evaluation multimap is the evaluation map of $\mathcal{B}$, and the currying of $f : \prod_{n+1}(A_1, \ldots, A_n, X) \to Y$ is the exponential transpose of

$$\prod\nolimits_2(\prod\nolimits_n(A_1, \ldots, A_n), X) \xrightarrow[\simeq]{e^{\star}_{A_\bullet, X}} \prod\nolimits_{n+1}(A_1, \ldots, A_n, X) \xrightarrow{f} Y$$

The counit ε_f is the following composite:

$$\begin{tikzcd}
\mathrm{eval}_{X,Y} \circ \left\langle \lambda(f \circ e^{\star}_{A_\bullet, X}) \circ \langle \pi_1, \ \ldots \ , \pi_n \rangle, \pi_{n+1} \right\rangle \arrow[r, "\varepsilon_f"] \arrow[d, "\cong"'] & f \\
\mathrm{eval}_{X,Y} \circ \left\langle \lambda(f \circ e^{\star}_{A_\bullet, X}) \circ \langle \pi_1, \ \ldots \ , \pi_n \rangle, \mathrm{Id}_X \circ \pi_{n+1} \right\rangle \arrow[d, "\mathrm{eval} \circ \mathsf{fuse}^{-1}"'] & f \circ \mathrm{Id}_{\prod(A_\bullet) \times X} \arrow[u, "\cong"'] \\
\mathrm{eval}_{X,Y} \circ \left(\left(\lambda(f \circ e^{\star}_{A_\bullet, X}) \times X \right) \circ e_{A_\bullet, X} \right) \arrow[d, "\cong"'] & f \circ (e^{\star}_{A_\bullet, X} \circ e_{A_\bullet, X}) \arrow[u, "f \circ \mathsf{w}_{A_\bullet, X}"'] \\
\left(\mathrm{eval}_{X,Y} \circ \left(\lambda(f \circ e^{\star}_{A_\bullet, X}) \times X \right) \right) \circ e_{A_\bullet, X} \arrow[r, "\varepsilon_{(f \circ e^{\star})} \circ e_{A_\bullet, X}"'] & (f \circ e^{\star}_{A_\bullet, X}) \circ e_{A_\bullet, X} \arrow[u, "\cong"']
\end{tikzcd}$$

For any 1-cell $g : \prod_n(A_1, \dots, A_n) \to (X \Rightarrow Y)$ and 2-cell $\alpha : \mathrm{eval}_{X,Y} \circ \langle g \circ \langle \pi_1, \dots, \pi_n \rangle, \pi_{n+1} \rangle \Rightarrow f$ the corresponding mediating 2-cell $g \Rightarrow \lambda(f \circ e^\star_{A_\bullet,X})$ is $\mathsf{e}^\dagger(\alpha^\circ)$, for α° defined by the diagram below.

$$\begin{tikzcd}
\mathrm{eval}_{X,Y} \circ (g \times X) \arrow[r, "\alpha^\circ"] \arrow[d, "\cong"'] & f \circ e^\star_{A_\bullet,X} \\
(\mathrm{eval}_{X,Y} \circ (g \times X)) \circ \mathrm{Id}_{\prod_2((\prod_n A_\bullet),B)} \arrow[d, "\mathrm{eval} \circ (g \times X) \circ \mathsf{v}_{\prod_2((\prod_n A_\bullet),B)}"'] & \\
(\mathrm{eval}_{X,Y} \circ (g \times X)) \circ \left(e_{A_\bullet,X} \circ e^\star_{A_\bullet,X}\right) \arrow[d, "\cong"'] & \\
(\mathrm{eval}_{X,Y} \circ ((g \times X)) \circ e_{A_\bullet,X}) \circ e^\star_{A_\bullet,X} \arrow[d, "\mathrm{eval} \circ \mathsf{fuse} \circ e^\star"'] & \\
(\mathrm{eval}_{X,Y} \circ \langle g \circ \langle \pi_1, \dots, \pi_n \rangle, \mathrm{Id}_X \circ \pi_{n+1} \rangle) \circ e^\star_{A_\bullet,X} \arrow[r, "\cong"'] & (\mathrm{eval}_{X,Y} \circ \langle g \circ \langle \pi_\bullet \rangle, \pi_{n+1} \rangle) \circ e^\star_{A_\bullet,X} \arrow[uuuu, "\alpha \circ e^\star"']
\end{tikzcd}$$

◂

The free cartesian closed biclone. In Chapters 3 and 4 we synthesised the required type theory from two principles: first, an appropriate notion of biclone, and second, the fact that the internal language of those biclones—when each rule is restricted to unary contexts—gives rise to an internal language for the corresponding bicategories. For the cartesian closed case, we cannot restrict every rule of the internal language to unary contexts without also discarding all curried morphisms (lambda abstractions). Nonetheless we can show that the nucleus of the free cartesian closed biclone is the free cartesian closed bicategory *up to biequivalence*. Thus, one obtains the internal language of cartesian closed bicategories (in a bicategorical sense) by synthesising the internal language of cartesian closed biclones.

We shall begin by defining an appropriate notion of signature and (strict) pseudofunctors of cartesian closed biclones. Then we shall construct the adjunctions of the following diagram, in which we write CCBiclone for the category of cartesian closed biclones and strict pseudofunctors and cc-Bicat for the category of cc-bicategories and strict pseudofunctors.

$$\begin{tikzcd}
& \mathrm{CCBiclone} \arrow[dl, bend right, "\mathrm{forget}"'] & \\
\Lambda^{\times,\to}_{\mathrm{ps}}\text{-sig} \arrow[ur, bend right, "\mathcal{FC}l^{\times,\to}(-)"'] \arrow[dr, bend right, "\tilde{\mathcal{L}}"'] \arrow[r, phantom, shift left=6, "\top"] \arrow[r, phantom, shift right=6, "\perp"] & & \text{cc-Bicat} \arrow[dl, bend left, "\mathrm{forget}"] \\
& \Lambda^{\times,\to}_{\mathrm{ps}}\text{-sig}\big|_1 \arrow[ul, bend right] \arrow[ur, bend left, "\mathcal{FB}ct^{\times,\to}(-)"] \arrow[r, phantom, shift left=6, "\perp"] &
\end{tikzcd} \tag{5.16}$$

Thereafter we shall extract our type theory $\Lambda^{\times,\to}_{\mathrm{ps}}$ from the free cartesian closed biclone over a signature, and use this to show that the nucleus of the free cartesian closed biclone is biequivalent to the free cc-bicategory over the same (unary) signature.

Definition 5.2.13. A $\Lambda^{\times,\to}_{\mathrm{ps}}$-*signature* $\mathcal{S} = (\mathfrak{B}, \mathcal{G})$ consists of

1. A set of base types $\mathfrak{B}$,
2. A 2-multigraph $\mathcal{G}$, with nodes generated by the grammar

$$A_1, \dots, A_n, C, D ::= B \mid \textstyle\prod_n(A_1, \dots, A_n) \mid C \Rightarrow D \qquad (B \in \mathfrak{B}, n \in \mathbb{N}) \tag{5.17}$$

If $\mathcal{G}$ is a 2-graph we call the signature *unary.* A *homomorphism* of $\Lambda^{\times,\to}_{\mathrm{ps}}$-signatures $h : \mathcal{S} \to \mathcal{S}'$ is a morphism $h : \mathcal{G} \to \mathcal{G}'$ of the underlying multigraphs such that

$$h(\textstyle\prod_n(A_1, \dots, A_n)) = \prod_n(hA_1, \dots, hA_n) \quad \text{and} \quad h(C \Rightarrow D) = (hC \Rightarrow hD)$$

for all $A_1, \dots, A_n, C, D \in \mathcal{G}_0$ $(n \in \mathbb{N})$. We denote the category of $\Lambda^{\times,\to}_{\mathrm{ps}}$-signatures and their homomorphisms by $\Lambda^{\times,\to}_{\mathrm{ps}}$-sig, and the full subcategory of unary $\Lambda^{\times,\to}_{\mathrm{ps}}$-signatures by $\Lambda^{\times,\to}_{\mathrm{ps}}\text{-sig}\big|_1$. $\blacktriangleleft$

Notation 5.2.14 (*c.f.* Notation 5.2.5)**.** For a $\Lambda^{\times,\to}_{\mathrm{ps}}$-signature $\mathcal{S} = (\mathfrak{B}, \mathcal{G})$, we write $\widetilde{\mathfrak{B}}$ for the set generated from $\mathfrak{B}$ by the grammar (5.17). In particular, when the signature is just a set (*i.e.* the graph $\mathcal{G}$ has no edges) we denote the signature $\mathcal{S} = (\mathfrak{B}, \mathcal{G})$ simply by $\widetilde{\mathfrak{B}}$. $\blacktriangleleft$

The embedding $\iota : \Lambda^{\times}\text{-sig}\big|_1 \hookrightarrow \Lambda^{\times}$-sig has a right adjoint by an argument similar to that for Lemma 4.2.24 (*c.f.* also Lemma 4.2.55).

The definition of cartesian closed pseudofunctor follows the template given by cartesian pseudofunctors of biclones, while the construction of the free cartesian closed biclone on a $\Lambda^{\times,\to}_{\mathrm{ps}}$-signature echoes that for the free cartesian closed clone on a $\Lambda^{\times,\to}$-signature (Lemma 5.2.7).

Definition 5.2.15. Let $(S, \mathcal{C}, \Pi_n(-), \Rightarrow)$ and $(T, \mathcal{D}, \Pi_n(-), \Rightarrow)$ be cartesian closed biclones. A *cartesian closed pseudofunctor* $(F, \mathrm{q}^{\times}, \mathrm{q}^{\Rightarrow}) : (S, \mathcal{C}, \Pi_n(-), \Rightarrow) \to (T, \mathcal{D}, \Pi_n(-), \Rightarrow)$ is a cartesian pseudofunctor $(F, \mathrm{q}^{\times}) : (S, \mathcal{C}, \Pi_n(-)) \to (T, \mathcal{C}, \Pi_n(-))$ equipped with a choice of equivalence $\mathrm{m}_{A,B} : F(A \Rightarrow B) \leftrightarrows FA \Rightarrow FB : \mathrm{q}^{\Rightarrow}_{A,B}$ for every $A, B \in S$, where $\mathrm{m}_{A,B} := \lambda\big(F\mathrm{eval}_{A,B}\big)$. We call $(F, \mathrm{q}^{\times}, \mathrm{q}^{\Rightarrow})$ *strict* if $(F, \mathrm{q}^{\times})$ is a strict cartesian pseudofunctor such that

$$\begin{aligned}
F(A \Rightarrow B) &= (FA \Rightarrow FB) \\
F(\mathrm{eval}_{A,B}) &= \mathrm{eval}_{FA,FB} \\
F(\lambda t) &= \lambda(Ft) \\
F(\varepsilon_t) &= \varepsilon_{Ft} \\
\mathrm{q}^{\Rightarrow}_{A,B} &= \mathrm{Id}_{FA \Rightarrow FB}
\end{aligned}$$

and the isomorphisms witnessing the adjoint equivalences are the canonical 2-cells

$$\mathrm{Id}_{(FA \Rightarrow FB)} \overset{\eta_{\mathrm{Id}}}{\Longrightarrow} \lambda\Big(\mathrm{eval}_{FA,FB}\Big[\mathrm{Id}_{(FA \Rightarrow FB)}[\mathsf{p}^{(1)}_{(FA \Rightarrow FB),FA}], \mathsf{p}^{(2)}_{(FA \Rightarrow FB),FA}\Big]\Big) \overset{\cong}{\Rightarrow} \lambda(\mathrm{eval}_{FA,FB})$$

obtained from the unit and the canonical structural isomorphism. $\blacktriangleleft$

For the construction of the free cc-biclone, it will be useful to introduce some notation. For $t : A \to B$ we define $t \times X := \mathsf{tup}(t[\pi_1], \mathrm{Id}_X[\pi_2]) : \prod_2(A, X) \to \prod_2(B, X)$, and similarly on 2-cells.

Construction 5.2.16. For any $\Lambda^{\times,\to}_{\mathrm{ps}}$-signature $\mathcal{S}$, define a cartesian closed biclone $\mathcal{FC}l^{\times,\to}(\mathcal{S})$ with sorts generated by the grammar

$$A_1, \ldots, A_n, C, D ::= B \mid \textstyle\prod_n(A_1, \ldots, A_n) \mid C \Rightarrow D \qquad (B \in \mathfrak{B}, n \in \mathbb{N})$$

by extending Construction 4.2.58 (page 124) with the following rules:

$$\frac{}{\mathsf{eval}_{B,C} \in \mathcal{FC}l^{\times,\to}(\mathcal{S})(B \Rightarrow C, B; C)} \qquad \frac{t \in \mathcal{FC}l^{\times,\to}(\mathcal{S})(A_1, \ldots, A_n, B; C)}{\lambda t \in \mathcal{FC}l^{\times,\to}(\mathcal{S})(A_1, \ldots, A_n; B \Rightarrow C)}$$

$$\frac{t \in \mathcal{FC}l^{\times,\to}(\mathcal{S})(A_1, \ldots, A_n, B; C)}{\varepsilon_t \in \mathcal{FC}l^{\times,\to}(\mathcal{S})(A_1, \ldots, A_n, B; C)\Big(\mathsf{eval}_{B,C}\Big[(\lambda t)[\mathsf{p}^{(1)}_{A_\bullet,B}, \ldots, \mathsf{p}^{(n)}_{A_\bullet,B}], \mathsf{p}^{(n+1)}_{A_\bullet,B}\Big], t\Big)}$$

$$\frac{\begin{array}{c} u \in \mathcal{FC}l^{\times,\to}(\mathcal{S})(A_1, \ldots, A_n; B \Rightarrow C) \\ \alpha \in \mathcal{FC}l^{\times,\to}(\mathcal{S})(A_1, \ldots, A_n, B; C)\Big(\mathsf{eval}_{B,C}\Big[u[\mathsf{p}^{(1)}_{A_\bullet,B}, \ldots, \mathsf{p}^{(n)}_{A_\bullet,B}], \mathsf{p}^{(n+1)}_{A_\bullet,B}\Big], t\Big) \end{array}}{\mathsf{e}^\dagger(\alpha) \in \mathcal{FC}l^{\times,\to}(\mathcal{S})(A_1, \ldots, A_n; A \Rightarrow B)(u, \lambda t)}$$

The equational theory $\equiv$ is that of Construction 4.2.58, extended by requiring that

- For every $\alpha : \mathsf{eval}_{B,C}\Big[u[\mathsf{p}^{(1)}_{A_\bullet,B}, \ldots, \mathsf{p}^{(n)}_{A_\bullet,B}], \mathsf{p}^{(n+1)}_{A_\bullet,B}\Big] \Rightarrow t : A_1, \ldots, A_n, B \to C$,

$$\alpha \equiv \varepsilon_t \bullet \mathsf{eval}_{B,C}\Big[\mathsf{e}^\dagger(\alpha)[\mathsf{p}^{(1)}_{A_\bullet,B}, \ldots, \mathsf{p}^{(n)}_{A_\bullet,B}], \mathsf{p}^{(n+1)}_{A_\bullet,B}\Big]$$

- For every $\gamma : u \Rightarrow \lambda t : A_1, \ldots, A_n \to (A \Rightarrow B)$,

$$\gamma \equiv \mathsf{e}^\dagger\Big(\varepsilon_t \bullet \mathsf{eval}_{B,C}\Big[\gamma[\mathsf{p}^{(1)}_{A_\bullet,B}, \ldots, \mathsf{p}^{(n)}_{A_\bullet,B}], \mathsf{p}^{(n+1)}_{A_\bullet,B}\Big]\Big)$$

- If $\alpha \equiv \alpha' : \mathsf{eval}_{B,C}[u \times B] \Rightarrow t : X_1, \ldots, X_n, B \to C$ then $\mathsf{e}^\dagger(\alpha) \equiv \mathsf{e}^\dagger(\alpha')$.

Finally we require that every ε_t and $\mathsf{e}^\dagger(\mathrm{id}_{\mathsf{eval}[\prod_2(u,B)]})$ is invertible. $\blacktriangleleft$

It follows that for any 2-cell $\alpha : \mathsf{eval}_{B,C}\Big[u[\mathsf{p}^{(1)}_{A_\bullet,B}, \ldots, \mathsf{p}^{(n)}_{A_\bullet,B}], \mathsf{p}^{(n+1)}_{A_\bullet,B}\Big] \Rightarrow t : A_1, \ldots, A_n, B \to C$, $\mathsf{e}^\dagger(\alpha)$ is the unique 2-cell γ of type $u \Rightarrow \lambda t$ such that $\alpha \equiv \varepsilon_t \bullet \mathsf{eval}_{B,C}\Big[\gamma[\mathsf{p}^{(1)}_{A_\bullet,B}, \ldots, \mathsf{p}^{(n)}_{A_\bullet,B}], \mathsf{p}^{(n+1)}_{A_\bullet,B}\Big]$. Existence is the first equation and uniqueness follows by the latter two (*c.f.* Lemma 4.2.59).

The required universal property extends that for cartesian biclones.

Lemma 5.2.17. For any $\Lambda^{\times,\to}_{\text{ps}}$-signature $\mathcal{S}$, cartesian closed biclone $(T, \mathcal{D}, \Pi_n(-), \mathbin{=\!\!\rhd})$ and $\Lambda^{\times,\to}_{\text{ps}}$-signature homomorphism $h : \mathcal{S} \to \mathcal{D}$ from $\mathcal{S}$ to the $\Lambda^{\times,\to}_{\text{ps}}$-signature underlying $\mathcal{D}$, there exists a unique strict cartesian closed pseudofunctor $h^{\#} : \mathcal{FCl}^{\times,\to}(\mathcal{S}) \to \mathcal{D}$ such that $h^{\#} \circ \iota = h$, for $\iota : \mathcal{S} \hookrightarrow \mathcal{FCl}^{\times,\to}(\mathcal{S})$ the inclusion.

Proof. We extend the strict cartesian pseudofunctor $h^{\#}$ defined in Lemma 4.2.60 (page 125) with the following rules:

$$
\begin{aligned}
h^{\#}(B \mathbin{=\!\!\rhd} C) &:= (h^{\#}A \mathbin{=\!\!\rhd} h^{\#}B) \\
h^{\#}(\mathsf{eval}_{B,C}) &:= \mathrm{eval}_{h^{\#}B, h^{\#}C} \\
h^{\#}(\lambda t) &:= \lambda\big(h^{\#}t\big) \\
h^{\#}(\varepsilon_t) &:= \varepsilon_{h^{\#}t} \\
h^{\#}(\mathsf{e}^{\dagger}(\alpha)) &:= \mathsf{e}^{\dagger}(h^{\#}\alpha)
\end{aligned}
$$

For uniqueness, it suffices to show that any strict cartesian closed pseudofunctor commutes with the $\mathsf{e}^{\dagger}(-)$ operation. For this we use the universal property. Let $F : \mathcal{FCl}^{\times,\to}(\mathcal{S}) \to \mathcal{D}$ be any cartesian closed pseudofunctor. Then, for any $\alpha : \mathsf{eval}_{B,C}\Big[u[\mathsf{p}^{(1)}_{A_\bullet,B}, \dots, \mathsf{p}^{(n)}_{A_\bullet,B}], \mathsf{p}^{(n+1)}_{A_\bullet,B}\Big] \Rightarrow t : A_1, \dots, A_n, B \to C$ in $\mathcal{FCl}^{\times,\to}(\mathcal{S})$,

$$
\begin{aligned}
&\varepsilon_{Ft} \bullet \mathrm{eval}_{FB,FC}\Big[\big(F\mathsf{e}^{\dagger}(\alpha)\big)\Big[\mathsf{p}^{(1)}_{FA_\bullet,FB}, \dots, \mathsf{p}^{(n)}_{FA_\bullet,FB}\Big], \mathsf{p}^{(n+1)}_{FA_\bullet,FB}\Big] \\
&\quad = F(\varepsilon_t) \bullet F\left(\mathsf{eval}_{B,C}\Big[\mathsf{e}^{\dagger}(\alpha)\Big[\mathsf{p}^{(1)}_{A_\bullet,B}, \dots, \mathsf{p}^{(n)}_{A_\bullet,B}\Big], \mathsf{p}^{(n+1)}_{A_\bullet,B}\Big]\right) && \text{by strict preservation} \\
&\quad = F\left(\varepsilon_t \bullet \mathsf{eval}_{B,C}\Big[\mathsf{e}^{\dagger}(\alpha)[\mathsf{p}^{(1)}_{A_\bullet,B}, \dots, \mathsf{p}^{(n)}_{A_\bullet,B}], \mathsf{p}^{(n+1)}_{A_\bullet,B}\Big]\right) \\
&\quad = F\alpha
\end{aligned}
$$

Hence $\mathsf{e}^{\dagger}(F\alpha)$ must equal $F\big(\mathsf{e}^{\dagger}(\alpha)\big)$. □

We saw in Example 4.2.63 (page 126) that the free fp-bicategory on a $\Lambda^{\times}_{\text{ps}}$-signature cannot arise as the nucleus of the free cartesian biclone over the same signature. We can now see that the addition of exponentials introduces a further obstacle (*c.f.* Remark 5.2.9). Let $\mathcal{S}$ be a unary $\Lambda^{\times,\to}_{\text{ps}}$-signature and $\overline{\mathcal{FCl}^{\times,\to}(\mathcal{S})}$ be its nucleus. Just as in the categorical case, the maps π_i in $\overline{\mathcal{FCl}^{\times,\to}(\mathcal{S})}$ are the biuniversal arrows defining products in $\mathcal{FCl}^{\times,\to}(\mathcal{S})$, but the evaluation map in $\overline{\mathcal{FCl}^{\times,\to}(\mathcal{S})}$ is $\mathsf{eval}_{B,C}[\pi_1, \pi_2]$ (recall Remark 5.2.11). It follows that for any cc-bicategory $(\mathcal{B}, \Pi_n(-), \mathbin{=\!\!\rhd})$ and strict cc-pseudofunctor $F : \overline{\mathcal{FCl}^{\times,\to}(\mathcal{S})} \to \mathcal{B}$ one must have

$$
\begin{aligned}
\mathrm{eval}_{FB,FC} &= F(\mathsf{eval}_{B,C}[\pi_1, \pi_2]) \\
&= F(\mathsf{eval}_{B,C} \circ \langle \pi_1, \pi_2 \rangle) && \text{by def. of products in } \overline{\mathcal{FCl}^{\times,\to}(\mathcal{S})} \\
&= F(\mathsf{eval}_{B,C}) \circ F\langle \pi_1, \pi_2 \rangle \\
&= F(\mathsf{eval}_{B,C}) \circ \langle \pi_1, \pi_2 \rangle && \text{by strict preservation}
\end{aligned}
\tag{5.18}
$$

In particular, since $h^{\#}(\mathsf{eval}_{B,C}) = \mathrm{eval}_{h^{\#}B,h^{\#}C}$, the restriction $\overline{h^{\#}}$ of $h^{\#}$ to unary multimaps cannot be strictly cartesian closed whenever $\mathrm{eval}_{h^{\#}B,h^{\#}C} \circ \langle \pi_1, \pi_2 \rangle \neq \mathrm{eval}_{h^{\#}B,h^{\#}C}$ in the target cc-bicategory. This occurs, for instance, in the cc-bicategories of generalised species [FGHW07] and concurrent games [Paq20].

One way to diagnose the problem is the chain of equivalences (5.12). The product structure in a cartesian closed biclone arises via the $\prod_n(-)$ operation, but exponentials are defined with respect to context extension. This mismatch makes it impossible for $\overline{h^{\#}}$ to strictly preserve both products and exponentials. To construct the free cc-bicategory over a unary signature, one must define exponentials directly with respect to products, resulting in a construction similar to that given in [Oua97].

The free cc-bicategory. As for Construction 5.2.16, we write $t \times B$ for the (derived) arrow $\mathsf{tup}(t[\pi_1], \mathrm{Id}[\pi_2])$, and likewise on 2-cells.

Construction 5.2.18. For any unary $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signature $\mathcal{S} = (\mathfrak{B}, \mathcal{G})$, define a cc-bicategory $\mathcal{FB}ct^{\times,\rightarrow}(\mathcal{S})$ as follows. The objects are generated by the grammar

$$A_1, \ldots, A_n, C, D ::= B \mid \textstyle\prod_n(A_1, \ldots, A_n) \mid C \Rightarrow D \qquad (B \in \mathfrak{B}, n \in \mathbb{N})$$

For 1-cells and 2-cells, one takes the deductive system defining the free fp-bicategory on $\mathcal{S}$ (Lemma 4.2.62, page 126), extended as follows. For 1-cells:

$$\frac{}{\mathsf{eval}_{B,C} \in \mathcal{FB}ct^{\times,\rightarrow}(\mathcal{S})(B \Rightarrow C \times B; C)} \qquad \frac{t \in \mathcal{FB}ct^{\times,\rightarrow}(\mathcal{S})(X \times B; C)}{\lambda t \in \mathcal{FB}ct^{\times,\rightarrow}(\mathcal{S})(X, B \Rightarrow C)}$$

For 2-cells:

$$\frac{t \in \mathcal{FB}ct^{\times,\rightarrow}(\mathcal{S})(X \times B, C)}{\varepsilon_t \in \mathcal{FB}ct^{\times,\rightarrow}(\mathcal{S})(X \times B, C)\big(\mathsf{eval}_{B,C}[\lambda t \times B], t\big)} \qquad \frac{\begin{array}{c} u \in \mathcal{FB}ct^{\times,\rightarrow}(\mathcal{S})(X, B \Rightarrow C) \\ \alpha \in \mathcal{FB}ct^{\times,\rightarrow}(\mathcal{S})(X \times B, C)\big(\mathsf{eval}_{B,C}[u \times B], t\big) \end{array}}{\mathsf{e}^{\dagger}(\alpha) \in \mathcal{FB}ct^{\times,\rightarrow}(\mathcal{S})(X, A \Rightarrow B)(u, \lambda t)}$$

Moreover, we extend the equational theory of Lemma 4.2.62 with the following three rules:

- For every $\alpha : \mathsf{eval}_{B,C}[u \times B] \Rightarrow t : X \times B \to C$,

$$\alpha \equiv \varepsilon_t \bullet \mathsf{eval}_{B,C}[\mathsf{e}^{\dagger}(\alpha) \times B]$$

- For every $\gamma : u \Rightarrow \lambda t : X \to (A \Rightarrow B)$,

$$\gamma \equiv \mathsf{e}^{\dagger}(\varepsilon_t \bullet \mathsf{eval}_{B,C}[\gamma \times B])$$

- If $\alpha \equiv \alpha' : \mathsf{eval}_{B,C}[u \times B] \Rightarrow t : X \times B \to C$ then $\mathsf{e}^{\dagger}(\alpha) \equiv \mathsf{e}^{\dagger}(\alpha')$.

Finally we require that every ε_t and $\mathsf{e}^{\dagger}(\mathrm{id}_{\mathsf{eval}[u \times B]})$ is invertible. ◂

The bicategory $\mathcal{FB}ct^{\times,\rightarrow}(\mathcal{S})$ is cartesian closed by exactly the same argument as for the biclone $\mathcal{FC}l^{\times,\rightarrow}(\mathcal{S})$. The associated free property is similarly straightforward.

Lemma 5.2.19. For any unary $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signature $\mathcal{S}$, cc-bicategory $(\mathcal{C}, \Pi_n(-), \Rightarrow)$ and $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signature homomorphism $h : \mathcal{S} \to \mathcal{C}$ from $\mathcal{S}$ to the $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signature underlying $\mathcal{C}$, there exists a unique strict cartesian closed pseudofunctor $h^{\#} : \mathcal{FB}ct^{\times,\rightarrow}(\mathcal{S}) \to \mathcal{C}$ such that $h^{\#} \circ \iota = h$, for $\iota : \mathcal{S} \hookrightarrow \mathcal{FB}ct^{\times,\rightarrow}(\mathcal{S})$ the inclusion.

Proof. We extend the strict cartesian pseudofunctor $h^{\#}$ defined in Lemma 4.2.62 (page 126) as follows:

$$h^{\#}(B \Rightarrow C) := (h^{\#}A \Rightarrow h^{\#}B)$$

$$\begin{aligned} h^{\#}(\mathsf{eval}_{B,C}) &:= \mathrm{eval}_{h^{\#}B, h^{\#}C} \\ h^{\#}(\lambda t) &:= \lambda\big(h^{\#}t\big) \end{aligned}$$

$$\begin{aligned} h^{\#}(\varepsilon_t) &:= \varepsilon_{h^{\#}t} \\ h^{\#}\left(\mathsf{e}^{\dagger}(\alpha)\right) &:= \mathsf{e}^{\dagger}(h^{\#}\alpha) \end{aligned}$$

For uniqueness, it suffices to show that any strict cartesian closed pseudofunctor commutes with the $\mathsf{e}^{\dagger}(-)$ operation. The proof is as in Lemma 5.2.17 (or, more abstractly, follows from Lemma 2.2.17). □

The preceding lemma entails that one may construct a type theory for cartesian closed bicategories by synthesising the internal language of $\mathcal{FB}ct^{\times,\rightarrow}(\mathcal{S})$. Within this 'bicategorical' (rather than *biclone-theoretic*) type theory the variables play almost no role. For instance, the lambda abstraction rule takes on the following form:

$$\frac{p : A \times B \vdash t : C \qquad q \text{ fresh}}{q : A \vdash \lambda(q, p \,.\, t) : B \Rightarrow C}\ \text{lam}$$

The variable p is bound, but q is free. It is possible to place such rules within the general framework of binding signatures, and the syntactic model of the resulting type theory is biequivalent to the syntactic model of the type theory extracted from the construction of $\mathcal{FC}l^{\times,\rightarrow}(\mathcal{S})$, restricted to unary contexts. However, the result is rather alien to the usual conception of a type theory. We therefore call the internal language of $\mathcal{FC}l^{\times,\rightarrow}(\mathcal{S})$ the 'type theory for cartesian closed bicategories'. In Section 5.3.3 we shall show that this terminology is warranted.

The freeness universal property of $\mathcal{FB}ct^{\times,\rightarrow}(\mathcal{S})$ also entails an up-to-equivalence uniqueness property we shall employ later. We begin by stating a result for the case where the signature is just a set; thereafter we employ slightly stronger hypotheses to handle constants. We write $t : A_1, \ldots, A_n \to B$ and $\tau : t \Rightarrow t' : A_1, \ldots, A_n \to B$ for 1-cells and 2-cells in $\mathcal{FB}ct^{\times,\rightarrow}(\mathcal{S})$.

Lemma 5.2.20. Let $\mathcal{S} = (\mathfrak{B}, \mathcal{G})$ be a unary $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signature for which $\mathcal{G}$ is a set, $(\mathcal{B}, \Pi_n(-), \mathrel{=\!\!\rhd})$ be a cc-bicategory and $h : \mathcal{S} \to \mathcal{C}$ be a $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signature homomorphism. Then, for any cc-pseudofunctor $(F, \mathrm{q}^{\times}, \mathrm{q}^{\mathrel{=\!\!\rhd}})$ such that the following diagram commutes,

$$\begin{tikzcd}
\mathcal{FB}ct^{\times,\rightarrow}(\mathcal{S}) \arrow[r, "F"] & \mathcal{C} \\
\mathcal{S} \arrow[u, hook] \arrow[ur, "h"'] &
\end{tikzcd} \tag{5.19}$$

there exists an equivalence $F \simeq h^{\#}$ between F and the canonical cc-pseudofunctor extending h.

Proof. We construct a pseudonatural transformation $(\mathsf{k}, \overline{\mathsf{k}}) : F \Rightarrow h^{\#}$ whose components are all equivalences. We define the components k_X and their pseudo-inverses $\mathsf{k}_X^{\star}$ by mutual induction as follows:

$$\begin{aligned}
\mathsf{k}_B &:= FB \xrightarrow{=} hB \xrightarrow{\mathrm{Id}_{hB}} hB \xrightarrow{=} h^{\#}B \qquad \text{for } B \in \mathfrak{B} \\
\mathsf{k}_B^{\star} &:= h^{\#}B \xrightarrow{=} hB \xrightarrow{\mathrm{Id}_{hB}} hB \xrightarrow{=} FB
\end{aligned}$$

$$\begin{aligned}
\mathsf{k}_{(\prod_n A_\bullet)} &:= F(\textstyle\prod_n A_\bullet) \xrightarrow{\langle F\pi_1, \dots, F\pi_n\rangle} \textstyle\prod_{i=1}^n F(A_i) \xrightarrow{\prod_{i=1}^n \mathsf{k}_{A_i}} \textstyle\prod_{i=1}^n h^{\#}A_i \\
\mathsf{k}_{(\prod_n A_\bullet)}^{\star} &:= \textstyle\prod_{i=1}^n h^{\#}A_i \xrightarrow{\prod_{i=1}^n \mathsf{k}_{A_i}^{\star}} \textstyle\prod_{i=1}^n F(A_i) \xrightarrow{\mathrm{q}_{A_\bullet}^{\times}} F(\textstyle\prod_n A_\bullet)
\end{aligned}$$

$$\begin{aligned}
\mathsf{k}_{(X \mathrel{=\!\!\rhd} Y)} &:= F(X \mathrel{=\!\!\rhd} Y) \xrightarrow{\mathrm{m}_{X,Y}} (FX \mathrel{=\!\!\rhd} FY) \xrightarrow{\mathsf{k}_X^{\star} \mathrel{=\!\!\rhd} \mathsf{k}_Y} \big(h^{\#}X \mathrel{=\!\!\rhd} h^{\#}Y\big) \\
\mathsf{k}_{(X \mathrel{=\!\!\rhd} Y)}^{\star} &:= \big(h^{\#}X \mathrel{=\!\!\rhd} h^{\#}Y\big) \xrightarrow{\mathsf{k}_X \mathrel{=\!\!\rhd} \mathsf{k}_Y^{\star}} (FX \mathrel{=\!\!\rhd} FY) \xrightarrow{\mathrm{q}_{X,Y}^{\mathrel{=\!\!\rhd}}} F(X \mathrel{=\!\!\rhd} Y)
\end{aligned}$$

We denote the unit and counit of the equivalence

$$\mathsf{k}_X : FX \leftrightarrows h^{\#}X : \mathsf{k}_X^{\star}$$

by $\mathsf{v}_X : \mathrm{Id}_{FX} \Rightarrow \mathsf{k}_X^{\star} \circ \mathsf{k}_X$ and $\mathsf{w}_X : \mathsf{k}_X \circ \mathsf{k}_X^{\star} \Rightarrow \mathrm{Id}_{h^{\#}X}$, respectively, and assume without loss of generality that they satisfy the two triangle laws.

We now construct the witnessing 2-cells $\overline{\mathsf{k}}_t : \mathsf{k}_B \circ Ft \Rightarrow h^{\#}(t) \circ \mathsf{k}_A$ by induction.

For identities, the definition is forced upon us by the unit law of a pseudonatural transformation. We define

$$\overline{\mathsf{k}}_{\mathrm{Id}_A} := \mathsf{k}_A \circ F(\mathrm{Id}_A) \xRightarrow{\mathsf{k}_A \circ (\psi_A^F)^{-1}} \mathsf{k}_A \circ \mathrm{Id}_{F(A)} \overset{\cong}{\Rightarrow} \mathrm{Id}_{h^{\#}(A)} \circ \mathsf{k}_A$$

For the product structure, we define $\overline{\mathsf{k}}_{\pi_k}$ and $\overline{\mathsf{k}}_{\mathsf{tup}(t_1, \dots, t_n)}$ by the commutativity of the following diagrams:

$$\begin{tikzcd}
\mathsf{k}_{A_k} \circ F\pi_k \arrow[rr, "\overline{\mathsf{k}}_{\pi_k}"] \arrow[d, "\mathsf{k}_{A_k} \circ \varpi^{(-k)}"'] & & h^{\#}(\pi_k) \circ \mathsf{k}_{(\prod_n A_\bullet)} \\
\mathsf{k}_{A_k} \circ (\pi_k \circ \langle F\pi_\bullet \rangle) \arrow[dr, "\cong"'] & & (\pi_k \circ \prod_{i=1}^{n} \mathsf{k}_{A_i}) \circ \langle F\pi_\bullet \rangle \arrow[u, "\cong"'] \\
& (\mathsf{k}_{A_k} \circ \pi_k) \circ \langle F\pi_\bullet \rangle \arrow[ur, "\varpi^{(-k)} \circ \langle F\pi_\bullet \rangle"'] &
\end{tikzcd}$$

$$\begin{tikzcd}
(\prod_{i=1}^{m} \mathsf{k}_{A_i} \circ \langle F\pi_\bullet \rangle) \circ F(\mathsf{tup}(t_1, \ldots, t_m)) \arrow[rr, "\overline{\mathsf{k}}_{\mathsf{tup}(t_1, \ldots, t_m)}"] \arrow[d, "\cong"'] & & h^{\#}(\mathsf{tup}(t_1, \ldots, t_m)) \circ \mathsf{k}_X \arrow[d, equal] \\
(\prod_{i=1}^{m} \mathsf{k}_{A_i}) \circ (\langle F\pi_\bullet \rangle \circ F(\mathsf{tup}(t_1, \ldots, t_m))) \arrow[d, "(\prod_i \mathsf{k}_{A_i}) \circ \mathsf{unpack}"'] & & \langle h^{\#}(t_\bullet) \rangle \circ \mathsf{k}_X \\
(\prod_{i=1}^{m} \mathsf{k}_{A_i}) \circ \langle F(t_\bullet) \rangle \arrow[r, "\mathsf{fuse}"'] & \langle \mathsf{k}_{A_\bullet} \circ F(t_\bullet) \rangle \arrow[r, "\langle \overline{\mathsf{k}}_{t_1}, \ldots, \overline{\mathsf{k}}_{t_m} \rangle"'] & \langle h^{\#}(t_\bullet) \circ \mathsf{k}_X \rangle \arrow[u, "\mathsf{post}^{-1}"']
\end{tikzcd}$$

The `eval` and `lam` cases require more work, but are in a similar spirit.

$\boxed{\texttt{eval}\text{ case.}}$ We are required to give an invertible 2-cell filling the diagram

$$\begin{tikzcd}
F\big((A \Rightarrow B) \times A\big) \arrow[rr, "F\mathsf{eval}_{A,B}"] \arrow[d, "\langle F\pi_1, F\pi_2 \rangle"'] \arrow[dd, bend right=90, "(\mathsf{k}_{(A \Rightarrow B)} \times \mathsf{k}_A) \circ \langle F\pi_1, F\pi_2 \rangle"'] & & FB \arrow[dd, "\mathsf{k}_B"] \\
F(A \Rightarrow B) \times F(A) \arrow[d, "\mathsf{k}_{(A \Rightarrow B)} \times \mathsf{k}_A"'] & \overset{\overline{\mathsf{k}}_{\mathrm{eval}}}{\Longleftarrow} & \\
h^{\#}(A \Rightarrow B) \times h^{\#}A \arrow[r, equal] & (h^{\#}A \Rightarrow h^{\#}B) \times h^{\#}A \arrow[r, "\mathrm{eval}"'] & h^{\#}B
\end{tikzcd}$$

To this end, first define an invertible 2-cell $\delta_{A,B}$ applying the counit ε as far as possible:

$$\begin{tikzcd}
\mathrm{eval}_{h^\# A, h^\# B} \circ \big(\mathsf{k}_{(A \Rightarrow B)} \times \mathsf{k}_A\big) \arrow[d, equal] \arrow[dddddddr, bend left, "\delta_{A,B}"] & \\
\mathrm{eval}_{h^\# A, h^\# B} \circ \big((\mathsf{k}^\star_A \Rightarrow \mathsf{k}_B) \circ \mathrm{m}^F_{A,B} \times \mathsf{k}_A\big) \arrow[d, "\cong"'] & \\
\big(\mathrm{eval}_{h^\# A, h^\# B} \circ \big((\mathsf{k}^\star_A \Rightarrow \mathsf{k}_B) \times h^\# A\big)\big) \circ (\mathrm{m}^F_{A,B} \times \mathsf{k}_A) \arrow[d, "\varepsilon_{(\mathrm{k} \circ \mathrm{eval} \circ (\mathrm{Id} \times \mathrm{k}^\star))} \circ (\mathrm{m}^F_{A,B} \times \mathrm{k}_A)"'] & \\
\big((\mathsf{k}_B \circ \mathrm{eval}_{FA,FB}) \circ \big(\mathrm{Id}_{(FA \Rightarrow FB)} \times \mathsf{k}^\star_A\big)\big) \circ (\mathrm{m}^F_{A,B} \times \mathsf{k}_A) \arrow[d, "\cong"'] & \\
\big(\mathsf{k}_B \circ \big(\mathrm{eval}_{FA,FB} \circ \big(\mathrm{m}^F_{A,B} \times FA\big)\big)\big) \circ \big(\mathrm{Id}_{(FA \Rightarrow FB)} \times \mathsf{k}^\star_A \mathsf{k}_A\big) \arrow[d, "\mathrm{k} \circ \varepsilon_{(F(\mathrm{eval}) \circ \mathrm{q}^\times)} \circ (\mathrm{Id} \times \mathrm{k}^\star \mathrm{k})"'] & \\
\Big(\mathsf{k}_B \circ \Big(F(\mathrm{eval}_{A,B}) \circ \mathrm{q}^\times_{A \Rightarrow B, A}\Big)\Big) \circ \big(\mathrm{Id}_{(FA \Rightarrow FB)} \times \mathsf{k}^\star_A \mathsf{k}_A\big) \arrow[d, "\mathrm{k} \circ F\mathrm{eval} \circ \mathrm{q}^\times \circ (\mathrm{Id} \times \mathrm{v}_A^{-1})"'] & \\
\Big(\mathsf{k}_B \circ \Big(F(\mathrm{eval}_{A,B}) \circ \mathrm{q}^\times_{A \Rightarrow B, A}\Big)\Big) \circ \big(\mathrm{Id}_{(FA \Rightarrow FB)} \times \mathrm{Id}_{FA}\big) \arrow[r, "\cong"'] & (\mathsf{k}_B \circ F(\mathsf{eval}_{A,B})) \circ \mathrm{q}^\times_{A \Rightarrow B, A}
\end{tikzcd}$$

Then define $\overline{\mathsf{k}}_{\mathsf{eval}}$ to be the composite

$$\begin{tikzcd}
\mathsf{k}_B \circ F(\mathrm{eval}_{A,B}) \arrow[r, "\overline{\mathsf{k}}_{\mathsf{eval}}"] \arrow[d, "\cong"'] & \mathrm{eval}_{h^\# A, h^\# B} \circ \big(\big(\mathsf{k}_{(A \Rightarrow B)} \times \mathsf{k}_A\big) \circ \langle F\pi_1, F\pi_2 \rangle\big) \\
(\mathsf{k}_B \circ F(\mathrm{eval}_{A,B})) \circ \mathrm{Id}_{F((A \Rightarrow B) \times A)} \arrow[d, "\big(\mathsf{k}_B \circ F(\mathrm{eval}_{A,B})\big) \circ (\mathrm{c}^\times_{A \Rightarrow B, A})^{-1}"'] & \\
(\mathsf{k}_B \circ F(\mathrm{eval}_{A,B})) \circ \Big(\mathrm{q}^\times_{A \Rightarrow B, A} \circ \langle F\pi_1, F\pi_2 \rangle\Big) \arrow[d, "\cong"'] & \\
\Big(\mathsf{k}_B \circ \big(F(\mathrm{eval}_{A,B}) \circ \mathrm{q}^\times_{A \Rightarrow B, A}\big)\Big) \circ \langle F\pi_1, F\pi_2 \rangle \arrow[dr, bend right, "\delta^{-1}_{A,B} \circ \langle F\pi_1, F\pi_2 \rangle"'] & \\
 & \big(\mathrm{eval}_{h^\# A, h^\# B} \circ \big(\mathsf{k}_{(A \Rightarrow B)} \times \mathsf{k}_A\big)\big) \circ \langle F\pi_1, F\pi_2 \rangle \arrow[uuuu, "\cong"]
\end{tikzcd}$$

$\boxed{\texttt{lam}\text{ case.}}$ Suppose $t : Z \times A \to B$. By induction we are given $\overline{\mathsf{k}}_t$ filling

$$\begin{tikzcd}
F(Z \times A) \arrow[rr, "Ft"] \arrow[d, "\langle F\pi_1, F\pi_2 \rangle"'] \arrow[dd, bend right=70, "(\mathsf{k}_Z \times \mathsf{k}_A) \circ \langle F\pi_1, F\pi_2 \rangle"'] & & FB \arrow[dd, "\mathsf{k}_B"] \\
FZ \times FA \arrow[d, "\mathsf{k}_Z \times \mathsf{k}_A"'] & \overset{\overline{\mathsf{k}}_t}{\Leftarrow} & \\
h^{\#}(Z) \times h^{\#}(A) \arrow[r, equal] & h^{\#}(Z \times A) \arrow[r, "h^{\#}t"'] & h^{\#}B
\end{tikzcd}$$

and we are required to fill the diagram

$$\begin{tikzcd}
FZ \arrow[r, "F(\lambda t)"] \arrow[dd, "\mathsf{k}_Z"'] & F(A \mathrel{=\!\!\rhd} B) \arrow[d, "\mathrm{m}^F_{A,B}"] \arrow[dd, bend left=70, "(\mathsf{k}^{\star}_A \mathrel{=\!\!\rhd} \mathsf{k}_B) \circ \mathrm{m}^F_{A,B}"] \\
\overset{\overline{\mathsf{k}}_{\lambda t}}{\Leftarrow} & \big(FA \mathrel{=\!\!\rhd} FB\big) \arrow[d, "(\mathsf{k}^{\star}_A \mathrel{=\!\!\rhd} \mathsf{k}_B)"] \\
h^{\#}Z \arrow[r, "h^{\#}(\lambda t)"'] & (h^{\#}A \mathrel{=\!\!\rhd} h^{\#}B)
\end{tikzcd}$$

Our strategy is the following. Writing cl for the clockwise composite around the preceding diagram, we define a 2-cell

$$\zeta_{A,B} : \mathrm{eval}_{h^{\#}A, h^{\#}B} \circ (cl \times h^{\#}A) \Rightarrow h^{\#}(t) \circ (\mathsf{k}_Z \times h^{\#}A)$$

so that $\mathsf{e}^{\dagger}(\zeta_{A,B}) : cl \Rightarrow \lambda\big(h^{\#}(t) \circ (\mathsf{k}_Z \times h^{\#}A)\big)$. We then define $\overline{\mathsf{k}}_{\lambda t}$ as the composite

$$cl \xRightarrow{\mathsf{e}^{\dagger}(\zeta_{A,B})} \lambda\big(h^{\#}(t) \circ (\mathsf{k}_Z \times h^{\#}A)\big) \xRightarrow{\mathsf{push}^{-1}} \lambda\big(h^{\#}t\big) \circ \mathsf{k}_Z = h^{\#}(\lambda t) \circ \mathsf{k}_Z$$

The 2-cell $\zeta_{A,B}$ is defined in stages. First we set $\upsilon_{A,B}$ to be the following composite, where we write $\cong$ for composites of Φ and structural isomorphisms:

$$\begin{tikzcd}
\mathrm{eval}_{h^{\#}A, h^{\#}B} \circ (cl \times h^{\#}A) \arrow[d, "\cong"'] \\
\big(\mathrm{eval}_{h^{\#}A, h^{\#}B} \circ \big((\mathsf{k}^{\star}_A \mathrel{=\!\!\rhd} \mathsf{k}_B) \times h^{\#}A\big)\big) \circ \big(\big(\mathrm{m}^F_{A,B} \circ F(\lambda t)\big) \times h^{\#}A\big) \arrow[d, "\varepsilon_{\mathrm{k} \circ \mathrm{eval} \circ (\mathrm{Id} \times \mathrm{k}^{\star})} \circ (\mathrm{m}^F_{A,B} F(\lambda t) \times h^{\#}A)"'] \\
\big((\mathsf{k}_B \circ \mathrm{eval}_{FA,FB}) \circ (\mathrm{Id}_{(FA \mathrel{=\!\!\rhd} FB)} \times \mathsf{k}^{\star}_A)\big) \circ \big(\big(\mathrm{m}^F_{A,B} \circ F(\lambda t)\big) \times h^{\#}A\big) \arrow[d, "\cong"'] \\
\big(\mathsf{k}_B \circ \big(\mathrm{eval}_{FA,FB} \circ \big(\mathrm{m}^F_{A,B} \times F(A)\big)\big)\big) \circ \big(F(\lambda t) \times \mathsf{k}^{\star}_A\big) \arrow[d, "\mathsf{k}_B \circ \varepsilon_{(F(\mathrm{eval}) \circ \mathrm{q}^{\times})} \circ (F(\lambda t) \times \mathsf{k}^{\star})"'] \\
\Big(\mathsf{k}_B \circ \Big(F(\mathrm{eval}_{A,B}) \circ \mathrm{q}^{\times}_{A \mathrel{=\!\!\rhd} B, A}\Big)\Big) \circ (F(\lambda t) \times \mathsf{k}^{\star}_A)
\end{tikzcd}$$

Next we define $\theta_{A,B}$ to be the composite

$$\begin{tikzcd}
F(\mathrm{eval}_{A,B}) \circ \Big(\mathrm{q}^{\times}_{A \Rightarrow B,A} \circ (F\lambda t \times FA)\Big) \arrow[rr, "\theta_{A,B}"] \arrow[d, swap, "F(\mathrm{eval}) \circ \mathrm{q}^{\times} \circ (F(\lambda t) \times \psi^{F}_{A})"] & & Ft \circ \mathrm{q}^{\times}_{Z,A} \\
F(\mathrm{eval}_{A,B}) \circ \Big(\mathrm{q}^{\times}_{A \Rightarrow B,A} \circ \big(\lambda t \times F\mathrm{Id}_{A}\big)\Big) \arrow[d, swap, "F(\mathrm{eval}) \circ \mathsf{nat}"] & & F(\mathrm{eval}_{A,B} \circ (\lambda t \times A)) \circ \mathrm{q}^{\times}_{Z,A} \arrow[u, swap, "F(\varepsilon_t) \circ \mathrm{q}^{\times}"] \\
F(\mathrm{eval}_{A,B}) \circ \Big(F(\lambda t \times A) \circ \mathrm{q}^{\times}_{Z,A}\Big) \arrow[rr, swap, "\cong"] & & (F(\mathrm{eval}_{A,B}) \circ F(\lambda t \times A)) \circ \mathrm{q}^{\times}_{Z,A} \arrow[u, swap, "\phi^{F}_{(\mathrm{eval}, \lambda t \times A)} \circ \mathrm{q}^{\times}"]
\end{tikzcd}$$

We can now define $\zeta_{A,B}$ as follows:

$$\begin{tikzcd}
\mathrm{eval}_{h^{\#}A, h^{\#}B} \circ (cl \times h^{\#}A) \arrow[r, "\zeta_{A,B}"] \arrow[d, swap, "\upsilon_{A,B}"] & h^{\#}(t) \circ \big(\mathsf{k}_{Z} \times A\big) \\
\Big(\mathsf{k}_{B} \circ \Big(F\mathrm{eval}_{A,B} \circ \mathrm{q}^{\times}_{A \Rightarrow B,A}\Big)\Big) \circ (F(\lambda t) \times \mathsf{k}^{\star}_{A}) \arrow[d, swap, "\cong"] & \\
\Big(\mathsf{k}_{B} \circ \Big(F\mathrm{eval}_{A,B} \circ \Big(\mathrm{q}^{\times}_{A \Rightarrow B,A} \circ (F(\lambda t) \times FA)\Big)\Big)\Big) \circ (FZ \times \mathsf{k}^{\star}_{A}) \arrow[d, swap, "\mathsf{k}_{B} \circ \theta_{A,B} \circ (FZ \times \mathsf{k}^{\star}_{A})"] & \\
\Big(\mathsf{k}_{B} \circ \Big(Ft \circ \mathrm{q}^{\times}_{Z,A}\Big)\Big) \circ (FZ \times \mathsf{k}^{\star}_{A}) \arrow[d, swap, "\cong"] & \\
(\mathsf{k}_{B} \circ Ft) \circ \Big(\mathrm{q}^{\times}_{Z,A} \circ (FZ \times \mathsf{k}^{\star}_{A})\Big) \arrow[d, swap, "\overline{\mathsf{k}}_{t} \circ \mathrm{q}^{\times} \circ (FZ \times \mathsf{k}^{\star}_{A})"] & \\
\big(h^{\#}(t) \circ ((\mathsf{k}_{Z} \times \mathsf{k}_{A}) \circ \langle F\pi_1, F\pi_2 \rangle)\big) \circ \Big(\mathrm{q}^{\times}_{Z,A} \circ (FZ \times \mathsf{k}^{\star}_{A})\Big) \arrow[d, swap, "\cong"] & \\
\Big(\big(h^{\#}(t) \circ (\mathsf{k}_{Z} \times \mathsf{k}_{A})\big) \circ \Big(\langle F\pi_1, F\pi_2 \rangle \circ \mathrm{q}^{\times}_{Z,A}\Big)\Big) \circ (FZ \times \mathsf{k}^{\star}_{A}) \arrow[d, swap, "h^{\#}(t) \circ (\mathsf{k}_{Z} \times \mathsf{k}_{A}) \circ (\mathsf{u}^{\times}_{Z,A})^{-1} \circ (FZ \times \mathsf{k}^{\star}_{A})"] & \\
h^{\#}(t) \circ (\mathsf{k}_{Z} \times \mathsf{k}_{A}) \circ \mathrm{Id}_{FZ \times FA} \circ \big(FZ \times \mathsf{k}^{\star}_{A}\big) \arrow[dr, swap, "\cong"] & \\
& h^{\#}(t) \circ (\mathsf{k}_{Z} \times \mathsf{k}_{A}\mathsf{k}^{\star}_{A}) \arrow[uuuuuuuu, swap, "h^{\#}(t) \circ (\mathsf{k}_{Z} \times \mathsf{w}_{A})"]
\end{tikzcd}$$

This completes the definition of $\overline{\mathsf{k}}_{\lambda t}$. The only remaining case is horizontal composition.

$\boxed{\texttt{hcomp}\text{ case.}}$ As was the case for identities, the definition for multimaps of the form $t \circ u : Z \to B$ is forced by the axioms of a pseudonatural transformation. Using that $h^{\#}$ is a strict pseudofunctor, we define

$$\begin{tikzcd}
\mathsf{k}_B \circ F(t \circ u) \arrow[rr, "\overline{\mathsf{k}}_{t \circ u}"] \arrow[d, "\mathsf{k}_B \circ (\phi^F_{t,u})^{-1}"'] & & \big(h^{\#}(t) \circ h^{\#}(u)\big) \circ \mathsf{k}_Z \\
\mathsf{k}_B \circ (F(t) \circ F(u)) \arrow[d, "\cong"'] & & h^{\#}(t) \circ \big(h^{\#}(u) \circ \mathsf{k}_Z\big) \arrow[u, "\cong"'] \\
(\mathsf{k}_B \circ Ft) \circ Fu \arrow[r, "\overline{\mathsf{k}}_t \circ F(u)"'] & \big(h^{\#}(t) \circ \mathsf{k}_A\big) \circ Fu \arrow[r, "\cong"'] & h^{\#}(t) \circ (\mathsf{k}_A \circ Fu) \arrow[u, "h^{\#}(t) \circ \overline{\mathsf{k}}_u"']
\end{tikzcd}$$

To show that $(\mathsf{k}, \overline{\mathsf{k}})$ is indeed a pseudonatural transformation, we need to check the naturality condition and two axioms. Naturality is a straightforward check for each case outlined above. The two axioms—corresponding to the identity and `hcomp` cases—hold by construction. □

Examining the construction of the pseudonatural transformation just given, one extracts the following result.

Corollary 5.2.21. For any unary $\Lambda^{\times,\rightarrow}_{\mathrm{ps}}$-signature $\mathcal{S} = (\mathfrak{B}, \mathcal{G})$, cc-bicategory $(\mathcal{B}, \Pi_n(-), \Rightarrow)$, $\Lambda^{\times,\rightarrow}_{\mathrm{ps}}$-signature homomorphism $h : \mathcal{S} \to \mathcal{C}$, and cc-pseudofunctor $(F, \mathrm{q}^{\times}, \mathrm{q}^{\Rightarrow})$ such that

1. Diagram (5.19) commutes, *i.e.*:

$$\begin{tikzcd}
\mathcal{FB}ct^{\times,\rightarrow}(\mathcal{S}) \arrow[r, "F"] & \mathcal{C} \\
\mathcal{S} \arrow[u, hook] \arrow[ur, "h"'] &
\end{tikzcd}$$

2. For every $A_1, \ldots, A_n, A, B \in \mathcal{FB}ct^{\times,\rightarrow}(\mathcal{S})$, the 1-cells $\langle F\pi_1, \ldots, F\pi_n \rangle$ and $\mathrm{m}_{A,B}$ are isomorphic to the identity,

there exists an equivalence $F \simeq h^{\#}$ between F and the canonical cc-pseudofunctor extending h.

Proof. One only needs to extend the pseudonatural equivalence $(\mathsf{k}, \overline{\mathsf{k}})$ constructed in the proof of Lemma 5.2.20 to cover constants. For these, one employs the second hypothesis. For any constant $c \in \mathcal{G}(A, B)$, condition (1) requires that $F(c) = h(c) = h^{\#}(c)$. Condition (2), on the other hand, entails that the components of $(\mathsf{k}, \overline{\mathsf{k}})$ are, inductively, each isomorphic to the identity. For the 2-cell filling

$$\begin{tikzcd}
FA \arrow[r, "Fc"] \arrow[d, "\mathsf{k}_A"'] \arrow[dr, phantom, "\overset{\overline{\mathsf{k}}_c}{\Leftarrow}"] & FB \arrow[d, "\mathsf{k}_B"] \\
h^{\#}(A) \arrow[r, "h^{\#}(c)"'] & h^{\#}(B)
\end{tikzcd}$$

one may therefore take the composite $\mathsf{k}_B \circ Fc \overset{\cong}{\Rightarrow} Fc = h^{\#}(c) \overset{\cong}{\Rightarrow} h^{\#}(c) \circ \mathsf{k}_A$ This definition is natural in c, and the two axioms of a pseudonatural transformation continue to hold. The claim follows. □

5.3 The type theory $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$

Fix a $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signature $\mathcal{S}$. The type theory $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}(\mathcal{S})$ is constructed as the internal language of $\mathcal{FC}l^{\times,\rightarrow}(\mathcal{S})$, with rules matching those of Construction 5.2.16. These are collected together in Figures 5.1–5.4. Recall that for a context renaming r we write $t\{r\}$ to denote the term $t\{x_i \mapsto r(x_i)\}$ (Figure 3.2), and that we write inc_x for the inclusion of contexts $\Gamma \hookrightarrow \Gamma, x : A$ extending Γ with a fresh variable x.

The lambda abstraction operation extends to a (functorial) mapping on rewrites, and the unit is derived as the mediating map corresponding to the identity (*c.f.* the discussion following Definition 5.1.1).

Definition 5.3.1.

1. For any derivable rewrite $(\Gamma, x : A \vdash \tau : t \Rightarrow t' : B)$ we define $\lambda x.\tau : \lambda x.t \Rightarrow \lambda x.t'$ to be the rewrite $\mathsf{e}^{\dagger}(x \,.\, \tau \bullet \varepsilon_t)$ in context Γ.
2. For any derivable term $(\Gamma \vdash u : A \Rightarrow B)$ we define the unit $\eta_u : u \Rightarrow \lambda x.\mathsf{eval}\{u\{\mathrm{inc}_x\}, x\}$ to be the rewrite $\mathsf{e}^{\dagger}(x \,.\, \mathrm{id}_{\mathsf{eval}\{u\{\mathrm{inc}_x\},x\}})$ in context Γ. ◂

The usual application operation becomes a derived rule:

$$\frac{\Gamma \vdash t : A \Rightarrow B \qquad \Gamma \vdash u : A}{\Gamma \vdash \mathsf{eval}\{t, u\} : B}$$

The ε-introduction rule only relates lambda abstractions and variables, but the general form of (explicit) β-reduction is derivable. In the definition we use the following notation. For a context $\Gamma := (x_i : A_i)_{i=1,\dots,n}$ and terms $\Gamma, x : A \vdash t : B$ and $\Gamma \vdash u : A$, we write $t\{\mathrm{id}_\Gamma, x \mapsto u\}$ to denote the term $t\{x_1 \mapsto x_1, \dots, x_n \mapsto x_n, x \mapsto u\}$ in context Γ.

Definition 5.3.2. For derivable terms $\Gamma, x : A \vdash t : B$ and $\Gamma \vdash u : A$ we define the *β-reduction rewrite* $\beta_{x.t,u} : \mathsf{eval}\{\lambda x.t, u\} \Rightarrow t\{\mathrm{id}_\Gamma, x \mapsto u\}$ to be $\varepsilon_t\{\mathrm{id}_\Gamma, x \mapsto u\} \bullet \tau$ in context Γ, where τ is the following composite of structural isomorphisms:

$$\begin{aligned}
\mathsf{eval}\{\lambda x.t, u\} &\cong \mathsf{eval}\{(\lambda x.t)\{\mathrm{inc}_x\}, u\} \\
&\cong \mathsf{eval}\big\{(\lambda x.t)\{\mathrm{inc}_x\{\mathrm{id}_\Gamma, x \mapsto u\}\}, u\big\} \\
&\cong \mathsf{eval}\big\{(\lambda x.t)\{\mathrm{inc}_x\}\{\mathrm{id}_\Gamma, x \mapsto u\}, x\{\mathrm{id}_\Gamma, x \mapsto u\}\big\} \\
&\cong \mathsf{eval}\{(\lambda x.t)\{\mathrm{inc}_x\}, x\}\{\mathrm{id}_\Gamma, x \mapsto u\}
\end{aligned}$$ ◂

In a similar vein, one may wish to introduce the counit via the following more explicit rule:

$$\frac{\Gamma, x : A \vdash t : B}{\Gamma, y : A \vdash \varepsilon_{x\,.\,t} : \mathsf{eval}\{(\lambda x.t)\{\mathrm{inc}_y\}, y\} \Rightarrow t\{\mathrm{id}_\Gamma, x \mapsto y\} : B}$$

In the presence of the structural rewrites, this definition is equivalent to that given in Figure 5.2.

We continue to work up to α-equivalence of terms and rewrites. Unlike the extension from $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$ to $\Lambda_{\mathrm{ps}}^{\times}$, the type theory $\Lambda_{\mathrm{ps}}^{\times,\to}$ has new *binding operations*: alongside the usual binding rules for lambda abstraction, we require that the variable x is bound in the rewrite $\mathsf{e}^{\dagger}(x\,.\,\alpha)$. This is reflected in the definition of α-equivalence.

α-equivalence and free variables For λ-abstraction we follow the usual conventions of the simply-typed lambda calculus (*c.f.* [Bar85]).

Definition 5.3.3. For any $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature $\mathcal{S}$ define the *α-equivalence relation* $=_\alpha$ on terms by extending Definition 4.3.2 with the rules

$$\frac{t[y/x] =_\alpha t'[y/x'] \qquad y \text{ fresh}}{\lambda x.t =_\alpha \lambda x'.t'} \qquad \frac{t =_\alpha t'}{\varepsilon_t =_\alpha \varepsilon_{t'}} \qquad \frac{\sigma[y/x] =_\alpha \sigma[y/x'] \qquad y \text{ fresh}}{\mathsf{e}^{\dagger}(x\,.\,\sigma) =_\alpha \mathsf{e}^{\dagger}(x'\,.\,\sigma)}$$

Similarly, the meta-operation of capture-avoiding substitution is that of Definition 4.3.2, extended by the rules

$$\mathsf{eval}(f,x)[t/f,u/x] := \mathsf{eval}\{t,u\} \quad \text{and} \quad (\lambda x.t)[u_i/x_i] := \lambda z.(t[z/x,u_i/x_i]) \text{ for } z \text{ fresh}$$

and

$$\varepsilon_t[u_i/x_i] := \varepsilon_{t[u_i/x_i]} \quad \text{and} \quad \mathsf{e}^{\dagger}(y.\alpha)[u_i/x_i] := \mathsf{e}^{\dagger}(z.\alpha[z/y,u_i/x_i]) \text{ for } z \text{ fresh}$$

These rules extend to the inverses of rewrites in the obvious fashion. ◂

Lemma 5.3.4. Let $\mathcal{S}$ be a $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature. Then in $\Lambda_{\mathrm{ps}}^{\times,\to}(\mathcal{S})$:

1. If $\Gamma \vdash t : B$ and $t =_\alpha t'$ then $\Gamma \vdash t' : B$,
2. If $\Gamma \vdash \tau : t \Rightarrow t' : B$ and $\tau =_\alpha \tau'$ then $\Gamma \vdash \tau' : t \Rightarrow t' : B$. □

The $=_\alpha$ relation is a congruence on the derived structure. In particular, one obtains the expected equality for the induced lambda abstraction operation on rewrites.

Lemma 5.3.5. Let $\mathcal{S}$ be a $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature. Then in $\Lambda_{\mathrm{ps}}^{\times,\to}(\mathcal{S})$:

1. If $\tau[y/x] =_\alpha \tau'[y/x']$ (for y fresh) then $\lambda x.\tau =_\alpha \lambda x'.\tau'$,
2. If $u =_\alpha u'$ then $\eta_u =_\alpha \eta_{u'}$,
3. If $t[y/x] =_\alpha t'[y/x']$ and $u =_\alpha u'$ then $\beta_{x.t,\,u} =_\alpha \beta_{x'.t',\,u'}$. □

As for $\Lambda_{\mathrm{ps}}^{\times}$, the type theory $\Lambda_{\mathrm{ps}}^{\times,\to}$ satisfies all the expected type-theoretic well-formedness properties.

$$\frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x.t : A \mathbin{=\!\rhd} B}\,\text{lam} \qquad \frac{}{f : A \mathbin{=\!\rhd} B, x : A \vdash \mathsf{eval}(f, x) : B}\,\text{eval}$$

Figure 5.1: Terms for cartesian closed structure

$$\frac{\Gamma, x : A \vdash t : B}{\Gamma, x : A \vdash \varepsilon_t : \mathsf{eval}\{(\lambda x.t)\{\mathrm{inc}_x\}, x\} \Rightarrow t : B}\,\varepsilon\text{-intro}$$

$$\frac{\Gamma, x : A \vdash t : B \qquad \Gamma \vdash u : A \mathbin{=\!\rhd} B \qquad \Gamma, x : A \vdash \alpha : \mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} \Rightarrow t : B}{\Gamma \vdash \mathsf{e}^{\dagger}(x \,.\, \alpha) : u \Rightarrow \lambda x.t : A \mathbin{=\!\rhd} B}\,\mathsf{e}^{\dagger}(x \,.\, \alpha)\text{-intro}$$

Figure 5.2: Rewrites for cartesian closed structure

$$\frac{\Gamma, x : A \vdash \alpha : \mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} \Rightarrow t : B}{\Gamma, x : A \vdash \alpha \equiv \varepsilon_t \bullet \mathsf{eval}\big\{\mathsf{e}^{\dagger}(x \,.\, \alpha)\{\mathrm{inc}_x\}, x\big\} : \mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} \Rightarrow t : B}\,\text{U1}$$

$$\frac{\Gamma \vdash \gamma : u \Rightarrow \lambda x.t : A \mathbin{=\!\rhd} B}{\Gamma \vdash \gamma \equiv \mathsf{e}^{\dagger}(x \,.\, \varepsilon_t \bullet \mathsf{eval}\{\gamma\{\mathrm{inc}_x\}, x\}) : u \Rightarrow \lambda x.t : A \mathbin{=\!\rhd} B}\,\text{U2}$$

$$\frac{\Gamma, x : A \vdash \alpha \equiv \alpha' : \mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} \Rightarrow t : B}{\Gamma \vdash \mathsf{e}^{\dagger}(x \,.\, \alpha) \equiv \mathsf{e}^{\dagger}(x \,.\, \alpha') : u \Rightarrow \lambda x.t : A \mathbin{=\!\rhd} B}\,\text{cong}$$

Figure 5.3: Universal property and congruence laws for $\mathsf{e}^{\dagger}(\alpha)$

$$\frac{\Gamma \vdash u : A \mathbin{=\!\rhd} B}{\Gamma \vdash \eta_u^{-1} : \lambda x.\mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} \Rightarrow u : A \mathbin{=\!\rhd} B}\,\eta^{-1}\text{-intro}$$

$$\frac{\Gamma, x : A \vdash t : B}{\Gamma, x : A \vdash \varepsilon_t^{-1} : t \Rightarrow \mathsf{eval}\{(\lambda x.t)\{\mathrm{inc}_x\}, x\} : B}\,\varepsilon^{-1}\text{-intro}$$

$$\frac{\Gamma \vdash u : A \mathbin{=\!\rhd} B}{\Gamma \vdash \eta_u \bullet \eta_u^{-1} \equiv \mathrm{id}_{\lambda x.\mathsf{eval}\{u\{\mathrm{inc}_x\},x\}} : \lambda x.\mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} \Rightarrow \lambda x.\mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} : A \mathbin{=\!\rhd} B}$$

$$\frac{\Gamma \vdash u : A \mathbin{=\!\rhd} B}{\Gamma \vdash \eta_u^{-1} \bullet \eta_u \equiv \mathrm{id}_u : u \Rightarrow u : A \mathbin{=\!\rhd} B} \qquad \frac{\Gamma, x : A \vdash t : B}{\Gamma, x : A \vdash \varepsilon_t \bullet \varepsilon_t^{-1} \equiv \mathrm{id}_t : t \Rightarrow t : B}$$

$$\frac{\Gamma, x : A \vdash t : B}{\Gamma, x : A \vdash \varepsilon_t^{-1} \bullet \varepsilon_t \equiv \mathrm{id}_{\mathsf{eval}\{(\lambda x.t)\{\mathrm{inc}_x\},x\}} : \mathsf{eval}\{(\lambda x.t)\{\mathrm{inc}_x\}, x\} \Rightarrow \mathsf{eval}\{(\lambda x.t)\{\mathrm{inc}_x\}, x\} : B}$$

Figure 5.4: Inverses for the unit and counit

Rules for $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}(\mathcal{S})$.

Definition 5.3.6. Fix a $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signature $\mathcal{S}$. We define the *free variables in a term t* in $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}(\mathcal{S})$ by extending Definition 4.3.3 as follows:

$$\mathrm{fv}(\lambda x.t) := \mathrm{fv}(t) - \{x\} \quad \text{and} \quad \mathrm{fv}(\mathsf{eval}\{p\}) := \{p\}$$

Similarly, we define the *free variables in a rewrite* τ in $\Lambda_{\mathrm{ps}}^{\times,\to}(\mathcal{S})$ by extending Definition 4.3.3 as follows:

$$\mathrm{fv}(\varepsilon_t) = \mathrm{fv}(t) \quad \text{and} \quad \mathrm{fv}\big(\mathsf{e}^{\dagger}(x\,.\,\alpha)\big) = \mathrm{fv}(\alpha) - \{x\},$$

We define the free variables of a specified inverse σ^{-1} to be exactly the free variables of σ. An occurrence of a variable in a term or rewrite is *bound* if it is not free. ◂

Lemma 5.3.7. Let $\mathcal{S}$ be a $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature. For any derivable judgements $\Gamma \vdash u : B$ and $\Gamma \vdash \tau : t \Rightarrow t' : B$ in $\Lambda_{\mathrm{ps}}^{\times,\to}(\mathcal{S})$,

1. $\mathrm{fv}(u) \subseteq \mathrm{dom}(\Gamma)$,
2. $\mathrm{fv}(\tau) \subseteq \mathrm{dom}(\Gamma)$,
3. The judgements $\Gamma \vdash t : B$ and $\Gamma \vdash t' : B$ are both derivable.

Moreover, whenever $(\Delta \vdash u_i : A_i)_{i=1,\dots,n}$ and $\Gamma := (x_i : A_i)_{i=1,\dots,n}$, then

1. If $\Gamma \vdash t : B$, then $\Delta \vdash t[u_i/x_i] : B$,
2. If $\Gamma \vdash \tau : t \Rightarrow t' : B$, then $\Delta \vdash \tau[u_i/x_i] : t[u_i/x_i] \Rightarrow t'[u_i/x_i] : B$. □

5.3.1 The syntactic model of $\Lambda_{\mathrm{ps}}^{\times,\to}$

We now turn to constructing the syntactic model for $\Lambda_{\mathrm{ps}}^{\times,\to}(\mathcal{S})$ and proving it is the free cartesian closed biclone on $\mathcal{S}$. The construction is a straightforward extension of Construction 4.3.6 (page 129).

Construction 5.3.8. For any $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature $\mathcal{S} = (\mathfrak{B}, \mathcal{G})$, define the *syntactic model* $\mathrm{Syn}^{\times,\to}(\mathcal{S})$ of $\Lambda_{\mathrm{ps}}^{\times,\to}(\mathcal{S})$ as follows. The sorts are nodes $A, B, \dots$ of $\mathcal{G}$. The 1-cells are α-equivalence classes of terms $(x_1 : A_1, \dots, x_n : A_n \vdash t : B)$ derivable in $\Lambda_{\mathrm{ps}}^{\times,\to}(\mathcal{S})$. We assume a fixed enumeration $x_1, x_2, \dots$ of variables, and that the variable name in the ith position is determined by this enumeration. The 2-cells are $\alpha\!\equiv$-equivalence classes of rewrites $(x_1 : A_1, \dots, x_n : A_n \vdash \tau : t \Rightarrow t' : B)$. Composition is vertical composition and and the identity on t is id_t; the substitution operation is explicit substitution and the structural rewrites are assoc, ι and $\varrho^{(i)}$. ◂

$\mathrm{Syn}^{\times,\to}(\mathcal{S})$ is a cartesian closed biclone. Products are as in $\mathrm{Syn}^{\times}(\mathcal{S})$ (Section 4.3.1) and for exponentials the biuniversal arrow is $\mathsf{eval}(f, x) : (f : (A \Rightarrow B), x : A) \to (y : B)$. Indeed, for any judgement $(\Gamma, x : A \vdash \alpha : \mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} \Rightarrow t : B)$ in $\Lambda_{\mathrm{ps}}^{\times,\to}(\mathcal{S})$, the rewrite $\mathsf{e}^{\dagger}(x\,.\,\alpha)$ is the unique γ (modulo $\alpha\!\equiv$) such that

$$\Gamma, x : A \vdash \alpha \equiv \varepsilon_t \bullet \mathsf{eval}\{\gamma\{\mathrm{inc}_x\}, x\} : \mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} \Rightarrow t : B \tag{5.20}$$

Existence is precisely rule U1. For uniqueness, for any γ satisfying (5.20) one has

$$\gamma \overset{\mathtt{U2}}{\equiv} \mathsf{e}^{\dagger}(x\,.\,\varepsilon_t \bullet \mathsf{eval}\{\gamma\{\mathrm{inc}_x\}, x\}) \overset{\mathtt{cong}}{\equiv} \mathsf{e}^{\dagger}(x\,.\,\alpha)$$

Moreover, $\mathrm{Syn}^{\times,\rightarrow}(\mathcal{S})$ is the free cartesian closed biclone on $\mathcal{S}$, which validates our claim that $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}(\mathcal{S})$ is the internal language of $\mathcal{FCl}^{\times,\rightarrow}(\mathcal{S})$.

Proposition 5.3.9. For any $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signature $\mathcal{S}$, cartesian closed biclone $(T, \mathcal{D}, \Pi_n(-), \Rightarrow)$, and $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signature homomorphism $h : \mathcal{S} \to \mathcal{D}$, there exists a unique strict cartesian closed pseudofunctor $h[\![-]\!] : \mathrm{Syn}^{\times,\rightarrow}(\mathcal{S}) \to \mathcal{D}$ such that $h[\![-]\!] \circ \iota = h$, for $\iota : \mathcal{S} \hookrightarrow \mathrm{Syn}^{\times,\rightarrow}(\mathcal{S})$ the inclusion.

Proof. We extend the pseudofunctor $h[\![-]\!]$ of Proposition 4.3.9 (page 130) with the following rules.

$$h[\![A \Rightarrow B]\!] := h[\![A]\!] \Rightarrow h[\![B]\!]$$

$$h[\![f : A \Rightarrow B, a : A \vdash \mathsf{eval}(f, a) : B]\!] := \mathrm{eval}_{A,B}$$

$$h[\![\Gamma \vdash \lambda x.t : A \Rightarrow B]\!] := \lambda(h[\![\Gamma, x : A \vdash t : B]\!])$$

$$h[\![\Gamma, x : A \vdash \varepsilon_t : \mathsf{eval}\{(\lambda x.t)\{\mathrm{inc}_x\}, x\} \Rightarrow t : B]\!] := \varepsilon_{h[\![\Gamma, x:A \vdash t:B]\!]}$$

$$h[\![\Gamma \vdash \mathsf{e}^{\dagger}(x\,.\,\alpha) : u \Rightarrow \lambda x.t : A \Rightarrow B]\!] := \mathsf{e}^{\dagger}(h[\![\Gamma, x : A \vdash \alpha : \mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} \Rightarrow t : B]\!])$$

Uniqueness follows because any strict cc-pseudofunctor must strictly preserve the $\lambda(-)$ and $\mathsf{e}^{\dagger}(-)$ operations (*c.f.* Lemma 5.2.17 and Lemma 2.2.17). □

Remark 5.3.10. As we saw for products (Remark 4.3.8), the universal property of the counit for exponentials gives rise to a nesting of (global) biuniversal arrows and (local) universal arrows. These are related by the following bijective correspondence, in which we write $(x : A)$ to indicate the variable x of type A is free in the context (*c.f.* [ML84]):

$$\frac{\begin{array}{c}(x : A)\\ \mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} \Rightarrow t : B\end{array}}{u \Rightarrow \lambda x.t : A \Rightarrow B}$$

We conjecture that a calculus for cartesian closed *tri*categories (cartesian closed ∞-categories) would have three (a countably infinite tower) of such correspondences. ◂

For a unary $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signature $\mathcal{S}$, the nucleus $\overline{\mathrm{Syn}^{\times,\rightarrow}(\mathcal{S})}$ of $\mathrm{Syn}^{\times,\rightarrow}(\mathcal{S})$ is cartesian closed with exponentials as described in Remark 5.2.11. We make this explicit in the next construction, which mirrors the syntactic model of the simply-typed lambda calculus (*e.g.* [Cro94, Chapter 4]).

Construction 5.3.11. For any $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signature $\mathcal{S}$, define a bicategory $\overline{\mathrm{Syn}^{\times,\rightarrow}(\mathcal{S})}$ as follows. The objects are unary contexts with a single *fixed* variable name. The 1-cells $(x : A) \to (x : B)$ are α-equivalence classes of terms $(x : A \vdash t : B)$ derivable in $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}(\mathcal{S})$. The 2-cells are $\alpha\equiv$-equivalence classes of rewrites $(x : A \vdash \tau : t \Rightarrow t' : B)$. Vertical composition is given by the $\bullet$ operation, and horizontal composition is given by explicit substitution. ◂

As we have seen, we cannot hope for $\overline{\mathrm{Syn}^{\times,\rightarrow}(\mathcal{S})}$ to satisfy a strict universal property (recall the discussion following Lemma 5.2.17 on page 159, as well as Example 4.2.63 on page 126). Nonetheless, we shall see in Section 5.3.3 that it is *weakly initial*: any morphism of $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signatures may be extended to a pseudofunctor out of $\overline{\mathrm{Syn}^{\times,\rightarrow}(\mathcal{S})}$, but this may not be unique. Hence, $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$ may be soundly interpreted in any cc-bicategory. We shall also see that $\overline{\mathrm{Syn}^{\times,\rightarrow}(\mathcal{S})}$ is biequivalent to the free cc-bicategory $\mathcal{FB}ct^{\times,\rightarrow}(\mathcal{S})$ on $\mathcal{S}$, yielding a bicategorical universal property. Before proceeding to these results, we first establish a series of lemmas that will simplify their proofs.

5.3.2 Reasoning within $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$

We begin by recovering the unit-counit presentation of exponentials (*c.f.* [See87, Hil96]) as a series of admissible rules. These are collected together in Figure 5.5, below. The proofs are similar to the case for products, so we omit them.

Lemma 5.3.12. For any $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signature $\mathcal{S}$, the rules of Figure 5.5 are admissible in $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}(\mathcal{S})$. □

A direct corollary is that the β-reduction rewrite of Definition 5.3.2 is natural.

Corollary 5.3.13. For any $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signature $\mathcal{S}$, if the judgements $(\Gamma, x : A \vdash \tau : t \Rightarrow t' : B)$ and $(\Gamma \vdash \sigma : u \Rightarrow u' : A)$ are derivable in $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}(\mathcal{S})$, then the following diagram of rewrites commutes:

$$\begin{tikzcd}
\mathsf{eval}\{\lambda x.t, u\} \arrow[r, Rightarrow, "\mathsf{eval}\{\lambda x.\tau, \sigma\}"] \arrow[d, Rightarrow, "\beta_{x.t,u}"'] & \mathsf{eval}\{\lambda x.t', u'\} \arrow[d, Rightarrow, "\beta_{x.t',u'}"] \\
t\{\mathrm{id}_\Gamma, x \mapsto u\} \arrow[r, Rightarrow, "\tau\{\mathrm{id}_\Gamma, x \mapsto \sigma\}"'] & t'\{\mathrm{id}_\Gamma, x \mapsto u'\}
\end{tikzcd}$$

□

Recall that for products we constructed a rewrite post of type

$$\mathsf{tup}(t_1, \ldots, t_m)\{u_1, \ldots, u_n\} \Rightarrow \mathsf{tup}(t_1\{u_1, \ldots, u_n\}, \ldots, t_m\{u_1, \ldots, u_n\})$$

For exponentials we call the corresponding rewrite push (*c.f.* Construction 5.1.5). Just as post witnesses that explicit substitutions and the tupling operation commute (up to isomorphism), so push witnesses that explicit substitutions and lambda abstractions can be permuted (up to isomorphism). Precisely, push relates the following two derivations (where $\Gamma := (x_i : A_i)_{i=1,\ldots,n}$):

$$\frac{\dfrac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x.t : A \Rightarrow B} \qquad (\Delta \vdash u_i : A_i)_{i=1,\ldots,n}}{\Delta \vdash (\lambda x.t)\{x_i \mapsto u_i\} : A \Rightarrow B}$$

and

$$\frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x.\mathrm{id}_t \equiv \mathrm{id}_{\lambda x.t} : \lambda x.t \Rightarrow \lambda x.t : A =\!\!\rhd B}$$

$$\frac{\Gamma, x : A \vdash \tau' : t' \Rightarrow t'' : B \qquad \Gamma, x : A \vdash \tau : t \Rightarrow t' : B}{\Gamma \vdash \lambda x.(\tau' \bullet \tau) \equiv (\lambda x.\tau') \bullet (\lambda x.\tau) : \lambda x.t \Rightarrow \lambda x.t'' : A =\!\!\rhd B}$$

$$\frac{\Gamma \vdash \sigma : u \Rightarrow u' : A =\!\!\rhd B}{\Gamma \vdash \eta_{u'} \bullet \sigma \equiv \lambda x.\mathsf{eval}\{\sigma\{\mathrm{inc}_x\}, x\} \bullet \eta_u : u \Rightarrow \lambda x.\mathsf{eval}\{u'\{\mathrm{inc}_x\}, x\} : A =\!\!\rhd B}\ \eta\text{-nat}$$

$$\frac{\Gamma, x : A \vdash \tau : t \Rightarrow t' : B}{\Gamma, x : A \vdash \tau \bullet \varepsilon_t \equiv \varepsilon_{t'} \bullet \mathsf{eval}\{(\lambda x.\tau)\{\mathrm{inc}_x\}, x\} : \mathsf{eval}\{(\lambda x.t)\{\mathrm{inc}_x\}, x\} \Rightarrow t' : B}\ \varepsilon\text{-nat}$$

$$\frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash (\lambda x.\varepsilon_t) \bullet \eta_t \equiv \mathrm{id}_{\lambda x.t} : \lambda x.t \Rightarrow \lambda x.t : A =\!\!\rhd B}\ \text{triangle-law-1}$$

$$\frac{\Gamma \vdash u : A =\!\!\rhd B}{\begin{array}{l}\Gamma, x : A \vdash \varepsilon_{\mathsf{eval}\{u\{\mathrm{inc}_x\}, x\}} \bullet \mathsf{eval}\{\eta_u\{\mathrm{inc}_x\}, x\} \equiv \mathrm{id}_{\mathsf{eval}\{u\{\mathrm{inc}_x\}, x\}} \\ \qquad\qquad\qquad : \mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} \Rightarrow \mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} : B\end{array}}\ \text{triangle-law-2}$$

Figure 5.5: Admissible rules for $\Lambda^{\times,\to}_{\mathrm{ps}}(\mathcal{G})$

$$\frac{\dfrac{\Gamma, x : A \vdash t : B \qquad \dfrac{(\Delta \vdash u_i : A_i)_{i=1,\dots,n}}{(\Delta, x : A \vdash u_i\{\mathrm{inc}_x\} : A_i)_{i=1,\dots,n}} \qquad \Delta, x : A \vdash x : A}{\Delta, x : A \vdash t\{x_i \mapsto u_i\{\mathrm{inc}_x\}, x \mapsto x\} : B}}{\Delta \vdash \lambda x.t\{x_i \mapsto u_i\{\mathrm{inc}_x\}, x \mapsto x\} : A =\!\!\rhd B}$$

From the perspective of the simply-typed lambda calculus, the rewrite

$$\mathsf{push} : (\lambda x.t)\{x_i \mapsto u_i\} \Rightarrow \lambda x.t\{x_i \mapsto u_i\{\mathrm{inc}_x\}, x \mapsto x\}$$

is an explicit version of the usual rule $(\lambda x.t)[u_i/x_i] = \lambda z.t[u_i/x_i, z/x]$ for the meta-operation of capture-avoiding substitution (*c.f.* [RdP97, Definition 4], where a similar operation is constructed for a version of the simply-typed lambda calculus with explicit substitution).

We construct push by emulating Construction 5.1.5 within $\Lambda^{\times,\to}_{\mathrm{ps}}$.

Construction 5.3.14. For any $\Lambda^{\times,\to}_{\mathrm{ps}}$-signature $\mathcal{S}$ we construct a rewrite $\mathsf{push}(t; u_\bullet)$ in $\Lambda^{\times,\to}_{\mathrm{ps}}(\mathcal{S})$ making the following rule is admissible:

$$\frac{\Gamma, x : A \vdash t : B \qquad (\Delta \vdash u_i : A_i)_{i=1,\dots,n}}{\Delta \vdash \mathsf{push}(t; u_\bullet) : (\lambda x.t)\{x_i \mapsto u_i\} \Rightarrow \lambda x.t\{x_i \mapsto u_i\{\mathrm{inc}_x\}, x \mapsto x\} : A =\!\!\rhd B}$$

Following Construction 5.1.5, we first need to construct the 2-cell Φ witnessing the pseudofunctorality of the product-former. From the judgements $\Gamma \vdash t : B$ and $(\Delta \vdash u_i : A_i)_{i=1,\dots,n}$ one obtains the terms

$$t\{\mathrm{inc}_x\}\{x_i \mapsto u_i\{\mathrm{inc}_x\}, x \mapsto x\} \quad \text{and} \quad t\{x_i \mapsto u_i\}\{\mathrm{inc}_x\}$$

of type B in context $\Delta, x : B$ by either performing explicit substitution or weakening first. These terms are related by the following composite, which we call $\Phi_{t,u_\bullet}$:

$$\begin{aligned} t\{\mathrm{inc}_x\}\{x_i \mapsto u_i\{\mathrm{inc}_x\}, x \mapsto x\} &\overset{\mathsf{assoc}}{\cong} t\{\mathrm{inc}_x\{x_i \mapsto u_i\{\mathrm{inc}_x\}, x \mapsto x\}\} \\ &\overset{t\{\varrho^{(\bullet)}\}}{\cong} t\{x_i \mapsto u_i\{\mathrm{inc}_x\}\} \\ &\overset{\mathsf{assoc}^{-1}}{\cong} t\{x_i \mapsto u_i\}\{\mathrm{inc}_x\} \end{aligned}$$

We therefore set $\mathsf{push}(t; u_\bullet)$ to be $\mathsf{e}^\dagger(x \,.\, \tau)$, for τ the composite

$$\begin{aligned} &\mathsf{eval}\big\{(\lambda x.t)\{x_i \mapsto u_i\}\{\mathrm{inc}_x\}, x\big\} \\ &\qquad \cong \mathsf{eval}\big\{(\lambda x.t)\{\mathrm{inc}_x\}\{x_i \mapsto u_i\{\mathrm{inc}_x\}, x \mapsto x\}, x\{x_i \mapsto u_i\{\mathrm{inc}_x\}, x \mapsto x\}\big\} \\ &\qquad \cong \mathsf{eval}\big\{(\lambda x.t)\{\mathrm{inc}_x\}, x\big\}\big\{x_i \mapsto u_i\{\mathrm{inc}_x\}, x \mapsto x\big\} \\ &\qquad \cong t\{x_i \mapsto u_i\{\mathrm{inc}_x\}, x \mapsto x\} \end{aligned}$$

where the first isomorphism is $\mathsf{eval}\Big\{(\Phi_{\lambda x.t, x_\bullet})^{-1}, \varrho^{(-(|\Delta|+1))}_{u_\bullet\{\mathrm{inc}_x\}, x}\Big\}$, the second is assoc^{-1} and the third is $\varepsilon_t\{u_i\{\mathrm{inc}_x\}, x\}$. ◂

Thinking of rewrites in $\Lambda_{\mathrm{ps}}^{\times,\to}$ as witnesses for equalities in the simply-typed lambda calculus, the following lemma is as expected (*c.f.* Lemma 5.1.6).

Lemma 5.3.15. For any $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature $\mathcal{S}$, if the judgements $\Gamma := (x_i : A_i)_{i=1,\dots,n}$ and $(\Delta \vdash \sigma_i : u_i \Rightarrow u_i' : A_i)$ are derivable in $\Lambda_{\mathrm{ps}}^{\times,\to}(\mathcal{S})$, then:

1. (Naturality). If $\Gamma, x : A \vdash \tau : t \Rightarrow t' : B$, then

$$\begin{tikzcd} (\lambda x.t)\{u_\bullet\} \arrow[r, Rightarrow, "\mathsf{push}"] \arrow[d, Rightarrow, "(\lambda x.\tau)\{\sigma_\bullet\}"'] & \lambda x.t\{u_\bullet\{\mathrm{inc}_x\}, x\} \arrow[d, Rightarrow, "\lambda x.\tau\{\sigma_\bullet\{\mathrm{inc}_x\}, x\}"] \\ (\lambda x.t')\{u_\bullet'\} \arrow[r, Rightarrow, "\mathsf{push}"'] & \lambda x.t'\{u_\bullet'\{\mathrm{inc}_x\}, x\} \end{tikzcd}$$

2. (Compatibility with ι). If $\Gamma, x : A \vdash t : B$, then

$$\begin{tikzcd} \lambda x.t \arrow[r, Rightarrow, "\iota"] \arrow[d, Rightarrow, "\lambda x.\iota"'] & (\lambda x.t)\{x_\bullet\} \arrow[d, Rightarrow, "\mathsf{push}"] \\ \lambda x.t\{x_\bullet\} & \lambda x.t\{x_\bullet\{\mathrm{inc}_x\}, x\} \arrow[l, Rightarrow, "\lambda x.t\{x, \varrho^{(\bullet)}\}"] \end{tikzcd}$$

3. (Compatibility with assoc). If $\Gamma, x : A \vdash t : C$, $\Delta := (y_j : B_j)_{j=1,\dots,m}$ and $(\Sigma \vdash v_j : B_j)_{j=1,\dots,m}$, then

$$\begin{tikzcd}
 & \big(\lambda x.t\{u_\bullet\{\mathrm{inc}_x\}, x\}\big)\{v_\bullet\} \arrow[dr, Rightarrow, "\mathsf{push}"] & \\
(\lambda x.t)\{u_\bullet\}\{v_\bullet\} \arrow[ur, Rightarrow, "\mathsf{push}\{v_\bullet\}"] \arrow[d, Rightarrow, "\mathsf{assoc}"'] & & \lambda x.t\{u_\bullet\{\mathrm{inc}_x\}, x\}\{v_\bullet\{\mathrm{inc}_x\}, x\} \arrow[d, Rightarrow, "\lambda x.\mathsf{assoc}"] \\
(\lambda x.t)\{u_\bullet\{v_\bullet\}\} \arrow[d, Rightarrow, "\mathsf{push}"'] & & \lambda x.t\big\{u_\bullet\{\mathrm{inc}_x\}\{v_\bullet\{\mathrm{inc}_x\}, x\}, x\{v_\bullet\{\mathrm{inc}_x\}, x\}\big\} \arrow[d, Rightarrow, "\lambda x.t\{\mathsf{assoc}, \varrho^{(m+1)}\}"] \\
\lambda x.t\big\{u_\bullet\{v_\bullet\}\{\mathrm{inc}_x\}, x\big\} \arrow[dr, Rightarrow, "\mathsf{assoc}"'] & & \lambda x.t\big\{u_\bullet\{y_\bullet\{v_\bullet\{\mathrm{inc}_x\}, x\}\}, x\big\} \arrow[dl, Rightarrow, "\lambda x.t\{u_\bullet\{\varrho^{(\bullet)}\}, x\}"] \\
 & \lambda x.t\big\{u_\bullet\{v_\bullet\{\mathrm{inc}_x\}\}, x\big\} &
\end{tikzcd}$$

4. (Compatibility with η). If $\Gamma, x : A \vdash t : B$ then

$$\begin{tikzcd}
t\{u_\bullet\} \arrow[rr, Rightarrow, "\eta\{u_\bullet\}"] \arrow[dd, Rightarrow, "\eta"'] & & \big(\lambda x.\mathsf{eval}\{t\{\mathrm{inc}_x\}, x\}\big)\{u_\bullet\} \arrow[d, Rightarrow, "\mathsf{push}"] \\
 & & \lambda x.\mathsf{eval}\{t\{\mathrm{inc}_x\}, x\}\{u_\bullet\{\mathrm{inc}_x\}, x\} \arrow[d, Rightarrow, "\lambda x.\mathsf{assoc}"] \\
\lambda x.\mathsf{eval}\big\{t\{u_\bullet\}\{\mathrm{inc}_x\}, x\big\} & & \lambda x.\mathsf{eval}\big\{t\{\mathrm{inc}_x\}\{u_\bullet\{\mathrm{inc}_x\}, x\}, x\{u_\bullet\{\mathrm{inc}_x\}, x\}\big\} \arrow[ll, Rightarrow, "\lambda x.\mathsf{eval}\{\Phi_{t;u_\bullet}, \varrho^{(m+1)}\}"]
\end{tikzcd}$$

Proof. Long but direct calculations using the universal property of $\mathsf{e}^{\dagger}(x \,.\, \alpha)$. □

The rewrite push is also compatible with the β-rewrite. In the simply-typed lambda calculus, for any terms $\Gamma, x : A \vdash t : B$ and $\Gamma \vdash u : A$ and any family $(\Delta \vdash v_i : A_i)_{i=1,\dots,n}$, then

$$(\mathsf{app}(\lambda x.t, u))[v_i/x_i] =_{\beta\eta} t[u/x][v_i/x_i] = t\big[u[v_i/x_i]/x, v_i/x_i\big] \tag{5.21}$$

In $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$ this corresponds to the two derivations

$$\dfrac{\dfrac{\dfrac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x.t : A \Rightarrow B} \qquad \Gamma \vdash u : A}{\Gamma \vdash \mathsf{eval}\{\lambda x.t, u\} : B} \qquad (\Delta \vdash v_i : A_i)_{i=1,\dots,n}}{\Delta \vdash \mathsf{eval}\{\lambda x.t, u\}\{x_i \mapsto v_i\} : B}$$

and

$$\dfrac{\Gamma, x : A \vdash t : B \qquad \dfrac{(\Delta \vdash v_i : A_i)_{i=1,\dots,n} \qquad \Gamma \vdash u : A}{\Delta \vdash u\{x_i \mapsto v_i\} : A} \qquad (\Delta \vdash v_i : A_i)_{i=1,\dots,n}}{\Delta \vdash t\{x_i \mapsto v_i, x \mapsto u\{x_i \mapsto v_i\}\} : B}$$

Continuing the equalities-as-rewrites perspective—which we make precise in Proposition 5.4.14—the equation (5.21) becomes the following lemma.

Lemma 5.3.16. Let $\mathcal{S}$ be any $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature and $\Gamma := (x_i : A_i)_{i=1,\dots,n}$ and $\Delta := (y_j : B_j)_{j=1,\dots,m}$ be contexts. If the judgements $(\Gamma, x : A \vdash t : B)$ and $(\Gamma \vdash u : A)$ and $(\Delta \vdash v_i : A_i)_{i=1,\dots,n}$ are derivable in $\Lambda_{\mathrm{ps}}^{\times,\to}(\mathcal{S})$, then

$$\begin{tikzcd}
\mathsf{eval}\{\lambda x.t, u\}\{v_\bullet\} \arrow[r, Rightarrow, "\mathsf{assoc}"] \arrow[d, Rightarrow, swap, "\beta_{x.t,u}\{v_\bullet\}"] & \mathsf{eval}\big\{(\lambda x.t)\{v_\bullet\}, u\{v_\bullet\}\big\} \arrow[d, Rightarrow, "\mathsf{eval}\{\mathsf{push}, u\{v_\bullet\}\}"] \\
t\{\mathrm{id}_\Gamma, x \mapsto u\}\{v_\bullet\} \arrow[d, Rightarrow, swap, "\cong"] & \mathsf{eval}\{\lambda x.t\{v_\bullet\{\mathrm{inc}_x\}, x\}, u\{v_\bullet\}\} \arrow[d, Rightarrow, "\beta_{x.t\{v_\bullet\{\mathrm{inc}_x\},x\},u\{v_\bullet\}}"] \\
t\{v_\bullet\{\mathrm{inc}_x\}, u\{v_\bullet\}\} & t\{v_\bullet\{\mathrm{inc}_x\}, x\}\{\mathrm{id}_\Delta, x \mapsto u\{v_\bullet\}\} \arrow[l, Rightarrow, "\cong"]
\end{tikzcd}$$

where the unlabelled isomorphisms are defined by commutativity of the following two diagrams:

$$\begin{tikzcd}
t\{\mathrm{id}_\Gamma, u\}\{v_\bullet\} \arrow[r, Rightarrow] \arrow[d, Rightarrow, swap, "\mathsf{assoc}"] & t\{v_\bullet\{\mathrm{inc}_x\}, u\{v_\bullet\}\} \\
t\{\mathrm{id}_\Gamma\{v_\bullet\}, u\{v_\bullet\}\} \arrow[r, Rightarrow, swap, "t\{\varrho^{(\bullet)}, u\{v_\bullet\}\}"] & t\{v_\bullet, u\{v_\bullet\}\} \arrow[u, Rightarrow, swap, "t\{\iota, u\{v_\bullet\}\}"]
\end{tikzcd}$$

$$\begin{tikzcd}
t\{v_\bullet\{\mathrm{inc}_x\}, x\}\{\mathrm{id}_\Delta, u\{v_\bullet\}\} \arrow[r, Rightarrow] \arrow[d, Rightarrow, swap, "\mathsf{assoc}"] & t\{v_\bullet\{\mathrm{inc}_x\}, u\{v_\bullet\}\} \\
t\big\{v_\bullet\{\mathrm{inc}_x\}\{\mathrm{id}_\Delta, u\{v_\bullet\}\}, x\{\mathrm{id}_\Delta, u\{v_\bullet\}\}\big\} \arrow[r, Rightarrow, swap, "t\{\mathsf{assoc}, \varrho^{(1)}\}"] & t\big\{v_\bullet\{y_\bullet\{\mathrm{id}_\Delta, u\{v_\bullet\}\}\}, u\{v_\bullet\}\big\} \arrow[u, Rightarrow, swap, "t\{v_\bullet\{\varrho^{(\bullet)}\}, u\{v_\bullet\}\}"]
\end{tikzcd}$$

Proof. Unfold the definitions and apply coherence. □

5.3.3 The free property of $\overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})}$

In this section we shall make precise the relationship between $\overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})}$ and the free cc-bicategory $\mathcal{FB}ct^{\times,\to}(\mathcal{S})$ on $\mathcal{S}$ (Construction 5.2.18). We establish two related results. First, we shall show that for any cc-bicategory $(\mathcal{B}, \Pi_n(-), \Rightarrow)$ and $\Lambda_{\mathrm{ps}}^{\times,\to}$-homomorphism $h : \mathcal{S} \to \mathcal{B}$, there exists a *semantic intepretation* cc-pseudofunctor $h[\![-]\!] : \overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})} \to \mathcal{B}$. Along the way, we shall observe that such an interpretation extends to the cc-bicategory defined by extending $\mathcal{T}_{\mathrm{ps}}^{@,\times}(\mathcal{S})$ (Construction 4.3.15) with exponentials. This cc-bicategory, in which every context appears as an object, will play an important role in the normalisation-by-evaluation proof of Chapter 8. Second, we shall show that $\overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})}$ is biequivalent $\mathcal{FB}ct^{\times,\to}(\mathcal{S})$. Thus, one does not obtain a strict universal property in the style of Theorem 3.2.17 (for $\Lambda_{\mathrm{ps}}^{\mathrm{bicat}}$) or Theorem 4.3.10 (for $\Lambda_{\mathrm{ps}}^{\times}$), but one does obtain such a universal property up to biequivalence.

Semantic interpretation. The semantic interpretation of $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$ follows the tradition of semantic interpretation of the simply-typed lambda calculus [Lam80, Lam86]. For a fixed cartesian closed category $(\mathbb{C}, \Pi_n(-), \Rightarrow)$ and $\Lambda^{\times,\rightarrow}$-signature homomorphism $h : \mathcal{S} \to \mathbb{C}$, the interpretation of a judgement $(\Gamma \vdash t : B)$ in the simply-typed lambda calculus over $\mathcal{S}$ is $h[\![\Gamma \vdash t : B]\!]$, where $h[\![-]\!]$ is the unique cartesian closed clone homomorphism extending h (so $h[\![-]\!]$ has domain the free cartesian closed clone on $\mathcal{S}$—namely, the syntactic model of the simply-typed lambda calculus—and codomain the cartesian closed clone $\mathrm{Cl}(\mathbb{C})$ constructed in Example 5.2.2 (page 147)).

Proposition 5.3.17. For any unary $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signature $\mathcal{S}$, cartesian closed bicategory $(\mathcal{B}, \Pi_n(-), \Rightarrow)$, and unary $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signature homomorphism $h : \mathcal{S} \to \mathcal{B}$, there exists a *semantic interpretation* $h[\![-]\!]$ assigning to every term $(\Gamma \vdash t : B)$ a 1-cell in $\mathcal{B}$ and to every rewrite $(\Gamma \vdash \tau : t \Rightarrow t' : B)$ a 2-cell in $\mathcal{B}$. Moreover, this interpretation is sound in the sense that if $(\Gamma \vdash \tau \equiv \tau' : t \Rightarrow t' : B)$ then $h[\![\Gamma \vdash \tau : t \Rightarrow t' : B]\!] = h[\![\Gamma \vdash \tau' : t \Rightarrow t' : B]\!]$.

Proof. The $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signature homomorphism h also defines a $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signature homomorphism $\mathcal{S} \to \mathrm{Bicl}(\mathcal{B})$ from $\mathcal{S}$ to the cartesian closed biclone arising from the cartesian closed structure of $\mathcal{B}$ (recall Example 5.2.12 on page 154). It follows from the universal property of $\mathrm{Syn}^{\times,\rightarrow}(\mathcal{S})$ (Proposition 5.3.9) that there exists a strict cartesian closed pseudofunctor of biclones $h[\![-]\!] : \mathrm{Syn}^{\times,\rightarrow}(\mathcal{S}) \to \mathrm{Bicl}(\mathcal{B})$. We take this to be the semantic interpretation. Soundness is then automatic. □

To avoid obstructing the flow of our discussion we leave the full description of the semantic interpretation to an appendix (Section C.2).

The following observation entails a weak universal property for $\overline{\mathrm{Syn}^{\times,\rightarrow}(\mathcal{S})}$.

Lemma 5.3.18. Let $(\mathcal{B}, \Pi_n(-), \Rightarrow)$ be a cc-bicategory and $(ob(\mathcal{B}), \mathrm{Bicl}(\mathcal{B}), \Pi_n(-), \Rightarrow)$ the associated cartesian closed biclone. Then, for any cartesian closed biclone $(S, \mathcal{C}, \Pi_n(-))$ and cartesian closed pseudofunctor of biclones $(F, \mathrm{q}^{\times}, \mathrm{q}^{\Rightarrow}) : \mathcal{C} \to \mathrm{Bicl}(\mathcal{B})$ such that $\mathrm{q}^{\times}_{X_\bullet} \cong \mathrm{Id}_{\prod_{i=1}^{n} FX_i}$ for all $X_1, \dots, X_n \in S$ $(n \in \mathbb{N})$, the restriction to unary multimaps $(\overline{F}, \mathrm{q}^{\times}, \mathrm{q}^{\Rightarrow}) : \overline{\mathcal{C}} \to \mathcal{B}$ is a cc-pseudofunctor of bicategories.

Proof. Define $\overline{F}(X) := FX$ and $\overline{F}_{X,Y} := F_{X;Y} : \overline{\mathcal{C}}(X,Y) = \mathcal{C}(X;Y) \to \mathcal{B}(X,Y)$. The 2-cells $\phi^{\overline{F}}$ and $\psi^{\overline{F}}$ are defined by restricting the 2-cells ϕ and $\psi^{(i)}$ of F to linear multimaps. The three axioms to check then follow from the three laws of a biclone pseudofunctor, restricted to linear multimaps.

For preservation of products, we are already given an equivalence

$$\langle F\pi_1, \dots, F\pi_n \rangle : F\big(\textstyle\prod_n(X_1, \dots, X_n)\big) \leftrightarrows \textstyle\prod_n(FX_1, \dots, FX_n) : \mathrm{q}^{\times}_{X_\bullet}$$

for every $X_1, \dots, X_n \in S$ $(n \in \mathbb{N})$ because tupling in $\mathrm{Bicl}(\mathcal{B})$ is tupling in $\mathcal{B}$. It follows that $(\overline{F}, \mathrm{q}^{\times})$ is an fp-pseudofunctor.

For preservation of exponentials, the cartesian closure of F provides an equivalence

$$\lambda\big(F(\mathsf{eval}_{A,B}) \circ \langle \pi_1, \pi_2 \rangle\big) : F(A \Rightarrow B) \leftrightarrows (FA \Rightarrow FB) : \mathrm{q}^{\Rightarrow}_{A,B}$$

for every $A, B \in S$ (recall from Example 5.2.12 the definition of currying in $\mathrm{Bicl}(\mathcal{B})$). On the other hand,

$$\begin{aligned} \mathrm{m}^{\overline{F}}_{A,B} &:= \lambda\big(\overline{F}(\mathsf{eval}_{A,B}) \circ \mathrm{q}^{\times}_{A,B}\big) \\ &\cong \lambda\big(\overline{F}(\mathsf{eval}_{A,B}) \circ \mathrm{Id}_{FA \times FB}\big) && \text{by assumption on } \mathrm{q}^{\times} \\ &\cong \lambda\big(\overline{F}(\mathsf{eval}_{A,B}) \circ \langle \pi_1, \pi_2 \rangle\big) \end{aligned}$$

Since $(f, g^{\star})$ is an equivalence whenever $(g, g^{\star})$ is an equivalence and $f \cong g$, it follows that $(\mathrm{m}^{\overline{F}}_{A,B}, \mathrm{q}^{\Rightarrow}_{A,B})$ is an equivalence for every $A, B \in S$. Hence, $(F, \mathrm{q}^{\times}, \mathrm{q}^{\Rightarrow})$ is a cc-pseudofunctor. □

Applying this lemma to the semantic interpretation $h[\![-]\!]$ of Proposition 5.3.17 immediately yields the following weak universal property of $\overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})}$.

Corollary 5.3.19. For any unary $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature $\mathcal{S}$, cc-bicategory $(\mathcal{B}, \Pi_n(-), \Rightarrow)$, and $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature homomorphism $h : \mathcal{S} \to \mathcal{B}$, there exists a cc-pseudofunctor $h[\![-]\!] : \overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})} \to \mathcal{B}$ such that $h[\![-]\!] \circ \iota = h$, for $\iota : \mathcal{S} \hookrightarrow \overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})}$ the inclusion. □

For the normalisation-by-evaluation argument in Chapter 8 we shall work with sets of terms indexed by types and contexts. We shall therefore require a syntactic model in which all contexts appear. For this purpose we extend $\mathcal{T}_{\mathrm{ps}}^{@,\times}(\mathcal{S})$ (Construction 4.3.15 on page 137) with exponentials. Recall from Section 4.3.3 that the resulting bicategory has two product structures: one from context extension, and the other from the type theory. We emphasise this fact in our notation.

Construction 5.3.20. For any $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature $\mathcal{S}$, define a bicategory $\mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\mathcal{S})$ as follows. The objects are contexts $\Gamma, \Delta, \dots$. The 1-cells $\Gamma \to (y_j : B_j)_{j=1,\dots,m}$ are m-tuples of α-equivalence classes of terms $(\Gamma \vdash t_j : B_j)_{j=1,\dots,m}$ derivable in $\Lambda_{\mathrm{ps}}^{\times,\to}(\mathcal{S})$, and the 2-cells $(\Gamma \vdash t_j : B_j)_{j=1,\dots,m} \Rightarrow (\Gamma \vdash t'_j : B_j)_{j=1,\dots,m}$ are m-tuples of $\alpha\equiv$-equivalence classes of rewrites $(\Gamma \vdash \tau : t_j \Rightarrow t'_j : B_j)_{j=1,\dots,m}$. Vertical composition is given pointwise by the $\bullet$ operation, and horizontal composition

$$\begin{aligned} (t_1, \dots, t_l), (u_1, \dots, u_m) &\mapsto (t_1\{x_i \mapsto u_i\}, \dots, t_m\{x_i \mapsto u_i\}) \\ (\tau_1, \dots, \tau_l), (\sigma_1, \dots, \sigma_m) &\mapsto (\tau_1\{x_i \mapsto \sigma_i\}, \dots, \tau_m\{x_i \mapsto \sigma_i\}) \end{aligned}$$

by explicit substitution. The identity on $\Delta = (y_j : B_j)_{j=1,\dots,m}$ is $(\Delta \vdash y_j : B_j)_{j=1,\dots,m}$. The structural isomorphisms l, r and a are given pointwise by ϱ, ι^{-1} and assoc, respectively. ◂

We define exponentials in a similar way to the type-theoretic product structure on $\mathcal{T}^{@,\times}_{\mathrm{ps}}(\mathcal{S})$ (Lemma 4.3.19): following Remark 5.1.4, the exponential $\Gamma \Rightarrow \Delta$ is defined to be

$$(p : \textstyle\prod_n(A_1, \dots, A_n)) \Rightarrow (q : \textstyle\prod_m(B_1, \dots, B_m))$$

for $\Gamma := (x_i : A_i)_{i=1,\dots,n}$ and $\Delta := (y_j : B_j)_{j=1,\dots,m}$.

Remark 5.3.21. Since Lemma 4.3.16 extends verbatim to $\mathcal{T}^{@,\times,\to}_{\mathrm{ps}}(\mathcal{S})$, one sees that $\mathcal{T}^{@,\times,\to}_{\mathrm{ps}}(\mathcal{S}) \simeq \overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})}$ for every unary $\Lambda^{\times,\to}_{\mathrm{ps}}$-signature $\mathcal{S}$ (*c.f.* Remark 4.3.17). Indeed, it is plain from the two definitions that the full sub-bicategory of $\mathcal{T}^{@,\times,\to}_{\mathrm{ps}}(\mathcal{S})$ consisting of just the unary contexts is exactly $\overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})}$. $\blacktriangleleft$

$\mathcal{T}^{@,\times}_{\mathrm{ps}}(\mathcal{S})$ satisfies a weak universal property akin to Corollary 5.3.19. However, since this bicategory does not arise from $\mathrm{Syn}^{\times,\to}(\mathcal{S})$ we must define the interpretation pseudofunctor by hand.

Proposition 5.3.22. For any unary $\Lambda^{\times,\to}_{\mathrm{ps}}$-signature $\mathcal{S}$, cc-bicategory $(\mathcal{B}, \Pi_n(-), \Rightarrow)$, and $\Lambda^{\times,\to}_{\mathrm{ps}}$-signature homomorphism $h : \mathcal{S} \to \mathcal{B}$, there exists a cc-pseudofunctor $h[\![-]\!] : \mathcal{T}^{@,\times,\to}_{\mathrm{ps}}(\mathcal{S}) \to \mathcal{B}$ (for the type-theoretic product structure of Lemma 4.3.18), such that $h[\![-]\!] \circ \iota = h$, for $\iota : \mathcal{S} \hookrightarrow \mathcal{T}^{@,\times,\to}_{\mathrm{ps}}(\mathcal{S})$ the inclusion.

Proof. As the notation suggests, we extend the interpretation $h[\![-]\!]$ of Proposition 5.3.17 to $\mathcal{T}^{@,\times,\to}_{\mathrm{ps}}(\mathcal{S})$ by setting

$$h[\![(\Gamma \vdash t_j : B_j)_{j=1,\dots,m}]\!] := \langle h[\![\Gamma \vdash t_1 : B_1]\!], \dots, h[\![\Gamma \vdash t_m : B_m]\!]\rangle$$
$$h\Big[\!\!\Big[(\Gamma \vdash \tau_j : t_j \Rightarrow t'_j : B_j)_{j=1,\dots,m}\Big]\!\!\Big] := \langle h[\![\Gamma \vdash \tau_1 : t_1 \Rightarrow t'_1 : B_1]\!], \dots, h[\![\Gamma \vdash \tau_m : t_m \Rightarrow t'_m : B_m]\!]\rangle$$

This is well-defined on $\alpha\equiv$-equivalence classes of rewrites by the soundness of the semantic interpretation. For preservation of composition, we define $\phi^{h[\![-]\!]}$ as follows (where $\Gamma := (x_i : A_i)_{i=1,\dots,n}$):

$$\begin{tikzcd}
h[\![(\Gamma \vdash t_j : B_j)_{j=1,\dots,m}]\!] \circ h[\![(\Delta \vdash u_i : A_i)_{i=1,..,n}]\!] \arrow[r, "\phi^{h[\![-]\!]}"] \arrow[d, equal] & h[\![(\Delta \vdash t_j\{x_i \mapsto u_i\} : B_j)_{j=1,\dots,m}]\!] \arrow[d, equal] \\
\big\langle h[\![t_j]\!]^{\Gamma}\big\rangle_j \circ \big\langle h[\![u_i]\!]^{\Delta}\big\rangle_i \arrow[r, "\mathsf{post}"'] & \big\langle h[\![t_j]\!]^{\Gamma} \circ \langle h[\![u_i]\!]^{\Delta}\rangle_i\big\rangle_j
\end{tikzcd}$$

For preservation of identities, we take

$$\psi^{h[\![\Gamma]\!]} := \mathrm{Id}_{h[\![\Gamma]\!]} \xRightarrow{\widehat{\varsigma}_{\mathrm{Id}_{h[\![\Gamma]\!]}}} \langle \pi_1, \dots, \pi_n\rangle = h[\![(\Gamma \vdash x_i : A_i)_{i=1,\dots,n}]\!]$$

where $\widehat{\varsigma}$ is defined in (5.15) on page 155. We check the three axioms of a pseudofunctor. For the left unit law, one derives the commutative diagram below, then applies the triangle law relating the unit ς and counit ϖ for products:

$$\begin{tikzcd}
\mathrm{Id}_{h[\![\Gamma]\!]} \circ \langle h[\![u_i]\!]^{\Gamma} \rangle_i \arrow[rr, "\cong"] \arrow[d, "\varsigma_{\mathrm{Id}} \circ \langle h[\![u_i]\!]^{\Gamma} \rangle_i"'] \arrow[dr, "\varsigma_{(\mathrm{Id} \circ \langle h[\![u_i]\!] \rangle_i)}"] \arrow[dd, bend right=60, "\widehat{\varsigma}_{\mathrm{Id}_{h[\![\Gamma]\!]}}"'] & & \langle h[\![u_i]\!]^{\Gamma} \rangle_i \arrow[dd, "\varsigma_{\langle h[\![u_i]\!] \rangle_i}"] \\
\big\langle \pi_{\bullet} \circ \mathrm{Id}_{h[\![\Gamma]\!]} \big\rangle \circ \langle h[\![u_i]\!]^{\Gamma} \rangle_i \arrow[d, "\cong"'] \arrow[dr, "\mathsf{post}"] & \big\langle \pi_{\bullet} \circ \big(\mathrm{Id}_{h[\![\Gamma]\!]} \circ \langle h[\![u_i]\!]^{\Gamma} \rangle_i\big) \big\rangle \arrow[d, "\cong"'] \arrow[r, phantom, "\overset{\text{nat.}}{=}"] & {} \\
\big\langle \pi_1, \ldots, \pi_n \big\rangle \circ \big\langle h[\![u_i]\!]^{\Gamma} \big\rangle_i \arrow[d, "\mathsf{post}"'] \arrow[r, phantom, "\overset{\text{nat.}}{=}"] & \big\langle \big(\pi_{\bullet} \circ \mathrm{Id}_{h[\![\Gamma]\!]}\big) \circ \langle h[\![u_i]\!]^{\Gamma} \rangle_i \big\rangle \arrow[r, "\cong"] \arrow[dl, "\cong"] & \big\langle \pi_{\bullet} \circ \langle h[\![u_i]\!]^{\Gamma} \rangle_i \big\rangle \arrow[dll, equal] \\
\big\langle \pi_{\bullet} \circ \langle h[\![u_i]\!]^{\Gamma} \rangle_i \big\rangle \arrow[d, "\langle \varpi^{(\bullet)} \rangle"'] & & \\
\langle h[\![u_i]\!]^{\Gamma} \rangle_i & &
\end{tikzcd}$$

The unlabelled triangular shape is an easily-verified property of post $\big($*c.f.* Lemma 4.1.7, diagram (4.5)$\big)$. The right unit law is similar, and the associativity law follows directly from the naturality of post and the observation that the following commutes $\big($*c.f.* Lemma 4.1.7(4.6)$\big)$:

$$\begin{tikzcd}
(\langle f_{\bullet} \rangle \circ g) \circ h \arrow[rr, "\mathsf{post} \circ h"] \arrow[d, "\cong"'] & & \langle f_{\bullet} \circ g \rangle \circ h \arrow[d, "\mathsf{post}"] \\
\langle f_{\bullet} \rangle \circ (g \circ h) \arrow[r, "\mathsf{post}"'] & \langle f_{\bullet} \circ (g \circ h) \rangle \arrow[r, "{\langle \cong, \ldots, \cong \rangle}"'] & \langle (f_{\bullet} \circ g) \circ h \rangle
\end{tikzcd}$$

Now we want to show that $h[\![-]\!]$ is a cc-pseudofunctor. We start with products. It is immediate from the definition that, for any family of unary contexts $(x_1 : A_1), \ldots, (x_n : A_n)$ $(n \in \mathbb{N})$, the pseudofunctor $h[\![-]\!]$ strictly preserves the data making $(p : \prod_n(A_1, \ldots, A_n)) = \prod_{i=1}^{n}(x_i : A_i)$ an n-ary product. More generally, for contexts $\Gamma^{(i)} := (x_j^{(i)} : A_j^{(i)})_{j=1,\ldots,|\Gamma^{(i)}|}$ $(i = 1, \ldots, n)$, the n-ary product $\Gamma^{(1)} \times \cdots \times \Gamma^{(n)}$ is interpreted as

$$h\Big[\!\!\Big[p : \textstyle\prod_n\big(\prod_{|\Gamma^{(1)}|} A_{\bullet}^{(1)}, \ldots, \prod_{|\Gamma^{(n)}|} A_{\bullet}^{(n)}\big)\Big]\!\!\Big] = \prod_{i=1}^{n} \prod_{j=1}^{|\Gamma^{(j)}|} h[\![A_j^{(i)}]\!] = \prod_{i=1}^{n} h[\![\Gamma^{(i)}]\!]$$

and the ith projection

$$\Big(p : \textstyle\prod_n\big(\prod_{|\Gamma^{(1)}|} A_{\bullet}^{(1)}, \ldots, \prod_{|\Gamma^{(n)}|} A_{\bullet}^{(n)}\big) \vdash \pi_j\{\pi_i(p)\} : A_j^{(i)}\Big)_{j=1,\ldots,|\Gamma^{(i)}|}$$

is interpreted as $\prod_{i=1}^{n} h[\![\Gamma^{(i)}]\!] \xrightarrow{\left\langle \pi_1 \circ \pi_i, \ldots, \pi_{|\Gamma^{(i)}|} \circ \pi_i \right\rangle} \prod_{j=1}^{|\Gamma^{(i)}|} h[\![A_j^{(i)}]\!] = h[\![\Gamma^{(i)}]\!]$. To witness that $h[\![-]\!]$ preserves products, then, one can take $\mathrm{q}_{\Gamma^{(\bullet)}}^{\times}$ to be the identity, with witnessing 2-cell

$$
\begin{aligned}
\big\langle\langle\pi_\bullet\circ\pi_1\rangle,\dots,\langle\pi_\bullet\circ\pi_n\rangle\big\rangle
&\xRightarrow{\langle\mathsf{post}^{-1},\dots,\mathsf{post}^{-1}\rangle}
\big\langle\langle\pi_1,\dots,\pi_{|\Gamma^{(1)}|}\rangle\circ\pi_1,\dots,\langle\pi_1,\dots,\pi_{|\Gamma^{(n)}|}\rangle\circ\pi_n\big\rangle \\
&\xRightarrow{\langle\widehat{\varsigma}^{-1},\dots,\widehat{\varsigma}^{-1}\rangle}
\langle\mathrm{Id}_{h\llbracket\Gamma^{(1)}\rrbracket}\circ\pi_1,\dots,\mathrm{Id}_{h\llbracket\Gamma^{(n)}\rrbracket}\circ\pi_n\rangle \\
&\cong\langle\pi_1,\dots,\pi_n\rangle \\
&\xRightarrow{\widehat{\varsigma}^{-1}}\mathrm{Id}_{h\llbracket\prod_i\Gamma^{(i)}\rrbracket}
\end{aligned}
$$

Note we once again use the 2-cell $\widehat{\varsigma}$ defined in (5.15) on page 155.

For exponentials, one sees that (where $\Delta := (y_j : B_j)_{j=1,\dots,m}$):

$$
\begin{aligned}
h\llbracket\Gamma\Rightarrow\Delta\rrbracket &= h\Big\llbracket\big(p:\textstyle\prod_n(A_1,\,\dots\,,A_n)\big)\Rightarrow\big(q:\prod_m(B_1,\,\dots\,,B_m)\big)\Big\rrbracket \\
&= h\llbracket f:\textstyle\prod_n(A_1,\,\dots\,,A_n)\Rightarrow\prod_m(B_1,\,\dots\,,B_m)\rrbracket \\
&= \big(\textstyle\prod_{i=1}^n h\llbracket A_i\rrbracket\big)\Rightarrow\big(\prod_{j=1}^n h\llbracket B_j\rrbracket\big)
\end{aligned}
$$

and

$$
\begin{aligned}
h\llbracket(\Gamma\Rightarrow\Delta)\times\Gamma\rrbracket &= h\big\llbracket p:\textstyle\prod_2\big(\prod_n A_\bullet\Rightarrow\prod_m B_\bullet,\prod_n A_\bullet\big)\big\rrbracket \\
&= \big(\textstyle\prod_{i=1}^n h\llbracket A_i\rrbracket\Rightarrow\prod_{j=1}^n h\llbracket B_j\rrbracket\big)\times\prod_{i=1}^n h\llbracket A_i\rrbracket
\end{aligned}
$$

It follows that $\mathrm{m}_{\Gamma,\Delta}^{h\llbracket-\rrbracket}$ is the currying of

$$
\begin{aligned}
h\big\llbracket p:\textstyle\prod_2\big(\prod_n A_\bullet\Rightarrow\prod_m B_\bullet,\prod_n A_\bullet\big)\vdash\mathsf{eval}\{\pi_1(p),\pi_2(p)\}:\prod_m B_\bullet\big\rrbracket &\circ\mathrm{Id}_{(h\llbracket\Gamma\Rightarrow\Delta\rrbracket\times h\llbracket\Gamma\rrbracket)} \\
= \big(\mathrm{eval}_{h\llbracket\Gamma\rrbracket,h\llbracket\Delta\rrbracket}\circ\langle\pi_1,\pi_2\rangle\big)&\circ\mathrm{Id}_{(h\llbracket\Gamma\Rightarrow\Delta\rrbracket\times h\llbracket\Gamma\rrbracket)}
\end{aligned}
$$

Hence, $\mathrm{m}_{\Gamma,\Delta}^{h\llbracket-\rrbracket}$ is naturally isomorphic to the identity via the composite

$$
\begin{aligned}
&\lambda\big(\,\big(\mathrm{eval}_{(h\llbracket\Gamma\rrbracket,h\llbracket\Delta\rrbracket)}\circ\langle\pi_1,\pi_2\rangle\big)\circ\mathrm{Id}_{(h\llbracket\Gamma\Rightarrow\Delta\rrbracket\times h\llbracket\Gamma\rrbracket)}\big) \\
&\quad\cong\lambda\big(\mathrm{eval}_{(h\llbracket\Gamma\rrbracket,h\llbracket\Delta\rrbracket)}\circ\big\langle\pi_1\circ\mathrm{Id}_{(h\llbracket\Gamma\Rightarrow\Delta\rrbracket\times h\llbracket\Gamma\rrbracket)},\mathrm{Id}_{\pi_2\circ(h\llbracket\Gamma\Rightarrow\Delta\rrbracket\times h\llbracket\Gamma\rrbracket)}\big\rangle\big) \\
&\quad\cong\lambda\big(\mathrm{eval}_{(h\llbracket\Gamma\rrbracket,h\llbracket\Delta\rrbracket)}\circ(\mathrm{Id}_{h\llbracket\Gamma\Rightarrow\Delta\rrbracket}\times\textstyle\prod_m h\llbracket B_\bullet\rrbracket)\big) \\
&\quad\overset{\eta}{\cong}\mathrm{Id}_{h\llbracket\Gamma\Rightarrow\Delta\rrbracket}
\end{aligned}
$$

and $h\llbracket-\rrbracket$ is a cc-pseudofunctor. □

Our aim now is to prove that $\overline{\mathrm{Syn}^{\times,\rightarrow}(\mathcal{S})}$ is biequivalent to the free cc-bicategory on the unary $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$-signature $\mathcal{S}$ (defined in Construction 5.2.18), and hence that $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$ is the internal language for cc-bicategories up to biequivalence.

$\overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})}$ **is biequivalent to $\mathcal{FB}ct^{\times,\to}(\mathcal{S})$.** Fix a unary $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature $\mathcal{S}$. We shall show that the canonical cc-pseudofunctors $\iota^{\#} : \mathcal{FB}ct^{\times,\to}(\mathcal{S}) \to \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\mathcal{S})$ and $\iota\llbracket - \rrbracket : \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\mathcal{S}) \to \mathcal{FB}ct^{\times,\to}(\mathcal{S})$ extending the respective inclusions $\mathcal{S} \hookrightarrow \mathcal{FB}ct^{\times,\to}(\mathcal{S})$ and $\mathcal{S} \hookrightarrow \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\mathcal{S})$ induce a biequivalence $\mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\mathcal{S}) \simeq \mathcal{FB}ct^{\times,\to}(\mathcal{S})$. (These cc-pseudofunctors are defined in Lemma 5.2.19 and Proposition 5.3.22, respectively.) One then obtains the required biequivalence by restricting $\mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\mathcal{S})$ to unary contexts (recall Remark 5.3.21).

Remark 5.3.23. Because the pseudofunctor $\iota^{\#}$ is defined inductively using the cartesian closed structure of $\mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\mathcal{S})$, we must be explicit about which cartesian closed structure we choose. We take the *type-theoretic* product structure, so that the composite $\iota^{\#} \circ \iota\llbracket - \rrbracket$ takes an arbitrary context Γ to an (equivalent) unary context. Because the restriction of $\mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\mathcal{S})$ to unary contexts is exactly $\overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})}$, this ensures that the biequivalence we construct will restrict to $\overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})}$ with its canonical cartesian closed structure (namely, that of Remark 5.2.11). Of course, up to biequivalence of the underlying bicategories, the uniqueness of products and exponentials ensures that the choice of cc-bicategory is immaterial (recall Remark 5.1.8 and Lemma 5.1.9). ◂

Our two-step approach reflects two intended applications. In this chapter we wish to prove a free property, so restrict to unary contexts, but in Chapter 8 we wish to interpret the syntax of $\Lambda_{\mathrm{ps}}^{\times,\to}$ varying over a (2-)category of contexts, and so require all contexts.

Remark 5.3.24. Although we present the argument indirectly here, it is also possible to prove directly that the canonical cc-pseudofunctors induce a biequivalence $\overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})} \simeq \mathcal{FB}ct^{\times,\to}(\mathcal{S})$. The calculations involved are similar to those we shall see below. ◂

We begin by showing that $\iota\llbracket - \rrbracket \circ \iota^{\#} \simeq \mathrm{id}_{\mathcal{FB}ct^{\times,\to}(\mathcal{S})}$. Recall from Proposition 5.3.22 that $\iota\llbracket - \rrbracket$ preserves products and exponentials up to equivalence in a particularly strong way, in the sense that $\langle \iota\llbracket \pi_1 \rrbracket, \dots, \iota\llbracket \pi_n \rrbracket \rangle \cong \mathrm{id}$ and $\mathrm{m}^{\iota\llbracket - \rrbracket} \cong \mathrm{id}$. One may therefore apply Corollary 5.2.21.

Proposition 5.3.25. For any unary $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature $\mathcal{S}$, the composite $\iota\llbracket - \rrbracket \circ \iota^{\#} : \mathcal{FB}ct^{\times,\to}(\mathcal{S}) \to \mathcal{FB}ct^{\times,\to}(\mathcal{S})$ induced by the following diagram is equivalent to $\mathrm{id}_{\mathcal{FB}ct^{\times,\to}(\mathcal{S})}$:

$$\begin{tikzcd}
\mathcal{FB}ct^{\times,\to}(\mathcal{S}) \arrow[r, "\iota^{\#}"] & \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\mathcal{S}) \arrow[r, "\iota\llbracket - \rrbracket"] & \mathcal{FB}ct^{\times,\to}(\mathcal{S}) \\
\mathcal{S} \arrow[u, hook, "\iota"] \arrow[r, equal] & \mathcal{S} \arrow[u, hook, "\iota"] \arrow[r, equal] & \mathcal{S} \arrow[u, hook, "\iota"']
\end{tikzcd}$$

Proof. The diagram commutes, and the composite $\iota\llbracket - \rrbracket \circ \iota^{\#}$ is certainly a cc-pseudofunctor. Since $\iota^{\#}$ is strict and $\iota\llbracket - \rrbracket$ has $\mathrm{q}^{\times}$ and $\mathrm{q}^{\Rightarrow}$ both given by the identity, Corollary 5.2.21 applies. Hence $\iota\llbracket - \rrbracket \circ \iota^{\#}$ is equivalent to the unique strict cc-pseudofunctor $\mathcal{FB}ct^{\times,\to}(\mathcal{S}) \to \mathcal{FB}ct^{\times,\to}(\mathcal{S})$ extending the inclusion $\mathcal{S} \hookrightarrow \mathcal{FB}ct^{\times,\to}(\mathcal{S})$. Since the identity is such a strict cc-pseudofunctor, it follows that $\iota\llbracket - \rrbracket \circ \iota^{\#} \simeq \mathrm{id}_{\mathcal{FB}ct^{\times,\to}(\mathcal{S})}$, as required. □

We shall see in Chapter 8 that this result is crucial to the normalisation-by-evaluation proof. Roughly speaking, it plays the same role as the 1-categorical observation that the canonical map from the free cartesian closed category to itself is the identity.

We now turn to showing that $\iota^{\#} \circ \iota\llbracket - \rrbracket$ is equivalent to the identity. To this end, observe that for any context $\Gamma := (x_i : A_i)_{i=1,\ldots,n}$,

$$\iota^{\#}(\iota\llbracket \Gamma \rrbracket) = \iota^{\#}(\textstyle\prod_n(A_1, \ldots, A_n)) = (p : \textstyle\prod_n(A_1, \ldots, A_n))$$

We define a pseudonatural transformation $(\mathsf{j}, \bar{\mathsf{j}}) : \iota^{\#} \circ \iota\llbracket - \rrbracket \Rightarrow \mathrm{id}_{\mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\mathcal{S})}$ with components $\mathsf{j}_\Gamma : \iota^{\#}(\iota\llbracket \Gamma \rrbracket) \to \Gamma$ given by the equivalence

$$\begin{tikzcd} \Gamma \arrow[r, shift left, "(\Gamma \vdash \mathsf{tup}(x_1, \ldots, x_n) : \prod_n A_\bullet)"] & \left(p : \prod_n(A_1, \ldots, A_n)\right) \arrow[l, shift left, "(p : \prod_n(A_1, \ldots, A_n) \vdash \pi_i(p) : A_i)_{i=1,\ldots,n}"] \end{tikzcd}$$

constructed in Lemma 4.3.16 (page 138). We are therefore required to provide an invertible 2-cell filling the diagram below for every judgement $(\Gamma \vdash t : B)$:

$$\begin{tikzcd} \iota^{\#}(\iota\llbracket \Gamma \rrbracket) \arrow[r, "\iota^{\#}(\iota\llbracket \Gamma \vdash t : B \rrbracket)"] \arrow[d, "\mathsf{j}_\Gamma"'] \arrow[dr, phantom, "\overset{\bar{\mathsf{j}}_t}{\Leftarrow}"] & \iota^{\#}(\iota\llbracket y : B \rrbracket) \arrow[d, "\mathsf{j}_B"] \\ \Gamma \arrow[r, "(\Gamma \vdash t : B)"'] & (y : B) \end{tikzcd} \tag{5.22}$$

Construction 5.3.26. For any $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature $\mathcal{S}$, we define a family of 2-cells $\bar{\mathsf{j}}_t$ filling (5.22) in $\mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\mathcal{S})$. Unfolding the anticlockwise composite, one sees that

$$\begin{aligned} (\Gamma \vdash t : B) \circ \mathsf{j}_\Gamma &= (\Gamma \vdash t : B) \circ \left(p : \textstyle\prod_n A_\bullet \vdash \pi_i(p) : A_i\right)_{i=1,\ldots,n} \\ &= \left(p : \textstyle\prod_n(A_1, \ldots, A_n) \vdash t\{x_i \mapsto \pi_i(p)\} : B\right) \end{aligned}$$

Thus, it suffices to define 2-cells $\bar{\mathsf{k}}_t$ of type $(p : \prod_n A_\bullet \vdash \bar{t} \Rightarrow t\{x_i \mapsto \pi_i(p)\} : B)$, where $\bar{t}$ is the term in the judgement $\iota^{\#}(\iota\llbracket \Gamma \vdash t : B \rrbracket)$. Since j_B is simply $(y : B \vdash y : B)$, one may then define the required 2-cell $\bar{\mathsf{j}}_t$ to be

$$\bar{\mathsf{j}}_t := y\{\bar{t}\} \xRightarrow{\varrho_{\bar{t}}^{(1)}} \bar{t} \overset{\bar{\mathsf{k}}_t}{\Rightarrow} t\{x_i \mapsto \pi_i(p)\}$$

We define $\bar{\mathsf{k}}_t$ by induction on the derivation of t.

$\boxed{\texttt{var}\text{ case.}}$ For $(\Gamma \vdash x_k : A_k)$ the corresponding term $\overline{x_k}$ is $\left(p : \prod_n A_\bullet \vdash \pi_k(p) : A_k\right)$, so we define

$$\bar{\mathsf{k}}_{x_i} := \left(p : \textstyle\prod_n A_\bullet \vdash \varrho_{\pi_\bullet(p)}^{(-k)} : \pi_k(p) \Rightarrow x_k\{x_i \mapsto \pi_i(p)\} : A_k\right)$$

$\boxed{\texttt{const}\text{ case.}}$ For any constant $c \in \mathcal{G}(A, B)$, the judgement $\iota^{\#}\iota\llbracket x : A \vdash c(x) : B \rrbracket$ is simply $(x : A \vdash c(x) : B)$. Since the context is unary, j_Γ is the identity and we may take $\bar{\mathsf{k}}_{c(x)}$ to be canonical structural isomorphism.

$\boxed{\texttt{proj}\text{ case.}}$ Observing that $\iota^{\#} \circ \iota\llbracket - \rrbracket$ is the identity on $(p : \prod_n(A_1, \ldots, A_n) \vdash \pi_i(p) : A_i)$, we take the canonical isomorphism

$$\begin{tikzcd}
\big(p : \prod_n(A_1, \dots, A_n)\big) \arrow[r, "(p:\prod_n A_\bullet\vdash\pi_i(p):A_i)"] \arrow[d, "(p:\prod_n A_\bullet\vdash p:\prod_n A_\bullet)"'] \arrow[dr, "(p:\prod_n A_\bullet\vdash\pi_i(p):A_i)" description] \arrow[dr, phantom, shift left=4ex, "\cong"] \arrow[dr, phantom, shift right=4ex, "\cong"] & (x_i : A_i) \arrow[d, "(x_i:A_i\vdash x_i:A_i)"] \\
\big(p : \prod_n(A_1, \dots, A_n)\big) \arrow[r, "(p:\prod_n A_\bullet\vdash\pi_i(p):A_i)"'] & (x_i : A_i)
\end{tikzcd}$$

$\boxed{\texttt{tup}\text{ case.}}$ From the induction hypothesis one obtains $\big(p : \prod_n A_\bullet \vdash \overline{\mathsf{k}}_{t_i} : \overline{t_j} \Rightarrow t_j\{x_i \mapsto \pi_i(p)\} : B_j\big)$ for $j = 1, \dots, m$. So for $\overline{\mathsf{k}}_{\mathsf{tup}(t_1, \dots, t_m)}$ we take the composite rewrite

$$\mathsf{tup}(\overline{t_1}, \dots, \overline{t_m}) \xRightarrow{\mathsf{tup}(\overline{\mathsf{k}}_{t_1}, \dots, \overline{\mathsf{k}}_{t_m})} \mathsf{tup}(t_1\{\pi_\bullet(p)\}, \dots, t_m\{\pi_\bullet(p)\}) \xRightarrow{\mathsf{post}^{-1}} \mathsf{tup}(t_1, \dots, t_m)\{\pi_\bullet(p)\}$$

of type $\prod_m(B_1, \dots, B_m)$ in context $\big(p : \prod_n(A_1, \dots, A_n)\big)$.

$\boxed{\texttt{eval}\text{ case.}}$ The evaluation 1-cell $(f : A \Rightarrow B) \times (x : A) \to (y : B)$ in $\mathcal{T}_{\mathrm{ps}}^{@,\times,\rightarrow}(\mathcal{S})$ with the type-theoretic product structure is $\big(p : (A \Rightarrow B) \times A \vdash \mathsf{eval}\{\pi_1(p), \pi_2(p)\} : B\big)$, so one obtains

$$\begin{aligned}
\iota^{\#}(\iota[\![f : A \Rightarrow B, x : A \vdash \mathsf{eval}(f, x) : B]\!]) &= \iota^{\#}(\mathrm{eval}_{\iota[\![A]\!],\iota[\![B]\!]}) \\
&= \big(p : (A \Rightarrow B) \times A \vdash \mathsf{eval}\{\pi_1(p), \pi_2(p)\} : B\big)
\end{aligned}$$

We therefore define $\overline{\mathsf{k}}_{\mathsf{eval}(f,x)}$ to be the identity.

$\boxed{\texttt{lam}\text{ case.}}$ The exponential transpose of a term $(p : Z \times B \vdash t : C)$ in $\mathcal{T}_{\mathrm{ps}}^{@,\times,\rightarrow}(\mathcal{S})$ is

$$(z : Z \vdash \lambda x.(t\{p \mapsto \mathsf{tup}(z, x)\}) : B \Rightarrow C)$$

It follows that

$$\begin{aligned}
\iota^{\#}(\iota[\![\Gamma \vdash \lambda x.t : B \Rightarrow C]\!]) &= \lambda\big(q : \textstyle\prod_2(\prod_n A_\bullet, B) \vdash \bar{t}\big\{\mathsf{tup}(\pi_\bullet\{\pi_1(q)\}, \pi_2(q))\big\} : C\big) \\
&= \big(p : \textstyle\prod_n A_\bullet \vdash \lambda x.\bar{t}\big\{\mathsf{tup}(\pi_\bullet\{\pi_1(q)\}, \pi_2(q))\big\}\{\mathsf{tup}(p, x)\} : B \Rightarrow C\big)
\end{aligned}$$

Now, the induction hypothesis provides the 2-cell $\big(s : \prod_n(A_1, \dots, A_n, B) \vdash \overline{\mathsf{k}}_t : \bar{t} \Rightarrow t\{x_i \mapsto \pi_i(s)\} : C\big)$ so for $\overline{\mathsf{k}}_{\lambda x.t}$ we begin by defining a composite ϑ_t by

$$\begin{tikzcd}
\bar{t}\big\{\mathsf{tup}(\pi_1\{\pi_1(q)\}, \dots, \pi_n\{\pi_1(q)\}, \pi_2(q))\big\}\{\mathsf{tup}(p, x)\} \arrow[d, Rightarrow, "\mathsf{assoc}"'] \arrow[ddr, Rightarrow, bend left, "\vartheta_t"] & \\
\bar{t}\big\{\mathsf{tup}\big(\pi_1\{\pi_1(q)\}, \dots, \pi_n\{\pi_1(q)\}, \pi_2(q)\big)\{\mathsf{tup}(p, x)\}\big\} \arrow[d, Rightarrow, "\bar{t}\{\mathsf{post}\}"'] & \\
\bar{t}\big\{\mathsf{tup}\big(\pi_1\{\pi_1(q)\}\{\mathsf{tup}(p, x)\}, \dots, \pi_n\{\pi_1(q)\}\{\mathsf{tup}(p, x)\}, \pi_2\{\mathsf{tup}(p, x)\}\big)\big\} \arrow[r, Rightarrow, bend right, "\bar{t}\big\{\mathsf{tup}(\gamma_1, \dots, \gamma_n, \varpi^{(2)}_{p,x})\big\}"'] & \bar{t}\big\{\mathsf{tup}(\pi_1\{p\}, \dots, \pi_n\{p\}, x)\big\}
\end{tikzcd}$$

in context $\big(p : \prod_n(A_1, \ldots, A_n), x : B\big)$, where γ_k is defined, in the same context, to be

$$\gamma_k := \pi_k\{\pi_1(q)\}\{\mathsf{tup}(p, x)\} \xRightarrow{\mathsf{assoc}} \pi_k\big\{\pi_1\{\mathsf{tup}(p, x)\}\big\} \xRightarrow{\pi_k\{\varpi^{(1)}_{p,x}\}} \pi_k\{p\}$$

for $k = 1, \ldots, n$. We then define $\overline{\mathsf{k}}_{\lambda x.t}$ to be the composite

$$\begin{tikzcd}
\lambda x.\overline{t}\big\{\mathsf{tup}(\pi_\bullet\{\pi_1(q)\}, \pi_2(q))\big\}\{\mathsf{tup}(p, x)\} \arrow[r, Rightarrow, "\overline{\mathsf{k}}_{\lambda x.t}"] \arrow[d, Rightarrow, "\lambda x.\vartheta_t"'] & (\lambda x.t)\{\pi_1(p), \ldots, \pi_n(p)\} \\
\lambda x.\overline{t}\big\{\mathsf{tup}(\pi_1\{p\}, \ldots, \pi_n\{p\}, x)\big\} \arrow[d, Rightarrow, "\lambda x.\overline{\mathsf{k}}_t\{\mathsf{tup}(\pi_1\{p\}, \ldots, \pi_n\{p\}, x)\}"'] & \\
\lambda x.t\{\pi_1(s), \ldots, \pi_n(s), \pi_{n+1}(s)\}\{\mathsf{tup}(\pi_1\{p\}, \ldots, \pi_n\{p\}, x)\} \arrow[d, Rightarrow, "\lambda x.\mathsf{assoc}"'] & \\
\lambda x.t\big\{\pi_\bullet\{\mathsf{tup}(\pi_1\{p\}, \ldots, \pi_n\{p\}, x)\}\big\} \arrow[r, Rightarrow, "\lambda x.t\{\varpi^{(\bullet)}\}"'] & \lambda x.t\{\pi_1\{p\}, \ldots, \pi_n\{p\}, x\} \arrow[uuu, Rightarrow, "\mathsf{push}^{-1}"']
\end{tikzcd}$$

It remains to consider the cases of explicit substitutions and n-tuples of terms. We take the latter first and then put it to work for explicit substitutions.

$\boxed{n\text{-tuples case.}}$ For contexts $\Gamma := (x_i : A_i)_{i=1, \ldots, n}$ and $\Delta := (z_j : Z_j)_{j=1, \ldots, m}$ and an n-tuple $(\Delta \vdash t_i : A_i)_{i=1, \ldots, n} : \Delta \to \Gamma$, we directly define the rewrite $\bar{\mathsf{j}}_{(t_j)_{j=1, \ldots, m}}$ filling

$$\begin{tikzcd}
\big(q : \prod_m(Z_1, \ldots, Z_m)\big) \arrow[rr, "\big(q:\prod_m Z_\bullet \vdash \mathsf{tup}(\overline{t_1}, \ldots, \overline{t_n}):\prod_n A_\bullet\big)"] \arrow[d, "\simeq"'] & & \big(p : \prod_n(A_1, \ldots, A_n)\big) \arrow[d, "\simeq"] \\
\Delta \arrow[rr, "(\Delta \vdash t_i : A_i)_{i=1, \ldots, n}"'] \arrow[rr, phantom, shift left=4, "\overset{\bar{\mathsf{j}}_{(t_i)_{i=1, \ldots, n}}}{\Leftarrow}"] & & \Gamma
\end{tikzcd}$$

to be the n-tuple with components

$$\bar{\mathsf{j}}_{(t_i)_{i=1, \ldots, n}} := \pi_k\big\{\mathsf{tup}(\overline{t_1}, \ldots, \overline{t_n})\big\} \xRightarrow{\varpi^{(k)}} \overline{t_k} \xRightarrow{\overline{\mathsf{k}}_{t_k}} t_k\{\pi_1(q), \ldots, \pi_m(q)\}$$

for $k = 1, \ldots, n$.

$\boxed{\texttt{hcomp}\text{ case.}}$ For explicit substitutions $(\Delta \vdash t\{x_i \mapsto u_i\} : B) = (\Gamma \vdash t : B) \circ (\Delta \vdash u_i : A_i)_{i=1,\ldots,n}$ we take the definition from the associativity law of a pseudonatural transformation. Thus, we define $\bar{\mathsf{j}}_{t\{x_i \mapsto u_i\}}$ to be the pasting diagram

$$\begin{tikzcd}
\big(q : \prod_m(B_1, \dots, B_m)\big) \arrow[rr, "\big(q{:}\prod_m B_\bullet \vdash \bar{t}\{\mathsf{tup}(\overline{u_1}, \dots, \overline{u_n})\}{:}C\big)"] \arrow[ddd, "\simeq"'] \arrow[dr, "(q{:}\prod_m B_\bullet \vdash \mathsf{tup}(\overline{u_1}, \dots, \overline{u_n}){:}\prod_n A_\bullet)" description] & & (z : C) \arrow[ddd, "(z{:}C \vdash z{:}C)"] \\
& \big(p : \prod_n(A_1, \dots, A_n)\big) \arrow[ur, "\big(p{:}\prod_n(A_1, \dots, A_n) \vdash \bar{t}{:}C\big)" description] \arrow[d, "\simeq" description] & \\
\overset{\bar{\mathsf{j}}_{(u_i)_{i=1, \dots, n}}}{\Leftarrow} & \Gamma \arrow[dr, "(\Gamma \vdash t{:}C)" description] & \overset{\bar{\mathsf{j}}_t}{\Leftarrow} \\
\Delta \arrow[rr, "(\Delta \vdash t\{x_i \mapsto u_i\}{:}C)"'] \arrow[ur, "(\Delta \vdash u_i{:}A_i)_{i=1, \dots, n}" description] & & (z : C)
\end{tikzcd}$$

◂

The preceding construction does indeed define a pseudonatural transformation. It is clear that each $\bar{\mathsf{j}}_t$ is natural, so it remains to check the unit and associativity laws. For the unit law, we are required to show the following equality of pasting diagrams for every context $\Gamma := (x_i : A_i)_{i=1,\dots,n}$:

$$\begin{tikzcd}
(p : \prod_n A_\bullet) \arrow[r, bend left, "\big(p{:}\prod_n A_\bullet \vdash p{:}\prod_n A_\bullet\big)"] \arrow[r, bend right, "\big(p{:}\prod_n A_\bullet \vdash \mathsf{tup}(\pi_1\{p\}, \dots, \pi_n\{p\})){:}\prod_n A_\bullet\big)"'] \arrow[r, phantom, "\overset{\psi^{\iota^{\#} \circ \iota[\![-]\!]}}{\cong}"] \arrow[d, "\simeq"'] & (p : \prod_n A_\bullet) \arrow[d, "\simeq"] \arrow[r, phantom, "="] & (p : \prod_n A_\bullet) \arrow[r, "\big(p{:}\prod_n A_\bullet \vdash p{:}\prod_n A_\bullet\big)"] \arrow[d, "\simeq"'] \arrow[dr, "\simeq" description] & (p : \prod_n A_\bullet) \arrow[d, "\simeq"] \\
\Gamma \arrow[r, "(\Gamma \vdash x_i{:}A_i)_{i=1, \dots, n}"'] \arrow[ur, phantom, "\overset{\bar{\mathsf{j}}_{(x_i)_{i=1, \dots, n}}}{\Leftarrow}"] & \Gamma & \Gamma \arrow[r, "(\Gamma \vdash x_i{:}A_i)_{i=1, \dots, n}"'] \arrow[ur, phantom, "\cong" near start] & \Gamma \arrow[ul, phantom, "\cong" near start]
\end{tikzcd}$$

Applying the definition of $\psi^{\iota[\![-]\!]}$ given in Proposition 5.3.22, this entails checking the outer edges of the following diagram commute for $k = 1, \dots, n$:

$$\begin{tikzcd}
\pi_k\{p\} \arrow[rr, "\iota^{-1}_{\pi_k(p)}"] \arrow[d, "\pi_k\{\varsigma_p\}"'] \arrow[dr, equal, bend left] \arrow[dr, phantom, "\overset{\text{triang. law}}{=}" near start] & & \pi_k(p) \arrow[dd, "\varrho^{(-k)}_{\pi_k(p)}"] \\
\pi_k\{\mathsf{tup}(\pi_1\{p\}, \dots, \pi_n\{p\})\} \arrow[r, "\varpi^{(k)}_{\pi_\bullet\{p\}}"'] \arrow[d, "\pi_k\big\{\mathsf{tup}(\iota^{-1}_{\pi_1(p)}, \dots, \iota^{-1}_{\pi_n(p)})\big\}"'] \arrow[dr, phantom, "\overset{\text{nat.}}{=}"] & \pi_k\{p\} \arrow[d, "\iota^{-1}_{\pi_k(p)}" description] \arrow[r, phantom, "="] & {} \\
\pi_k\{\mathsf{tup}(\pi_1(p), \dots, \pi_n(p))\} \arrow[r, "\varpi^{(k)}_{\pi_\bullet(p)}"'] & \pi_k(p) \arrow[r, "\varrho^{(-k)}_{\pi_k(p)}"'] & x_k\{x_i \mapsto \pi_i(p)\}
\end{tikzcd}$$

Hence, the unit law does indeed hold. The associativity law holds by construction for composites of terms in unary contexts. For the general case, one instantiates the definition of $\phi^{\iota\llbracket - \rrbracket}$ from Proposition 5.3.22 and applies the definition of post to get exactly the required composite. This completes the proof of the next lemma.

Lemma 5.3.27. For any unary $\Lambda^{\times,\to}_{\mathrm{ps}}$-signature $\mathcal{S}$, the composite $\iota^{\#} \circ \iota\llbracket - \rrbracket : \mathcal{T}^{@,\times,\to}_{\mathrm{ps}}(\mathcal{S}) \to \mathcal{T}^{@,\times,\to}_{\mathrm{ps}}(\mathcal{S})$ induced by the following diagram is equivalent to $\mathrm{id}_{\mathcal{T}^{@,\times,\to}_{\mathrm{ps}}(\mathcal{S})}$:

$$\begin{tikzcd}
\mathcal{T}^{@,\times,\to}_{\mathrm{ps}}(\mathcal{S}) \arrow[r, "\iota\llbracket - \rrbracket"] & \mathcal{FB}ct^{\times,\to}(\mathcal{S}) \arrow[r, "\iota^{\#}"] & \mathcal{T}^{@,\times,\to}_{\mathrm{ps}}(\mathcal{S}) \\
\mathcal{S} \arrow[u, hook] \arrow[r, equal] & \mathcal{S} \arrow[u, hook] \arrow[r, equal] & \mathcal{S} \arrow[u, hook]
\end{tikzcd} \tag{5.23}$$

□

Putting this lemma together with Proposition 5.3.25, one obtains the biequivalence between $\mathcal{T}^{@,\times,\to}_{\mathrm{ps}}(\mathcal{S})$ and $\mathcal{FB}ct^{\times,\to}(\mathcal{S})$:

Proposition 5.3.28. For any unary $\Lambda^{\times,\to}_{\mathrm{ps}}$-signature $\mathcal{S}$, the cc-pseudofunctors $\iota\llbracket - \rrbracket$ and $\iota^{\#}$ extending the inclusion as in the diagram

$$\begin{tikzcd}
\mathcal{FB}ct^{\times,\to}(\mathcal{S}) \arrow[r, "\iota^{\#}"] & \mathcal{T}^{@,\times,\to}_{\mathrm{ps}}(\mathcal{S}) \arrow[r, "\iota\llbracket - \rrbracket"] & \mathcal{FB}ct^{\times,\to}(\mathcal{S}) \\
\mathcal{S} \arrow[u, hook, "\iota"] \arrow[r, equal] & \mathcal{S} \arrow[u, hook, "\iota"] \arrow[r, equal] & \mathcal{S} \arrow[u, hook, "\iota"]
\end{tikzcd}$$

form a biequivalence $\mathcal{FB}ct^{\times,\to}(\mathcal{S}) \simeq \mathcal{T}^{@,\times,\to}_{\mathrm{ps}}(\mathcal{S})$. □

It is not hard to see that the pseudonatural transformation $(\mathsf{j}, \bar{\mathsf{j}})$ defined in Construction 5.3.26 restricts to a pseudonatural transformation $\iota\llbracket - \rrbracket \circ \iota^{\#} \simeq \mathrm{id}_{\overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})}}$ for $\iota\llbracket - \rrbracket$ the restriction of the interpretation pseudofunctor of Proposition 5.3.22 to $\overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})}$. Since the proof of Proposition 5.3.25 also restricts to the unary case, one obtains the following.

Corollary 5.3.29. For any unary $\Lambda^{\times,\to}_{\mathrm{ps}}$-signature $\mathcal{S}$, the cc-pseudofunctors $\iota\llbracket - \rrbracket$ and $\iota^{\#}$ extending the inclusion as in the diagram

$$\begin{tikzcd}
\mathcal{FB}ct^{\times,\to}(\mathcal{S}) \arrow[r, "\iota^{\#}"] & \overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})} \arrow[r, "\iota\llbracket - \rrbracket"] & \mathcal{FB}ct^{\times,\to}(\mathcal{S}) \\
\mathcal{S} \arrow[u, hook, "\iota"] \arrow[r, equal] & \mathcal{S} \arrow[u, hook, "\iota"] \arrow[r, equal] & \mathcal{S} \arrow[u, hook, "\iota"]
\end{tikzcd}$$

form a biequivalence $\mathcal{FB}ct^{\times,\to}(\mathcal{S}) \simeq \overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})}$. □

Hence, up to canonical biequivalence, the syntactic model of $\Lambda_{\mathrm{ps}}^{\times,\to}(\mathcal{S})$ is the free cc-bicategory on the $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature $\mathcal{S}$. We are therefore justified in calling $\Lambda_{\mathrm{ps}}^{\times,\to}$ the internal language of cartesian closed bicategories.

It further follows that the canonical pseudofunctor is unique up to equivalence.

Corollary 5.3.30. For any cc-bicategory $(\mathcal{B}, \Pi_n(-), {\Rightarrow})$, unary $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature $\mathcal{S}$ and $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature homomorphism $h : \mathcal{S} \to \mathcal{B}$, there exists a strict cc-pseudofunctor $h[\![-]\!] : \overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})} \to \mathcal{B}$. Up to equivalence, this is the unique strict cc-pseudofunctor $F : \overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})} \to \mathcal{B}$ such that $F \circ \iota = h$, for ι the inclusion.

Proof. Existence is Corollary 5.3.19 so it suffices to show uniqueness. To this end, consider the diagram

$$\begin{tikzcd}
\mathcal{FB}ct^{\times,\to}(\mathcal{S}) \arrow[r, "\iota^{\#}"] & \overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})} \arrow[r, "F"] & \mathcal{B} \\
& \mathcal{S} \arrow[ul, hook, "\iota"] \arrow[u, hook, "\iota"] \arrow[ur, "h"'] &
\end{tikzcd}$$

where F is any strict cc-pseudofunctor. By the free property of $\mathcal{FB}ct^{\times,\to}(\mathcal{S})$ (Lemma 5.2.19), $h^{\#} = F \circ \iota^{\#}$. Then, applying Corollary 5.3.29, one sees that

$$F \simeq F \circ (\iota^{\#} \circ \iota[\![-]\!]) \simeq (F \circ \iota^{\#}) \circ \iota[\![-]\!] = h^{\#} \circ \iota[\![-]\!]$$

It follows that any strict cc-pseudofunctor extending h is equivalent to $h^{\#} \circ \iota[\![-]\!]$. Hence, $h[\![-]\!]$ is unique up to equivalence. □

We finish this section with a corollary relating the semantic interpretation of Proposition 5.3.17 to the free property of the free cc-bicategory (Lemma 5.2.19).

Corollary 5.3.31. For any cc-bicategory $(\mathcal{X}, \Pi_n(-), {\Rightarrow})$, set of base types $\mathfrak{B}$, and $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature homomorphism $h : \mathcal{S} \to \mathcal{X}$, there exists an equivalence $h^{\#} \circ \iota[\![-]\!] \simeq h[\![-]\!] : \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}}) \to \mathcal{X}$.

Proof. Observe that the composite $\widetilde{\mathfrak{B}} \hookrightarrow \mathcal{FB}ct^{\times,\to}(\widetilde{\mathfrak{B}}) \xrightarrow{\iota^{\#}} \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}}) \xrightarrow{h[\![-]\!]} \mathcal{X}$ is equal to simply h. Thus, applying Lemma 5.2.20, there exists an equivalence $h^{\#} \simeq h[\![-]\!] \circ \iota^{\#}$. But by Proposition 5.3.28 there also exists an equivalence $\iota^{\#} \circ \iota[\![-]\!] \simeq \mathrm{id}_{\mathcal{FB}ct^{\times,\to}(\widetilde{\mathfrak{B}})}$. Hence,

$$h^{\#} \circ \iota[\![-]\!] \simeq (h[\![-]\!] \circ \iota^{\#}) \circ \iota[\![-]\!] \simeq h[\![-]\!]$$

as claimed. □

5.4 Normal forms in $\Lambda^{\times,\to}_{\mathrm{ps}}$

In this final section we shall make precise the sense in which $\Lambda^{\times,\to}_{\mathrm{ps}}$ is the simply-typed lambda calculus 'up to isomorphism', which will enable us to port the notion of (long-$\beta\eta$) normal form from the simply-typed lambda calculus into $\Lambda^{\times,\to}_{\mathrm{ps}}$. Our approach is to extend the mappings defined in Section 3.3 for $\Lambda^{\mathrm{bicl}}_{\mathrm{ps}}$ to include cartesian closed structure. One could go further, and prove that the syntactic model of $\Lambda^{\times,\to}_{\mathrm{ps}}$ is biequivalent to the syntactic model of the strict language H^{cl} extended with pseudo cartesian closed structure. Such a result provides a constructive proof that the free cartesian closed bicategory on a $\Lambda^{\times,\to}_{\mathrm{ps}}$-signature $\mathcal{S}$ is biequivalent to the free 2-category with bicategorical products and exponentials on $\mathcal{S}$. Since this follows from the Mac Lane-Paré coherence theorem [MP85], together with fact that biequivalences preserve bilimits and biadjunctions, we restrict ourselves to mappings on terms. However, we shall present certain results one requires in order to construct this biequivalence, as they turn out to be of importance in the proof of our main theorem in Chapter 8.

To fix notation, let $\Lambda^{\times,\to}(\mathcal{S})$ denote the simply-typed lambda calculus with constants and base types specified by a $\Lambda^{\times,\to}$-signature $\mathcal{S} = (\mathfrak{B}, \mathcal{G})$. This is defined in Figure 5.6 below. As for $\Lambda^{\times,\to}_{\mathrm{ps}}$, we present products in an n-ary style which is equivalent to the usual presentation in terms of binary products and a terminal object. The equational theory is the usual $\alpha\beta\eta$-equality for the simply-typed lambda calculus (*e.g.* [Bar85, Cro94]).

$$\frac{}{x_1 : A_1, \dots, x_n : A_n \vdash x_k : A_k}\ \text{var}\ (1 \leqslant k \leqslant n)$$

$$\frac{c \in \mathcal{G}(A_1, \dots, A_n; B) \qquad (\Delta \vdash u_i : A_i)_{i=1,\dots,n}}{\Delta \vdash c(u_1, \dots, u_n) : B}\ \text{const}$$

$$\frac{\Gamma \vdash t_1 : A_1 \quad \dots \quad \Gamma \vdash t_n : A_n}{\Gamma \vdash \langle t_1, \dots, t_n \rangle : \prod_n (A_1, \dots, A_n)}\ n\text{-tuple} \qquad \frac{\Gamma \vdash t : \prod_n (A_1, \dots, A_n)}{\Gamma \vdash \pi_k(t) : A_k}\ k\text{-proj}\ (1 \leqslant k \leqslant n)$$

$$\frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x.t : A \Rightarrow B}\ \text{lam} \qquad \frac{\Gamma \vdash t : A \Rightarrow B \qquad \Gamma \vdash u : A}{\Gamma \vdash \mathsf{app}(t, u) : B}\ \text{app}$$

Figure 5.6: Rules for $\Lambda^{\times,\to}(\mathcal{S})$.

We shall not distinguish notationally between the type theory $\Lambda^{\times,\to}$ (resp. $\Lambda^{\times,\to}_{\mathrm{ps}}$) and its set of terms (or set of terms and rewrites) up to α-equivalence. We employ the following notation:

$$\Lambda^{\times,\to}(\mathcal{S})(\Gamma; B) := \{t \mid \Gamma \vdash_{\mathrm{STLC}} t : B\} \,/ =_\alpha$$
$$\Lambda^{\times,\to}_{\mathrm{ps}}(\mathcal{S})(\Gamma; B) := \{t \mid \Gamma \vdash_{\Lambda^{\times,\to}_{\mathrm{ps}}} t : B\} \,/ =_\alpha$$

Similarly, we write $\Lambda^{\times,\to}(\mathcal{S})$ to denote the set of all $\Lambda^{\times,\to}$-terms modulo α-equivalence, and $\Lambda^{\times,\to}_{\mathrm{ps}}(\mathcal{S})$ to denote the set of all $\Lambda^{\times,\to}_{\mathrm{ps}}$-terms modulo α-equivalence. (Precisely, these are sets indexed by (context, type) pairs.) We drop the decorations on the turnstile symbol unless the type theory in question is ambiguous.

Relating $\Lambda^{\times,\to}_{\mathrm{ps}}$ and $\Lambda^{\times,\to}$. We define a pair of maps $\llparenthesis - \rrparenthesis : \Lambda^{\times,\to}(\mathcal{S}) \leftrightarrows \Lambda^{\times,\to}_{\mathrm{ps}}(\mathcal{S}) : \overline{(-)}$ for a fixed $\Lambda^{\times,\to}$-signature $\mathcal{S}$. These maps extend those constructed in Section 3.3 for biclones; indeed, the terms of $\mathrm{H}^{\mathrm{cl}}(\mathcal{S})$ are exactly the variables and constants in $\Lambda^{\times,\to}(\mathcal{S})$.

Construction 5.4.1. For any $\Lambda^{\times,\to}$-signature $\mathcal{S}$, define a mapping $\overline{(-)} : \Lambda^{\times,\to}_{\mathrm{ps}}(\mathcal{S}) \to \Lambda^{\times,\to}(\mathcal{S})$ as follows:

$$\begin{array}{rlrl}
\overline{x_i} & := x_i & \overline{c(x_1, \dots, x_n)} & := c(x_1, \dots, x_n) \\
\overline{\pi_k(p)} & := \pi_k(p) & \overline{\mathsf{tup}(t_1, \dots, t_n)} & := \langle \overline{t_1}, \dots, \overline{t_n} \rangle \\
\overline{\mathsf{eval}(f, a)} & := \mathsf{app}(f, a) & \overline{\lambda x.t} & := \lambda x.\overline{t}
\end{array}$$

◂

It is elementary to check this definition respects α-equivalence and the equational theory $\equiv$.

Lemma 5.4.2. For any $\Lambda^{\times,\to}$-signature $\mathcal{S}$,

1. For all derivable terms t, t' in $\Lambda^{\times,\to}_{\mathrm{ps}}(\mathcal{S})$, if $t =_\alpha t'$ then $\overline{t} =_\alpha \overline{t'}$,
2. If $\Gamma \vdash t : B$ in $\Lambda^{\times,\to}_{\mathrm{ps}}(\mathcal{S})$ then $\Gamma \vdash \overline{t} : B$ in $\Lambda^{\times,\to}(\mathcal{S})$, *i.e.* one obtains maps of indexed sets. □

As we did for biclones, we think of $\overline{t}$ as the *strictification* of a term in $\Lambda^{\times,\to}_{\mathrm{ps}}$. The map $\llparenthesis - \rrparenthesis$ interprets $\Lambda^{\times,\to}$-terms in $\Lambda^{\times,\to}_{\mathrm{ps}}$.

Construction 5.4.3. For any $\Lambda^{\times,\to}$-signature $\mathcal{S}$, define a mapping $\llparenthesis - \rrparenthesis : \Lambda^{\times,\to}(\mathcal{S}) \to \Lambda^{\times,\to}_{\mathrm{ps}}(\mathcal{S})$ as follows:

$$\begin{array}{rlrl}
\llparenthesis x_k \rrparenthesis & := x_k & \llparenthesis c(u_1, \dots, u_n) \rrparenthesis & := c\{\llparenthesis u_1 \rrparenthesis, \dots, \llparenthesis u_n \rrparenthesis\} \\
\llparenthesis \pi_k(t) \rrparenthesis & := \pi_k\{\llparenthesis t \rrparenthesis\} & \llparenthesis \mathsf{app}(t, u) \rrparenthesis & := \mathsf{eval}\{\llparenthesis t \rrparenthesis, \llparenthesis u \rrparenthesis\} \\
\llparenthesis \langle t_1, \dots, t_n \rangle \rrparenthesis & := \mathsf{tup}(\llparenthesis t_1 \rrparenthesis, \dots, \llparenthesis t_n \rrparenthesis) & \llparenthesis \lambda x.t \rrparenthesis & := \lambda x.\llparenthesis t \rrparenthesis
\end{array}$$

◂

This mapping also respects typing and α-equivalence.

Lemma 5.4.4. For any $\Lambda^{\times,\rightarrow}$-signature $\mathcal{S}$,

1. For all derivable terms t, t' in $\Lambda^{\times,\rightarrow}(\mathcal{S})$, if $t =_\alpha t'$ then $\llparenthesis t \rrparenthesis =_\alpha \llparenthesis t' \rrparenthesis$,
2. If $\Gamma \vdash t : B$ in $\Lambda^{\times,\rightarrow}(\mathcal{S})$ then $\Gamma \vdash \llparenthesis t \rrparenthesis : B$ in $\Lambda^{\times,\rightarrow}_{\mathrm{ps}}(\mathcal{S})$, *i.e.* one obtains maps of indexed sets. □

As in Section 3.3, strictifying a $\Lambda^{\times,\rightarrow}$-term does nothing.

Lemma 5.4.5. The composite mapping $\overline{(-)} \circ \llparenthesis - \rrparenthesis$ is exactly the identity on $\Lambda^{\times,\rightarrow}(\mathcal{S})$.

Proof. The claim holds by induction, using the usual laws of capture-avoiding substitution for the simply-typed lambda calculus:

$$x_k \mapsto x_k \mapsto x_k$$

$$c(u_1, \ldots, u_n) \mapsto c\{\llparenthesis u_1 \rrparenthesis, \ldots, \llparenthesis u_n \rrparenthesis\} \mapsto c(x_1, \ldots, x_n)[\overline{\llparenthesis u_i \rrparenthesis}/x_i]$$

$$\pi_k(t) \mapsto \pi_k\{\llparenthesis t \rrparenthesis\} \mapsto \pi_k(p)[\overline{\llparenthesis t \rrparenthesis}/p]$$

$$\langle t_1, \ldots, t_n \rangle \mapsto \mathsf{tup}(\llparenthesis t_1 \rrparenthesis, \ldots, \llparenthesis t_n \rrparenthesis) \mapsto \langle \overline{\llparenthesis t_1 \rrparenthesis}, \ldots, \overline{\llparenthesis t_n \rrparenthesis} \rangle$$

$$\mathsf{app}(t, u) \mapsto \mathsf{eval}\{\llparenthesis t \rrparenthesis, \llparenthesis u \rrparenthesis\} \mapsto (\mathsf{app}(f, a))[\overline{\llparenthesis t \rrparenthesis}/f, \overline{\llparenthesis u \rrparenthesis}/a]$$

$$\lambda x.t \mapsto \lambda x.\llparenthesis t \rrparenthesis \mapsto \lambda x.\overline{\llparenthesis t \rrparenthesis}$$

□

We shall require a rewrite reducing explicit substitutions to the meta-operation of capture-avoiding substitution. As in the biclone case, this is the extra data required to make $\llparenthesis - \rrparenthesis$ into a pseudofunctor. Unlike the biclone case, however, we must now deal with variable binding. This entails an extra step in our construction. To inductively prove a lemma about substitution in the simply-typed lambda calculus, it is common to first prove a lemma about weakening. This auxiliary result allows one to deal with the fresh variable appearing in the lambda abstraction step. We shall do something similar. First, we shall define a rewrite reducing context renamings (in particular, weakenings) to actual syntactic substitutions. Then, we shall use this to construct our rewrite handling arbitrary substitutions.

We call the auxiliary rewrite cont for *context renaming.*

Construction 5.4.6. For any $\Lambda^{\times,\rightarrow}$-signature $\mathcal{S}$ and context renaming r, we construct a rewrite $\mathsf{cont}(t; r)$ making the following rule admissible:

$$\frac{\Gamma \vdash \llparenthesis t \rrparenthesis : B \qquad r : \Gamma \to \Delta}{\Delta \vdash \mathsf{cont}(t; r) : \llparenthesis t \rrparenthesis\{x_i \mapsto r(x_i)\} \Rightarrow \llparenthesis t[r(x_i)/x_i] \rrparenthesis : B}$$

The definition is by induction on the derivation of t:

$$\mathsf{cont}(x_k; r) := x_k\{x_i \mapsto r(x_i)\} \xRightarrow{\varrho^{(r(x_i))}} \llparenthesis\, r(x_i) \,\rrparenthesis$$

$$\mathsf{cont}(c(u_\bullet); r) := c\{\llparenthesis\, u_1 \,\rrparenthesis, \dots, \llparenthesis\, u_n \,\rrparenthesis\}\{r\} \xRightarrow{\mathsf{assoc}} c\{\llparenthesis\, u_\bullet \,\rrparenthesis\{r\}\} \xRightarrow{c\{\mathsf{cont}, \dots, \mathsf{cont}\}} c\{\llparenthesis\, u_\bullet[r(x_i)/x_i] \,\rrparenthesis\}$$

$$\mathsf{cont}(\pi_k(t); r) := \pi_k\{\llparenthesis\, t \,\rrparenthesis\}\{r\} \xRightarrow{\mathsf{assoc}} \pi_k\{\llparenthesis\, t \,\rrparenthesis\{r\}\} \xRightarrow{\pi_k\{\mathsf{cont}\}} \pi_k\{\llparenthesis\, t[r(x_i)/x_i] \,\rrparenthesis\}$$

$$\mathsf{cont}(\langle t_1, \dots, t_n \rangle; u_\bullet) := \mathsf{tup}(\llparenthesis\, t_1 \,\rrparenthesis, \dots, \llparenthesis\, t_n \,\rrparenthesis)\{\llparenthesis\, u_\bullet \,\rrparenthesis\} \xRightarrow{\mathsf{post}} \mathsf{tup}(\llparenthesis\, t_\bullet \,\rrparenthesis\{\llparenthesis\, u_\bullet \,\rrparenthesis\}) \xRightarrow{\mathsf{tup}(\mathsf{cont}, \dots, \mathsf{cont})} \mathsf{tup}(\llparenthesis\, t_\bullet[u_i/x_i] \,\rrparenthesis)$$

$$\begin{aligned}\mathsf{cont}(\mathsf{app}(t, u); r) := \mathsf{eval}\{\llparenthesis\, t \,\rrparenthesis, \llparenthesis\, u \,\rrparenthesis\}\{r\} &\xRightarrow{\mathsf{assoc}} \mathsf{eval}\big\{\llparenthesis\, t \,\rrparenthesis\{r\}, \llparenthesis\, u \,\rrparenthesis\{r\}\big\} \\ &\xRightarrow{\mathsf{eval}\{\mathsf{cont},\mathsf{cont}\}} \mathsf{eval}\big\{\llparenthesis\, t[r(x_i)/x_i] \,\rrparenthesis, \llparenthesis\, u[r(x_i)/x_i] \,\rrparenthesis\big\}\end{aligned}$$

$$\begin{aligned}\mathsf{cont}(\lambda x.t; r) := (\lambda x.\llparenthesis\, t \,\rrparenthesis)\{r\} &\xRightarrow{\mathsf{push}} \lambda x.\llparenthesis\, t \,\rrparenthesis\{x \mapsto x, x_i \mapsto r(x_i)\{\mathrm{inc}_x\}\} \\ &\xRightarrow{\lambda x.\llparenthesis\, t \,\rrparenthesis\{x,\mathsf{cont}(r(x_i);\mathrm{inc}_x)\}} \lambda x.\llparenthesis\, t \,\rrparenthesis\{x \mapsto x, x_i \mapsto r(x_i)\} \\ &\xRightarrow{\lambda x.\mathsf{cont}} \lambda x.\llparenthesis\, t[x/x, r(x_i)/x_i] \,\rrparenthesis\end{aligned}$$

◀

We can now define sub. The construction extends its biclone counterpart, Construction 3.3.14.

Construction 5.4.7. For any $\Lambda^{\times,\rightarrow}$-signature $\mathcal{S}$, we construct a rewrite $\mathsf{sub}(t; u_\bullet)$ so that the following rule is admissible:

$$\frac{x_1 : A_1, \dots, x_n : A_n \vdash \llparenthesis\, t \,\rrparenthesis : B \qquad (\Delta \vdash \llparenthesis\, u_i \,\rrparenthesis : A_i)_{i=1,\dots,n}}{\Delta \vdash \mathsf{sub}(t; u_\bullet) : \llparenthesis\, t \,\rrparenthesis\{x_i \mapsto \llparenthesis\, u_i \,\rrparenthesis\} \Rightarrow \llparenthesis\, t[u_i/x_i] \,\rrparenthesis : B}$$

The definition is by induction on the derivation of t:

$$\mathsf{sub}(x_k; u_\bullet) := x_k\{x_i \mapsto \llparenthesis\, u_i \,\rrparenthesis\} \xRightarrow{\varrho^{(k)}} \llparenthesis\, u_k \,\rrparenthesis$$

$$\mathsf{sub}(c(u_\bullet); v_\bullet) := c\{\llparenthesis\, u_1 \,\rrparenthesis, \dots, \llparenthesis\, u_n \,\rrparenthesis\}\{\llparenthesis\, v_\bullet \,\rrparenthesis\} \xRightarrow{\mathsf{assoc}} c\{\llparenthesis\, u_\bullet \,\rrparenthesis\{\llparenthesis\, v_\bullet \,\rrparenthesis\}\} \xRightarrow{c\{\mathsf{sub}, \dots, \mathsf{sub}\}} c\{\llparenthesis\, u_\bullet[v_j/y_j] \,\rrparenthesis\}$$

$$\mathsf{sub}(\pi_k(t); u_\bullet) := \pi_k\{\llparenthesis\, t \,\rrparenthesis\}\{\llparenthesis\, u_\bullet \,\rrparenthesis\} \xRightarrow{\mathsf{assoc}} \pi_k\{\llparenthesis\, t \,\rrparenthesis\{\llparenthesis\, u_\bullet \,\rrparenthesis\}\} \xRightarrow{\pi_k\{\mathsf{sub}\}} \pi_k\{\llparenthesis\, t[u_i/x_i] \,\rrparenthesis\}$$

$$\mathsf{sub}(\langle t_1, \dots, t_n \rangle; u_\bullet) := \mathsf{tup}(\llparenthesis\, t_1 \,\rrparenthesis, \dots, \llparenthesis\, t_n \,\rrparenthesis)\{\llparenthesis\, u_\bullet \,\rrparenthesis\} \xRightarrow{\mathsf{post}} \mathsf{tup}(\llparenthesis\, t_\bullet \,\rrparenthesis\{\llparenthesis\, u_\bullet \,\rrparenthesis\}) \xRightarrow{\mathsf{tup}(\mathsf{sub}, \dots, \mathsf{sub})} \mathsf{tup}(\llparenthesis\, t_\bullet[u_i/x_i] \,\rrparenthesis)$$

$$\begin{aligned}\mathsf{sub}(\mathsf{app}(t, u); v_\bullet) := \mathsf{eval}\{\llparenthesis\, t \,\rrparenthesis, \llparenthesis\, u \,\rrparenthesis\}\{\llparenthesis\, v_\bullet \,\rrparenthesis\} &\xRightarrow{\mathsf{assoc}} \mathsf{eval}\big\{\llparenthesis\, t \,\rrparenthesis\{\llparenthesis\, v_\bullet \,\rrparenthesis\}, \llparenthesis\, u \,\rrparenthesis\{\llparenthesis\, v_\bullet \,\rrparenthesis\}\big\} \\ &\xRightarrow{\mathsf{eval}\{\mathsf{sub},\mathsf{sub}\}} \mathsf{eval}\big\{\llparenthesis\, t[v_j/y_j] \,\rrparenthesis \llparenthesis\, u[v_j/y_j] \,\rrparenthesis\big\}\end{aligned}$$

$$\begin{aligned}\mathsf{sub}(\lambda x.t; u_\bullet) := (\lambda x.\llparenthesis\, t \,\rrparenthesis)\{\llparenthesis\, v_\bullet \,\rrparenthesis\} &\xRightarrow{\mathsf{push}} \lambda x.\llparenthesis\, t \,\rrparenthesis\big\{x, \llparenthesis\, u \,\rrparenthesis\{\mathrm{inc}_x\}\big\} \\ &\xRightarrow{\lambda x.\llparenthesis\, t \,\rrparenthesis\{x,\mathsf{cont}(u;\mathrm{inc}_x)\}} \lambda x.\llparenthesis\, t \,\rrparenthesis\{x, \llparenthesis\, u \,\rrparenthesis\} \\ &\xRightarrow{\lambda x.\mathsf{sub}} \lambda x.\llparenthesis\, t[x/x, u_i/x_i] \,\rrparenthesis\end{aligned}$$

◀

Note the use of cont in the lambda abstraction step. As one would expect, sub and cont coincide where the terms being substituted are all variables.

Lemma 5.4.8. For any $\Lambda^{\times,\rightarrow}$-signature $\mathcal{S}$, judgement $(\Gamma \vdash \llparenthesis\, t \,\rrparenthesis : B)$ in $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}(\mathcal{S})$, and context renaming $r : \Gamma \to \Delta$, then

$$\Delta \vdash \mathsf{sub}(t; r(x_\bullet)) \equiv \mathsf{cont}(t; r) : \llparenthesis\, t \,\rrparenthesis\{x_i \mapsto r(x_i)\} \Rightarrow \llparenthesis\, t \,\rrparenthesis : B$$

Proof. By induction on the derivation of t: comparing the cases one-by-one, the equality is immediate. □

Let us note some of other the ways in which cont and sub behave as expected (*c.f.* Lemma 3.3.17). We shall not need these results immediately, but they will play an important role in the normalisation-by-evaluation proof of Chapter 8.

Lemma 5.4.9. For any $\Lambda^{\times,\rightarrow}$-signature $\mathcal{S}$ and any contexts $\Gamma := (x_i : A_i)_{i=1,\dots,n}$ and $\Delta := (y_j : B_j)_{j=1,\dots,m}$,

1. If $\Gamma \vdash \llparenthesis\, t \,\rrparenthesis : B$ then

$$\begin{tikzcd}
\llparenthesis\, t \,\rrparenthesis \arrow[d, Rightarrow, "\iota_{\llparenthesis t \rrparenthesis}"'] \arrow[dr, equals, bend left] & \\
\llparenthesis\, t \,\rrparenthesis\{x_i \mapsto x_i\} \arrow[r, Rightarrow, "\mathsf{cont}(t;\mathrm{id}_\Gamma)"'] & \llparenthesis\, t[x_i/x_i] \,\rrparenthesis
\end{tikzcd} \tag{5.24}$$

2. If $\Gamma \vdash \llparenthesis\, t \,\rrparenthesis : B$ and $(\Delta \vdash \llparenthesis\, u_i \,\rrparenthesis : A_i)_{i=1,\dots,n}$ then

$$\begin{tikzcd}
\llparenthesis\, t \,\rrparenthesis\{x_i \mapsto \llparenthesis\, u_i \,\rrparenthesis\}\{\mathrm{id}_\Delta\} \arrow[r, Rightarrow, "\mathsf{assoc}"] \arrow[d, Rightarrow, "\mathsf{sub}(t;u_\bullet)\{\mathrm{id}_\Delta\}"'] & \llparenthesis\, t \,\rrparenthesis\big\{x_i \mapsto \llparenthesis\, u_i \,\rrparenthesis\{\mathrm{id}_\Delta\}\big\} \arrow[r, Rightarrow, "\llparenthesis t \rrparenthesis\{\mathsf{sub}(u_i;\mathrm{id}_\Delta)\}"] & \llparenthesis\, t \,\rrparenthesis\{x_i \mapsto \llparenthesis\, u_i \,\rrparenthesis\} \arrow[d, Rightarrow, "\mathsf{sub}(t;u_\bullet)"] \\
\llparenthesis\, t[u_i/x_i] \,\rrparenthesis\{\mathrm{id}_\Delta\} \arrow[rr, Rightarrow, "\mathsf{sub}(t[u_i/x_i];\mathrm{id}_\Delta)"'] & & \llparenthesis\, t[u_i/x_i] \,\rrparenthesis
\end{tikzcd} \tag{5.25}$$

3. If $(\Gamma \vdash \llparenthesis\, t \,\rrparenthesis : B)$, $(\Delta \vdash \llparenthesis\, u_i \,\rrparenthesis : A_i)_{i=1,\dots,n}$ and $(\Sigma \vdash \llparenthesis\, v_j \,\rrparenthesis : B_j)_{j=1,\dots,m}$, then

$$\begin{tikzcd}
\llparenthesis\, t \,\rrparenthesis\{\llparenthesis\, u_\bullet \,\rrparenthesis\}\{\llparenthesis\, v_\bullet \,\rrparenthesis\} \arrow[r, Rightarrow, "\mathsf{assoc}"] \arrow[d, Rightarrow, "\mathsf{sub}(t;u_\bullet)\{v_\bullet\}"'] & \llparenthesis\, t \,\rrparenthesis\big\{\llparenthesis\, u_\bullet \,\rrparenthesis\{\llparenthesis\, v_\bullet \,\rrparenthesis\}\big\} \arrow[r, Rightarrow, "\llparenthesis t \rrparenthesis\{\mathsf{sub}(u_i;v_\bullet)\}"] & \llparenthesis\, t \,\rrparenthesis\{\llparenthesis\, u_\bullet[v_j/y_j] \,\rrparenthesis\} \arrow[d, Rightarrow, "\mathsf{sub}(t;u_\bullet)"] \\
\llparenthesis\, t[u_i/x_i] \,\rrparenthesis\{\llparenthesis\, v_\bullet \,\rrparenthesis\} \arrow[rr, Rightarrow, "\mathsf{sub}(t[u_i/x_i];v_\bullet)"'] & & \llparenthesis\, t\,[u_i[v_j/y_j]/x_i] \,\rrparenthesis
\end{tikzcd} \tag{5.26}$$

Proof. Each of the claims is proven by induction. Most of the cases for (1) are almost immediate, except for lambda abstraction. There one uses Lemma 5.3.15(2).

For (2) and (3), all the cases except for lambda abstraction are relatively simple. One can prove (3) and derive (2) as a special case. For lambda abstraction, *i.e.* for judgements of the form $(\Gamma \vdash t : A \Rightarrow B)$, one must deal with fresh variables. For this we take the claims in order.

To prove the `lam` case of (2) one first proves three further lemmas building towards the target result. The first is that whenever $(\Delta \vdash \llparenthesis u_i \rrparenthesis : A_i)$, then

$$\begin{tikzcd}
\llparenthesis u_i \rrparenthesis\{\mathrm{id}_\Delta\}\{\mathrm{id}_\Delta\} \arrow[r, Rightarrow, "\mathsf{assoc}"] \arrow[d, Rightarrow, "\mathsf{sub}(t;\mathrm{id}_\Delta)\{\mathrm{id}_\Delta\}"'] & \llparenthesis u_i \rrparenthesis\{y_j\{\mathrm{id}_\Delta\}\} \arrow[r, Rightarrow, "\llparenthesis u_i \rrparenthesis\{\varrho^{(\bullet)}_{y_\bullet}\}"] & \llparenthesis u_i \rrparenthesis\{\mathrm{id}_\Delta\} \arrow[d, Rightarrow, "\mathsf{sub}(u_i;y_\bullet)"] \\
\llparenthesis u_i \rrparenthesis\{\mathrm{id}_\Delta\} \arrow[rr, Rightarrow, "\mathsf{sub}(t;\mathrm{id}_\Delta)"'] & & \llparenthesis u_i \rrparenthesis
\end{tikzcd} \tag{5.27}$$

To show this diagram commutes, one inducts on the derivation of $\llparenthesis t \rrparenthesis$; all the cases but `lam` follow as for (3). For the `lam` case one uses the inductive hypothesis, the coherence of $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$, and Lemma 5.3.15(3).

Next we show that, whenever $(\Gamma \vdash \llparenthesis t \rrparenthesis : B)$ and $(\Delta \vdash \llparenthesis u_i \rrparenthesis : A_i)_{i=1,\dots,n}$, then

$$\begin{tikzcd}
\llparenthesis t \rrparenthesis\{\llparenthesis u_\bullet \rrparenthesis\}\{\mathrm{id}_\Delta\} \arrow[r, Rightarrow, "\mathsf{assoc}"] \arrow[d, Rightarrow, "\mathsf{sub}(t;u_\bullet)\{\mathrm{id}_\Delta\}"'] & \llparenthesis t \rrparenthesis\big\{x_i \mapsto \llparenthesis u_i \rrparenthesis\{\mathrm{id}_\Delta\}\big\} \arrow[r, Rightarrow, "\llparenthesis t \rrparenthesis\{\mathsf{sub}(u_i;\mathrm{id}_\Delta)\}"] & \llparenthesis t \rrparenthesis\{\llparenthesis u_\bullet \rrparenthesis\} \arrow[d, Rightarrow, "\mathsf{sub}(t;u_\bullet)"] \\
\llparenthesis t[u_i/x_i] \rrparenthesis\{\mathrm{id}_\Delta\} \arrow[rr, Rightarrow, "\mathsf{sub}(t[u_i/x_i];\mathrm{id}_\Delta)"'] & & \llparenthesis t[u_i/x_i] \rrparenthesis
\end{tikzcd} \tag{5.28}$$

Once again all the cases but `lam` follow from the generality of (3). For the lambda abstraction case the proof is similar to that for (5.27): one applies the inductive hypothesis, Lemma 5.3.15(3) and (5.27).

The final lemma required is the following. For any judgements $(\Gamma \vdash \llparenthesis t \rrparenthesis : B)$, $(\Delta \vdash \llparenthesis u_i \rrparenthesis : A_i)_{i=1,\dots,n}$ and $(\Sigma, x : A \vdash \llparenthesis v_j \rrparenthesis : B_j)_{j=1,\dots,m}$, one shows that

$$\begin{tikzcd}
\llparenthesis t \rrparenthesis\{\llparenthesis u_i \rrparenthesis\}\{\llparenthesis \mathrm{id}_\Delta \rrparenthesis\} \arrow[r, Rightarrow, "\mathsf{assoc}"] \arrow[d, Rightarrow, "\mathsf{sub}(t;u_\bullet)\{\mathrm{id}_\Delta\}"'] & \llparenthesis t \rrparenthesis\{x_i \mapsto \llparenthesis u_i \rrparenthesis\{\mathrm{id}_\Delta\}\} \arrow[r, Rightarrow, "\llparenthesis t \rrparenthesis\{\mathsf{sub}(u_i;\mathrm{id}_\Delta)\}"] & \llparenthesis t \rrparenthesis\{\llparenthesis u_\bullet \rrparenthesis\} \arrow[d, Rightarrow, "\mathsf{sub}(t;u_\bullet)"] \\
\llparenthesis t[u_i/x_i] \rrparenthesis\{\mathrm{id}_\Delta\} \arrow[rr, Rightarrow, "\mathsf{sub}(t[u_i/x_i];\mathrm{id}_\Delta)"'] & & \llparenthesis t[u_i/x_i] \rrparenthesis
\end{tikzcd} \tag{5.29}$$

We are finally in a position to prove the `lam` case of (3). Unwinding the clockwise route around the claim, one obtains the left-hand edge of Figure 5.7 below (page 199), in which we abbreviate the term

$$\lambda x.\llparenthesis t \rrparenthesis^{\Gamma,x:A}\Big\{\llparenthesis u_\bullet \rrparenthesis\{\mathrm{inc}_x\}^{\Delta,x:A}\big\{\llparenthesis v_\bullet \rrparenthesis\{\mathrm{inc}_x\}^{\Sigma,x:A}, x^{\Sigma,x:A}\big\}, x^{\Delta,x:A}\big\{\llparenthesis v_\bullet \rrparenthesis\{\mathrm{inc}_x\}^{\Sigma,x:A}, x^{\Sigma,x:A}\big\}\Big\}$$

by $\lambda x.\llparenthesis t \rrparenthesis\{(*)\}$ and write $\varrho^{(x)}_{u_\bullet,x}$ for the rewrite $\varrho^{(x)}_{u_\bullet,x} : x\{x_i \mapsto u_i, x \mapsto v\} \Rightarrow v$ taking the projection at the variable x. One then unfolds the anticlockwise route and applies the inductive hypothesis to obtain the outer edge of Figure 5.7, completing the proof. □

STLC up to isomorphism. One approach in the field of game semantics is to quotient a (putative) cc-bicategory to obtain a cartesian closed category (see *e.g.* [Paq20, Chapter 2]). Doing so loses intensional information, but makes calculations simpler. This suggests that one ought to be able to quotient $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$ (up to the existence of an invertible rewrite) to obtain $\Lambda^{\times,\rightarrow}$ (up to $\beta\eta$-equality).

We begin by making precise the sense in which the $\llparenthesis - \rrparenthesis$ mapping respects $\beta\eta$-equality up to isomorphism.

Lemma 5.4.10. Let $\mathcal{S}$ be a $\Lambda^{\times,\rightarrow}$-signature.

1. If $\Gamma \vdash \tau : t \Rightarrow t' : A$ in $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}(\mathcal{S})$, then $\overline{t} =_{\beta\eta} \overline{t'}$.
2. If $t =_{\beta\eta} t'$ for $t, t' \in \Lambda^{\times,\rightarrow}(\mathcal{S})(\Gamma; A)$, then there exists a rewrite $\Gamma \vdash \mathsf{BE}(t, t') : \llparenthesis t \rrparenthesis \Rightarrow \llparenthesis t' \rrparenthesis : A$ in $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}(\mathcal{S})$.

Proof. For (1) we induct on the derivation of τ. For the structural rewrites and the identity the result is trivial, while for $\tau' \bullet \tau$ it follows immediately from the inductive hypothesis. For $\varpi^{(k)}$ one obtains $\overline{\pi_k\{\mathsf{tup}(t_1, \dots, t_n)\}} = \pi_k(\langle \overline{t_1}, \dots, \overline{t_n} \rangle) =_{\beta\eta} \overline{t_k}$, while for $\mathsf{p}^{\dagger}(\alpha_1, \dots, \alpha_n)$ one has $\overline{u} =_{\beta\eta} \langle \pi_1(\overline{u}), \dots, \pi_n(\overline{u}) \rangle \overset{\mathrm{IH}}{=}_{\beta\eta} \langle \overline{t_1}, \dots, \overline{t_n} \rangle$. The cases for exponential structure are similar: for ε_t one sees that $\overline{\mathsf{eval}\{(\lambda x.t)\{\mathrm{inc}_x\}, x\}} = \mathsf{app}\big(\lambda x.\overline{t}, x\big) =_{\beta\eta} \overline{t}$, while for $\mathsf{e}^{\dagger}(x \,.\, \tau)$ one finds that $\overline{u} =_{\beta\eta} \lambda x.\mathsf{app}(u, x) \overset{\mathrm{IH}}{=}_{\beta\eta} \lambda x.\overline{t}$.

For (2) we induct on the definition of $\beta\eta$-equality (*e.g.* [Cro94, Figure 4.2]).

$\boxed{\beta\text{-rules}}$ For the $\pi_k(\langle t_1, \dots, t_n \rangle) =_{\beta\eta} t_k$ rule one takes $\pi_k\{\mathsf{tup}(\llparenthesis t_1 \rrparenthesis, \dots, \llparenthesis t_n \rrparenthesis)\} \xrightarrow{\varpi^{(k)}} \llparenthesis t_k \rrparenthesis$. For $\mathsf{app}(\lambda x.t, u) =_{\beta\eta} t[u/x]$ one takes $\mathsf{eval}\{\lambda x.\llparenthesis t \rrparenthesis, \llparenthesis u \rrparenthesis\} \overset{\beta}{\Rightarrow} \llparenthesis t \rrparenthesis\{\mathrm{id}_\Gamma, x \mapsto \llparenthesis u \rrparenthesis\} \xrightarrow{\mathsf{sub}} \llparenthesis t[u/x] \rrparenthesis$.

$\boxed{\eta\text{-rules}}$ In a similar fashion, for $t =_{\beta\eta} \langle \pi_1(t), \dots, \pi_n(t) \rangle$ one takes $\llparenthesis t \rrparenthesis \overset{\varsigma}{\Rightarrow} \mathsf{tup}(\pi_1\{\llparenthesis t \rrparenthesis\}, \dots, \pi_n\{\llparenthesis t \rrparenthesis\})$ while for $t =_{\beta\eta} \lambda x.\mathsf{app}(t, x)$ one takes $\llparenthesis t \rrparenthesis \overset{\eta}{\Rightarrow} \lambda x.\mathsf{eval}\{\llparenthesis t \rrparenthesis\{\mathrm{inc}_x\}, x\} \xrightarrow{\lambda x.\mathsf{eval}\{\mathsf{sub}, x\}} \lambda x.\mathsf{eval}\{\llparenthesis t \rrparenthesis, x\}$.

The rules for an equivalence relation hold by the categorical rules on vertical composition. The congruence rules hold by the functoriality of explicit substitution and the functoriality of the $\mathsf{tup}(-, \dots, =)$ and $\lambda x.(-)$ operations. □

The preceding lemma motivates the following definition.

Definition 5.4.11. Fix a $\Lambda^{\times,\rightarrow}$-signature $\mathcal{S}$. For every context Γ and type A, define an equivalence relation $\cong_A^\Gamma$ on $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}(\mathcal{S})(\Gamma; A)$ by setting $t \cong_A^\Gamma t'$ if and only if there exists a (necessarily invertible) rewrite τ such that $\Gamma \vdash \tau : t \Rightarrow t' : A$. ◂

We can therefore rephrase Lemma 5.4.10 as follows. For any pair of terms $t, t' \in \Lambda^{\times,\rightarrow}(\Gamma; A)$ such that $t =_{\beta\eta} t'$, then $\llparenthesis t \rrparenthesis \cong_A^\Gamma \llparenthesis t' \rrparenthesis$; moreover, if $t \cong_A^\Gamma t'$ then $\overline{t} =_{\beta\eta} \overline{t'}$. To show that $\Lambda^{\times,\rightarrow}(\mathcal{S})(\Gamma; A)$-terms modulo-$\beta\eta$ are in bijection with $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}(\mathcal{S})(\Gamma; A)$-terms modulo-$\cong_A^\Gamma$, it remains to show how to *reduce* a term of the form $\llparenthesis \overline{t} \rrparenthesis$ to the original term t.

Construction 5.4.12. Define an invertible rewrite reduce with typing

$$\frac{\Gamma \vdash t : A}{\Gamma \vdash \mathsf{reduce}(t) : t \Rightarrow \llparenthesis \overline{t} \rrparenthesis : A}$$

by extending Construction 3.3.20 with the following rules:

$$\begin{aligned}
\mathsf{reduce}(\pi_k(p)) &:= \pi_k(p) \overset{\iota}{\Rightarrow} \pi_k\{p\} \\
\mathsf{reduce}(\mathsf{tup}(t_1, \ldots, t_n)) &:= \mathsf{tup}(t_1, \ldots, t_n) \xRightarrow{\mathsf{tup}(\mathsf{reduce}, \ldots, \mathsf{reduce})} \mathsf{tup}(\llparenthesis \overline{t_1} \rrparenthesis, \ldots, \llparenthesis \overline{t_n} \rrparenthesis) \\
\mathsf{reduce}(\mathsf{eval}(f, x)) &:= \mathsf{eval}(f, x) \overset{\iota}{\Rightarrow} \mathsf{eval}\{f, x\} \\
\mathsf{reduce}(\lambda x.t) &:= \lambda x.t \xRightarrow{\lambda x.\mathsf{reduce}(t)} \lambda x.\llparenthesis \overline{t} \rrparenthesis
\end{aligned}$$

◂

Thought of as syntax trees, the term $\llparenthesis \overline{t} \rrparenthesis$ is constructed by evaluating explicit substitutions as far as possible and pushing them as far as possible to the left. The reduce rewrites reach a fixpoint on terms of form $\llparenthesis \overline{t} \rrparenthesis$, thereby providing a notion of normalisation in the sense of abstract rewriting systems (*e.g.* [BN98]).

Lemma 5.4.13. For any $\Lambda^{\times,\to}$-signature $\mathcal{S}$ and any term $(\Gamma \vdash t : A)$ derivable in $\Lambda^{\times,\to}(\mathcal{S})$, the judgement $\big(\Gamma \vdash \mathsf{reduce}(\llparenthesis t \rrparenthesis) \equiv \mathrm{id}_{\llparenthesis t \rrparenthesis} : \llparenthesis t \rrparenthesis \Rightarrow \llparenthesis t \rrparenthesis : A\big)$ is derivable in $\Lambda_{\text{ps}}^{\times,\to}(\mathcal{S})$.

Proof. Induction on the structure of t. □

We are now in a position to make precise the sense in which $\Lambda_{\text{ps}}^{\times,\to}$ is $\Lambda^{\times,\to}$ up to isomorphism.

Proposition 5.4.14. For any $\Lambda^{\times,\to}$-signature $\mathcal{S}$, the maps $\llparenthesis - \rrparenthesis : \Lambda^{\times,\to}(\mathcal{S}) \leftrightarrows \Lambda_{\text{ps}}^{\times,\to}(\mathcal{S}) : \overline{(-)}$ descend to a bijection

$$\Lambda^{\times,\to}(\mathcal{S})(\Gamma; A)/\beta\eta \cong \Lambda_{\text{ps}}^{\times,\to}(\mathcal{S})(\Gamma; A)/{\cong_A^\Gamma}$$

between $\alpha\beta\eta$-equivalence classes of $\Lambda^{\times,\to}(\mathcal{S})$-terms and $\alpha{\cong_A^\Gamma}$-equivalence classes of $\Lambda_{\text{ps}}^{\times,\to}(\mathcal{S})$-terms.

Proof. The maps are well-defined on equivalence classes by Lemma 5.4.10 and respect typing by Lemmas 5.4.2 and 5.4.4, so it suffices to check the isomorphism. By Lemma 5.4.5, the composite $\overline{(-)} \circ \llparenthesis - \rrparenthesis$ is the identity. For the other composite, one needs to construct an invertible rewrite $\llparenthesis \overline{t} \rrparenthesis \cong t$ for every derivable term t: we take reduce. □

In particular, every typeable term $(\Gamma \vdash t : A)$ in $\Lambda_{\text{ps}}^{\times,\to}(\mathcal{S})$ has a natural choice of *normal form*, namely the long-$\beta\eta$ normal form (*e.g.* [Hue76]) of $\overline{t}$ as an $\Lambda^{\times,\to}$-term.

Corollary 5.4.15. Let $\mathcal{S}$ be a $\Lambda^{\times,\rightarrow}$-signature. For any derivable term $\Gamma \vdash t : B$ in $\Lambda^{\times,\rightarrow}_{\mathrm{ps}}(\mathcal{S})$, there exists a unique long-$\beta\eta$ normal form term N in $\Lambda^{\times,\rightarrow}(\mathcal{S})$ such that $t \cong^{\Gamma}_{B} (\!| N |\!)$ and $\mathsf{reduce}((\!| N |\!)) \equiv \mathrm{id}_{(\!| N |\!)}$.

Proof. We take N to be the long-$\beta\eta$ normal form of $\overline{t}$. Then $N =_{\beta\eta} \overline{t}$ so, by Proposition 5.4.14,

$$(\!| N |\!) \cong^{\Gamma}_{B} (\!| \overline{t} |\!) \cong^{\Gamma}_{B} t$$

For uniqueness, suppose that N and N' are long-$\beta\eta$ normal terms such that $(\!| N |\!) \cong^{\Gamma}_{B} t \cong^{\Gamma}_{B} (\!| N' |\!)$. Then $\overline{(\!| N |\!)} =_{\beta\eta} \overline{(\!| N' |\!)}$, so that $N =_{\beta\eta} N'$, and hence $N = N'$ by the uniqueness of long $\beta\eta$-normal forms. □

We end this chapter by recording the bicategorical statement of the work in this section.

Theorem 5.4.16. Fix a unary $\Lambda^{\times,\rightarrow}_{\mathrm{ps}}$-signature $\mathcal{S}$. The mappings $(\!| - |\!)$ and $\overline{(-)}$ extend to pseudofunctors between the free cartesian closed bicategory on $\mathcal{S}$ and the free 2-category with bicategorical cartesian closed structure on $\mathcal{S}$. Together with the pseudonatural transformation $(\mathrm{Id}, \mathsf{reduce})$, they form a biequivalence. □

$$\begin{tikzcd}
(\lambda x.\llparenthesis t\rrparenthesis)^{\Gamma}\{\llparenthesis u_\bullet\rrparenthesis^{\Delta}\}\{\llparenthesis v_\bullet\rrparenthesis^{\Sigma}\} \arrow[r, "\mathsf{push}\{v_\bullet\}"] \arrow[dd, "\mathsf{assoc}"'] & \lambda x.\llparenthesis t\rrparenthesis^{\Gamma,x:A}\Big\{\llparenthesis u_\bullet\rrparenthesis\{\mathrm{inc}_x\}^{\Delta,x:A}, x^{\Delta,x:A}\Big\}\{\llparenthesis v_\bullet\rrparenthesis^{\Sigma}\} \arrow[r, "\mathsf{push}"] & \lambda x.\llparenthesis t\rrparenthesis^{\Gamma,x:A}\Big\{\llparenthesis u_\bullet\rrparenthesis\{\mathrm{inc}_x\}^{\Delta,x:A}, x^{\Delta,x:A}\Big\}\Big\{\llparenthesis v_\bullet\rrparenthesis\{\mathrm{inc}_x\}^{\Sigma,x:A}, x^{\Sigma,} \arrow[d, "\lambda x.\mathsf{assoc}"] \\
 & & \lambda x.\llparenthesis t\rrparenthesis^{\Gamma,x:A}\{(*)\} \arrow[d, "\lambda x.\llparenthesis t\rrparenthesis\big\{\llparenthesis u_\bullet\rrparenthesis\{\varrho^{(\bullet)}\}\bullet\mathsf{assoc},\varrho^{(x)}\big\}"] \arrow[dddddl, rounded corners, "\lambda x.\llparenthesis t\rrparenthesis\Big\{\llparenthesis u_\bullet\rrparenthesis\{\mathrm{inc}_x\}\{\mathsf{sub}(\llparenthesis v_\bullet\rrparenthesis;\mathrm{inc}_x),\mathrm{id}_x\},\varrho^{(x)}\Big\}"] \\
(\lambda x.\llparenthesis t\rrparenthesis)^{\Gamma}\Big\{\llparenthesis u_\bullet\rrparenthesis\{\llparenthesis v_\bullet\rrparenthesis\}^{\Sigma}\Big\} \arrow[d, "(\lambda x.\llparenthesis t\rrparenthesis)\{\mathsf{sub}(u_i;v_\bullet)\}"'] \arrow[dr, "\mathsf{push}"] & \overset{\text{Lemma 5.3.15(3)}}{=} & \lambda x.\llparenthesis t\rrparenthesis^{\Gamma,x:A}\Big\{\llparenthesis u_\bullet\rrparenthesis\{\llparenthesis v_\bullet\rrparenthesis\{\mathrm{inc}_x\}\}^{\Sigma,x:A}, x^{\Sigma,x:A}\Big\} \arrow[dddl, "\lambda x.\llparenthesis t\rrparenthesis\{\llparenthesis u_\bullet\rrparenthesis\{\mathsf{sub}(\llparenthesis v_\bullet\rrparenthesis;\mathrm{inc}_x)\},\mathrm{id}_x\}"] \\
(\lambda x.\llparenthesis t\rrparenthesis^{\Gamma,x:A})\{\llparenthesis u_i[v_j/y_j]\rrparenthesis^{\Sigma}\} \arrow[d, "\mathsf{push}"'] \arrow[r, phantom, "\overset{\text{nat.}}{=}"] & \lambda x.\llparenthesis t\rrparenthesis^{\Gamma,x:A}\Big\{\llparenthesis u_\bullet\rrparenthesis\{\llparenthesis v_\bullet\rrparenthesis\}\{\mathrm{inc}_x\}^{\Sigma,x:A}, x^{\Sigma,x:A}\Big\} \arrow[ur, "\lambda x.\llparenthesis t\rrparenthesis\{\mathsf{assoc},\mathrm{id}_x\}"] \arrow[dl, "\lambda x.\llparenthesis t\rrparenthesis\{\mathsf{sub}(u_i;v_\bullet)\{\mathrm{inc}_x\},\mathrm{id}_x\}"] & \\
\lambda x.\llparenthesis t\rrparenthesis^{\Gamma,x:A}\Big\{\llparenthesis u_i[v_j/y_j]\rrparenthesis\{\mathrm{inc}_x\}^{\Sigma,x:A}, x^{\Sigma,x:A}\Big\} \arrow[dd, "\lambda x.\llparenthesis t\rrparenthesis\{\mathsf{sub}(u_i[v_j/y_j];\mathrm{inc}_x),\mathrm{id}_x\}"'] & \overset{(5.28)}{=} & \\
 & \lambda x.\llparenthesis t\rrparenthesis^{\Gamma,x:A}\Big\{\llparenthesis u_\bullet\rrparenthesis\{\llparenthesis v_\bullet\rrparenthesis\}^{\Sigma,x:A}, x^{\Sigma,x:A}\Big\} \arrow[dl, "\lambda x.\llparenthesis t\rrparenthesis\{\mathsf{sub}(u_\bullet;v_\bullet),\mathrm{id}_x\}"] & \\
\lambda x.\llparenthesis t\rrparenthesis^{\Gamma,x:A}\{\llparenthesis u_i[v_j/y_j]\rrparenthesis^{\Sigma,x:A}, x^{\Sigma,x:A}\} \arrow[ddd, "\lambda x.\mathsf{sub}(t;u_\bullet[v_j/y_j],x)"'] & \lambda x.\llparenthesis t\rrparenthesis\Big\{\llparenthesis u_\bullet\rrparenthesis\{y_\bullet\{\llparenthesis v_\bullet\rrparenthesis, x\}\}^{\Sigma,x:A}, x^{\Sigma,x:A}\Big\} \arrow[u, "\lambda x.\llparenthesis t\rrparenthesis\big\{\llparenthesis u_\bullet\rrparenthesis\{\varrho^{(\bullet)}\},\mathrm{id}_x\big\}"'] & \overset{\text{nat.}}{=} \\
\overset{(5.27)}{=} & \lambda x.\llparenthesis t\rrparenthesis^{\Gamma,x:A}\Big\{\llparenthesis u_\bullet\rrparenthesis\{\mathrm{inc}_x\}\{\llparenthesis v_\bullet\rrparenthesis, x\}^{\Sigma,x:A}, x^{\Sigma,x:A}\Big\} \arrow[u, "\lambda x.\llparenthesis t\rrparenthesis\{\mathsf{assoc},\mathrm{id}_x\}"'] \arrow[d, "\lambda x.\llparenthesis t\rrparenthesis^{\Gamma,x:A}\big\{\mathsf{sub}(\llparenthesis u_\bullet\rrparenthesis;\mathrm{inc}_x)\{\llparenthesis v_\bullet\rrparenthesis,x\},\mathrm{id}_x\big\}"] & \\
 & \lambda x.\llparenthesis t\rrparenthesis^{\Gamma,x:A}\Big\{\llparenthesis u_\bullet\rrparenthesis\{\llparenthesis v_\bullet\rrparenthesis, x\}^{\Sigma,x:A}, x^{\Sigma,x:A}\Big\} \arrow[uul, bend left, "\lambda x.\llparenthesis t\rrparenthesis\{\mathsf{sub}(u_\bullet;\llparenthesis v_\bullet\rrparenthesis,x)\},\mathrm{id}_x"] & \\
\lambda x.\llparenthesis t\,[u_i[v_j/y_j]/x_i]\rrparenthesis^{\Sigma,x:A} & &
\end{tikzcd}$$

Figure 5.7: Diagram for the proof of Lemma 5.4.9(3)

Part II

Glueing and normalisation-by-evaluation

Chapter 6

Indexed categories as bicategorical presheaves

Categories of (pre)sheaves are often useful as a kind of 'completion', allowing one to employ extra structure that may not exist in the original category. The aim of this chapter is to show that bicategorical versions of some of these properties extend to the bicategory $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$ of pseudofunctors from a bicategory $\mathcal{B}$ to the 2-category $\mathbf{Cat}$. (Pseudofunctors $\mathcal{B}^{\mathrm{op}} \to \mathbf{Cat}$ are also called *indexed categories* [MP85].) Recall that, since $\mathbf{Cat}$ is a 2-category, so is $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$, and that we write $\mathbf{Cat}$ for the 2-category of small categories (Notation 2.1.10).

Specifically, we shall prove three results which will be used in later chapters:

1. $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$ has all small bilimits, which are given pointwise,
2. $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$ is cartesian closed, and the value of the exponential $[P, Q]$ at $X \in \mathcal{B}$ can be taken to be $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(\mathrm{Y}X \times P, Q) : \mathcal{B} \to \mathbf{Cat}$, for $\mathrm{Y}X := \mathcal{B}(X, -)$ the covariant Yoneda embedding,
3. For any $X \in \mathcal{B}$ the exponential $[\mathrm{Y}X, P]$ in $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$ may be given by $P(- \times X)$.

The proofs are rather technical. The reader willing to take these three statements on trust—for example, by analogy with the case of presheaves—may safely skip this chapter. For reference, the cartesian closed structures we construct here are summarised in an appendix (Tables B.1 and B.2).

Our first result is that $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$ is bicomplete. For brevity, we provide an abstract argument which relies on the notions of *pseudolimit* [Str80] and *flexible limit* [BKP89]. We will not use these concepts anywhere else, so do not delve into the details here: an excellent overview of the various forms of limit and their relationship is available in [Lac10].

Proposition 6.0.1. For any bicategory $\mathcal{B}$, the 2-category $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$ is bicomplete, with bilimits given pointwise.

Proof. We may assume without loss of generality that $\mathcal{B}$ is a 2-category. To see this is the case, observe that if $\mathcal{V} \simeq \mathcal{V}'$ are biequivalent bicategories then $\mathrm{Hom}(\mathcal{V}, \mathbf{Cat}) \simeq \mathrm{Hom}(\mathcal{V}', \mathbf{Cat})$ (see Lemma 6.1.1), and hence $\mathrm{Hom}(\mathcal{V}, \mathbf{Cat})$ has all small bilimits if and only if $\mathrm{Hom}(\mathcal{V}', \mathbf{Cat})$ does. By the coherence theorem for bicategories [MP85] every bicategory is biequivalent to a 2-category, so the claim follows.

Now, by [Pow89b, Proposition 3.6] for any 2-category $\mathcal{C}$ the 2-category $\mathrm{Hom}(\mathcal{C}, \mathbf{Cat})$ admits all flexible limits, calculated pointwise. The so-called 'PIE limits' are flexible ([BKPS89, Proposition 4.7]) and suffice to construct all pseudolimits ([Kel89, Proposition 5.2]), so $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$ has all pseudolimits. But, as explained in [Lac10, §6.12], a 2-category with all pseudolimits has all bilimits, completing the proof. □

This result may also be obtained directly, in a manner similar to the categorical argument, as a corollary of the following proposition. We do not pursue the point any further here for reasons of space.

Proposition 6.0.2. Let $F : \mathcal{B} \to \mathcal{W}$ and $D : \mathcal{V} \to \mathcal{W}$ (D for 'diagram') be pseudofunctors equipped with a chosen biuniversal arrow $(LB, u_B : D(LB) \to FB)$ from D to FB for every $B \in \mathcal{B}$. Then

1. The mapping $L : ob(\mathcal{B}) \to ob(\mathcal{V})$ extends canonically to a pseudofunctor $\mathcal{B} \to \mathcal{V}$, and
2. The biuniversal arrows u_B are the components of a biuniversal arrow $DL \Rightarrow F$ from $D \circ (-) : \mathrm{Hom}(\mathcal{B}, \mathcal{V}) \to \mathrm{Hom}(\mathcal{B}, \mathcal{W})$ to F. □

6.1 $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$ is cartesian closed

It follows immediately from Proposition 6.0.1 that, for any bicategory $\mathcal{B}$, the 2-category $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$ has all finite products. In this section we confront the construction of exponentials. The usual Yoneda argument (see *e.g.* [Awo10, §8.7]), expressed bicategorically, gives us a canonical choice of exponential to check. For any pseudofunctors $P, Q : \mathcal{B} \to \mathbf{Cat}$, putative exponential $[P, Q]$ and object $X \in \mathcal{B}$ one must have

$$\begin{aligned} [P, Q](X) &\simeq \mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(\mathrm{Y}X, [P, Q]) && \text{by the Yoneda lemma} \\ &\simeq \mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(\mathrm{Y}X \times P, Q) && \text{by definition of an exponential} \end{aligned}$$

So it remains to show that the pseudofunctor $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})\big(\mathrm{Y}(-) \times P, Q\big) : \mathcal{B} \to \mathbf{Cat}$ is indeed the exponential $[P, Q]$ in $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$, where $\mathrm{Y}X := \mathcal{B}(X, -)$ denotes the covariant Yoneda embedding.

To simplify the presentation we assume throughout this section that $\mathcal{B}$ is a 2-category. The following lemma guarantees that this entails no loss of generality.

Lemma 6.1.1. Suppose that $\mathcal{B} \simeq \mathcal{B}'$ are biequivalent bicategories and $\mathcal{V}$ is any bicategory. Then:

1. The hom-bicategories $\mathrm{Hom}(\mathcal{B}, \mathcal{V})$ and $\mathrm{Hom}(\mathcal{B}', \mathcal{V})$ are biequivalent, and
2. If $\mathcal{B}$ is cartesian closed, so is $\mathcal{B}'$.

Proof. For (1), suppose the biequivalence is given by pseudofunctors $P : \mathcal{B} \leftrightarrows \mathcal{B}' : Q$. Define pseudofunctors $Q_* : \mathrm{Hom}(\mathcal{B}, \mathcal{V}) \leftrightarrows \mathrm{Hom}(\mathcal{B}', \mathcal{V}) : P_*$ by setting $Q_*(H) := H \circ Q$ and $P_*(F) := F \circ P$. From the biequivalence $\mathcal{B} \simeq \mathcal{B}'$ one obtains equivalences $PQ \simeq \mathrm{id}_{\mathcal{B}'}$ and $QP \simeq \mathrm{id}_{\mathcal{B}}$ and hence equivalences $P_*Q_* \simeq \mathrm{id}_{\mathrm{Hom}(\mathcal{B},\mathcal{V})}$ and $Q_*P_* \simeq \mathrm{id}_{\mathrm{Hom}(\mathcal{B}',\mathcal{V})}$, as required.

For (2), one applies Lemma 2.2.13 to carry the required biuniversal arrows from $\mathcal{B}$ to $\mathcal{B}'$ (*c.f.* also Corollary 2.3.3). □

We now turn to the construction of exponentials in $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$. This entails constructing an adjoint equivalence $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(R, [P, Q]) \simeq \mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(R \times P, Q)$ for every triple of pseudofunctors $P, Q, R : \mathcal{B} \to \mathbf{Cat}$. Since the definition of $[P, Q]$ is also in terms of hom-categories, working with the 1- and 2-cells in $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(R, [P, Q])$ and $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(R \times P, Q)$ quickly becomes complex, with several layers of data to consider. We therefore take the time to unwind some of the definitions we shall be using; as well as serving as a quick-reference on the details of the various definitions, this will fix notation for what follows.

6.1.1 A quick-reference summary

The pseudofunctor $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})\big(\mathrm{Y}(-) \times P, Q\big)$**.** Suppose $f : X \to X'$ in $\mathcal{B}$. The functor $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(\mathrm{Y}f \times P, Q) : \mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(\mathrm{Y}X \times P, Q) \to \mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(\mathrm{Y}X' \times P, Q)$ takes a pseudonatural transformation $(\mathsf{k}, \overline{\mathsf{k}}) : \mathrm{Y}X \times P \to Q$ to the pseudonatural transformation with components $\mathsf{k}(- \circ f, =)$ and witnessing 2-cell given by the following composite for every $g : B \to B'$:

$$\begin{tikzcd}
\mathcal{B}(X', B) \times PB \arrow[r, "\mathcal{B}(X',g) \times Pg"] \arrow[d, "\mathcal{B}(f,B) \times PB"'] \arrow[dr, phantom, "="] & \mathcal{B}(X', B') \times PB' \arrow[d, "\mathcal{B}(f,B') \times PB'"] \\
\mathcal{B}(X, B) \times PB \arrow[r, "\mathcal{B}(X,g) \times Pg"] \arrow[d, "\mathsf{k}_B"'] \arrow[dr, phantom, "\overset{\overline{\mathsf{k}}_g}{\Leftarrow}"] & \mathcal{B}(X, B') \times PB' \arrow[d, "\mathsf{k}_{B'}"] \\
QB \arrow[r, "Qg"'] & QB'
\end{tikzcd}$$

The top square commutes because products in **Cat** are strict and we have assumed that $\mathcal{B}$ is a 2-category.

Remark 6.1.2. We shall write both k_B and $\mathsf{k}(B,-,=)$ to denote the component of a pseudonatural transformation $(\mathsf{k},\overline{\mathsf{k}})$ at an object B. These are just two notations for the same concept: the choice in any particular context is only dependent on which is clearest for exposition. Similar remarks apply to the 2-cells $\overline{\mathsf{k}}$ and to modifications. ◂

Pseudonatural transformations $R \Rightarrow [P,Q]$. To give a pseudonatural transformation $(\mathsf{k},\overline{\mathsf{k}}) : R \Rightarrow \mathrm{Hom}(\mathcal{B},\mathbf{Cat})\big(\mathrm{Y}(-)\times P, Q\big)$ is to give

- For every $X \in \mathcal{B}$ a functor $\mathsf{k}_X : RX \to \mathrm{Hom}(\mathcal{B},\mathbf{Cat})(\mathrm{Y}X \times P, Q)$,
- For every $f : X \to X'$ in $\mathcal{B}$ an invertible 2-cell (that is, a natural isomorphism) $\overline{\mathsf{k}}_f$ as in the following diagram:

$$\begin{tikzcd}
RX \arrow[r, "Rf"] \arrow[d, "\mathsf{k}_X"'] \arrow[dr, phantom, "\overset{\overline{\mathsf{k}}_f}{\Leftarrow}"] & RX' \arrow[d, "\mathsf{k}_{X'}"] \\
\mathrm{Hom}(\mathcal{B},\mathbf{Cat})(\mathrm{Y}X \times P, Q) \arrow[r, "\mathrm{Hom}(\mathcal{B},\mathbf{Cat})(\mathrm{Y}f\times P,Q)"'] & \mathrm{Hom}(\mathcal{B},\mathbf{Cat})(\mathrm{Y}X' \times P, Q)
\end{tikzcd}$$

Thus, for every $r \in RX$ one obtains a pseudonatural transformation $\mathsf{k}(X,r,-) : \mathrm{Y}X\times P \Rightarrow Q$ and an invertible 2-cell (modification) $\overline{\mathsf{k}}(f,r) : \mathsf{k}(X',(Rf)(r),-) \to \mathrm{Hom}(\mathcal{B},\mathbf{Cat})(\mathrm{Y}f \times P, Q)\big(\mathsf{k}(X,r,-)\big)$. The components of this modification are natural isomorphisms $\overline{\mathsf{k}}(f,r,B)$, with components

$$\lambda(h,x)^{\mathcal{B}(X',B)\times PB}\,.\,\mathsf{k}(X',(Rf)(r),B)(h,x) \xrightarrow{\overline{\mathsf{k}}(f,r,B)(h,x)} \mathsf{k}(X,r,B)(h\circ f,x) \tag{6.1}$$

indexed by $B \in \mathcal{B}$. (Note that we use the λ-notation $\lambda(h,x)^{\mathcal{B}(X',B)\times PB}\,.\,\mathsf{k}(X,r,B)(h,x)$ to anonymously refer to the action on objects $(h,x) \in \mathcal{B}(X',B)\times PB$.) The modification axiom on $\overline{\mathsf{k}}(f,r)$ requires that the diagram below commutes for every $(h,p) \in \mathcal{B}(X,B)\times PB$, $g : B \to B'$ and $f : X \to X'$ in $\mathcal{B}$:

$$\begin{tikzcd}
\mathsf{k}(X',(Rf)(r),B')\big(gh,(Pf)(p)\big) \arrow[r, "\overline{\mathsf{k}}(X',(Rf)(r),g)(h,(Pf)(p))"] \arrow[d, "\overline{\mathsf{k}}(f,r)(gh,(Pf)(p))"'] & (Qg)\big(\mathsf{k}(X',(Rf)(r),B)(h,p)\big) \arrow[d, "(Qg)(\overline{\mathsf{k}}(f,r)(h,p))"] \\
\mathsf{k}(X,r,B')\big(ghf,(Pf)(p)\big) \arrow[r, "\overline{\mathsf{k}}(X,r,g)(hf,(Pf)(p))"'] & (Qg)\big(\mathsf{k}(X,r,B)(hf,p)\big)
\end{tikzcd} \tag{6.2}$$

We can unfold the pseudonatural transformation $\mathsf{k}(X,r,-)$ further. It has components given by functors $\mathsf{k}(X,r,B) : \mathcal{B}(X,B)\times PB \to QB$ (for $B \in \mathcal{B}$), and for every $g : B \to B'$ one obtains an invertible 2-cell (that is, a natural isomorphism) $\overline{\mathsf{k}}(X,r,g)$ as in

$$\begin{tikzcd}
\mathcal{B}(X,B)\times PB \arrow[r, "\mathcal{B}(X,g)\times Pg"] \arrow[d, "\mathsf{k}(X,r,B)"'] \arrow[dr, phantom, "\overset{\overline{\mathsf{k}}(X,r,g)}{\Leftarrow}"] & \mathcal{B}(X,B')\times PB' \arrow[d, "\mathsf{k}(X,r,B')"] \\
QB \arrow[r, "Qg"'] & QB'
\end{tikzcd} \tag{6.3}$$

Examining the components of this 2-cell, one sees that for each $(h, p) \in \mathcal{B}(X, B) \times PB$ one obtains an invertible 1-cell $\overline{\mathsf{k}}(X, r, g)(h, p) : \mathsf{k}(X, r, B')\big(g \circ h, (Pg)(p)\big) \to (Qg)\big(\mathsf{k}(X, r, B)(h, p)\big)$.

There are then two levels of naturality at play, related via (6.2). The naturality condition making $\overline{k}(X, r, -)$ a pseudonatural transformation requires that for every 2-cell $\tau : g \Rightarrow g' : B \to B'$ the following commutes:

$$\begin{tikzcd}
\mathsf{k}(X, r, B')\big(g \circ h, (Pg)(p)\big) \arrow[r, "\mathsf{k}(X,r,B')(\tau \circ h, (P\tau)(p))"] \arrow[d, "\overline{\mathsf{k}}(X,r,g)(h,p)"'] & \mathsf{k}(X, r, B')\big(g' \circ h, (Pg)(p)\big) \arrow[d, "\overline{\mathsf{k}}(X,r,g')(h,p)"] \\
(Qg)\big(\mathsf{k}(X, r, B)(h, p)\big) \arrow[r, "(Q\tau)(\mathsf{k}(X,r,B)(h,p))"'] & (Qg')\big(\mathsf{k}(X, r, B)(h, p)\big)
\end{tikzcd}$$

On the other hand, the naturality condition making $\overline{\mathsf{k}}(X, r, g)$ a natural transformation requires that for every $\rho : h \Rightarrow h'$ in $\mathcal{B}(X, B)$ and $t : p \to p'$ in PB, the following commutes:

$$\begin{tikzcd}
\mathsf{k}(X, r, B')\big(g \circ h, (Pg)(p)\big) \arrow[r, "\mathsf{k}(X,r,B')(g \circ \rho, (Pg)(t))"] \arrow[d, "\overline{\mathsf{k}}(X,r,g)(h,p)"'] & \mathsf{k}(X, r, B')\big(g \circ h', (Pg)(p')\big) \arrow[d, "\overline{\mathsf{k}}(X,r,g)(h',p')"] \\
(Qg)\big(\mathsf{k}(X, r, B)(h, p)\big) \arrow[r, "(Qg)(\mathsf{k}(X,r,B)(\rho,t))"'] & (Qg)\big(\mathsf{k}(X, r, B)(h', p')\big)
\end{tikzcd}$$

Modifications $(\mathsf{j}, \bar{\mathsf{j}}) \to (\mathsf{m}, \overline{\mathsf{m}}) : R \Rightarrow [P, Q]$**.** To give a modification $\Psi : (\mathsf{j}, \bar{\mathsf{j}}) \to (\mathsf{m}, \overline{\mathsf{m}})$ between pseudonatural transformations $R \Rightarrow [P, Q]$ is to give a natural transformation $\Psi_X : \mathsf{j}_X \Rightarrow \mathsf{m}_X$ between functors of type $RX \to \mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(\mathrm{Y}X \times P, Q)$ for every $X \in \mathcal{B}$, such that the whole X-indexed family of natural transformations satisfies the modification axiom.

Unwinding the definition of natural transformation, Ψ_X is a family of 2-cells (that is, modifications) $\Psi(X, r, -) : \mathsf{j}(X, r, -) \Rightarrow \mathsf{m}(X, r, -)$, natural in $r \in \mathcal{B}$ and such that every $\Psi(X, r, -)$ satisfies the modification axiom. In particular, since every $\Psi(X, r, -)$ is a modification between pseudonatural transformations $\mathrm{Y}X \times P \Rightarrow Q$, for every $B \in \mathcal{B}$ we have a natural transformation $\Psi(X, r, B) : \mathsf{j}(X, r, B) \Rightarrow \mathsf{m}(X, r, B) : \mathcal{B}(X, B) \times PB \to QB$.

6.1.2 The cartesian closed structure of $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$

To construct exponentials in $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$ we are required to give:

- A biuniversal arrow $\mathrm{eval}_{P,Q} : [P, Q] \times P \to Q$ for each $P, Q : \mathcal{B} \to \mathbf{Cat}$,
- A mapping $\Lambda : ob\big(\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(R \times P, Q)\big) \to ob\big(\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(R, [P, Q])\big)$,
- An invertible universal 2-cell $\mathrm{eval}_{P,Q} \circ \Lambda(\mathsf{j}, \bar{\mathsf{j}}) \Rightarrow (\mathsf{j}, \bar{\mathsf{j}})$ defining the counit, such that the unit is also invertible.

We take these components in turn. The main difficulty of the proof is maintaining a clear view of what one is required to construct, and ensuring that all the relevant axioms have been checked.

The biuniversal arrow. Our first step is the construction of the biuniversal arrow $\mathrm{eval}_{P,Q} : [P, Q] \times P \to Q$. To be a 1-cell in $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$, this needs to be a pseudonatural transformation for which each component is a functor $e_X : \mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(\mathrm{Y}X \times P, Q) \times PX \to QX$.

Let $X \in \mathcal{B}$ be fixed; we define e_X. Consider a pair $\big((\mathsf{k}, \overline{\mathsf{k}}), p\big) \in \mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(\mathrm{Y}X \times P, Q)$ consisting of a pseudonatural transformation $(\mathsf{k}, \overline{\mathsf{k}}) : \mathrm{Y}X \times P \Rightarrow Q$ and an element $p \in PX$. Noting that, in particular, the component of $(\mathsf{k}, \overline{\mathsf{k}})$ at $X \in \mathcal{B}$ has type $\mathcal{B}(X, X) \times PX \to QX$, one obtains a functor $\mathsf{k}(X, \mathrm{Id}_X, -) : PX \to QX$. We therefore define $e_X\big((\mathsf{k}, \overline{\mathsf{k}}), p\big) := \mathsf{k}(X, \mathrm{Id}_X, p)$.

To extend this to morphisms, we need to define a morphism $\mathsf{k}(X, \mathrm{Id}_X, p) \to \mathsf{k}'(X, \mathrm{Id}_X, p')$ for every pair (Ξ, f) consisting of a modification $\Xi : (\mathsf{k}, \overline{\mathsf{k}}) \to (\mathsf{k}', \overline{\mathsf{k}}')$ and morphism $f : p \to p'$. The modification Ξ is a family of natural transformations $\Xi_X : \mathsf{k}(X, -, =) \Rightarrow \mathsf{k}'(X, -, =)$ for $X \in \mathcal{B}$, where naturality amounts to the following commutative diagram for every $\tau : h \Rightarrow h' : X \to B$ and $f : p \to p'$ in PB:

$$\begin{tikzcd}
\mathsf{k}(X, h, p) \arrow[r, "\mathrm{k}(X,\tau,f)"] \arrow[d, "\Xi_X(h,p)"'] & \mathsf{k}(X, h', p') \arrow[d, "\Xi_X(h',p')"] \\
\mathsf{k}'(X, h, p) \arrow[r, "\mathrm{k}(X,\tau,f)"'] & \mathsf{k}'(X, h', p')
\end{tikzcd}$$

We define $e_X(\Xi, f)$ to be the composite

$$e_X(\Xi, f) := \mathsf{k}(X, \mathrm{Id}_X, p) \xRightarrow{\Xi_X(\mathrm{Id}_X, p)} \mathsf{k}'(X, \mathrm{Id}_X, p) \xRightarrow{\mathrm{k}'(X, \mathrm{Id}_X, f)} \mathsf{k}'(X, \mathrm{Id}_X, p')$$

This definition is functorial.

Next we need to provide invertible 2-cells witnessing that the mappings e_X are pseudonatural. That is, for every $f : X \to X'$ in $\mathcal{B}$ we need to provide a natural isomorphism as in the following diagram:

$$\begin{tikzcd}
\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(\mathrm{Y}X \times P, Q) \times PX \arrow[rr, "\mathrm{Hom}(\mathcal{B},\mathbf{Cat})(\mathrm{Y}f \times P, Q) \times Pf"] \arrow[d, "e_X"'] & & \mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(\mathrm{Y}X' \times P, Q) \times PX' \arrow[d, "e_{X'}"] \\
QX \arrow[rr, "Qf"'] & \overset{\overline{e}_f}{\Leftarrow} & QX'
\end{tikzcd}$$

Chasing an arbitrary element $\big((\mathsf{k}, \overline{\mathsf{k}}), p\big) \in \mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(\mathrm{Y}X \times P, Q) \times PX$ through this diagram, one sees that we need to provide an isomorphism $\mathsf{k}\big(X', f, (Pf)(p)\big) \cong (Qf)(\mathsf{k}(X, \mathrm{Id}_X, p))$ in QX'. We take

$$\overline{e}_f\big((\mathsf{k}, \overline{\mathsf{k}}), p\big) := \mathsf{k}(X', f, (Pf)(p)) = \mathsf{k}(X', f \circ \mathrm{Id}_X, (Pf)(p)) \xRightarrow{\overline{\mathrm{k}}(X,r,f)(\mathrm{Id}_X, p)} (Qf)\big(\mathsf{k}(X, r, B)(\mathrm{Id}_X, p)\big)$$

using the natural isomorphism provided by diagram (6.3).

Lemma 6.1.3. The pair $(e, \overline{e})$ defined above is a pseudonatural transformation $[P, Q] \times P \Rightarrow Q$.

Proof. The naturality condition follows directly from that for $\overline{\mathsf{k}}$. Similarly, the unit and associativity and unit laws hold immediately because they hold for $(\mathsf{k},\overline{\mathsf{k}})$. □

We now have a candidate for the biuniversal arrow $\mathrm{eval}_{P,Q}$ defining exponentials. The next step is to define a mapping $\Lambda : ob\big(\mathrm{Hom}(\mathcal{B},\mathbf{Cat})(R \times P, Q)\big) \to ob\big(\mathrm{Hom}(\mathcal{B},\mathbf{Cat})(R,[P,Q])\big)$.

The mapping Λ. Let $(\mathsf{j},\bar{\mathsf{j}})$ be a pseudonatural transformation $R \times P \Rightarrow Q$. We define $\Lambda(\mathsf{j},\bar{\mathsf{j}}) : R \Rightarrow [P,Q]$ in stages. For the 1-cell components we need to define a functor $RX \to \mathrm{Hom}(\mathcal{B},\mathbf{Cat})(\mathrm{Y}X \times P, Q)$ for every $X \in \mathcal{B}$. We do this first.

Fix some $X \in \mathcal{B}$ and $r \in RX$. We define a pseudonatural transformation $(\Lambda\mathsf{j})(X,r,-) : \mathrm{Y}X \times P \Rightarrow Q$. For every $B \in \mathcal{B}$ we take the functor

$$\begin{aligned} \mathcal{B}(X,B) \times PB &\to QB \\ (h,p) &\mapsto \mathsf{j}\big(X,(Rh)(r),p\big) \end{aligned}$$

This is well-defined because $\mathsf{j}_X : RX \times PX \to QX$, so $(Rh)(r) \in RB$. We take the evident functorial action on 2-cells: $(\Lambda\mathsf{j})(X,r,B)(\tau,f) := \mathsf{j}\big(X,(R\tau)(r),f\big)$.

To extend these 1-cells to a pseudonatural transformation we need to provide a natural isomorphism $\overline{(\Lambda\mathsf{j})}(X,r,g)$ as in

$$\begin{tikzcd}
\mathcal{B}(X,B) \times PB \arrow[r, "\mathcal{B}(X{,}g) \times Pg"] \arrow[d, "(\Lambda\mathsf{j})(X{,}r{,}B)"'] \arrow[dr, phantom, "\overset{\overline{(\Lambda\mathsf{j})}(X{,}r{,}g)}{\Leftarrow}"] & \mathcal{B}(X,B') \times PB' \arrow[d, "(\Lambda\mathsf{j})(X{,}r)_{B'}"] \\
QB \arrow[r, "Qg"'] & QB'
\end{tikzcd}$$

for every $g : B \to B'$ in $\mathcal{B}$. So for every $(h,p) \in \mathcal{B}(X,B) \times PB$ we need to give an isomorphism $\mathsf{j}\big(X,(Rgh)(r),(Pg)(p)\big) \cong (Qg)\big(\mathsf{j}\big(X,(Rh)(r),p\big)\big)$, for which we take the composite defined by commutativity of

$$\begin{tikzcd}
\mathsf{j}\big(X,(Rgh)(r),(Pg)(p)\big) \arrow[rr, "\overline{(\Lambda\mathsf{j})}(X{,}r{,}g)"] \arrow[dr, "\mathsf{j}\left(X{,}(\phi^R_{g,h})^{-1}(r){,}(Pg)(p)\right)"'] & & (Qg)\big(\mathsf{j}\big(X,(Rh)(r),p\big)\big) \\
& \mathsf{j}\big(X,(Rg)(Rh)(r),(Pg)(p)\big) \arrow[ur, "\bar{\mathsf{j}}(g{,}(Rh)(r){,}p)"'] &
\end{tikzcd}$$

This definition is natural in g because $\phi^R_{g,h}$ and $\bar{\mathsf{j}}_g$ both are. The unit and associativity laws follow easily from those of $(\mathsf{j},\bar{\mathsf{j}})$, yielding the following.

Lemma 6.1.4. For every $X \in \mathcal{B}$, $r \in RX$ and pseudonatural transformation $(\mathsf{j},\bar{\mathsf{j}}) : R \times P \Rightarrow Q$, the pair $\big((\Lambda\mathsf{j})(X,r,-),\overline{(\Lambda\mathsf{j})}(X,r,-)\big)$ is a pseudonatural transformation $\mathrm{Y}X \times P \Rightarrow Q$. □

The preceding lemma defines a mapping $ob(RX) \to ob\big(\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(\mathrm{Y}X \times P, Q)\big)$. Our next task is to extend this to a functor. So suppose $f : r \to r'$ in RX. To give a modification $(\Lambda \mathsf{j})(X, f, -) : (\Lambda \mathsf{j})(X, r, -) \to (\Lambda \mathsf{j})(X, r', -)$, one must provide a family of natural transformations $(\Lambda \mathsf{j})(X, r, B) \Rightarrow (\Lambda \mathsf{j})(X, r', B)$ indexed by $B \in \mathcal{B}$. For a fixed choice of B and $(h, p) \in \mathcal{B}(X, B) \times PB$, we take the 1-cell

$$(\Lambda \mathsf{j})(X, f, B)(h, p) := \mathsf{j}(X, (Rh)(r), p) \xRightarrow{\mathsf{j}(X,(Rh)(f),p)} \mathsf{j}(X, (Rh)(r'), p)$$

This is natural in h and p by functoriality. The modification law for $(\Lambda \mathsf{j})(X, f, -)$ is a consequence of the naturality properties. For (h, p) as above and $f : r \to r'$, one has

$$\begin{tikzcd}
\mathsf{j}\big(X', (Rgh)(r), (Pg)(p)\big) \arrow[r, "\mathsf{j}(X',(Rgh)(f),(Pg)(p))"] \arrow[d, "\mathsf{j}\big(X',(\phi^R_{g,h})^{-1}(r),(Pg)(p)\big)"'] & \mathsf{j}\big(X', (Rgh)(r'), (Pg)(p)\big) \arrow[d, "\mathsf{j}\big(X',(\phi^R_{g,h})^{-1}(r'),(Pg)(p)\big)"] \\
\mathsf{j}\big(X', (Rg)(Rh)(r), (Pg)(p)\big) \arrow[r, "\mathsf{j}(X',(Rg)(Rh)(f),(Pg)(p))"] \arrow[d, "\bar{\mathsf{j}}(g,(Rh)(r),p)"'] & \mathsf{j}\big(X', (Rg)(Rh)(r'), (Pg)(p)\big) \arrow[d, "\bar{\mathsf{j}}(g,(Rh)(r'),p)"] \\
(Qg)\big(\mathsf{j}(X, (Rh)(r), p)\big) \arrow[r, "(Qg)(\mathsf{j}(X,(Rh)(f),p))"'] & (Qg)\big(\mathsf{j}(X, (Rh)(r'), p)\big)
\end{tikzcd}$$

in which the top square commutes by naturality of ϕ^R and the bottom square by the fact that $\bar{\mathsf{j}}_g$ is a natural transformation.

We have now defined a functor $(\Lambda \mathsf{j})(X, -, =) : RX \to \mathrm{Hom}(\mathcal{B}, \mathbf{Cat})\big(\mathrm{Y}X \times P, Q\big)$ for each $X \in \mathcal{B}$. It remains to show these functors are the components of a pseudonatural transformation. Thus, for every $f : X \to X'$ we need to provide invertible 2-cells $\overline{(\Lambda \mathsf{j})}(f, -, =)$ as in

$$\begin{tikzcd}
RX \arrow[r, "Rf"] \arrow[d, "(\Lambda \mathsf{j})(X,-,=)"'] \arrow[dr, phantom, "\overset{\overline{(\Lambda \mathsf{j})}(f,-,=)}{\Leftarrow}"] & RX' \arrow[d, "(\Lambda \mathsf{j})(X',-,=)"] \\
\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(\mathrm{Y}X \times P, Q) \arrow[r, "\mathrm{Hom}(\mathcal{B},\mathbf{Cat})(\mathrm{Y}f \times P,Q)"'] & \mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(\mathrm{Y}X' \times P, Q)
\end{tikzcd}$$

This diagram requires an isomorphism

$$\lambda B^{\mathcal{B}} \,.\, \lambda(h, p)^{\mathcal{B}(X',B) \times PB} \,.\, \mathsf{j}(X, (Rh)(Rf)(r), p) \cong \mathsf{j}(X, (Rhf)(r), p) \tag{6.4}$$

for each $r \in RX$, for which we take simply $\lambda B^{\mathcal{B}} \,.\, \lambda(h, p)^{\mathcal{B}(X'B) \times PB} \,.\, \mathsf{j}(X, \phi^R_{h,f}(r), p)$. The unit and associativity laws then follow from the unit and associativity laws of the pseudofunctor R.

We record our progress in the following lemma.

Lemma 6.1.5. The pair $\big((\Lambda \mathsf{j})(X, -, =), \overline{(\Lambda \mathsf{j})}(f, -, =)\big)$ is a pseudonatural transformation $R \Rightarrow \mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(\mathrm{Y}X \times P, Q)$. □

We therefore define the required mapping as follows:

$$\begin{aligned}\Lambda : ob\big(\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(R \times P, Q)\big) &\to ob\big(\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(R, [P, Q])\big) \\ (\mathsf{j}, \bar{\mathsf{j}}) &\mapsto \big((\Lambda\mathsf{j})(X, -, =), \overline{(\Lambda\mathsf{j})}(f, -, =)\big)\end{aligned}$$

Our next task is to define the universal arrow, which will act as the counit.

The counit E. We begin by calculating $\mathrm{eval}_{P,Q} \circ \big((\mathsf{k}, \overline{\mathsf{k}}) \times P\big) : R \times P \Rightarrow Q$ for any $(\mathsf{k}, \overline{\mathsf{k}}) : R \Rightarrow [P, Q]$. The component at $X \in \mathcal{B}$ is the functor acting on $(r, p) \in RX \times PX$ by

$$\begin{aligned}\big(e_X \circ (\mathsf{k}_X \times PX)\big)(X, r, p) &= e_X\big(\mathsf{k}(X, r, -), p\big) \\ &= e_X\big(\lambda B^{\mathcal{B}} \,.\, \lambda(h, x)^{\mathcal{B}(X,B) \times PB} \,.\, \mathsf{k}(X, r, B)(h, x), p\big) \\ &= \mathsf{k}(X, r, X)(\mathrm{Id}_X, p)\end{aligned}$$

For any $f : X \to X'$ and $(r, p) \in RX \times PX$, the witnessing 2-cell is defined by the following commutative diagram:

$$\begin{tikzcd}
\mathsf{k}(X', (Rf)(r), X')\big(\mathrm{Id}_{X'}, (Pf)(p)\big) \arrow[r, "\overline{(\mathrm{eval}_{P,Q} \circ ((\mathsf{k},\overline{\mathsf{k}}) \times P))}_f(r,p)"] \arrow[d, "\overline{\mathsf{k}}(f,r)(\mathrm{Id}_{X'},(Pf)(p))"'] & (Qf)\big(\mathsf{k}(X, r, X)(\mathrm{Id}_X, p)\big) \\
\mathsf{k}(X, r, X')\big(\mathrm{Id}_{X'} \circ f, (Pf)(p)\big) \arrow[r, equal] & \mathsf{k}(X, r, X')\big(f \circ \mathrm{Id}_X, (Pf)(p)\big) \arrow[u, "\overline{\mathsf{k}}(X,r,f)(\mathrm{Id}_X,p)"']
\end{tikzcd} \tag{6.5}$$

Note that both levels of naturality appear in this definition: the first arrow arises from the components of the modification $\overline{\mathsf{k}}(f, r)$ given in (6.1), while the second arises from the 2-cell witnessing the naturality of k_X in diagram (6.3).

Now suppose that $(\mathsf{j}, \bar{\mathsf{j}}) : R \times P \Rightarrow Q$ and consider $\mathrm{eval}_{P,Q} \circ \big(\Lambda(\mathsf{j}, \bar{\mathsf{j}}) \times P\big) : R \times P \Rightarrow Q$. The 1-cell components of this pseudonatural transformation act by

$$\begin{aligned}RX \times PX &\to QX \\ (r, p) &\mapsto \mathsf{j}\big(X, (R\mathrm{Id}_X)(r), p\big)\end{aligned} \tag{6.6}$$

and for $f : X \to X'$ and $(r, p) \in RX \times PX$ the witnessing 2-cell is the composite

$$\begin{tikzcd}
\mathsf{j}\big(X', (R\mathrm{Id}_{X'})(Rf)(r), (Pf)(p)\big) \arrow[r, "\overline{(\mathrm{eval}_{P,Q} \circ (\Lambda(\mathsf{j},\bar{\mathsf{j}}) \times P))}_f"] \arrow[d, "\mathsf{j}\big(X',\phi^R_{\mathrm{Id},f}(r),(Pf)(p)\big)"'] & (Qf)\big(\mathsf{j}(X, R(\mathrm{Id}_X)(r), p)\big) \\
\mathsf{j}\big(X', R(\mathrm{Id}_{X'} \circ f)(r), (Pf)(p)\big) \arrow[d, equal] & \\
\mathsf{j}\big(X', R(f \circ \mathrm{Id}_X)(r), (Pf)(p)\big) \arrow[r, "\mathsf{j}\big(X',(\phi^R_{f,\mathrm{Id}})^{-1}(r),(Pf)(p)\big)"'] & \mathsf{j}\big(X', R(f)R(\mathrm{Id}_X)(r), (Pf)(p)\big) \arrow[uu, "\bar{\mathsf{j}}(f,(R\mathrm{Id}_X)(r),p)"']
\end{tikzcd}$$

By the identification (6.6), to define the counit modification $\mathrm{E} : \mathrm{eval}_{P,Q} \circ \big(\Lambda(\mathsf{j}, \bar{\mathsf{j}}) \times P\big) \to (\mathsf{j}, \bar{\mathsf{j}})$ we need to provide a natural transformation $\mathrm{E}_X : \mathsf{j}\big(X, (R\mathrm{Id}_X)(-), =\big) \Rightarrow \mathsf{j}(X, -, =) : RX \times PX \to QX$ for every $X \in \mathcal{B}$. We take the obvious choice, namely $\lambda(r,p)^{RX \times PX} \,.\, \mathsf{j}\big(X, (\psi^R_X)^{-1}(r), p\big)$. Since $\psi^R_X : \mathrm{Id}_{RX} \Rightarrow R\mathrm{Id}_X$ is a 2-cell in **Cat**, *i.e.* a natural transformation, it only remains to check the modification axiom.

Lemma 6.1.6. The family of 2-cells $\mathrm{E}_X := \mathsf{j}\big(X, (\psi^R_X)^{-1}(-), =\big)$ (for $X \in \mathcal{B}$) form a modification $\mathrm{eval}_{P,Q} \circ \Lambda(\mathsf{j}, \bar{\mathsf{j}}) \to (\mathsf{j}, \bar{\mathsf{j}})$.

Proof. We need to verify that the following diagram commutes for every $f : X \to X'$ in $\mathcal{B}$:

$$\begin{tikzcd}
\mathsf{j}\big(X', (Rf)(r), (Pf)(p)\big) \arrow[r, "\bar{\mathsf{j}}(f,r,p)"] \arrow[d, swap, "\mathrm{E}_{X'}((Rf)(r),(Pf)(p))"] & (Qf)\big(\mathsf{j}(X, r, p)\big) \arrow[ddddd, "(Qf)(\mathrm{E}_X(r,p)))"] \\
\mathsf{j}\big(X', (R\mathrm{Id}_{X'})(Rf)(r), (Pf)(p)\big) \arrow[d, swap, "\mathsf{j}(X'{,}\phi^R_{\mathrm{Id},f}(r){,}(Pf)(p))"] \arrow[dddd, bend right=60, swap, "\overline{(\mathrm{eval}_{P,Q}\circ(\Lambda(\mathsf{j},\bar{\mathsf{j}})\times P))}(f,r,p)"] & \\
\mathsf{j}\big(X', R(\mathrm{Id}_{X'} \circ f)(r), (Pf)(p)\big) \arrow[d, equal] & \\
\mathsf{j}\big(X', R(f \circ \mathrm{Id}_X)(r), (Pf)(p)\big) \arrow[d, swap, "\mathsf{j}\big(X'{,}(\phi^R_{f,\mathrm{Id}})^{-1}(r){,}(Pf)(p)\big)"] & \\
\mathsf{j}\big(X', R(f)R(\mathrm{Id}_X)(r), (Pf)(p)\big) \arrow[d, swap, "\bar{\mathsf{j}}(f{,}R(\mathrm{Id}_X)(r){,}p))"] & \\
(Qf)\big(\mathsf{j}(X, R(\mathrm{Id}_X)(r), p)\big) \arrow[r, equal] & (Qf)\big(\mathsf{j}(X, R(\mathrm{Id}_X)(r), p)\big)
\end{tikzcd} \tag{6.7}$$

To this end, one uses the two unit laws of a pseudofunctor to see that the following commutes:

$$\begin{tikzcd}
 & & \mathsf{j}_{X'} \circ \big(Rf \times Pf\big) \arrow[dll, swap, "\mathsf{j}_{X'}\circ(\psi^R_{X'}\times Pf)"] \arrow[ddd, equal] \arrow[ddddr, "\mathsf{j}_{X'}\circ((Rf\circ\psi^R_X)\times Pf)"] & \\
\mathsf{j}_{X'} \circ \big((R\mathrm{Id}_{X'} \circ Rf) \times Pf\big) \arrow[d, swap, "\mathsf{j}_{X'}\circ(\phi^R_{\mathrm{Id},f}\times Pf)"] & & & \\
\mathsf{j}_{X'} \circ \big(R(\mathrm{Id}_{X'} \circ f) \times Pf\big) \arrow[drr, equal] \arrow[ddrr, equal, bend right] & & & \\
 & & \mathsf{j}_{X'} \circ \big(Rf \times Pf\big) \arrow[d, equal] & \\
 & & \mathsf{j}_{X'} \circ \big(R(f \circ \mathrm{Id}_X) \times Pf\big) \arrow[r, swap, "\mathsf{j}\big(X'{,}(\phi^R_{f,\mathrm{Id}})^{-1}{,}Pf\big)"] & \mathsf{j}_{X'} \circ \big((Rf \circ R\mathrm{Id}_X) \times Pf\big)
\end{tikzcd}$$

Diagram (6.7) therefore reduces to

$$\begin{tikzcd}
\mathsf{j}\big(X', (Rf)(r), (Pf)(p)\big) \arrow[r, "\bar{\mathsf{j}}(f,r,p)"] \arrow[d, "\mathsf{j}(X',(Rf)(\psi^R_X)(r),(Pf)(p))"'] & (Qf)\big(\mathsf{j}(X, r, p)\big) \arrow[dd, "(Qf)(\mathsf{j}(X,\psi^R_X(r),p))"] \\
\mathsf{j}\big(X', R(f)R(\mathrm{Id}_X)(r), (Pf)(p)\big) \arrow[d, "\bar{\mathsf{j}}(f,R(\mathrm{Id}_X)(r),p))"'] & \\
(Qf)\big(\mathsf{j}(X, R(\mathrm{Id}_X)(r), p)\big) \arrow[r, equal] & (Qf)\big(\mathsf{j}(X, R(\mathrm{Id}_X)(r), p)\big)
\end{tikzcd}$$

which commutes by the naturality of $\bar{\mathsf{j}}(f, -, =)$ in r. □

We have constructed our candidate counit E; now we need to show it is universal. For the existence part of this claim, we need to construct a modification $\Xi^\dagger : (\mathsf{k}, \overline{\mathsf{k}}) \to \Lambda(\mathsf{j}, \bar{\mathsf{j}})$ for every pair of pseudonatural transformations $(\mathsf{j}, \bar{\mathsf{j}}) : R \times P \Rightarrow Q$ and $(\mathsf{k}, \overline{\mathsf{k}}) : R \Rightarrow [P, Q]$ and every modification $\Xi : \mathrm{eval}_{P,Q} \circ \big((\mathsf{k}, \overline{\mathsf{k}}) \times P\big) \to (\mathsf{j}, \bar{\mathsf{j}})$.

The modification $\Xi^\dagger$. We begin by unwinding the definition of a modification $\mathrm{eval}_{P,Q} \circ \big((\mathsf{k}, \overline{\mathsf{k}}) \times P\big) \to (\mathsf{j}, \bar{\mathsf{j}})$. For every $X \in \mathcal{B}$ and $(r, p) \in RX \times PX$, we are given a 1-cell $\Xi(X, r, p) : \mathsf{k}(X, r, X)(\mathrm{Id}_X, p) \to \mathsf{j}(X, r, p)$ in QX. These are natural in the sense that, for any $g : r \to r'$ and $h : p \to p'$ in $RX \times PX$, the following commutes:

$$\begin{tikzcd}
\mathsf{k}(X, r, X)(\mathrm{Id}_X, p) \arrow[r, "\mathsf{k}(X,g,X)(\mathrm{Id}_X,h)"] \arrow[d, "\Xi(X,r,p)"'] & \mathsf{k}(X, r', X)(\mathrm{Id}_X, p') \arrow[d, "\Xi(X,r',p')"] \\
\mathsf{j}(X, r, p) \arrow[r, "\mathsf{j}(X,g,h)"'] & \mathsf{j}(X, r', p')
\end{tikzcd}$$

The X-indexed family of natural transformations $\Xi(X, -, =)$ is subject to the modification axiom, which requires that the following commutes for every $f : X \to X'$ in $\mathcal{B}$ (recall the definition of $\overline{(\mathrm{eval}_{P,Q} \circ ((\mathsf{k}, \overline{\mathsf{k}}) \times P)}_f$ from (6.5)):

$$\begin{tikzcd}
\mathsf{k}\big(X', (Rf)(r), X'\big)\big(\mathrm{Id}_{X'}, (Pf)(p)\big) \arrow[r, "\Xi(X',(Rf)(r),(Pf)(r))"] \arrow[d, "\overline{\mathsf{k}}(f,r,B)(\mathrm{Id}_{X'},(Pf)(p))"'] & \mathsf{j}\big(X', (Rf)(r), (Pf)(p)\big) \arrow[ddd, "\bar{\mathsf{j}}(f,r,p)"] \\
\mathsf{k}(X, r, X')\big(\mathrm{Id}_{X'} \circ f, (Pf)(p)\big) \arrow[d, equal] & \\
\mathsf{k}(X, r, X')\big(f \circ \mathrm{Id}_X, (Pf)(p)\big) \arrow[d, "\overline{\mathsf{k}}(X,r,f)(\mathrm{Id}_X,p)"'] & \\
(Qf)\big(\mathsf{k}(X, r, X)(\mathrm{Id}_X, p)\big) \arrow[r, "(Qf)(\Xi(X,r,p))"'] & (Qf)\big(\mathsf{j}(X, r, p)\big)
\end{tikzcd} \tag{6.8}$$

Now, to define $\Xi^{\dagger}$ we are required to provide a 2-cell $\Xi^{\dagger}_{X} : \mathsf{k}_X \to (\Lambda \mathsf{j})_X$ for every $X \in \mathcal{B}$, subject to the modification axiom. Since k_X and $(\Lambda\mathsf{j})_X$ are functors $RX \to [P, Q]X$, such a natural transformation consists of a family of 1-cells (modifications) $\Xi^{\dagger}(X, r, -) : \mathsf{k}(X, r, -) \to (\Lambda\mathsf{j})(X, r, -)$ that is natural in r. We build this data in stages.

Fix $X \in \mathcal{B}$ and $r \in RX$. We begin by defining the modifications $\Xi^{\dagger}(X, r, -)$. For the components, we define a natural transformation $\Xi^{\dagger}(X, r, B) : \mathsf{k}(X, r, B) \Rightarrow (\Lambda\mathsf{j})(X, r, B)$ for each $B \in \mathcal{B}$ as follows. For $(h, p) \in \mathcal{B}(X, B) \times PB$, we take the 1-cell defined by commutativity of the diagram below, where the bottom arrow arises from the fact that each $\overline{\mathsf{k}}_f$ is a modification with type given in (6.1):

$$\begin{tikzcd}
\mathsf{k}(X, r, B)(h, p) \arrow[r, "\Xi^{\dagger}(X,r,B)(h,p)"] \arrow[d, equal] & \mathsf{j}\big(B, (Rh)(r), p\big) \\
\mathsf{k}(X, r, B)(\mathrm{Id}_B \circ h, p) \arrow[r, "\overline{\mathsf{k}}(h,r,B)(\mathrm{Id}_B,p)^{-1}"'] & \mathsf{k}\big(B, (Rh)(r), B\big)(\mathrm{Id}_B, p) \arrow[u, "\Xi(B,(Rh)(r),p)"']
\end{tikzcd} \tag{6.9}$$

The family of 1-cells thus defined is natural in (h, p) because each component is. We claim that the family of natural transformations $\Xi^{\dagger}(X, r, -)$ is a modification. This entails checking that the following commutes for every $f : B \to B'$ in $\mathcal{B}$:

$$\begin{tikzcd}
\mathsf{k}(X, r, B) \circ \big(\mathcal{B}(X, f) \times Pf\big) \arrow[r, "\Xi^{\dagger}(X,r,B)\circ(\mathcal{B}(X,f)\times Pf)"] \arrow[d, "\overline{\mathsf{k}}(X,r,f)"'] & (\Lambda\mathsf{j})(X, r, B) \circ \big(\mathcal{B}(X, f) \times Pf\big) \arrow[d, "\overline{(\Lambda\mathsf{j})}(X,r,f)"] \\
(Qf)\big(\mathsf{k}(X, r, B)\big) \arrow[r, "(Qf)(\Xi^{\dagger}(X,r,B))"'] & (\Lambda\mathsf{j})(X, r, B)
\end{tikzcd}$$

To prove this, fix some $(h, p) \in \mathcal{B}(X, B) \times PB$. Applying the naturality of Ξ with respect to the map $\phi^{R}_{f,h}(r) : (Rf)(Rh)(r) \to R(f \circ h)(r)$, and the modification axiom (6.8), one reduces the claim to showing that

$$\begin{tikzcd}
 & \mathsf{k}(X,r,B')(\mathrm{Id}_{B'}\circ f\circ h,(Pf)(p)) \arrow[dr, equal] & \\
\mathsf{k}(B',R(fh)(r),B')(\mathrm{Id}_{B'},(Pf)(p)) \arrow[ur, "\overline{\mathsf{k}}(f\circ h,r)(\mathrm{Id}_{B'},(Pf)(p))"] & & \mathsf{k}(X,r,B')(f\circ h,(Pf)(p)) \arrow[dd, "\overline{\mathsf{k}}(X,r,f)(h,p)"] \\
\mathsf{k}(B',(Rf)(Rh)(r),B')(\mathrm{Id}_{B'},(Pf)(p)) \arrow[u, "\mathsf{k}\big(B',\phi^R_{f,h}(r),B'\big)(\mathrm{Id}_{B'},(Pf)(p)\big)"] \arrow[d, "\overline{\mathsf{k}}(B,R(h)(r),f)(\mathrm{Id}_{B'},(Pf)(p))"'] & & \\
\mathsf{k}(B,(Rh)(r),B')(\mathrm{Id}_{B'}\circ f,(Pf)(p)) \arrow[d, equal] & & (Qf)\big(\mathsf{k}(X,r,B)(h,p)\big) \arrow[d, equal] \\
\mathsf{k}(B,(Rh)(r),B')(f\circ\mathrm{Id}_B,(Pf)(p)) \arrow[dr, "\overline{\mathsf{k}}(B,(Rh)(r),f)(\mathrm{Id}_B,p)"'] & & (Qf)\big(\mathsf{k}(X,r,B)(\mathrm{Id}_B\circ h,p)\big) \\
 & (Qf)\big(\mathsf{k}(B',(Rh)(r),B')(\mathrm{Id}_B,p)\big) \arrow[ur, "(Qf)(\overline{\mathsf{k}}(h,r)(\mathrm{Id}_B,p))"'] &
\end{tikzcd}$$

This commutes by an application of the associativity law for R and the modification axiom (6.2) for $\overline{\mathsf{k}}(f,r)$.

Thus, $\Xi^\dagger(X,r)$ is a modification $\big(\mathsf{k}(X,r,-),\overline{\mathsf{k}}(X,r,-)\big)\to\big((\Lambda\mathsf{j})(X,r,-),\overline{(\Lambda\mathsf{j})}(X,r,-)\big)$ for every $X\in\mathcal{B}$ and $r\in RX$. Moreover, since each of the components in the definition of $\Xi^\dagger(X,r)$ is natural in r, this r-indexed family of 1-cells forms a natural transformation $\Xi^\dagger_X:\mathsf{k}_X\Rightarrow(\Lambda\bar{\mathsf{j}})_X$.

To show that $\Xi^\dagger$ is a modification $(\mathsf{k},\overline{\mathsf{k}})\to(\Lambda\mathsf{j},\overline{\Lambda\mathsf{j}})$, it remains to check the following modification law for every $f:X\to X'$ and $(h,p)\in\mathcal{B}(X',B)\times PB$:

$$\begin{tikzcd}
\mathsf{k}\big(X',(Rf)(r),B\big)(h,p) \arrow[r, "\overline{\mathsf{k}}(f,r)"] \arrow[d, "\Xi^\dagger(X,(Rf)(r),B)(h,p)"'] & \mathsf{k}\big(X,r,B\big)(h\circ f,p) \arrow[d, "\Xi^\dagger(X,r,B)(hf,p)"] \\
(\Lambda\mathsf{j})\big(X',(Rf)(r),B\big)(h,p) \arrow[r, "(\Lambda\bar{\mathsf{j}})(f)"'] & (\Lambda\mathsf{j})\big(X,r,B\big)(h\circ f,p)
\end{tikzcd} \tag{6.10}$$

This follows from the associativity law for $\mathrm{eval}_{P,Q}\circ\big((\mathsf{k},\overline{\mathsf{k}})\times P\big)$, namely

$$\begin{tikzcd}
\mathsf{k}\big(B,(Rh)(Rf)(r),B\big)(\mathrm{Id}_B,p) \arrow[rr, "\mathsf{k}\big(B,\phi^R_{h,f}(r),B\big)(\mathrm{Id}_B,p)"] \arrow[d, "\overline{\mathsf{k}}(h,(Rf)(r))(\mathrm{Id}_B,p)"'] & & \mathsf{k}\big(B,R(hf)(r),B\big)(\mathrm{Id}_B,p) \arrow[dd, "\overline{\mathsf{k}}(h\circ f,r)(\mathrm{Id}_B,p)"] \\
\mathsf{k}\big(X',(Rf)(r),B\big)(\mathrm{Id}_B\circ h,p) \arrow[d, equal] & & \\
\mathsf{k}\big(X',(Rf)(r),B\big)(h,p) \arrow[r, "\overline{\mathsf{k}}(f,r)(h,p)"'] & \mathsf{k}\big(X,r,B\big)(h\circ f,p) \arrow[r, equal] & \mathsf{k}\big(X,r,B\big)(\mathrm{Id}_B\circ h\circ f,p)
\end{tikzcd}$$

together with the naturality of Ξ_X with respect to the morphism $\phi^R_{h,f}(r) : (Rh)(Rf)(r) \to R(hf)(r)$. We summarise the result:

Lemma 6.1.7. The family of natural transformations $\Xi^\dagger(X, -, =)$ defined in (6.9) forms a modification $(\mathsf{k}, \overline{\mathsf{k}}) \to (\Lambda\mathsf{j}, \overline{\Lambda\mathsf{j}})$. □

The final part of the proof is showing that $\Xi^\dagger$ is the unique modification Ψ such that

$$\begin{tikzcd}
\mathrm{eval}_{P,Q} \circ \big((\mathsf{k}, \overline{\mathsf{k}}) \times P\big) \arrow[rr, "\mathrm{eval}_{P,Q} \circ (\Psi \times P)"] \arrow[dr, "\Xi"'] & & \mathrm{eval}_{P,Q} \circ \big(\Lambda(\mathsf{j}, \bar{\mathsf{j}}) \times P\big) \arrow[dl, "\mathrm{E}"] \\
& (\mathsf{j}, \bar{\mathsf{j}}) &
\end{tikzcd} \tag{6.11}$$

We turn to this next.

The universal property of E. The existence part of the claim follows from the unit law of a pseudonatural transformation and the fact that $\Xi(X, r, p)$ is a natural transformation:

$$\begin{tikzcd}
\mathsf{k}(X, r, X)(\mathrm{Id}_X, p) \arrow[dr, equal] \arrow[dd, "\Xi^\dagger(X,r,X)(\mathrm{Id}_X,p)"'] \arrow[ddr, phantom, "\overset{\mathrm{def}}{=}"] & & \\
& \mathsf{k}(X, r, X)(\mathrm{Id}_X \circ \mathrm{Id}_X, p) \arrow[d, "\bar{\mathsf{k}}(\mathrm{Id}_X,r)(\mathrm{Id}_X,p)^{-1}"] \arrow[dd, equal, bend left=60] & \\
\mathsf{j}\big(X, R(\mathrm{Id}_X)(r), p\big) \arrow[dd, "\mathsf{j}\left(X,(\psi^R_X)^{-1}(r),p\right)"'] \arrow[dr, phantom, "\overset{\mathrm{nat}}{=}"] & \mathsf{k}\big(X, R(\mathrm{Id}_X)(r), X\big)(\mathrm{Id}_X, p) \arrow[l, "\Xi(X,R(\mathrm{Id}_X)(r),p)"] \arrow[d, "\mathsf{k}\left(X,(\psi^R_X)^{-1}(r),X\right)(\mathrm{Id}_X,p)"] & \overset{\text{unit law}}{=} \\
& \mathsf{k}(X, r, X)(\mathrm{Id}_X, p) \arrow[dl, "\Xi(X,r,p)"] & \\
\mathsf{j}(X, r, p) & &
\end{tikzcd}$$

For uniqueness, suppose that Ψ is a modification filling (6.11). Then, applying the definition of $(\Lambda\mathsf{j})(f, -, =)$ from (6.4), one obtains the diagram below, in which one uses the modification axiom $\big($*c.f.* (6.10)$\big)$, the assumption on Ψ and the unit law of a pseudofunctor:

$$\begin{tikzcd}
\mathsf{k}(X,r,B)(h,p) \arrow[d, equal] \arrow[dr, bend left, "\Psi(X,r,B)(h,p)"] \arrow[dr, phantom, "="] & \\
\mathsf{k}(X,r,B)(\mathrm{Id}_B \circ h, p) \arrow[r, "\Psi(X,r,B)(\mathrm{Id}_B \circ h,p)"] \arrow[d, "\bar{\mathsf{k}}(h,r)(\mathrm{Id}_B,p)^{-1}"'] \arrow[dr, phantom, "\overset{\text{modif. law}}{=}"] & \mathsf{j}\big(B, R(\mathrm{Id}_B \circ h)(r), p\big) \arrow[d, "\mathsf{j}\left(B,(\phi^R_{\mathrm{Id},h})^{-1}(r),p\right)"] \arrow[dd, equal, bend left=80, "\overset{\text{unit law}}{=}"] \\
\mathsf{k}\big(B,(Rh)(r),B\big)(\mathrm{Id}_B,p) \arrow[r, "\Psi(B,(Rh)(r),B)(\mathrm{Id}_B,p)"] \arrow[dr, bend right, "\Xi(B,(Rh)(r),B)(\mathrm{Id}_B,p)"'] \arrow[dr, phantom, "\overset{(6.11)}{=}"] & \mathsf{j}\big(B,(R\mathrm{Id}_B)(Rh)(r),p\big) \arrow[d, "\mathsf{j}\left(B,(\psi^R_B)^{-1}(Rh)(r),p\right)"] \\
 & \mathsf{j}\big(B,(Rh)(r),p\big)
\end{tikzcd}$$

Since the left-hand leg of this diagram is the definition of $\Xi^\dagger$ (6.9), one obtains the required universal property:

Lemma 6.1.8. For any modification $\Xi : \mathrm{eval}_{P,Q} \circ \big((\mathsf{k}, \bar{\mathsf{k}}) \times P\big) \to (\mathsf{j}, \bar{\mathsf{j}})$ the modification $\Xi^\dagger$ of Lemma 6.1.7 is the unique such filling (6.11). □

Putting together everything we have seen in this section, for every $P, Q : \mathcal{B} \to \mathbf{Cat}$ the pseudofunctor $[P, Q] := \mathrm{Hom}(\mathcal{B}, \mathbf{Cat})\big(\mathrm{Y}(-) \times P, Q\big)$ satisfies an adjoint equivalence

$$\Lambda : \big(\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(R \times P, Q)\big) \leftrightarrows \big(\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(R, [P, Q])\big) : \mathrm{eval}_{P,Q} \circ (- \times P)$$

with evaluation map defined as in Lemma 6.1.3 and counit E defined as in Lemma 6.1.6. The universality of the counit is witnessed by the mapping $(-)^\dagger$ of Lemma 6.1.7. Moreover, it is clear that $\Xi^\dagger$ is invertible if Ξ is, so in particular the unit is invertible. Thus, $[P, Q]$ is an exponential in $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$.

Proposition 6.1.9. For any 2-category $\mathcal{B}$ and pseudofunctors $P, Q : \mathcal{B} \to \mathbf{Cat}$, the exponential $[P, Q]$ exists and may be given by $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})\big(\mathrm{Y}(-) \times P, Q\big)$. □

Hence, $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$ is cartesian closed for any 2-category $\mathcal{B}$. Applying Lemma 6.1.1 yields our final result.

Theorem 6.1.10. For any bicategory $\mathcal{B}$, the 2-category $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$ is cartesian closed. □

6.2 Exponentiating by a representable

For any 2-category $\mathcal{B}$ with pseudo-products, object $X \in \mathcal{B}$ and pseudofunctor $P : \mathcal{B}^{\mathrm{op}} \to \mathbf{Cat}$, the exponential $[\mathrm{Y}X, P]$ may be given as $P(- \times X)$. This follows immediately from the the uniqueness of exponentials up to equivalence (Remark 5.1.4), together with the following chain of equivalences:

$$\begin{aligned}
[\mathrm{Y}X, P] &\simeq \mathrm{Hom}(\mathcal{B}, \mathbf{Cat})\big(\mathrm{Y}(-) \times \mathrm{Y}X, P\big) && \text{by Proposition 6.1.9} \\
&\simeq \mathrm{Hom}(\mathcal{B}, \mathbf{Cat})\big(\mathrm{Y}(- \times X), P\big) && \\
&\simeq P(- \times X) && \text{by the Yoneda Lemma}
\end{aligned} \tag{6.12}$$

For the second line we use the fact that birepresentables preserve bilimits (Lemma 2.3.4).

In the normalisation-by-evaluation argument (Chapter 8) we shall require an explicit description of the evaluation map witnessing $P(- \times X)$ as the exponential $[\mathrm{Y}X, P]$. In this section, therefore, we outline the exponential structure of $P(- \times X)$ and briefly show that it satisfies the required universal property. Since this structure may be extracted from the work of the preceding section by chasing through the equivalences (6.12), our presentation will be less detailed than before.

Note that, for the rest of this chapter, we work contravariantly. Since we are assuming $\mathcal{B}$ is a 2-category, the Yoneda pseudofunctor is now both strict (in fact, a 2-functor) and contravariant: $\mathrm{Y}X = \mathcal{B}^{\mathrm{op}}(X, -) = \mathcal{B}(-, X)$.

The evaluation map. We begin with the pseudonatural transformation $P(- \times X) \times \mathrm{Y}X \Rightarrow P$ that will act as the evaluation map. For the component at $B \in \mathcal{B}$ we take the functor

$$\begin{aligned}
e_B : P(B \times X) \times \mathcal{B}(B, X) &\to PB \\
(p, h) &\mapsto P(\langle \mathrm{Id}_B, h \rangle)(p)
\end{aligned}$$

with the evident action on 2-cells. To turn this into a pseudonatural transformation we need to provide an invertible 2-cell $\overline{e}_f$ as in the diagram below for every $f : B' \to B$ in $\mathcal{B}$:

$$\begin{tikzcd}
P(B \times X) \times \mathcal{B}(B, X) \arrow[r, "P(f \times X) \times \mathcal{B}(f, X)"] \arrow[d, "e_B"'] \arrow[dr, phantom, "\overset{\overline{e}_f}{\Leftarrow}"] & P(B' \times X) \times \mathcal{B}(B', X) \arrow[d, "e_{B'}"] \\
PB \arrow[r, "Pf"'] & PB'
\end{tikzcd}$$

At $h : B \to X$ we define $\overline{e}_f(h, -)$ to be the composite

$$\begin{tikzcd}
P(\langle \mathrm{Id}_B, h \circ f \rangle) \circ P(f \times X) \arrow[r, "\overline{e}_f(h{,}-)"] \arrow[d, "\phi^P_{\langle \mathrm{Id}, hf \rangle, f \times X}"'] & P(f) \circ P\langle \mathrm{Id}_B, h \rangle \\
P\big((f \times X)\langle \mathrm{Id}_B, hf \rangle\big) \arrow[r, "P\mathsf{swap}_{h,f}"'] & P(\langle \mathrm{Id}_{B'}, h \rangle \circ f) \arrow[u, "(\phi^P_{\langle \mathrm{Id}, h \rangle, f})^{-1}"']
\end{tikzcd}$$

where the isomorphism $\mathsf{swap}_{h,f}$ is $(f \times X) \circ \langle \mathrm{Id}_B, hf \rangle \xRightarrow{\mathsf{fuse}} \langle f, hf \rangle \xRightarrow{\mathsf{post}^{-1}} \langle \mathrm{Id}_{B'}, h \rangle \circ f$. The whole composite is a natural isomorphism because each component is, so it remains to check the two axioms of a pseudonatural transformation. The unit law is a short diagram chase using the unit law for P and the fact that

$$\mathrm{Id}_{B \times X} \circ \langle \mathrm{Id}_B, h \rangle \xRightarrow{\varsigma_{\mathrm{Id}} \circ \langle \mathrm{Id}, h \rangle} \langle \mathrm{Id}_B, h \rangle \circ \mathrm{Id}_B \xRightarrow{\mathsf{swap}} \mathrm{Id}_{B \times X} \circ \langle \mathrm{Id}_B, h \rangle$$

is the identity.

To prove the associativity law, on the other hand, one uses the naturality of the ϕ^P 2-cells and the associativity law of a pseudofunctor to reduce the problem to a diagram in the image of P, whereupon one can apply standard properties of the product structure (recall Lemma 4.1.7).

Lemma 6.2.1. For any $X \in \mathcal{B}$ and pseudofunctor $P : \mathcal{B}^{\mathrm{op}} \to \mathbf{Cat}$, the pair $(e, \overline{e})$ defined above forms a pseudonatural transformation $P(- \times X) \times \mathrm{Y}X \Rightarrow P$. □

The mapping Λ. Next we define the mapping $\Lambda : ob\big(\mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})(R \times \mathrm{Y}X, P)\big) \to ob\big(\mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})(R, P$ Let $(\mathsf{k}, \overline{\mathsf{k}}) : R \times \mathrm{Y}X \Rightarrow P$ be a pseudonatural transformation. We define $\Lambda(\mathsf{k}, \overline{\mathsf{k}}) := (\Lambda\mathsf{k}, \overline{\Lambda\mathsf{k}}) : R \Rightarrow P(- \times X)$ as follows. For $B \in \mathcal{B}$ we take the functor

$$\begin{aligned} (\Lambda\mathsf{k})_B : RB &\to P(B \times X) \\ r &\mapsto \mathsf{k}_{B \times X}\big(R(\pi_1)(r), \pi_2\big) \end{aligned}$$

Thus, $(\Lambda\mathsf{k})_B$ is the composite $RB \xrightarrow{R\pi_1} R(B \times X) \xrightarrow{\mathsf{k}_{B \times X}(-, \pi_2)} P(B \times X)$. To define $\overline{(\Lambda\mathsf{k})}_f$, where $f : B' \to B$, we need to give an invertible 2-cell as in

$$\begin{tikzcd}
RB \arrow[r, "Rf"] \arrow[d, "(\Lambda\mathsf{k})_B"'] \arrow[dr, phantom, "\overset{\overline{(\Lambda\mathsf{k})}_f}{\Leftarrow}"] & RB' \arrow[d, "(\Lambda\mathsf{k})_{B'}"] \\
P(B \times X) \arrow[r, "P(f \times X)"'] & P(B' \times X)
\end{tikzcd}$$

This must be a natural isomorphism $\mathsf{k}_{B' \times X}\big(R(\pi_1)R(f)(-), \pi_2\big) \overset{\cong}{\Rrightarrow} P(f \times X)\big(\mathsf{k}_{B \times X}(R(\pi_1)(-), \pi_2)\big)$, for which we take the following composite:

$$\begin{tikzcd}
\mathsf{k}_{B' \times X}(R(\pi_1) \circ R(f), \pi_2) \arrow[rr, "(\overline{\Lambda\mathsf{k}})_f"] \arrow[d, "\mathsf{k}_{B' \times X}(\phi^R_{f, \pi_1}, \pi_2)"'] & & P(f \times X)\big(\mathsf{k}_{B \times X}(R\pi_1, \pi_2)\big) \\
\mathsf{k}_{B' \times X}\big(R(f \circ \pi_1), \pi_2\big) \arrow[d, "\mathsf{k}_{B' \times X}(R\varpi^{(-1)}, \varpi^{(-2)})"'] & & \\
\mathsf{k}_{B' \times X}\big(R\left(\pi_1(f \times X)\right), \pi_2(f \times X)\big) \arrow[rr, "\mathsf{k}_{B' \times X}((\phi^R_{\pi_1, f \times X})^{-1}, \pi_2(f \times X))"'] & & \mathsf{k}_{B' \times X}\big(R(f \times X) \circ R(\pi_1), \pi_2(f \times X)\big) \arrow[uu, "\overline{\mathsf{k}}_{f \times X}(R\pi_1, \pi_2)"']
\end{tikzcd}$$

To see that this is a pseudonatural transformation, observe that we have actually defined $\Lambda(\mathsf{k}, \overline{\mathsf{k}})$ as a composite

$$\begin{tikzcd}
RB \arrow[r, "Rf"] \arrow[d, "\mathsf{n}_B"'] \arrow[dr, phantom, "\overset{\overline{\mathsf{n}}_f}{\Leftarrow}"] & RB' \arrow[d, "\mathsf{n}_{B'}"] \\
R(B \times X) \times \mathcal{B}(B \times X, X) \arrow[r, "R(f\times X)\times\mathcal{B}(f\times X,X)"'] \arrow[d, "\mathsf{k}_{B\times X}"'] \arrow[dr, phantom, "\overset{\overline{\mathsf{k}}_{f\times X}}{\Leftarrow}"] & R(B' \times X) \times \mathcal{B}(B' \times X, X) \arrow[d, "\mathsf{k}_{B'\times X}"] \\
P(B \times X) \arrow[r, "P(f\times X)"'] & P(B' \times X)
\end{tikzcd} \tag{6.13}$$

where $\mathsf{n}_B(r) := \big(R(\pi_1)(r), \pi_2\big)$ and $\overline{\mathsf{n}}_f$ has first component

$$R\pi_1 \circ Rf \xRightarrow{\phi^R_{f,\pi_1}} R(f \circ \pi_1) \xRightarrow{R\varpi^{(-1)}} R\big(\pi_1 \circ (f \times X)\big) \xRightarrow{(\phi^R_{\pi_1, f\times X})^{-1}} R(f \times X) \circ R\pi_1 \tag{6.14}$$

and second component $\pi_2 \xRightarrow{\varpi^{(-2)}} \pi_2 \circ (f \times X)$. So it suffices to show that $(\mathsf{n}, \overline{\mathsf{n}})$ defines a pseudonatural transformation $R \Rightarrow R(- \times X) \times \mathcal{B}(- \times X, X)$. Naturality follows immediately from the fact each component in the definition is natural. For the unit law, the first component is the triangle law for products, and the second component is a short diagram chase.

For the associativity law, it is once again the second component that is more difficult. As for $(e, \overline{e})$ (Lemma 6.2.1), the proof consists of using the associativity axiom of a pseudofunctor and the naturality of ϕ^R. Once the calculation has been pushed 'inside' R, what remains is a relatively easy diagram chase. This completes the proof that $(\mathsf{n}, \overline{\mathsf{n}})$ is a pseudonatural transformation, and hence the definition of the mapping Λ.

Lemma 6.2.2. The pair $(\mathsf{n}, \overline{\mathsf{n}})$ defined in (6.14) forms a pseudonatural transformation $R \Rightarrow R(- \times X) \times \mathcal{B}(- \times X, X)$. □

Corollary 6.2.3. The pair $(\Lambda\mathsf{k}, \overline{\Lambda\mathsf{k}})$ defined in (6.13) forms a pseudonatural transformation $R \Rightarrow P(- \times X)$ for every $(\mathsf{k}, \overline{\mathsf{k}}) : R \times \mathrm{Y}X \Rightarrow P$. □

The counit E. For every $(\mathsf{k}, \overline{\mathsf{k}}) : R \times \mathrm{Y}X \Rightarrow P$ we need to provide an invertible modification $\mathrm{E}^{(\mathsf{k},\overline{\mathsf{k}})} : (e, \overline{e}) \circ \big(\Lambda(\mathsf{k}, \overline{\mathsf{k}}) \times \mathrm{Y}X\big) \to (\mathsf{k}, \overline{\mathsf{k}})$.

Unwrapping the definition of $(e, \overline{e}) \circ \big(\Lambda(\mathsf{k}, \overline{\mathsf{k}}) \times \mathrm{Y}X\big)$ at $B \in \mathcal{B}$ and $(r, h) \in RB \times \mathcal{B}(B, X)$, one sees that

$$\begin{aligned}
\Big(e_B \circ \big((\Lambda\mathsf{k})_B \times \mathrm{Y}X\big)\Big)(r, h) &= e_B\big(\mathsf{k}_{B\times X}(R(\pi_1)(r), \pi_2), h\big) \\
&= P\left(\langle \mathrm{Id}_B, h\rangle\right)\big(\mathsf{k}_{B\times X}(R(\pi_1)(r), \pi_2)\big)
\end{aligned}$$

Furthermore, for $f : B' \to B$ the corresponding 2-cell $\overline{\big(e_B \circ \big((\Lambda\mathsf{k})_B \times \mathrm{Y}X\big)\big)_f}$ is defined by

$$\begin{tikzcd}
P(\langle \mathrm{Id}_B, hf\rangle)\big(\mathsf{k}_{B'\times X}(R(\pi_1)R(f)(r), \pi_2)\big) \arrow[r, "\overline{(e_B\circ((\Lambda\mathsf{k})_B\times\mathrm{Y}X))}_f(r,h)"] \arrow[d, "P(\langle\mathrm{Id}_B,hf\rangle)(\bar{\mathsf{j}}_f(r))"'] & P(f)P(\langle \mathrm{Id}_B, h\rangle)\big(\mathsf{k}_{B\times X}(R(\pi_1)(r), \pi_2)\big) \\
P(\langle \mathrm{Id}_B, hf\rangle)\big(\mathsf{k}_{B\times X}(R(f\times X)R(\pi_1)(r), \pi_2(f\times X)\big) \arrow[r, "P(\langle\mathrm{Id}_B,hf\rangle)(\overline{\mathsf{k}}_{f\times X}(R(\pi_1)(r),\pi_2))"'] & P(\langle \mathrm{Id}_B, hf\rangle)P(f\times X)\big(\mathsf{k}_{B\times X}(R(\pi_1)(r), \pi_2)\big) \arrow[u, "\overline{e}_f(h,\mathsf{k}_{B\times X}(R(\pi_1)(r),\pi_2))"']
\end{tikzcd}$$

We therefore take the component at $B \in \mathcal{B}$ of $\mathrm{E}_B^{(\mathsf{k},\overline{\mathsf{k}})}$ to be the natural isomorphism defined by

$$\begin{tikzcd}
P(\langle \mathrm{Id}_B, h\rangle)\big(\mathsf{k}_{B'\times X}(R(\pi_1)(r), \pi_2)\big) \arrow[rr, "\mathrm{E}_B^{(\mathsf{k},\overline{\mathsf{k}})}(r,h)"] \arrow[d, "\overline{\mathsf{k}}^{-1}_{\langle\mathrm{Id},h\rangle}(R(\pi_1)(r),\pi_2)"'] & & \mathsf{k}_B(r, h) \\
\mathsf{k}_B\big(R(\langle\mathrm{Id}_B, h\rangle)R(\pi_1)(r), \pi_2\langle\mathrm{Id}_B, h\rangle\big) \arrow[r, "\mathsf{k}_B(\phi^R_{\pi_1,\langle\mathrm{Id},h\rangle}(r),\varpi^{(2)})"'] & \mathsf{k}_B\big(R(\pi_1\langle\mathrm{Id}_B, h\rangle)(r), h\big) \arrow[r, "\mathsf{k}_B(R\varpi^{(1)},h)"'] & \mathsf{k}_B\big(R(\mathrm{Id}_B)(r), h\big) \arrow[u, "\mathsf{k}_B((\psi^R_B)^{-1},h)"']
\end{tikzcd} \tag{6.15}$$

We need to check the B-indexed family of 2-cells $\mathrm{E}^{(\mathsf{k},\overline{\mathsf{k}})}$ satisfies the modification axiom, namely that

$$\begin{tikzcd}
P(\langle \mathrm{Id}_B, hf\rangle)\big(\mathsf{k}_{B\times X}(R(\pi_1)R(f)(r), \pi_2)\big) \arrow[r, "\mathrm{E}_B^{(\mathsf{k},\overline{\mathsf{k}})}(R(\pi_1)R(f)(r),\pi_2)"] \arrow[d, "\overline{\big(e_B\circ\big((\Lambda\mathsf{k})_B\times\mathrm{Y}X\big)\big)}_f(r,h)"'] & \mathsf{k}_B\big(R(f)(r), hf\big) \arrow[d, "\overline{\mathsf{k}}_f(r,h)"] \\
P(f)P(\langle \mathrm{Id}_B, h\rangle)\big(\mathsf{k}_{B\times X}(R(\pi_1)(r), \pi_2)\big) \arrow[r, "P(f)(\mathrm{E}_B^{(\mathsf{k},\overline{\mathsf{k}})}(r,h))"'] & P(f)\big(\mathsf{k}_B(r, h)\big)
\end{tikzcd}$$

Unfolding all the data results in a long exercise in diagram chasing. The second component is relatively straightforward. For the first component, one applies the naturality properties and associativity law of a pseudofunctor to reduce the claim to the following:

$$\begin{tikzcd}
R(\langle\mathrm{Id}_{B'}, hf\rangle)\circ R(\pi_1)\circ R(f) \arrow[r, "\phi^R_{\pi_1,\langle\mathrm{Id},hf\rangle}\circ R(f)"] \arrow[d, "R(\langle\mathrm{Id}_{B'},hf\rangle)\circ\phi^R_{f,\pi_1}"'] & R(\pi_1\langle\mathrm{Id}_{B'}, hf\rangle)\circ R(f) \arrow[r, "R(\varpi^{(1)})\circ R(f)"] & R(\mathrm{Id}_{B'})\circ R(f) \\
R(\langle\mathrm{Id}_{B'}, hf\rangle)\circ R(f\circ\pi_1) \arrow[d, "R(\langle\mathrm{Id}_{B'},hf\rangle)\circ R(\varpi^{(-1)})"'] & & R(f) \arrow[u, "\psi^R_{B'}\circ R(f)"'] \arrow[d, equal] \\
R(\langle\mathrm{Id}_{B'}, hf\rangle)\circ R\big(\pi_1\circ(f\times X)\big) \arrow[d, "R(\langle\mathrm{Id}_{B'},hf\rangle)\circ(\phi^R_{\pi_1,f\times X})^{-1}"'] & & R(\mathrm{Id}_B\circ f) \\
R(\langle\mathrm{Id}_{B'}, hf\rangle)\circ R(f\times X)\circ R(\pi_1) \arrow[d, "\phi^R_{f\times X,\langle\mathrm{Id},hf\rangle}\circ R(\pi_1)"'] & & R(\pi_1\circ\langle\mathrm{Id}_B, h\rangle\circ f) \arrow[u, "R(\varpi^{(1)}\circ f)"'] \\
R\big((f\times X)\circ\langle\mathrm{Id}_{B'}, hf\rangle\big)\circ R(\pi_1) \arrow[r, "R(\mathsf{fuse})\circ R(\pi_1)"'] & R(\langle f, hf\rangle)\circ R(\pi_1) \arrow[r, "R(\mathsf{post}^{-1})\circ R(\pi_1)"'] & R(\langle\mathrm{Id}_B, h\rangle\circ f)\circ R(\pi_1) \arrow[u, "\phi^R_{\pi_1,\langle\mathrm{Id}_B,h\rangle\circ f}"']
\end{tikzcd}$$

The strategy is now familiar: one applies naturality and the associativity law to bring together all the morphisms in the image of R, and then unwraps the definition of post and fuse to reduce the long anticlockwise claim to the top row.

We have therefore constructed a modification to act as the counit.

Lemma 6.2.4. The 2-cells $\mathrm{E}_B^{(\mathsf{k},\bar{\mathsf{k}})}$ $(B \in \mathcal{B})$ defined in (6.15) form an invertible modification $(e, \bar{e}) \circ \big(\Lambda(\mathsf{k}, \bar{\mathsf{k}}) \times \mathrm{Y}X\big) \to (\mathsf{k}, \bar{\mathsf{k}})$. □

All that remains is to show the modification $\mathrm{E}^{(\mathsf{k},\bar{\mathsf{k}})}$ is a universal arrow.

The modification $\Xi^\dagger$. We aim to construct a modification $\Xi^\dagger$ for every pseudonatural transformation $(\mathsf{j}, \bar{\mathsf{j}}) : R \Rightarrow P(- \times X)$ and modification $\Xi : (e, \bar{e}) \circ \big((\mathsf{j}, \bar{\mathsf{j}}) \times \mathrm{Y}X\big) \to (\mathsf{k}, \bar{\mathsf{k}})$, such that $\Xi^\dagger$ is the unique modification filing

$$\begin{tikzcd}
(e, \bar{e}) \circ \big((\mathsf{j}, \bar{\mathsf{j}}) \times \mathrm{Y}X\big) \arrow[rr, "(e,\bar{e})\circ(\Xi^\dagger\times \mathrm{Y}X)"] \arrow[dr, "\Xi"'] & & (e, \bar{e}) \circ \big(\Lambda(\mathsf{k}, \bar{\mathsf{k}}) \times \mathrm{Y}X\big) \arrow[dl, "\mathrm{E}^{(\mathsf{k},\bar{\mathsf{k}})}"] \\
& (\mathsf{k}, \bar{\mathsf{k}}) &
\end{tikzcd} \tag{6.16}$$

Because the definitions of $(e, \bar{e})$, $\Lambda(\mathsf{k}, \bar{\mathsf{k}})$ and $\mathrm{E}^{(\mathsf{k},\bar{\mathsf{k}})}$ are all composites, the proof requires working with a large accumulation of data. Nonetheless the diagram chases—although long—are not especially difficult.

Suppose that $\Xi : (e, \bar{e}) \circ \big((\mathsf{j}, \bar{\mathsf{j}}) \times \mathrm{Y}X\big) \to (\mathsf{k}, \bar{\mathsf{k}})$. Since

$$\big(e_B \circ (\mathsf{j}_B \times \mathrm{Y}X)\big)(r, h) = e_B(\mathsf{j}_B(r), h) = P(\langle \mathrm{Id}_B, h\rangle)(\mathsf{j}_B(r))$$

for every $B \in \mathcal{B}$ we are provided with a natural transformation with components $\Xi_B(r, h) : (P\langle \mathrm{Id}_B, h\rangle)(\mathsf{j}_B(r)) \to \mathsf{k}_B(r, h)$ for $(r, h) \in RB \times \mathcal{B}(B, X)$. We define $\Xi_B^\dagger$ to be the composite

$$\begin{tikzcd}
\mathsf{j}_B \arrow[rr, "\Xi_B^\dagger"] \arrow[d, "\psi^P_{B\times X}\circ \mathsf{j}_B"'] & & \mathsf{k}_{B\times X}(R\pi_1, \pi_2) \\
P(\mathrm{Id}_{B\times X}) \circ \mathsf{j}_B \arrow[d, "P(\varsigma_{\mathrm{Id}})\circ \mathsf{j}_B"'] & & P(\langle \mathrm{Id}_{B\times X}, \pi_2\rangle) \circ \mathsf{j}_{B\times X} \circ R\pi_1 \arrow[u, "\Xi_B(R\pi_1,\pi_2)"'] \\
P(\langle \pi_1, \pi_2\rangle) \circ \mathsf{j}_B \arrow[r, "P(\mathsf{fuse}^{-1})\circ \mathsf{j}_B"'] & P\big((\pi_1 \times X) \circ \langle \mathrm{Id}_{B\times X}, \pi_2\rangle\big) \circ \mathsf{j}_B \arrow[r, "(\phi^P_{\pi_1\times X,\langle \mathrm{Id},\pi_2\rangle})^{-1}\circ \mathsf{j}_B"'] & P(\langle \mathrm{Id}_{B\times X}, \pi_2\rangle) \circ P(\pi_1 \times X) \circ \mathsf{j}_B \arrow[u, "P(\langle \mathrm{Id}_{B\times X},\pi_2\rangle)\circ \bar{\mathsf{j}}_{\pi_1}^{-1}"']
\end{tikzcd} \tag{6.17}$$

and claim this does indeed define a modification. We therefore need to verify the following diagram of functors commutes for every $f : B' \to B$ in $\mathcal{B}$:

$$\begin{tikzcd}
\mathsf{j}_{B'}\big(R(f)\big) \arrow[r, "\Xi^{\dagger}_{B'}(R(f))"] \arrow[d, "\bar{\mathsf{j}}_f"'] & \mathsf{k}_{B'\times X}\big(R(\pi_1)R(f), \pi_2\big) \arrow[d, "(\overline{\Lambda\mathsf{k}})_f"] \\
P(f \times X)\big(\mathsf{j}_B\big) \arrow[r, "P(f\times X)(\Xi^{\dagger}_B)"'] & P(f \times X)\big(\mathsf{k}_{B\times X}(R(\pi_1), \pi_2)\big)
\end{tikzcd}$$

Unfolding all the various composites results in a very large diagram. We give the strategy for proving it commutes. One begins by using naturality until one can apply the modification axiom for Ξ to relate the final term in the composite defining $\overline{(\Lambda\mathsf{k})}_f$ with $P(f \times X)\big(\Xi_{B\times X}(R(\pi_1)(r), \pi_2)\big)$. Next one applies the associativity law for $(\mathsf{j}, \bar{\mathsf{j}})$ in order to push the 2-cells ϕ^P as early as possible. One then observes that the following diagram commutes, and hence that its image under P commutes:

$$\begin{tikzcd}
f \times X \arrow[r, "\varsigma_{\mathrm{Id}}\circ(f\times X)"] \arrow[d, "(f\times X)\circ\varsigma_{\mathrm{Id}}"'] & \langle \pi_1, \pi_2 \rangle \circ (f \times X) \\
(f \times X) \circ \langle \pi_1, \pi_2 \rangle \arrow[d, "(f\times X)\circ\langle\pi_1,\varpi^{(-2)}\rangle"'] & \\
(f \times X) \circ \langle \pi_1, \pi_2(f \times X) \rangle \arrow[d, "(f\times X)\circ\mathsf{fuse}^{-1}"'] & (\pi_1 \times X) \circ \langle \mathrm{Id}_{B\times X}, \pi_2 \rangle \circ (f \times X) \arrow[uu, "\mathsf{fuse}\circ(f\times X)"'] \\
(f \times X) \circ (\pi_1 \times X) \circ \langle \mathrm{Id}_{B'\times X}, \pi_2(f \times X) \rangle \arrow[d, "\Phi_{f,\pi_1;\mathrm{Id}_X}\circ\langle\mathrm{Id},\pi_2(f\times X)\rangle"'] & (\pi_1 \times X) \circ \big((f \times X) \times X\big) \circ \langle \mathrm{Id}_{B'\times X}, \pi_2(f \times X) \rangle \arrow[u, "(\pi_1\times X)\circ\mathsf{swap}"'] \\
\big((f \circ \pi_1) \times X\big) \circ \langle \mathrm{Id}_{B'\times X}, \pi_2(f \times X) \rangle \arrow[r, "(\varpi^{(-1)}\times X)\circ\langle\mathrm{Id},\pi_2(f\times X)\rangle"'] & \big((\pi_1(f \times X)) \times X\big) \circ \langle \mathrm{Id}_{B'\times X}, \pi_2(f \times X) \rangle \arrow[u, "\Phi^{-1}_{\pi_1,f\times X;\mathrm{Id}_X}\circ\langle\mathrm{Id},\pi_2(f\times X)\rangle"']
\end{tikzcd}$$

From this point the rest of the proof is a manageable diagram chase. Hence, $\Xi^{\dagger}$ is a modification.

Lemma 6.2.5. For every modification $\Xi : (e, \overline{e}) \circ \big((\mathsf{j}, \bar{\mathsf{j}}) \times \mathrm{Y}X\big) \to (\mathsf{k}, \overline{\mathsf{k}})$ between pseudonatural transformations $R \times \mathrm{Y}X \Rightarrow P$, the 2-cells $\Xi^{\dagger}_B$ form a modification $(\mathsf{j}, \bar{\mathsf{j}}) \to \Lambda(\mathsf{k}, \overline{\mathsf{k}})$. □

The last part of the proof is checking that $\Xi^{\dagger}$ is the unique modification filling the diagram (6.16).

The universal property of E. The existence and uniqueness parts of (6.16) also entail long but not especially difficult diagram chases. In each case one unfolds the various composites and applies the modification axiom for Ξ. The rest of the proof is an exercise in applying the various naturality properties and the two laws of a pseudofunctor.

Putting together all the work of this section, one obtains the following.

Proposition 6.2.6. For any 2-category $\mathcal{B}$ with pseudo-products, pseudofunctor $P : \mathcal{B}^{\mathrm{op}} \to \mathbf{Cat}$ and object $X \in \mathcal{B}$, the modification E of Lemma 6.2.4 is the counit of an adjoint equivalence

$$\Lambda : \mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})(R \times \mathrm{Y}X, P) \leftrightarrows \mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})\big(R, P(- \times X)\big) : (e, \overline{e}) \circ (- \times \mathrm{Y}X)$$

in which the pseudonatural transformation $(e, \overline{e})$ and mapping Λ are as in Lemma 6.2.1 and Corollary 6.2.3, respectively. □

Theorem 6.2.7. For any 2-category $\mathcal{B}$ with pseudo-products, pseudofunctor $P : \mathcal{B}^{\mathrm{op}} \to \mathbf{Cat}$ and object $X \in \mathcal{B}$, the pseudofunctor $P(- \times X)$ is (up to equivalence) the exponential $[\mathrm{Y}X, P]$ in $\mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$. □

Setting $\mathcal{C} := \mathcal{B}^{\mathrm{op}}$ recovers the covariant statement.

Chapter 7

Bicategorical glueing

Glueing is a powerful technique which may be used to leverage semantic arguments in order to prove syntactic results. Intuitively, one 'glues together' syntactic and semantic information, allowing one to extract proofs of syntactic properties from semantic arguments. The breadth and utility of this approach has led to its being discovered in various forms, with correspondingly various names: the notions of logical relation [Plo73, Sta85], sconing [FS90], Freyd covers and glueing (*e.g.* [LS86]) are all closely related (see *e.g.* [MS93] for an overview of the connections). Taylor identifies the basic apparatus as going back to Groethendieck [Tay99, Section 7.7], while versions of logical relations appear as early as Gandy's thesis (who, in turn, attributes some of the theory to Turing) [Gan53]. Originally presented in the set-theoretic setting, the technique was quickly given categorical expression [MR92, MS93], for which Hermida provided an account in terms of fibrations in his thesis [Her93]. Such techniques are now a standard component of the armoury for studying type theories.

In this chapter we define a notion of glueing for bicategories and prove a bicategorical version of the fundamental result establishing mild conditions for the glueing category to be cartesian closed. (For reference, the construction is summarised in the appendix on page 310.) This will form the core of our normalisation-by-evaluation proof in the next chapter.

We begin by recalling the categorical glueing construction and giving a precise statement of the cartesian closure result we wish to prove. These will provide a template for our bicategorical work.

7.1 Categorical glueing

The most succinct description of categorical glueing is as a special kind of comma category.

Definition 7.1.1.

1. Let $F : \mathbb{A} \to \mathbb{C}$ and $G : \mathbb{B} \to \mathbb{C}$ be functors. The *comma category* $(F \downarrow G)$ has objects triples (A, f, B), where $A \in \mathbb{A}$ and $B \in \mathbb{B}$ are objects and $f : FA \to GB$ is a morphism in

$\mathbb{C}$. Morphisms $(A, f, B) \to (A', f', B')$ are pairs of morphisms (p, q) such that the following square commutes:

$$\begin{tikzcd}
FA \arrow[r, "Fp"] \arrow[d, "f"'] & FA' \arrow[d, "f'"] \\
GB \arrow[r, "Gq"'] & GB'
\end{tikzcd} \tag{7.1}$$

2. The *glueing* $\mathrm{gl}(\mathfrak{J})$ of $\mathbb{B}$ to $\mathbb{C}$ along a functor $\mathfrak{J} : \mathbb{B} \to \mathbb{C}$ is the comma category $(\mathrm{id}_{\mathbb{C}} \downarrow \mathfrak{J})$. We denote the objects and morphisms following the vertical order of their appearance in diagram (7.1), as $(C \in \mathbb{C}, c : C \to \mathfrak{J}B, B \in \mathbb{B})$ and $(q : C \to C', p : B \to B')$. ◂

There are evident *projection functors* $\mathbb{B} \xleftarrow{\pi_{\mathrm{dom}}} \mathrm{gl}(\mathfrak{J}) \xrightarrow{\pi_{\mathrm{cod}}} \mathbb{C}$. We wish to bicategorify the following folklore result (*c.f.* [MR92, Proposition 2]):

Proposition 7.1.2. Let $\mathfrak{J} : \mathbb{B} \to \mathbb{C}$ be a functor between cartesian closed categories, such that $\mathfrak{J}$ preserves products and $\mathbb{C}$ has all pullbacks. Then the glueing category $\mathrm{gl}(\mathfrak{J})$ is cartesian closed, and the projection π_{dom} strictly preserves the cartesian closed structure.

Proof. For $n \in \mathbb{N}$ the n-ary product of objects (C_i, c_i, B_i) $(i = 1, \dots, n)$ is the composite

$$\textstyle\prod_{i=1}^n C_i \xrightarrow{\prod_i c_i} \prod_{i=1}^n (\mathfrak{J}B_i) \xrightarrow{\cong} \mathfrak{J}\big(\prod_{i=1}^n B_i\big)$$

Projections are given pointwise, as $(\pi_i^{\mathbb{C}}, \pi_i^{\mathbb{B}})$, and the n-ary tupling of a family of 1-cells $(f_i, g_i) : (X, x, Y) \to (C_i, c_i, B_i)$ $(i = 1, \dots, n)$ is the pair $(\langle f_1, \dots, f_n\rangle, \langle g_1, \dots, g_n\rangle)$. Hence both π_{dom} and π_{cod} strictly preserve products.

The exponential $(C, c, B) \Rightarrow (C', c', B')$ is defined to be the left-hand vertical map in the pullback diagram

$$\begin{tikzcd}
C \supset C' \arrow[rr, "q_{c,c'}"] \arrow[d, "p_{c,c'}"'] \arrow[dr, phantom, "\lrcorner", very near start] & & (C \Rightarrow C') \arrow[d, "C \Rightarrow c'"] \\
\mathfrak{J}(B \Rightarrow B') \arrow[r, "\mathrm{m}_{B,B'}"'] & (\mathfrak{J}B \Rightarrow \mathfrak{J}B') \arrow[r, "(c \Rightarrow \mathfrak{J}B')"'] & (C \Rightarrow \mathfrak{J}B')
\end{tikzcd} \tag{7.2}$$

where $\mathrm{m}_{B,B'}$ is the exponential transpose of $\big(\mathfrak{J}(B \Rightarrow B') \times \mathfrak{J}B \xrightarrow{\cong} \mathfrak{J}((B \Rightarrow B') \times B) \xrightarrow{\mathfrak{J}\mathrm{eval}_{B,B'}} \mathfrak{J}B'\big)$. The evaluation map has first component $(C \supset C') \times C \xrightarrow{q_{c,c'} \times C} (C \Rightarrow C') \times C \xrightarrow{\mathrm{eval}_{C,C'}} C'$ and second component simply $\mathrm{eval}_{B,B'}$. The currying operation is given by the universal property of pullbacks. □

The rest of the chapter is dedicated to proving a bicategorical version of this proposition.

7.2 Bicategorical glueing

We bicategorify Definition 7.1.1 in the usual way: by replacing commuting squares with invertible 2-cells, subject to coherence conditions.

Definition 7.2.1. Let $F : \mathcal{A} \to \mathcal{C}$ and $G : \mathcal{B} \to \mathcal{C}$ be pseudofunctors of bicategories. The *comma bicategory* $(F \downarrow G)$ has objects triples $(A \in \mathcal{A}, f : FA \to GB, B \in \mathcal{B})$. The 1-cells $(A, f, B) \to (A', f', B')$ are triples (p, α, q), where $p : A \to A'$ and $q : B \to B'$ are 1-cells and α is an invertible 2-cell $\alpha : f' \circ Fp \Rightarrow Gq \circ f$ witnessing the commutativity of (7.1):

$$\begin{tikzcd}
FA \arrow[r, "Fp"] \arrow[d, "f"'] \arrow[dr, phantom, "\overset{\alpha}{\Leftarrow}"] & FA' \arrow[d, "f'"] \\
GB \arrow[r, "Gq"'] & GB'
\end{tikzcd} \tag{7.3}$$

The 2-cells $(p, \alpha, q) \Rightarrow (p', \alpha', q')$ are pairs of 2-cells $(\sigma : p \Rightarrow p', \tau : q \Rightarrow q')$ such that the following diagram commutes:

$$\begin{tikzcd}
f' \circ F(p) \arrow[r, "f' \circ F(\sigma)"] \arrow[d, "\alpha"'] & f' \circ F(p') \arrow[d, "\alpha'"] \\
G(q) \circ f \arrow[r, "G(\tau) \circ f"'] & G(q') \circ f
\end{tikzcd} \tag{7.4}$$

The horizontal composite of $(A, f, B) \xrightarrow{(p,\alpha,q)} (A', f', B') \xrightarrow{(r,\beta,s)} (A'', f'', B'')$ is $(r \circ p, \cong, s \circ q)$, where the isomorphism is the composite on the left below:

$$\begin{tikzcd}
f'' \circ F(r \circ p) \arrow[r] \arrow[d, "f'' \circ (\phi^F_{r,p})^{-1}"'] & G(s \circ q) \circ f \\
f'' \circ (Fr \circ Fp) \arrow[d, "\cong"'] & (Gs \circ Gq) \circ f \arrow[u, "\phi^G_{s,q} \circ f"'] \\
(f'' \circ Fr) \circ Fp \arrow[d, "\beta \circ Fp"'] & Gs \circ (Gq \circ f) \arrow[u, "\cong"'] \\
(Gs \circ f') \circ Fp \arrow[r, "\cong"'] & Gs \circ (f' \circ Fp) \arrow[u, "Gs \circ \alpha"']
\end{tikzcd}
\qquad
\begin{tikzcd}
f \circ F\mathrm{Id}_A \arrow[r] \arrow[d, "f \circ (\psi^F_A)^{-1}"'] & G\mathrm{Id}_B \circ f \\
f \circ \mathrm{Id}_{FA} \arrow[r, "\cong"'] & \mathrm{Id}_{GB} \circ f \arrow[u, "\psi^G_B \circ f"']
\end{tikzcd}$$

In a similar fashion, the identity 1-cell on (A, f, B) is $(\mathrm{Id}_A, \cong, \mathrm{Id}_B)$ with isomorphism $\cong$ as on the right above.

Vertical composition and the identity 2-cell are given component-wise, as are the structural isomorphisms a, l and r. ◂

The identities and composition may be expressed as the following pasting diagrams:

$$\begin{tikzcd}
FA \arrow[r, "F\mathrm{Id}_A"] \arrow[d, "f"'] \arrow[dr, "f"] \arrow[dr, phantom, "\cong", shift left=3ex] \arrow[dr, phantom, "\cong"', shift right=3ex] & FA \arrow[d, "f"] \\
GB \arrow[r, "G\mathrm{Id}_B"'] & GB
\end{tikzcd}
\qquad
\begin{tikzcd}
FA \arrow[r, "Fp"] \arrow[rr, bend left=50, "F(r \circ p)"] \arrow[rr, phantom, bend left=25, "\overset{\phi^F}{\cong}"] \arrow[d, "f"'] \arrow[dr, phantom, "\overset{\alpha}{\Leftarrow}"] & FA' \arrow[r, "Fr"] \arrow[d, "f'"'] \arrow[dr, phantom, "\overset{\beta}{\Leftarrow}"] & FA'' \arrow[d, "f''"] \\
GB \arrow[r, "Gq"'] \arrow[rr, bend right=50, "G(s \circ q)"'] \arrow[rr, phantom, bend right=25, "\overset{\phi^G}{\cong}"'] & GB' \arrow[r, "Gs"'] & GB''
\end{tikzcd}$$

We call axiom (7.4) the *cylinder condition* due to its shape when viewed as a (3-dimensional) pasting diagram (*c.f.* the *cylinders* of [Bén67, § 8]). From this perspective, the axiom requires that if one passes across the top of the cylinder and then down the front, the result is the same as passing first down the back of the cylinder and then the bottom (*c.f.* the definition of *transformation* between T-algebra morphisms in 2-dimensional universal algebra [Lac10, § 4.1]):

$$\begin{tikzcd}
FA \arrow[r, bend left] \arrow[r, bend right] \arrow[r, phantom, "\Downarrow"] \arrow[d] \arrow[dr, phantom, "\overset{\alpha}{\Leftarrow}"] & FA' \arrow[d] \\
GA \arrow[r, bend right] & GB'
\end{tikzcd}
=
\begin{tikzcd}
FA \arrow[r, bend left] \arrow[d] \arrow[dr, phantom, "\overset{\alpha'}{\Leftarrow}"] & FA' \arrow[d] \\
GA \arrow[r, bend left, dashed] \arrow[r, bend right] \arrow[r, phantom, "\Downarrow"] & GB'
\end{tikzcd}$$

The following lemma, which mirrors the categorical statement, helps assure us the preceding definition is correct. For the proof one simply unwinds the two universal properties.

Lemma 7.2.2. For any pseudofunctor $F : \mathcal{B} \to \mathcal{C}$ and $C \in \mathcal{C}$, the following are equivalent:

1. (R, u) is a biuniversal arrow from F to C,
2. $(FR \xrightarrow{u} C)$ is the terminal object in $(F \downarrow \mathrm{const}_C)$, where const_C denotes the constant pseudofunctor at C. □

The glueing construction is an instance of the comma construction.

Definition 7.2.3. The *glueing bicategory* $\mathrm{gl}(\mathfrak{J})$ of bicategories $\mathcal{B}$ and $\mathcal{C}$ along a pseudofunctor $\mathfrak{J} : \mathcal{B} \to \mathcal{C}$ is the comma bicategory $(\mathrm{id}_{\mathcal{C}} \downarrow \mathfrak{J})$. ◂

As in Definition 7.1.1, we order the tuples in a comma bicategory as they are read down the page. In the particular case of a glueing bicategory, therefore, the objects, 1-cells and 2-cells have the following form:

$$\begin{aligned} \text{objects} &: (C \in \mathcal{C}, c : C \to \mathfrak{J}B, B \in \mathcal{B}) \\ \text{1-cells} &: (q : C \to C', \alpha : c' \circ q \Rightarrow \mathfrak{J}(p) \circ c, p : B \to B') \\ \text{2-cells} &: (\tau : q \Rightarrow q', \sigma : p \Rightarrow p') \end{aligned}$$

One now obtains projection *pseudo*functors $\mathcal{B} \xleftarrow{\pi_{\mathrm{dom}}} \mathrm{gl}(\mathfrak{J}) \xrightarrow{\pi_{\mathrm{cod}}} \mathcal{C}$. Note also that there is a 'weakest link' property at play: the bicategory $\mathrm{gl}(\mathfrak{J})$ is a 2-category only if $\mathcal{B}$, $\mathcal{C}$ and $\mathfrak{J}$ are all strict.

Remark 7.2.4. The preceding definitions are *pseudo*. One obtains a *lax* comma bicategory (and hence lax glueing bicategory) by dropping the requirement that the 2-cells filling (7.3) are invertible. ◂

7.3 Cartesian closed structure on gl($\mathfrak{J}$)

We now turn to a bicategorical version of Proposition 7.1.2. The construction for products is relatively easy.

7.3.1 Finite products in gl($\mathfrak{J}$)

Recall from Definition 4.1.1 that a bicategory with finite products—an *fp-bicategory*—is a bicategory $\mathcal{B}$ equipped with a chosen object $\prod_n(A_1, \dots, A_n)$ and a biuniversal arrow $(\pi_1, \dots, \pi_n) : \Delta\big(\prod_n(A_1, \dots, A_n)\big) \to (A_1, \dots, A_n)$ for every $A_1, \dots, A_n \in \mathcal{B}$ $(n \in \mathbb{N})$. An *fp-pseudofunctor* is then a pseudofunctor of the underlying bicategories that preserves these biuniversal arrows (Definition 4.1.9).

We claim the following:

Proposition 7.3.1. Let $(\mathcal{B}, \Pi_n(-))$ and $(\mathcal{C}, \Pi_n(-))$ be fp-bicategories and $(\mathfrak{J}, \mathrm{q}^{\times}) : \mathcal{B} \to \mathcal{C}$ an fp-pseudofunctor. Then $\mathrm{gl}(\mathfrak{J})$ is an fp-bicategory with both projection pseudofunctors π_{dom} and π_{cod} strictly preserving products. □

We construct the data in stages and then verify the required equivalence on hom-categories. Recall that we denote the 2-cells witnessing the fact that $\mathfrak{J}$ preserves products by

$$\begin{aligned} \mathrm{u}^{\times}_{B_\bullet} &: \mathrm{Id}_{(\prod_i \mathfrak{J}B_i)} \Rightarrow \langle \mathfrak{J}\pi_1, \dots, \mathfrak{J}\pi_n \rangle \circ \mathrm{q}^{\times}_{B_\bullet} \\ \mathrm{c}^{\times}_{B_\bullet} &: \mathrm{q}^{\times}_{B_\bullet} \circ \langle \mathfrak{J}\pi_1, \, \dots \, , \mathfrak{J}\pi_n \rangle \Rightarrow \mathrm{Id}_{\mathfrak{J}(\prod_i B_i)} \end{aligned}$$

We begin with the product mapping. For a family of objects $(C_i, c_i, B_i)_{i=1,\dots,n}$ we define the n-ary product $\prod_{i=1}^{n}(C_i, c_i, B_i)$ to be the tuple $\big(\prod_{i=1}^{n} C_i, \mathrm{q}^{\times}_{B_\bullet} \circ \prod_{i=1}^{n} c_i, \prod_{i=1}^{n} B_i\big)$. We set the k-th projection $\underline{\pi}_k$ to be (π_k, μ_k, π_k), where μ_k is defined by commutativity of the following diagram:

$$\begin{tikzcd}
c_k \circ \pi_k \arrow[r, "\mu_k"] \arrow[d, "\varpi^{(-k)}"'] & \mathfrak{J}(\pi_k) \circ \Big(\mathrm{q}^{\times}_{B_\bullet} \circ \prod_i c_i\Big) \\
\pi_k \circ \prod_i c_i \arrow[d, "\cong"'] & (\mathfrak{J}\pi_k \circ \mathrm{q}^{\times}_{B_\bullet}) \circ \prod_i c_i \arrow[u, "\cong"'] \\
(\pi_k \circ \mathrm{Id}_{(\prod_i \mathfrak{J}B_i)}) \circ \prod_i c_i \arrow[dr, bend right, "\pi_k \circ \mathrm{u}^{\times}_{B_\bullet} \circ \prod_i c_i"'] & \Big((\pi_k \circ \langle \mathfrak{J}\pi_1, \, \dots \, , \mathfrak{J}\pi_n \rangle) \circ \mathrm{q}^{\times}_{B_\bullet}\Big) \circ \prod_i c_i \arrow[u, "\varpi^{(k)} \circ \mathrm{q}^{\times}_{B_\bullet} \circ \prod_i c_i"'] \\
 & \Big(\pi_k \circ (\langle \mathfrak{J}\pi_1, \, \dots \, , \mathfrak{J}\pi_n \rangle \circ \mathrm{q}^{\times}_{B_\bullet})\Big) \circ \prod_i c_i \arrow[u, "\cong"']
\end{tikzcd} \tag{7.5}$$

Next we define the n-ary tupling map. For an n-ary family of 1-cells $(g_i, \alpha_i, f_i) : (Y, y, X) \to (C_i, c_i, B_i)$ $(i = 1, \dots, n)$, we set the n-ary tupling to be $(\langle g_1, \, \dots \, , g_n \rangle, \{\alpha_1, \, \dots \, , \alpha_n\}, \langle f_1, \, \dots \, , f_n \rangle)$, where $\{\alpha_1, \, \dots \, , \alpha_n\}$ is the composite

$$\begin{tikzcd}
\Big(\mathrm{q}^{\times}_{B_\bullet} \circ \prod_i c_i\Big) \circ \langle g_1, \, \dots \, , g_n \rangle \arrow[r, "\{\alpha_1, \dots, \alpha_n\}"] \arrow[d, "\cong"'] & \mathfrak{J}\langle f_1, \, \dots \, , f_n \rangle \circ y \\
\mathrm{q}^{\times}_{B_\bullet} \circ (\prod_i c_i \circ \langle g_1, \, \dots \, , g_n \rangle) \arrow[d, "\mathrm{q}^{\times}_{B_\bullet} \circ \mathsf{fuse}"'] & \mathrm{Id}_{\mathfrak{J}(\prod B_i)} \circ (\mathfrak{J}\langle f_1, \, \dots \, , f_n \rangle \circ y) \arrow[u, "\cong"'] \\
\mathrm{q}^{\times}_{B_\bullet} \circ \langle c_1 \circ g_1, \, \dots \, , c_n \circ g_n \rangle \arrow[d, "\mathrm{q}^{\times}_{B_\bullet} \circ \langle \alpha_1, \dots, \alpha_n \rangle"'] & \Big(\mathrm{q}^{\times}_{B_\bullet} \circ \langle \mathfrak{J}\pi_1, \, \dots \, , \mathfrak{J}\pi_n \rangle\Big) \circ (\mathfrak{J}\langle f_1, \, \dots \, , f_n \rangle \circ y) \arrow[u, "\mathrm{c}^{\times}_{B_\bullet} \circ \mathfrak{J}\langle f_1, \dots, f_n \rangle) \circ y"'] \\
\mathrm{q}^{\times}_{B_\bullet} \circ \langle \mathfrak{J}f_1 \circ y, \, \dots \, , \mathfrak{J}f_n \circ y \rangle \arrow[dr, bend right, "\mathrm{q}^{\times}_{B_\bullet} \circ \mathsf{post}^{-1}"'] & \mathrm{q}^{\times}_{B_\bullet} \circ ((\langle \mathfrak{J}\pi_1, \, \dots \, , \mathfrak{J}\pi_n \rangle \circ \mathfrak{J}\langle f_1, \, \dots \, , f_n \rangle) \circ y) \arrow[u, "\cong"'] \\
 & \mathrm{q}^{\times}_{B_\bullet} \circ (\langle \mathfrak{J}f_1, \, \dots \, , \mathfrak{J}f_n \rangle \circ y) \arrow[u, "\mathrm{q}^{\times}_{B_\bullet} \circ \mathsf{unpack}^{-1}_{f_\bullet} \circ y"']
\end{tikzcd} \tag{7.6}$$

Finally, we are required to provide a universal arrow to act as the counit. For every family of 1-cells $(g_i, \alpha_i, f_i) : (Y, y, X) \to (C_i, c_i, B_i)$ $(i = 1, \ldots, n)$ we require a glued 2-cell

$$\underline{\pi}_k \circ (\langle g_1, \ldots, g_n \rangle, \{\alpha_1, \ldots, \alpha_n\}, \langle f_1, \ldots, f_n \rangle) \Rightarrow (g_k, \alpha_k, f_k)$$

for which we take simply $(\varpi^{(k)}_{g_\bullet}, \varpi^{(k)}_{f_\bullet})$. The next lemma establishes that this is a 2-cell in gl($\mathfrak{J}$).

Lemma 7.3.2. For every family of 1-cells $(g_i, \alpha_i, f_i) : (Y, y, X) \to (C_i, c_i, B_i)$ $(i = 1, \ldots, n)$, the cylinder condition holds for $(\varpi^{(k)}_{g_\bullet}, \varpi^{(k)}_{f_\bullet})$. That is, the following diagram commutes:

$$\begin{tikzcd}
c_k \circ (\pi_k \circ \langle g_1, \ldots, g_n \rangle) \arrow[r, "c_k \circ \varpi^{(k)}"] \arrow[d, "\cong"'] & c_k \circ g_k \arrow[r, "\alpha_k"] & \mathfrak{J}(f_k) \circ y \\
(c_k \circ \pi_k) \circ \langle g_1, \ldots, g_n \rangle \arrow[d, "\mu_k \circ \langle g_1, \ldots, g_n \rangle"'] & & \mathfrak{J}\left(\pi_k \circ \langle f_1, \ldots, f_n \rangle\right) \circ y \arrow[u, "\mathfrak{J}(\varpi^{(k)}) \circ y"'] \\
\left(\mathfrak{J}(\pi_k) \circ \left(\mathrm{q}^{\times}_{B_\bullet} \circ \prod_i c_i\right)\right) \circ \langle g_1, \ldots, g_n \rangle \arrow[d, "\cong"'] & & (\mathfrak{J}\pi_k \circ \mathfrak{J}\langle f_1, \ldots, f_n \rangle) \circ y \arrow[u, "\phi^{\mathfrak{J}}_{\pi_k ; \langle f_\bullet \rangle} \circ y"'] \\
\mathfrak{J}\pi_k \circ \left(\left(\mathrm{q}^{\times}_{B_\bullet} \circ \prod_i c_i\right) \circ \langle g_1, \ldots, g_n \rangle\right) \arrow[rr, "\mathfrak{J}(\pi_k) \circ \{\alpha_1, \ldots, \alpha_n\}"'] & & \mathfrak{J}\pi_k \circ (\mathfrak{J}\langle f_1, \ldots, f_n \rangle \circ y) \arrow[u, "\cong"']
\end{tikzcd}$$

Proof. Unfolding the definition of fuse and applying the functoriality of composition as far as possible, the claim reduces to commutative diagram below, in which the unlabelled cells are all instances of functoriality of composition or naturality. To improve readability we neglect the bracketing and corresponding associativity constraints; the coherence theorem for bicategories guarantees that one can translate to the 'fully bicategorical' version as required.

$$\begin{tikzcd}
\pi_k \circ \langle c_\bullet \circ g_\bullet \rangle \arrow[rrr, "\varpi^{(k)}"] \arrow[d, "\cong"'] \arrow[rr, "\pi_k \circ \langle \alpha_1, \dots, \alpha_n \rangle"', start anchor=south east, end anchor=north west] & & & c_k \circ g_k \arrow[dddddddd, "\alpha_k"] \\
\pi_k \circ \mathrm{Id}_{(\prod_i \mathfrak{J} B_i)} \circ \langle c_\bullet \circ g_\bullet \rangle \arrow[d, "\pi_k \circ \mathrm{u}^{\times}_{B_\bullet} \circ \langle c_\bullet \circ g_\bullet \rangle"'] \arrow[rrdd, bend left, "\pi_k \circ \mathrm{Id} \circ \langle \alpha_1, \dots, \alpha_n \rangle"] & & \pi_k \circ \langle \mathfrak{J}(f_\bullet) \circ y \rangle \arrow[dd, "\cong"] \arrow[ddddd, bend left=60, "\pi_k \circ \mathsf{post}^{-1}"] & \\
\pi_k \circ \langle \mathfrak{J}\pi_\bullet \rangle \circ \mathrm{q}^{\times}_{B_\bullet} \circ \langle c_\bullet \circ g_\bullet \rangle \arrow[d, "\pi_k \circ \langle \mathfrak{J}\pi_\bullet \rangle \circ \mathrm{q}^{\times}_{B_\bullet} \circ \langle \alpha_1, \dots, \alpha_n \rangle"'] & \text{triang. law} \arrow[d, phantom, "="] & & \\
\pi_k \circ \langle \mathfrak{J}\pi_\bullet \rangle \circ \mathrm{q}^{\times}_{B_\bullet} \circ \langle \mathfrak{J}(f_\bullet) \circ y \rangle \arrow[rr, "\pi_k \circ \mathrm{c}^{\times}_{B_\bullet} \circ \langle \mathfrak{J}(f_\bullet) \circ y \rangle"] \arrow[d, "\pi_k \circ \mathsf{post}^{-1}"'] & & \pi_k \circ \mathrm{Id}_{(\prod_i \mathfrak{J} B_i)} \circ \langle \mathfrak{J}(f_\bullet) \circ y \rangle \arrow[d, "\pi_k \circ \mathrm{Id} \circ \mathsf{post}^{-1}"] & \\
\pi_k \circ \langle \mathfrak{J}\pi_\bullet \rangle \circ \mathrm{q}^{\times}_{B_\bullet} \circ \langle \mathfrak{J} f_\bullet \rangle \circ y \arrow[rr, "\pi_k \circ \mathrm{c}^{\times}_{B_\bullet} \circ \langle \mathfrak{J} f_\bullet \rangle \circ y"] \arrow[d, "\pi_k \circ \langle \mathfrak{J}\pi_\bullet \rangle \circ \mathrm{q}^{\times}_{B_\bullet} \circ \mathsf{unpack}^{-1}_{f_\bullet} \circ y"'] & & \pi_k \circ \mathrm{Id}_{(\prod_i \mathfrak{J} B_i)} \circ \langle \mathfrak{J} f_\bullet \rangle \circ y \arrow[dd, "\cong"] \arrow[lldd, bend right, "\pi_k \circ \langle \mathfrak{J}\pi_\bullet \rangle \circ \mathsf{unpack}^{-1}_{f_\bullet} \circ y"'] & \\
\pi_k \circ \langle \mathfrak{J}\pi_\bullet \rangle \circ \mathrm{q}^{\times}_{B_\bullet} \circ \langle \mathfrak{J}\pi_\bullet \rangle \circ \mathfrak{J}\langle f_\bullet \rangle \circ y \arrow[d, "\pi_k \circ \langle \mathfrak{J}\pi_\bullet \rangle \circ \mathrm{c}^{\times}_{B_\bullet} \circ \mathfrak{J}\langle f_\bullet \rangle \circ y"'] & & & \\
\pi_k \circ \langle \mathfrak{J}\pi_\bullet \rangle \circ \mathrm{Id}_{(\mathfrak{J} \prod_i B_i)} \circ \mathfrak{J}\langle f_\bullet \rangle \circ y \arrow[d, "\cong"'] & & \pi_k \circ \langle \mathfrak{J} f_\bullet \rangle \circ y \arrow[lld, "\pi_k \circ \mathsf{unpack}^{-1}_{f_\bullet} \circ y"'] \arrow[rdd, "\varpi^{(k)} \circ y"] & \overset{\mathsf{post\ def.}}{=} \\
\pi_k \circ \langle \mathfrak{J}\pi_\bullet \rangle \circ \mathfrak{J}\langle f_\bullet \rangle \circ y \arrow[d, "\varpi^{(k)} \circ \mathfrak{J}\langle f_\bullet \rangle \circ y"'] & & \overset{\mathsf{unpack\ def.}}{=} & \\
\mathfrak{J}(\pi_k) \circ \mathfrak{J}\langle f_\bullet \rangle \circ y \arrow[rr, "\phi^{\mathfrak{J}}_{(\pi_k, \langle f_\bullet \rangle)} \circ y"'] & & \mathfrak{J}(\pi_k \circ \langle f_\bullet \rangle) \circ y \arrow[r, "\mathfrak{J}(\varpi^{(k)}) \circ y"'] & \mathfrak{J}(f_k) \circ y
\end{tikzcd}$$

□

It remains to check the universal property. Taking arbitrary 1-cells

$$\begin{aligned} (v, \gamma, u) &: (Y, y, X) \to \textstyle\prod_{i=1}^n (C_i, c_i, B_i) \\ (t_i, \tau_i, s_i) &: (Y, y, X) \to (C_i, c_i, B_i) \qquad\qquad (i = 1, \dots, n) \end{aligned}$$

related by 2-cells

$$(\beta_i, \alpha_i) : \underline{\pi_i} \circ (v, \gamma, u) \Rightarrow (t_i, \tau_i, s_i) \qquad\qquad (i = 1, \dots, n)$$

we observe that $\beta_i : \pi_i \circ v \Rightarrow t_i$ and $\alpha_i : \pi_i \circ u \Rightarrow s_i$ for each i. We therefore claim that $\big(\mathsf{p}^\dagger(\beta_1, \dots, \beta_n), \mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n)\big)$ is the unique 2-cell in $\mathrm{gl}(\mathfrak{J})$ such that the following commutes for $i = 1, \dots, n$:

$$\begin{tikzcd}
\underline{\pi_i} \circ (v, \gamma, u) \arrow[rr, "\underline{\pi_i} \circ (\mathsf{p}^\dagger(\beta_\bullet), \mathsf{p}^\dagger(\alpha_\bullet))"] \arrow[dr, "(\beta_i, \alpha_i)"'] & & \underline{\pi_i} \circ (\langle t_\bullet \rangle, \{\tau_\bullet\}, \langle s_\bullet \rangle) \arrow[dl, "(\varpi^{(i)}_{t_\bullet}, \varpi^{(i)}_{s_\bullet})"] \\
& (t_i, \tau_i, s_i) &
\end{tikzcd}$$

Of course, it suffices to show that $\big(\mathsf{p}^\dagger(\beta_\bullet), \mathsf{p}^\dagger(\alpha_\bullet)\big)$ is a 2-cell in gl($\mathfrak{I}$): the rest of the claim follows from the (bi)universality of products in $\mathcal{B}$ and $\mathcal{C}$.

Lemma 7.3.3. For any 1-cells (v, γ, u) and (t_i, τ_i, s_i) and any 2-cells $(\beta_i, \alpha_i) : \underline{\pi_i} \circ (v, \gamma, u) \Rightarrow (t_i, \tau_i, s_i)$ $(i = 1, \dots, n)$ as above, the pair $\big(\mathsf{p}^\dagger(\beta_1, \dots, \beta_n), \mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n)\big)$ is a 2-cell in gl($\mathfrak{I}$).

Proof. We need to check the cylinder condition, which in this case is the following:

$$\begin{tikzcd}
\Big(\mathrm{q}^{\times}_{B_\bullet} \circ \prod_i c_i\Big) \circ v \arrow[r, "\mathrm{q}^{\times}_{B_\bullet} \circ (\prod_i c_i) \circ \mathsf{p}^\dagger(\beta_1, \dots, \beta_n)"] \arrow[d, "\gamma"'] & \Big(\mathrm{q}^{\times}_{B_\bullet} \circ \prod_i c_i\Big) \circ \langle t_1, \dots, t_n \rangle \arrow[d, "\{\tau_1, \dots, \tau_n\}"] \\
\mathfrak{I}(u) \circ y \arrow[r, "\mathfrak{I}(\mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n)) \circ y"'] & \mathfrak{I}(\langle s_1, \dots, s_n \rangle) \circ y
\end{tikzcd}$$

For this, one begins by observing that the following commutes for every $k = 1, \dots, n$:

$$\begin{tikzcd}
\pi_k \circ (\prod_i c_i \circ v) \arrow[r, "\cong"] \arrow[d, "\pi_k \circ (\prod_i c_i) \circ \mathsf{p}^\dagger(\beta_1, \dots, \beta_n)"'] & (\pi_k \circ \prod_i c_i) \circ v \arrow[r, "\varpi^{(k)} \circ v"] \arrow[d, "\pi_k \circ \prod_i c_i \circ \mathsf{p}^\dagger(\beta_1, \dots, \beta_n)"] & (c_k \circ \pi_k) \circ v \arrow[d, "\cong"] \\
\pi_k \circ (\prod_i c_i \circ \langle t_\bullet \rangle) \arrow[r, "\cong"'] \arrow[dd, "\pi_k \circ \mathsf{fuse}"'] & (\pi_k \circ \prod_i c_i) \circ \langle t_\bullet \rangle \arrow[d, "\varpi^{(k)} \circ \langle t_\bullet \rangle"] & c_k \circ (\pi_k \circ v) \arrow[d, "c_k \circ \pi_k \circ \mathsf{p}^\dagger(\beta_1, \dots, \beta_n)"] \arrow[dd, bend left=60, "c_k \circ \beta_k"] \\
\arrow[dr, phantom, "\text{def. } \underset{=}{\text{of}} \text{ fuse}"] & (c_k \circ \pi_k) \circ \langle t_\bullet \rangle \arrow[r, "\cong"'] & c_k \circ (\pi_k \circ \langle t_\bullet \rangle) \arrow[d, "c_k \circ \varpi^{(k)}"] \\
\pi_k \circ \langle c_\bullet \circ t_\bullet \rangle \arrow[rr, "\varpi^{(k)}"'] \arrow[d, "\pi_k \circ \langle \tau_\bullet \rangle"'] & & c_k \circ t_k \arrow[d, "\tau_k"] \\
\pi_k \circ \langle \mathfrak{I}(s_\bullet) \circ y \rangle \arrow[rr, "\varpi^{(k)}"'] \arrow[d, "\pi_k \circ \mathsf{post}^{-1}"'] & \arrow[d, phantom, "\text{def. } \underset{=}{\text{of}} \text{ post}"] & \mathfrak{I}(s_k) \circ y \arrow[d, equal] \\
\pi_k \circ (\langle \mathfrak{I} s_\bullet \rangle \circ y) \arrow[r, "\cong"'] & (\pi_k \circ \langle \mathfrak{I} s_\bullet \rangle) \circ y \arrow[r, "\varpi^{(k)} \circ y"'] & \mathfrak{I}(s_k) \circ y
\end{tikzcd}$$

and that the following commutes:

$$\begin{tikzcd}
c_k \circ (\pi_k \circ v) \arrow[rrrr, "\cong"] \arrow[dd, "c_k\circ\beta_k"'] & & & & (c_k \circ \pi_k) \circ v \arrow[d, "\mu_k\circ v"] \\
& & & & \Big(\mathfrak{J}(\pi_k) \circ (\mathrm{q}^{\times}_{B_\bullet} \circ \prod_i c_i)\Big) \circ v \arrow[d, "\cong"] \\
\pi_k \circ t_k \arrow[dd, "\tau_k"'] & & \overset{\text{cylinder condition}}{=} & & \mathfrak{J}(\pi_k) \circ \Big(\Big(\mathrm{q}^{\times}_{B_\bullet} \circ \prod_i c_i\Big) \circ v\Big) \arrow[d, "\mathfrak{J}(\pi_k)\circ\gamma"] \\
& & \mathfrak{J}(\pi_k \circ u) \circ y \arrow[dll, "\mathfrak{J}(\alpha_k)\circ y"'] \arrow[ddll, bend left, "\mathfrak{J}(\pi_k\circ\mathsf{p}^{\dagger}(\alpha_\bullet))\circ y"] & & \mathfrak{J}(\pi_k) \circ (\mathfrak{J}(u) \circ y) \arrow[d, "\cong"] \\
\mathfrak{J}(s_k) \circ y \arrow[d, "\mathfrak{J}(\varpi^{(-k)})\circ y"'] & \overset{\text{def.}}{=} & & \overset{\text{nat.}}{=} & (\mathfrak{J}\pi_k \circ \mathfrak{J}u) \circ y \arrow[ull, "\phi^{\mathfrak{J}}_{\pi_k,u}\circ y"'] \arrow[d, "\mathfrak{J}(\pi_k)\circ\mathfrak{J}(\mathsf{p}^{\dagger}(\alpha_\bullet))\circ y"] \\
\mathfrak{J}(\pi_k \circ \langle s_\bullet \rangle) \circ y \arrow[rrrr, "(\phi^{\mathfrak{J}}_{\pi_k,\langle s_\bullet\rangle})^{-1}\circ y"'] & & & & (\mathfrak{J}(\pi_k) \circ \mathfrak{J}\langle s_\bullet \rangle) \circ y
\end{tikzcd}$$

Putting these two together and applying the definition of unpack, one obtains the following commuting diagram:

$$\begin{tikzcd}
\pi_k \circ (\prod_i c_i \circ v) \arrow[rr, "\cong"] \arrow[d, "\pi_k\circ\prod_i c_i\circ\mathsf{p}^{\dagger}(\beta_\bullet)"'] & & (\pi_k \circ \mathrm{Id}_{(\prod_i \mathfrak{J}B_i)}) \circ (\prod_i c_i \circ v) \arrow[d, "\pi_k\circ\mathrm{u}^{\times}_{B_\bullet}\circ\prod_i c_i\circ v"] \\
\pi_k \circ (\prod_i c_i \circ \langle t_\bullet \rangle) \arrow[d, "\pi_k\circ\mathsf{fuse}"'] & & \Big(\pi_k \circ \Big(\langle \mathfrak{J}\pi_\bullet \rangle \circ \mathrm{q}^{\times}_{B_\bullet}\Big)\Big) \circ (\prod_i c_i \circ v) \arrow[d, "\cong"] \\
\pi_k \circ \langle c_\bullet \circ t_\bullet \rangle \arrow[d, "\pi_k\circ\langle\tau_\bullet\rangle"'] & & (\pi_k \circ \langle \mathfrak{J}\pi_\bullet \rangle) \circ \Big(\Big(\mathrm{q}^{\times}_{B_\bullet} \circ \prod_i c_i\Big) \circ v\Big) \arrow[d, "\varpi^{(k)}\circ\mathrm{q}^{\times}_{B_\bullet}\circ\prod_i c_i\circ v"] \\
\pi_k \circ \langle \mathfrak{J}(s_\bullet) \circ y \rangle \arrow[d, "\pi_k\circ\mathsf{post}^{-1}"'] & & \mathfrak{J}(\pi_k) \circ \Big(\Big(\mathrm{q}^{\times}_{B_\bullet} \circ \prod_i c_i\Big) \circ v\Big) \arrow[d, "\mathfrak{J}(\pi_k)\circ\gamma"] \\
\pi_k \circ (\langle \mathfrak{J}s_\bullet \rangle \circ y) \arrow[d, "\pi_k\circ\mathsf{unpack}^{-1}_{s_\bullet}\circ y"'] & & \mathfrak{J}(\pi_k) \circ (\mathfrak{J}(u) \circ y) \arrow[d, "\mathfrak{J}(\pi_k)\circ\mathfrak{J}(\mathsf{p}^{\dagger}(\alpha_\bullet))\circ y"] \\
\pi_k \circ ((\langle \mathfrak{J}\pi_\bullet \rangle \circ \mathfrak{J}\langle s_\bullet \rangle) \circ y) \arrow[r, "\cong"'] & (\pi_k \circ \langle \mathfrak{J}\pi_\bullet \rangle) \circ (\mathfrak{J}\langle s_\bullet \rangle \circ y) \arrow[r, "\varpi^{(k)}\circ\mathfrak{J}\langle s_\bullet\rangle\circ y"'] & \mathfrak{J}(\pi_k) \circ (\mathfrak{J}\langle s_\bullet \rangle \circ y)
\end{tikzcd}$$

With this lemma in hand, the rest of the proof is a diagram chase applying naturality and the definition of post. □

Lemma 7.3.3 completes the proof that $\mathrm{gl}(\mathfrak{J})$ does indeed have finite products, and hence the proof of Proposition 7.3.1. For the construction of exponentials we will require morphisms of the form $f \times A$. We briefly check that such morphisms appear in gl($\mathfrak{J}$) in the way one would expect, namely as pasting diagrams of the form

$$\begin{tikzcd}[column sep=huge, row sep=huge]
C \times Y \arrow[rr, "g \times Y"] \arrow[d, "c \times y"'] \arrow[drr, bend left=10, "c'g \times y" description] \arrow[drr, bend right=10, "\mathfrak{J}(f)c \times y" description] \arrow[dd, bend right=60, "\mathrm{q}^{\times}_{B,X} \circ (c \times y)"'] \arrow[rrd, phantom, bend left=30, "\overset{\Phi}{\cong}"] \arrow[rrd, phantom, "\overset{\alpha \times y}{\cong}"] & & C' \times Y \arrow[d, "c' \times y"] \arrow[dd, bend left=60, "\mathrm{q}^{\times}_{B',X} \circ (c' \times y)"] \\
\mathfrak{J}B \times \mathfrak{J}X \arrow[rr, bend right=20, "\mathfrak{J}f \times \mathfrak{J}X"'] \arrow[d, "\mathrm{q}^{\times}_{B,X}"'] \arrow[rr, phantom, "\overset{\Phi}{\cong}"] \arrow[rrd, phantom, "\overset{\mathsf{nat}}{\cong}"] & & \mathfrak{J}B' \times \mathfrak{J}X \arrow[d, "\mathrm{q}^{\times}_{B',X}"] \\
\mathfrak{J}(B \times X) \arrow[rr, "\mathfrak{J}(f \times X)"'] & & \mathfrak{J}(B' \times X)
\end{tikzcd}$$

In particular, when the bicategories $\mathcal{B}$ and $\mathcal{C}$ are 2-categories with strict products and $\mathfrak{J} : \mathcal{B} \to \mathcal{C}$ is a strict fp-pseudofunctor, this 2-cell is simply $\alpha \times y$.

Lemma 7.3.4. For every 1-cell $\underline{g} := (g, \alpha, f) : (C, c, B) \to (C', c', B')$ and object $\underline{Y} := (Y, y, X)$ in gl($\mathfrak{J}$), the 1-cell $\underline{g} \times \underline{Y} : (C, c, B) \times (Y, y, X) \to (C', c', B') \times (Y, y, X)$ is equal to $(g \times Y, \underline{\alpha_{Y}}, f \times Y)$, where $\underline{\alpha_{Y}}$ is the composite

$$\begin{tikzcd}
\left(\mathrm{q}^{\times}_{B',X} \circ (c' \times y)\right) \circ (g \times Y) \arrow[r, "\underline{\alpha_{Y}}"] \arrow[d, "\cong"'] & \mathfrak{J}(f \times X) \circ \left(\mathrm{q}^{\times}_{B,X} \circ (c \times y)\right) \\
\mathrm{q}^{\times}_{B',X} \circ ((c' \times y) \circ (g \times Y)) \arrow[d, "\mathrm{q}^{\times}_{B',X} \circ \Phi_{c',g;y,\mathrm{Id}}"'] & \left(\mathfrak{J}(f \times X) \circ \mathrm{q}^{\times}_{B,X}\right) \circ (c \times y) \arrow[u, "\cong"'] \\
\mathrm{q}^{\times}_{B',X} \circ ((c' \circ g) \times (y \circ \mathrm{Id}_{Y})) \arrow[d, "\cong"'] & \left(\mathrm{q}^{\times}_{B',X} \circ (\mathfrak{J}f \times \mathfrak{J}\mathrm{Id}_{X})\right) \circ (c \times y) \arrow[u, "\mathsf{nat}_{f,\mathrm{Id}_X} \circ (c \times y)"'] \\
\mathrm{q}^{\times}_{B',X} \circ ((c' \circ g) \times (\mathrm{Id}_{\mathfrak{J}X} \circ y)) \arrow[d, "\mathrm{q}^{\times}_{B',X} \circ (\alpha \times (\mathrm{Id}_{\mathfrak{J}X} \circ y))"'] & \mathrm{q}^{\times}_{B',X} \circ ((\mathfrak{J}f \times \mathfrak{J}\mathrm{Id}_{X}) \circ (c \times y)) \arrow[u, "\cong"'] \\
\mathrm{q}^{\times}_{B',X} \circ ((\mathfrak{J}f \circ c) \times (\mathrm{Id}_{\mathfrak{J}X} \circ y)) \arrow[r, "\mathrm{q}^{\times}_{B',X} \circ ((\mathfrak{J}f \circ c) \times (\psi^{\mathfrak{J}}_{X} \circ y))"'] & \mathrm{q}^{\times}_{B',X} \circ ((\mathfrak{J}f \circ c) \times (\mathfrak{J}\mathrm{Id}_{X} \circ y)) \arrow[u, "\mathrm{q}^{\times}_{B',X} \circ \Phi^{-1}_{\mathfrak{J}f,c;\mathfrak{J}\mathrm{Id},y}"']
\end{tikzcd} \tag{7.7}$$

Proof. The proof amounts to unfolding the definition and checking that it does indeed equal the composite given in the claim. Let τ_1 and τ_2 respectively denote the 2-cells defined by the pasting diagrams on the left and right below:

$$\begin{tikzcd}
C \times Y \arrow[rr, "\pi_1"] \arrow[d, "c\times y"'] \arrow[rrrr, bend left, "g\circ\pi_1"] \arrow[dd, bend right=60, "\mathrm{q}^{\times}_{B,X}\circ(c\times y)"'] & & C \arrow[rr, "g"] \arrow[dd, "c"] & & C' \arrow[dd, "c'"] \\
\mathfrak{J}B \times \mathfrak{J}X \arrow[d, "\mathrm{q}^{\times}_{B,X}"'] & \overset{\mu_1}{\Leftarrow} & & \overset{\alpha}{\Leftarrow} & \\
\mathfrak{J}(B \times X) \arrow[rr, "\mathfrak{J}\pi_1"'] \arrow[rrrr, bend right, "\mathfrak{J}(f\circ\pi_1)"', "{\overset{\phi^{\mathfrak{J}}_{f,\pi_1}}{\cong}}"] & & \mathfrak{J}B \arrow[rr, "\mathfrak{J}f"'] & & \mathfrak{J}B'
\end{tikzcd}
\qquad
\begin{tikzcd}
C \times Y \arrow[rr, "\pi_2"] \arrow[d, "c\times y"'] \arrow[rrrr, bend left, "\mathrm{Id}_Y\circ\pi_2"] \arrow[dd, bend right=60, "\mathrm{q}^{\times}_{B,X}\circ(c\times y)"'] & & Y \arrow[rr, "\mathrm{Id}_Y"] \arrow[dd, "y"] \arrow[ddrr, "y"] & \cong & Y \arrow[dd, "y"] \\
\mathfrak{J}B \times \mathfrak{J}X \arrow[d, "\mathrm{q}^{\times}_{B,X}"'] & \overset{\mu_2}{\Leftarrow} & & \cong & \\
\mathfrak{J}(B \times X) \arrow[rr, "\mathfrak{J}\pi_2"'] \arrow[rrrr, bend right=50, "\mathfrak{J}(\mathrm{Id}_X\circ\pi_2)"', "{\overset{\phi^{\mathfrak{J}}_{\mathrm{Id},\pi_2}}{\cong}}"] & & \mathfrak{J}X \arrow[rr, bend left, "\mathrm{Id}_{\mathfrak{J}X}"] \arrow[rr, bend right, "\mathfrak{J}\mathrm{Id}_X"'] \arrow[rr, phantom, "{\overset{\psi^{\mathfrak{J}}_X}{\cong}}"] & & \mathfrak{J}X
\end{tikzcd}$$

By definition, the 1-cell $\underline{g} \times \underline{Y}$ has a witnessing 2-cell given by the following composite, in which we write $(*)$ for $\mathrm{q}^{\times}_{B',X} \circ \Big\langle \Big(\mathfrak{J}(f \circ \pi_1) \circ \mathrm{q}^{\times}_{B',X}\Big) \circ (c \times y), \Big(\mathfrak{J}(\mathrm{Id}_X \circ \pi_2) \circ \mathrm{q}^{\times}_{B',X}\Big) \circ (c \times y)\Big\rangle$:

$$\begin{tikzcd}
\Big(\mathrm{q}^{\times}_{B',X} \circ (c' \times y)\Big) \circ \langle g \circ \pi_1, \mathrm{Id}_Y \circ \pi_2 \rangle \arrow[r, "\{\tau_1,\tau_2\}"] \arrow[d, "\cong"'] & \mathfrak{J}(f \times B) \circ \Big(\mathrm{q}^{\times}_{B,X} \circ (c \times y)\Big) \\
\mathrm{q}^{\times}_{B',X} \circ ((c' \times y) \circ \langle g \circ \pi_1, \mathrm{Id}_Y \circ \pi_2 \rangle) \arrow[d, "\mathrm{q}^{\times}_{B',X}\circ\mathsf{fuse}"'] & \\
\mathrm{q}^{\times}_{B',X} \circ \langle c' \circ (g \circ \pi_1), y \circ (\mathrm{Id}_Y \circ \pi_2) \rangle \arrow[d, "\mathrm{q}^{\times}_{B',X}\circ\langle\tau_1,\tau_2\rangle"'] & \\
(*) \arrow[d, "\mathrm{q}^{\times}_{B',X}\circ\mathsf{post}^{-1}"'] & \mathrm{Id}_{\mathfrak{J}(B'\times X)} \circ \mathfrak{J}(f \times X) \circ (c \times y) \arrow[uuu, "\cong"'] \\
\mathrm{q}^{\times}_{B',X} \circ \Big(\langle \mathfrak{J}(f \circ \pi_1), \mathfrak{J}(\mathrm{Id}_X \circ \pi_2) \rangle \circ \Big(\mathrm{q}^{\times}_{B',X} \circ (c \times y)\Big)\Big) \arrow[d, "\mathrm{q}^{\times}_{B',X}\circ\mathsf{unpack}^{-1}_{f\circ\pi_1,\mathrm{Id}\circ\pi_2}\circ(c\times y)"'] & \\
\mathrm{q}^{\times}_{B',X} \circ \Big((\langle \mathfrak{J}\pi_1, \mathfrak{J}\pi_2 \rangle \circ \mathfrak{J}(f \times X)) \circ \Big(\mathrm{q}^{\times}_{B'\times X} \circ (c \times y)\Big)\Big) \arrow[dr, "\cong"'] & \\
& \Big(\mathrm{q}^{\times}_{B',X} \circ \langle \mathfrak{J}\pi_1, \mathfrak{J}\pi_2 \rangle\Big) \circ \Big(\mathfrak{J}(f \times X) \circ \Big(\mathrm{q}^{\times}_{B'\times X} \circ (c \times y)\Big)\Big) \arrow[uuu, "\mathrm{c}^{\times}_{B',X}\circ\mathfrak{J}(f\times X)\circ(c\times y)"']
\end{tikzcd}$$

Applying naturality and the lemma relating unpack with $\mathsf{u}^{\times}$ (Lemma 4.1.13), a long diagram chase transforms this to the composite in the claim. □

7.3.2 Exponentials in $\mathrm{gl}(\mathfrak{J})$

As in the 1-categorical case, the definition of currying in $\mathrm{gl}(\mathfrak{J})$ employs pullbacks. We therefore take a brief diversion to spell out their universal property.

Pullbacks in a bicategory. The notion of pullback we employ is sometimes referred to as a *bipullback* (*e.g.* [Lac10]) to distinguish it from pullbacks defined as a pseudolimit. Since the only limits we work with in this thesis are bilimits, we omit the prefix.

Definition 7.3.5. Let C (for 'cospan') denote the category $(1 \xrightarrow{h_1} 0 \xleftarrow{h_2} 2)$ and $\mathcal{B}$ be any bicategory. A *pullback* of the cospan $(X_1 \xrightarrow{f_1} X_0 \xleftarrow{f_2} X_2)$ in $\mathcal{B}$ is a bilimit for the strict pseudofunctor $\mathrm{C} \to \mathcal{B}$ determined by this cospan. ◀

This characterisation of pullbacks, while precise, must be unfolded to obtain a universal property one can use for calculations. The next lemma establishes such a property. The proof is not especially hard, and the result appears to be known—although not explicitly proven—in the literature, so we leave it for an appendix (Appendix D).

Lemma 7.3.6. For any bicategory $\mathcal{B}$ and cospan $(X_1 \xrightarrow{f_1} X_0 \xleftarrow{f_2} X_2)$ in $\mathcal{B}$, the pullback of $(X_1 \xrightarrow{f_1} X_0 \xleftarrow{f_2} X_2)$ is determined, up to equivalence, by the following universal property: there exists a chosen object $P \in \mathcal{B}$, span $(X_1 \xleftarrow{\gamma_1} P \xrightarrow{\gamma_2} X_2)$ and invertible 2-cell $\overline{\gamma}$ filling the diagram on the left below

$$\begin{tikzcd} & P \arrow[dl, "\gamma_1"'] \arrow[dr, "\gamma_2"] & & & & Q \arrow[dl, "\mu_1"'] \arrow[dr, "\mu_2"] & \\ X_1 \arrow[dr, "f_1"'] & \overset{\overline{\gamma}}{\cong} & X_2 \arrow[dl, "f_2"] & & X_1 \arrow[dr, "f_1"'] & \overset{\overline{\mu}}{\cong} & X_2 \arrow[dl, "f_2"] \\ & X_0 & & & & X_0 & \end{tikzcd} \tag{7.8}$$

such that for any other such square as on the right above there exists an invertible *fill-in* (u, Ξ_1, Ξ_2) (*c.f.* [Vit10]), namely a 1-cell $u : Q \to P$ and invertible 2-cells $\Xi_i : \gamma_i \circ u \Rightarrow \mu_i$ $(i = 1, 2)$ such that

$$\begin{tikzcd} (f_2 \circ \gamma_2) \circ u \arrow[r, "\cong"] \arrow[d, "\overline{\gamma} \circ u"'] & f_2 \circ (\gamma_2 \circ u) \arrow[r, "f_2 \circ \Xi_2"] & f_2 \circ \mu_2 \arrow[d, "\overline{\mu}"] \\ (f_1 \circ \gamma_1) \circ u \arrow[r, "\cong"'] & f_1 \circ (\gamma_1 \circ u) \arrow[r, "f_1 \circ \Xi_1"'] & f_1 \circ \mu_1 \end{tikzcd} \tag{7.9}$$

This fill-in is universal in the following sense. For any other fill-in

$$(v : Q \to P, \Psi_1 : \gamma_1 \circ v \Rightarrow \mu_1, \Psi_2 : \gamma_2 \circ v \Rightarrow \mu_2)$$

there exists a 2-cell $\Psi^{\dagger} : v \Rightarrow u$, unique such that

$$\begin{tikzcd} \gamma_i \circ v \arrow[rr, "\gamma_i \circ \Psi^{\dagger}"] \arrow[dr, "\Psi_i"'] & & \gamma_i \circ u \arrow[dl, "\Xi_i"] \\ & \mu_i & \end{tikzcd} \tag{7.10}$$

for $i = 1, 2$. Finally, it is required that for any $w : Q \to P$ the 2-cell $\mathrm{id}^{\dagger}$ obtained by applying the universal property to $(w, \mathrm{id}_{\gamma_1 \circ w}, \mathrm{id}_{\gamma_2 \circ w})$ is invertible. □

Remark 7.3.7. The universal property of pullbacks can be stated in a slightly different way, which is more useful for some calculations. The pullback of a cospan $(X_1 \xrightarrow{f_1} X_0 \xleftarrow{f_2} X_2)$ is determined by a biuniversal arrow $(\gamma, \overline{\gamma}) : \Delta P \Rightarrow F$, for F the pseudofunctor determined by the cospan, P the pullback, and $(\gamma, \overline{\gamma})$ an iso-commuting square as in (7.8). It follows that the functor $(\gamma, \overline{\gamma}) \circ \Delta(-) : \mathcal{B}(Z, P) \to \mathrm{Hom}(\mathrm{C}, \mathcal{B})(\Delta Z, F)$ is fully-faithful and essentially surjective for every $Z \in \mathcal{B}$. Being essentially surjective is exactly the existence of a fill-in for every iso-commuting square, as in the preceding lemma. Being full and faithful entails that, for every pair of 1-cells $t, u : Z \to P$ equipped with 2-cells $\Gamma_i : \gamma_i \circ t \Rightarrow \gamma_i \circ u$ $(i = 1, 2)$ satisfying the fill-in law (7.9), there exists a unique 2-cell $\Gamma^{\dagger} : t \Rightarrow u$ such that $\gamma_i \circ \Gamma^{\dagger} = \Gamma_i$ for $i = 1, 2$. ◂

The following is an example of where it is convenient to use the universal property of Remark 7.3.7. The lemma guarantees that one may define objects in a glueing bicategory (up to equivalence) by pullback.

Lemma 7.3.8. For any pseudofunctor $\mathfrak{J} : \mathcal{B} \to \mathcal{C}$ and any pullbacks

$$\begin{tikzcd} P \arrow[r, "q"] \arrow[d, "p"'] \arrow[dr, phantom, "\lrcorner", very near start] \arrow[dr, phantom, "\overset{\pi}{\cong}"] & B \arrow[d, "b"] \\ \mathfrak{J}A \arrow[r, "a"'] & C \end{tikzcd} \qquad \begin{tikzcd} X \arrow[r, "y"] \arrow[d, "x"'] \arrow[dr, phantom, "\lrcorner", very near start] \arrow[dr, phantom, "\overset{\chi}{\cong}"] & B \arrow[d, "b"] \\ \mathfrak{J}A \arrow[r, "a"'] & C \end{tikzcd}$$

in $\mathcal{C}$, the objects $(P \xrightarrow{p} \mathfrak{J}A)$ and $(X \xrightarrow{x} \mathfrak{J}A)$ are equivalent in $\mathrm{gl}(\mathfrak{J})$.

Proof. It is immediate from the uniqueness of bilimits that there exists a canonical equivalence $P \simeq X$. The only question is whether this equivalence lifts to a 1-cell in $\mathrm{gl}(\mathfrak{J})$. If one constructs the equivalence using the universal property of Remark 7.3.7, this follows immediately. □

Preliminaries complete, we can now give the data for defining exponentials in the glueing bicategory. Precisely, we extend Proposition 7.3.1 to the following. Recall that a cartesian closed bicategory—a *cc-bicategory*—is an fp-bicategory equipped with a right biadjoint to $(-) \times A$ for every object A (Definition 5.1.1).

Theorem 7.3.9. Let $(\mathcal{B}, \Pi_n(-), \Rightarrow)$ and $(\mathcal{C}, \Pi_n(-), \Rightarrow)$ be cc-bicategories and suppose that $\mathcal{C}$ has all pullbacks. Then for any fp-pseudofunctor $(\mathfrak{J}, \mathrm{q}^{\times}) : (\mathcal{B}, \Pi_n(-)) \to (\mathcal{C}, \Pi_n(-))$ the glueing bicategory $\mathrm{gl}(\mathfrak{J})$ is cartesian closed with forgetful pseudofunctor $\pi_{\mathrm{dom}} : \mathrm{gl}(\mathfrak{J}) \to \mathcal{B}$ strictly preserving products and exponentials. □

Much of the complication in the definitions that follow arises from the invertible 2-cells moving 1-cells in and out of products; where the product structure is strict, the exponentials in $\mathrm{gl}(\mathfrak{J})$ are given similarly to the 1-categorical case. The reader happy to employ Power's coherence result for fp-bicategories (Proposition 4.1.8) may therefore greatly simplify the definitions just given and the calculations to come. Because we wish to prove an independent coherence result, we do not take this approach.

We begin by defining the mapping $(-) \Rightarrow (=)$ and the evaluation 1-cell $\underline{\mathrm{eval}}$.

Defining $(-) \Rightarrow (=)$ and $\underline{\mathrm{eval}}$. For $\underline{C} := (C, c, B)$ and $\underline{C'} := (C', c', B')$ in gl($\mathfrak{J}$) we set the exponential $\underline{C} \Rightarrow \underline{C'}$ to be the left-hand vertical leg of the following pullback diagram, in which $\mathrm{m}_{B,B'}$ is the exponential transpose of $\mathfrak{J}(\mathrm{eval}_{B,B'}) \circ \mathrm{q}^{\times}$ $\big($*c.f.* the definition in the 1-categorical case (7.2)$\big)$:

$$\begin{tikzcd}
C \supset C' \arrow[rr, "q_{c,c'}"] \arrow[d, "p_{c,c'}"'] \arrow[dr, phantom, very near start, "\lrcorner"] & \overset{\omega_{c,c'}}{\Leftarrow} & (C \Rightarrow C') \arrow[d, "\lambda(c' \circ \mathrm{eval}_{C,C'})"] \\
\mathfrak{J}(B \Rightarrow B') \arrow[r, "\mathrm{m}_{B,B'}"'] \arrow[rr, bend right, "\lambda(\mathrm{eval}_{\mathfrak{J}B,\mathfrak{J}B'} \circ ((\mathfrak{J}B \Rightarrow \mathfrak{J}B') \times c)) \circ \mathrm{m}_{B,B'}"'] & (\mathfrak{J}B \Rightarrow \mathfrak{J}B') \arrow[r, "\lambda(\mathrm{eval}_{\mathfrak{J}B,\mathfrak{J}B'} \circ ((\mathfrak{J}B \Rightarrow \mathfrak{J}B') \times c))"'] & (C \Rightarrow \mathfrak{J}B')
\end{tikzcd} \tag{7.11}$$

We use $\lambda(c' \circ \mathrm{eval}_{C,C'})$ and $\lambda(\mathrm{eval}_{\mathfrak{J}B,\mathfrak{J}B'} \circ ((\mathfrak{J}B \Rightarrow \mathfrak{J}B') \times c))$ instead of $(\mathfrak{J}B \Rightarrow c)$ and $(C \Rightarrow c')$ as a simplifying measure: doing so avoids the need to apply the isomorphisms $(\mathfrak{J}B \Rightarrow c) \cong \lambda(c' \circ \mathrm{eval}_{C,C'})$ and $(C \Rightarrow c') \cong \lambda(\mathrm{eval}_{\mathfrak{J}B,\mathfrak{J}B'} \circ ((\mathfrak{J}B \Rightarrow \mathfrak{J}B') \times c))$ removing the redundant identities in the left-hand side (recall the comment after Notation 5.1.3).

Notation 7.3.10. For reasons of space—particularly for fitting pasting diagrams onto a single page—we will sometimes write $\widetilde{c} := \mathrm{eval}_{\mathfrak{J}B,\mathfrak{J}B'} \circ ((\mathfrak{J}B \Rightarrow \mathfrak{J}B') \times c)$ where $c : C \to \mathfrak{J}B$ in $\mathcal{C}$ (see, for example, (7.12)). ◂

For the evaluation 1-cell $\underline{\mathrm{eval}}$ we take the 1-cell with components

$$(C \supset C') \times C \xrightarrow{q_{c,c'} \times C} (C \Rightarrow C') \times C \xrightarrow{\mathrm{eval}_{C,C'}} C'$$

$$(B \Rightarrow B') \times B \xrightarrow{\mathrm{eval}_{B,B'}} B'$$

The witnessing 2-cell $\mathrm{E}_{\underline{C},\underline{C'}}$ is given by the following pasting diagram.

$$\begin{tikzcd}
(C \supset C') \times C \arrow[rrr, "q_{c,c'} \times C"] \arrow[rrrr, bend left=20, "\mathrm{eval}_{C,C'} \circ (q_{c,c'} \times C)"] \arrow[dd, "p_{c,c'} \times c"] \arrow[dr, "p_{c,c'} \times C"] \arrow[ddd, bend right=50, swap, "\mathrm{q}^{\times}_{(B \Rightarrow B', B)} \circ (p_{c,c'} \times c)"] & & \arrow[d, phantom, "\overset{\omega_{c,c'} \times C}{\cong}"] & (C \Rightarrow C') \times C \arrow[r] \arrow[d, "\lambda(c' \circ \mathrm{eval}_{C,C'}) \times C"] & C' \arrow[ddd, "c'"] \\
\arrow[r, phantom, "\overset{\Phi}{\cong}"] & \mathfrak{J}(B \Rightarrow B') \times C \arrow[r, "\mathrm{m}_{B,B'} \times C"] \arrow[dl, "\mathfrak{J}(B \Rightarrow B') \times c"] \arrow[dr, phantom, "\cong"] & (\mathfrak{J}B \Rightarrow \mathfrak{J}B') \times C \arrow[r, "\lambda\tilde{c} \times C"] \arrow[d, "(\mathfrak{J}B \Rightarrow \mathfrak{J}B') \times c"] & (C \Rightarrow \mathfrak{J}B') \times C \arrow[r, phantom, "\overset{\varepsilon}{\cong}"] \arrow[ddr, "\mathrm{eval}_{C,\mathfrak{J}B'}"] \arrow[d, phantom, "\overset{\varepsilon}{\cong}"] & {} \\
\mathfrak{J}(B \Rightarrow B') \times \mathfrak{J}B \arrow[rr, swap, "\mathrm{m}_{B,B'} \times \mathfrak{J}B"] \arrow[d, "\mathrm{q}^{\times}_{(B \Rightarrow B', B)}"] & & (\mathfrak{J}B \Rightarrow \mathfrak{J}B') \times \mathfrak{J}B \arrow[drr, "\mathrm{eval}_{\mathfrak{J}B,\mathfrak{J}B'}"] \arrow[d, phantom, "\overset{\varepsilon}{\cong}"] & {} & \\
\mathfrak{J}\left((B \Rightarrow B') \times B\right) \arrow[rrrr, swap, "\mathfrak{J}\mathrm{eval}_{B,B'}"] & & {} & & \mathfrak{J}B'
\end{tikzcd} \tag{7.12}$$

Here we omit the canonical 2-cells for the product structure: thus, the shape labelled $\omega_{c,c'} \times C$ is actually the composite

$$\begin{tikzcd}
\big(\lambda(c' \circ \mathrm{eval}_{C,C'}) \times C\big) \circ \big(q_{c,c'} \times C\big) \arrow[r] \arrow[d, "\Phi_{\lambda(c' \circ \mathrm{eval}),q;\mathrm{Id}}"'] & (\lambda\tilde{c} \times C) \circ ((\mathrm{m}_{B,B'} \times C) \circ (p_{c,c'} \times C)) \\
\big(\lambda(c' \circ \mathrm{eval}_{C,C'}) \circ q_{c,c'}\big) \times (\mathrm{Id}_C \circ \mathrm{Id}_C) \arrow[d, "\cong"'] & \\
\big(\lambda(c' \circ \mathrm{eval}_{C,C'}) \circ q_{c,c'}\big) \times C \arrow[r, swap, "\omega_{c,c'} \times C"] & \big(\lambda\tilde{c} \circ \mathrm{m}_{B,B'} \circ p_{c,c'}\big) \times C \arrow[uu, "\cong"']
\end{tikzcd}$$

in which the unlabelled isomorphism employs two applications of Φ^{-1}, together with the evident structural isomorphisms.

Notation 7.3.11. For the rest of this chapter we will adopt the convention just employed, and write simply $\cong$ for instances of either Φ or its inverse, composed with structural isomorphisms. Power's coherence result guarantees that this is valid as an explanatory shorthand: of course, the masochistic reader could work explicitly with all the instances of Φ and prove exactly the same set of diagrams commute. Thus, while Power's result is useful *for reasons of exposition and presentation*, the proofs we present do not rely on it. ◂

With this convention, $\mathrm{E}_{\underline{C},\underline{C'}}$ is the following composite:

$$\begin{tikzcd}
c' \circ (\mathrm{eval}_{C,C'} \circ (q_{c,c'} \times C)) \arrow[r, "\mathrm{E}_{\underline{C},\underline{C'}}"] \arrow[d, "\cong"'] & \mathfrak{J}(\mathrm{eval}_{B,B'}) \circ \big(\mathrm{q}^{\times}_{(B \Rightarrow B',B)} \circ (p_{c,c'} \times c)\big) \\
(c' \circ \mathrm{eval}_{C,C'}) \circ (q_{c,c'} \times C) \arrow[d, "\varepsilon^{-1}_{(c' \circ \mathrm{eval})} \circ (q_{c,c'} \times C)"'] & \\
(\mathrm{eval}_{C,C'} \circ (\lambda(c' \circ \mathrm{eval}_{C,C'}) \times C)) \circ (q_{c,c'} \times C) \arrow[d, "\cong"'] & \big(\mathfrak{J}(\mathrm{eval}_{B,B'}) \circ \mathrm{q}^{\times}_{(B \Rightarrow B',B)}\big) \circ (p_{c,c'} \times c) \arrow[uu, "\cong"'] \\
\mathrm{eval}_{C,C'} \circ \big(\lambda(c' \circ \mathrm{eval}_{C,C'}) \circ q_{c,c'}\big) \times C \arrow[d, "\mathrm{eval} \circ (\omega_{c,c'} \times C)"'] & (\mathrm{eval}_{\mathfrak{J}B,\mathfrak{J}B'} \circ (\mathrm{m}_{B,B'} \times \mathfrak{J}B)) \circ (p_{c,c'} \times c) \arrow[u, "\varepsilon_{(\mathfrak{J}\mathrm{eval} \circ \mathrm{q}^{\times})} \circ (p_{c,c'} \times c)"'] \\
\mathrm{eval}_{C,C'} \circ ((\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}) \circ p_{c,c'}) \times C \arrow[d, "\cong"'] & \\
(\mathrm{eval}_{C,C'} \circ (\lambda\widetilde{c} \times C)) \circ (\mathrm{m}_{B,B'} p_{c,c'} \times C) \arrow[r, "\varepsilon_{\widetilde{c}} \circ (\mathrm{m}_{B,B'} p_{c,c'} \times C)"'] & \widetilde{c} \circ (\mathrm{m}_{B,B'} p_{c,c'} \times C) \arrow[uu, "\cong"']
\end{tikzcd} \tag{7.13}$$

The mapping $\underline{\lambda}$. Next we need to provide a mapping $\underline{\lambda}$ assigning a 1-cell of type $\underline{R} \to (\underline{C} \Rightarrow \underline{C'})$ to every 1-cell $\underline{R} \times \underline{C} \to \underline{C'}$. Let $\underline{R} := (R, r, Q)$, $\underline{C} := (C, c, B)$ and $\underline{C'} := (C', c', B')$. As our starting point, suppose given a 1-cell $(t, \alpha, s) : \underline{R} \times \underline{C} \to \underline{C'}$, as on the left below:

$$\begin{tikzcd}
R \times C \arrow[r, "t"] \arrow[d, "r \times c"'] \arrow[dd, bend right=60, "\mathrm{q}^{\times}_{Q,B} \circ (r \times c)"'] & C' \arrow[dd, "c'"] \\
\mathfrak{J}Q \times \mathfrak{J}B \arrow[d, "\mathrm{q}^{\times}_{Q,B}"'] \arrow[r, phantom, "\overset{\alpha}{\Leftarrow}"] & \\
\mathfrak{J}(Q \times B) \arrow[r, "\mathfrak{J}s"'] & \mathfrak{J}B'
\end{tikzcd}
\qquad
\begin{tikzcd}
R \arrow[rr, "\lambda t"] \arrow[d, "r"'] \arrow[dd, bend right=60, "\mathfrak{J}(\lambda s) \circ r"'] & & C \Rightarrow C' \arrow[dd, "\lambda(c' \circ \mathrm{eval}_{C,C'})"] \\
\mathfrak{J}Q \arrow[d, "\mathfrak{J}\lambda s"'] & \overset{\mathrm{L}_{\alpha}}{\Leftarrow} & \\
\mathfrak{J}(B \Rightarrow B') \arrow[r, "\mathrm{m}_{B,B'}"'] \arrow[rr, bend right=20, "\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}"'] & \mathfrak{J}B \Rightarrow \mathfrak{J}B' \arrow[r, "\lambda\widetilde{c}"'] & C \Rightarrow \mathfrak{J}B'
\end{tikzcd}$$

We construct a 2-cell L_{α} as on the right above and apply the universal property of the pullback (7.11). To this end, let us define two invertible composites, which we denote by T_{α} and U_{α}. For T_{α} we take

$$\begin{tikzcd}
\mathrm{eval}_{C,\mathfrak{J}B'} \circ \big(\lambda(c' \circ \mathrm{eval}_{C,C'}) \circ \lambda t\big) \times C \arrow[rr, "\mathrm{T}_{\alpha}"] \arrow[d, "\cong"'] & & c' \circ t \\
(\mathrm{eval}_{C,\mathfrak{J}B'} \circ (\lambda(c' \circ \mathrm{eval}_{C,C'}) \times C)) \circ (\lambda t \times C) \arrow[dr, "\varepsilon_{(c' \circ \mathrm{eval})} \circ (\lambda t \times C)"'] & & c' \circ (\mathrm{eval}_{C,C'} \circ (\lambda t \times C)) \arrow[u, "c' \circ \varepsilon_{t}"'] \\
& (c' \circ \mathrm{eval}_{C,C'}) \circ (\lambda t \times C) \arrow[ur, "\cong"'] &
\end{tikzcd}$$

and for U_α we take

$$\begin{tikzcd}
\mathrm{eval}_{C,\mathfrak{J}B} \circ \big((\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}) \circ (\mathfrak{J}(\lambda s) \circ r) \big) \times C \arrow[r, "\mathrm{U}_\alpha"] \arrow[d, "\cong"'] & \mathfrak{J}s \circ \Big(\mathrm{q}^{\times}_{Q,B} \circ (r \times c) \Big) \\
\mathrm{eval}_{C,\mathfrak{J}B} \circ (\lambda\widetilde{c} \times C) \circ \big(\mathrm{m}_{B,B'} \circ (\mathfrak{J}(\lambda s) \circ r) \big) \times C \arrow[d, "\varepsilon_{\widetilde{c}} \circ (\mathrm{m}_{B,B'} \circ \mathfrak{J}(\lambda s) \circ r) \times C"'] & \\
\widetilde{c} \circ \big(\mathrm{m}_{B,B'} \circ (\mathfrak{J}(\lambda s) \circ r) \big) \times C \arrow[d, "\cong"'] & \mathfrak{J}\big(\mathrm{eval}_{B,B'} \circ (\lambda s \times B)\big) \circ \Big(\mathrm{q}^{\times}_{Q,B} \circ (r \times c) \Big) \arrow[uu, "\mathfrak{J}\varepsilon_s \circ \mathrm{q}^{\times} \circ (r \times c)"'] \\
\big(\mathrm{eval}_{\mathfrak{J}B,\mathfrak{J}B'} \circ (\mathrm{m}_{B,B'} \times \mathfrak{J}B) \big) \circ ((\mathfrak{J}(\lambda s) \times \mathfrak{J}B) \circ (r \times c)) \arrow[d, "\varepsilon_{(\mathfrak{J}\mathrm{eval} \circ \mathrm{q}^{\times})} \circ (\mathfrak{J}(\lambda s) \times \mathfrak{J}B) \circ (r \times c)"'] & \big(\mathfrak{J}(\mathrm{eval}_{B,B'}) \circ \mathfrak{J}(\lambda s \times B) \big) \circ \Big(\mathrm{q}^{\times}_{Q,B} \circ (r \times c) \Big) \arrow[u, "\phi^{\mathfrak{J}}_{\mathrm{eval}, \lambda s \times B} \circ \mathrm{q}^{\times} \circ (r \times c)"'] \\
\Big(\mathfrak{J}(\mathrm{eval}_{B,B'}) \circ \mathrm{q}^{\times}_{(B \Rightarrow B', B)} \Big) \circ ((\mathfrak{J}(\lambda s) \times \mathfrak{J}\mathrm{Id}_B) \circ (r \times c)) \arrow[dr, bend right, "\cong"'] & \mathfrak{J}(\mathrm{eval}_{B,B'}) \circ \Big(\Big(\mathfrak{J}(\lambda s \times B) \circ \mathrm{q}^{\times}_{Q,B} \Big) \circ (r \times c) \Big) \arrow[u, "\cong"'] \\
& \mathfrak{J}(\mathrm{eval}_{B,B'}) \circ \big(\big(\mathrm{q}^{\times}_{(B \Rightarrow B', B)} \circ (\mathfrak{J}(\lambda s) \times \mathfrak{J}\mathrm{Id}_B) \big) \circ (r \times c) \big) \arrow[u, "\mathfrak{J}(\mathrm{eval}_{B,B'}) \circ \mathsf{nat} \circ (r \times c)"']
\end{tikzcd}$$

We may therefore define a 2-cell K_α as the composite

$$\begin{tikzcd}
\mathrm{eval}_{C,\mathfrak{J}B'} \circ (\lambda(c' \circ \mathrm{eval}_{C,C'}) \circ \lambda t) \times C \arrow[r, "\mathrm{K}_\alpha"] \arrow[d, "\mathrm{T}_\alpha"'] & \mathrm{eval}_{C,\mathfrak{J}B} \circ ((\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}) \circ (\mathfrak{J}(\lambda s) \circ r)) \times C \\
c' \circ t \arrow[r, "\alpha"'] & \mathfrak{J}s \circ \Big(\mathrm{q}^{\times}_{Q,B} \circ (r \times c) \Big) \arrow[u, "\mathrm{U}_\alpha^{-1}"']
\end{tikzcd}$$

and, finally, L_α as

$$\begin{tikzcd}
\lambda(c' \circ \mathrm{eval}_{C,C'}) \circ \lambda t \arrow[rr, "\mathrm{L}_\alpha"] \arrow[dr, bend left, "\mathrm{e}^{\dagger}(\mathrm{K}_\alpha)"'] & & (\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}) \circ (\mathfrak{J}(\lambda s) \circ r) \\
& \lambda\big(\mathrm{eval}_{C,\mathfrak{J}B} \circ ((\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}) \circ (\mathfrak{J}(\lambda s) \circ r)) \times C\big) \arrow[ur, bend right, "\eta^{-1}"'] &
\end{tikzcd}$$

Since we work in the pseudo setting, U_α, T_α, K_α—and hence L_α—are all invertible.

Now, L_α fills the following diagram:

$$\begin{tikzcd}
R \arrow[r, "\lambda t"] \arrow[d, "\mathfrak{J}(\lambda s) \circ r"'] \arrow[dr, phantom, "\overset{\mathrm{L}_\alpha}{\cong}"] & (C \Rightarrow C') \arrow[d, "\lambda(c' \circ \mathrm{eval}_{C,C'})"] \\
\mathfrak{J}(B \Rightarrow B') \arrow[r, "\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}"'] & (C \Rightarrow \mathfrak{J}B')
\end{tikzcd} \tag{7.14}$$

Hence, by the universal property of the pullback (7.11), one obtains a 1-cell $\underline{\mathsf{lam}}(t)$ and a pair of invertible 2-cells $\Gamma_{c,c'}$ and $\Delta_{c,c'}$ filling the diagram

$$\begin{tikzcd}
R \arrow[dr, dashed, "\underline{\mathsf{lam}}(t)"] \arrow[drr, bend left, "\lambda t"] \arrow[ddr, bend right, "\mathfrak{J}(\lambda s)\circ r"'] \arrow[dr, phantom, shift right=6ex, "\overset{\Gamma_{c,c'}}{\Leftarrow}"] & & \\
& C \supset C' \arrow[r, "q_{c,c'}"] \arrow[d, "p_{c,c'}"'] \arrow[dr, phantom, very near start, "\lrcorner"] \arrow[ur, phantom, shift left=3ex, "\overset{\Delta_{c,c'}}{\Rightarrow}"] \arrow[dr, phantom, "\overset{\omega_{c,c'}}{\Leftarrow}"] & (C \Rightarrow C') \arrow[d, "\lambda(c'\circ\mathrm{eval}_{C,C'})"] \\
& \mathfrak{J}(B \Rightarrow B') \arrow[r, "\lambda\widetilde{c}\circ\mathrm{m}_{B,B'}"'] & (C \Rightarrow \mathfrak{J}B')
\end{tikzcd} \tag{7.15}$$

such that the pasting diagrams (7.14) and (7.15) are equal, *i.e.* the following commutes:

$$\begin{tikzcd}
& \lambda(c' \circ \mathrm{eval}_{C,C'}) \circ (q_{c,c'} \circ \underline{\mathsf{lam}}(t)) \arrow[dr, "\lambda(c'\circ\mathrm{eval}_{C,C'})\circ\Delta_{c,c'}"] & \\
(\lambda(c' \circ \mathrm{eval}_{C,C'}) \circ q_{c,c'}) \circ \underline{\mathsf{lam}}(t) \arrow[ur, "\cong"] \arrow[d, "\omega_{c,c'}\circ\underline{\mathsf{lam}}(t)"'] & & \lambda(c' \circ \mathrm{eval}_{C,C'}) \circ \lambda t \arrow[d, "\mathrm{L}_\alpha"] \\
((\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}) \circ p_{c,c'}) \circ \underline{\mathsf{lam}}(t) \arrow[dr, "\cong"'] & & (\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}) \circ (\mathfrak{J}(\lambda s) \circ r) \\
& (\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}) \circ (p_{c,c'} \circ \underline{\mathsf{lam}}(t)) \arrow[ur, "\lambda\widetilde{c}\circ\mathrm{m}_{B,B'}\circ\Gamma_{c,c'}"'] &
\end{tikzcd} \tag{7.16}$$

Moreover, $\Gamma_{c,c'}$ and $\Delta_{c,c'}$ are universal in the sense of Lemma 7.3.6. We define $\underline{\lambda}(t, \alpha, s) := \big(\underline{\mathsf{lam}}(t), \Gamma_{c,c'}, \lambda s\big)$.

The counit $\underline{\varepsilon}$. Finally we come to the counit. Let us first calculate $\underline{\mathrm{eval}} \circ \big(\underline{\lambda}(t, \alpha, s) \times (C, c, B)\big)$ for a 1-cell $\underline{t} := (t, \alpha, s) : (R, r, Q) \times (C, c, B) \to (C', c', B')$. Using Lemma 7.3.4, one unwinds this 1-cell to the following pasting diagram, in which we omit the canonical isomorphisms for the product structure as well as the structural isomorphisms:

$$\begin{tikzcd}[column sep=large, row sep=large]
R \times C \arrow[r, "\underline{\mathsf{lam}}(t)\times C"] \arrow[d, "r\times c"] \arrow[dd, bend right=60, "\overset{\times}{\mathrm{q}}_{Q,B}\circ(r\times c)"'] \arrow[rr, bend left=30, "(\mathrm{eval}_{C,C'} \circ (q_{c,c'} \times C)) \circ (\underline{\mathsf{lam}}(t) \times C)"] \arrow[from=1-1, to=2-2, phantom, "\overset{\Gamma_{c,c'}\times c}{\Leftarrow}"] & (C \supset C') \times C \arrow[r, "\mathrm{eval}_{C,C'}\circ(q_{c,c'}\times C)"] \arrow[d, "q_{c,c'}\times c"] & C' \arrow[dd, "c'"] \\
\mathfrak{J}Q \times \mathfrak{J}B \arrow[r, bend left, "\mathfrak{J}(\lambda s)\times\mathfrak{J}B"] \arrow[r, bend right, "\mathfrak{J}(\lambda s)\times\mathfrak{J}\mathrm{Id}_B"'] \arrow[r, phantom, "\overset{\mathfrak{J}(\lambda s)\times\psi^{\mathfrak{J}}_B}{\cong}"] \arrow[d, "\overset{\times}{\mathrm{q}}_{Q,B}"] \arrow[from=2-1, to=3-2, phantom, "\overset{\mathsf{nat}}{\cong}"] & \mathfrak{J}(B \Rightarrow B') \times \mathfrak{J}B \arrow[d, "\overset{\times}{\mathrm{q}}_{(B \Rightarrow B', B)}"] \arrow[r, phantom, "\overset{\mathrm{E}_{\underline{C,C'}}}{\cong}"] & {} \\
\mathfrak{J}(Q \times B) \arrow[r, "\mathfrak{J}(\lambda s\times B)"'] \arrow[rr, bend right=30, "\mathfrak{J}(\mathrm{eval}_{B,B'} \circ (\lambda s \times B))"'] \arrow[rr, phantom, bend right=12, "\overset{\phi^{\mathfrak{J}}}{\cong}"] & \mathfrak{J}\big((B \Rightarrow B') \times B\big) \arrow[r, "\mathfrak{J}\mathrm{eval}_{B,B'}"'] & \mathfrak{J}B'
\end{tikzcd}$$

For the counit $\underline{\varepsilon}_{\underline{t}}$ we therefore take the 2-cell with first component $\underline{\mathsf{e}}_t$ defined by

$$\begin{tikzcd}
(\mathrm{eval}_{C,C'} \circ (q_{c,c'} \times C)) \circ (\underline{\mathsf{lam}}(t) \times C) \arrow[r, "\mathsf{e}_t"] \arrow[d, "\cong"'] & t \\
\mathrm{eval}_{C,C'} \circ (q_{c,c'} \circ \underline{\mathsf{lam}}(t)) \times C \arrow[r, "\mathrm{eval}_{C,C'}\circ(\Delta_{c,c'}\times C)"'] & \mathrm{eval}_{C,C'} \circ (\lambda t \times C) \arrow[u, "\varepsilon_t"']
\end{tikzcd} \tag{7.17}$$

and second component simply

$$\mathrm{eval}_{B,B'} \circ (\lambda s \times B) \overset{\varepsilon_s}{\Rightarrow} s$$

We need to check that this to be a legitimate 2-cell in $\mathrm{gl}(\mathfrak{J})$, *i.e.* that the cylinder condition holds.

Lemma 7.3.12. For any objects $\underline{R} := (R, r, Q)$, $\underline{C} := (C, c, B)$ and $\underline{C'} := (C', c', B')$ and 1-cell $\underline{t} := (t, \alpha, s) : \underline{R} \times \underline{C} \to \underline{C'}$ in gl($\mathfrak{J}$), the pasting diagram

$$\begin{tikzcd}[column sep=huge, row sep=large]
R\times C \arrow[r, "\underline{\mathsf{lam}}(t)\times C"] \arrow[d, "r\times c"'] \arrow[dd, bend right=60, "\overset{\times}{\mathrm{q}}_{Q,B}\circ(r\times c)"'] \arrow[rr, bend left=40, "(\mathrm{eval}_{C,C'}\circ(q_{c,c'}\times C))\circ(\underline{\mathsf{lam}}(t)\times C)"] \arrow[dr, phantom, "\overset{\Gamma_{c,c'}\times c}{\Leftarrow}"] & (C\supset C')\times C \arrow[r, "\mathrm{eval}_{C,C'}\circ(q_{c,c'}\times C)"] \arrow[d, "q_{c,c'}\times c"] & C' \arrow[dd, "c'"] \\
\mathfrak{J}Q\times\mathfrak{J}B \arrow[r, bend left, "\mathfrak{J}(\lambda s)\times\mathfrak{J}B"] \arrow[r, bend right, "\mathfrak{J}(\lambda s)\times\mathfrak{J}\mathrm{Id}_B"'] \arrow[r, phantom, "\overset{\mathfrak{J}(\lambda s)\times\psi^{\mathfrak{J}}_B}{\cong}"] \arrow[d, "\overset{\times}{\mathrm{q}}_{Q,B}"] \arrow[dr, phantom, "\overset{\mathsf{nat}}{\cong}"] & \mathfrak{J}(B\Rightarrow B')\times\mathfrak{J}B \arrow[d, "\overset{\times}{\mathrm{q}}_{(B\Rightarrow B',B)}"] \arrow[r, phantom, "\overset{\mathrm{E}_{\underline{C},\underline{C'}}}{\cong}"] & \\
\mathfrak{J}(Q\times B) \arrow[r, "\mathfrak{J}(\lambda s\times B)"'] \arrow[rr, bend right=30, "\mathfrak{J}(\mathrm{eval}_{B,B'}\circ(\lambda s\times B))"'] \arrow[rr, bend right=50, "\mathfrak{J}s"'] \arrow[rr, phantom, bend right=15, "\overset{\phi^{\mathfrak{J}}}{\cong}"] \arrow[rr, phantom, bend right=40, "\overset{\mathfrak{J}\varepsilon_s}{\cong}"] & \mathfrak{J}\big((B\Rightarrow B')\times B\big) \arrow[r, "\mathfrak{J}\mathrm{eval}_{B,B'}"'] & \mathfrak{J}B'
\end{tikzcd}$$

is equal to

$$\begin{tikzcd}[column sep=large, row sep=large]
\overset{}{\cong} & (C\supset C')\times C \arrow[r, "q_{c,c'}\times C"] \arrow[d, phantom, "\Downarrow\Delta_{c,c'}\times C"] & (C\Rightarrow C')\times C \arrow[dr, "\mathrm{eval}_{C,C'}"] \arrow[d, phantom, "\overset{\varepsilon_t}{\cong}"] & \cong \\
R\times C \arrow[ur, "\underline{\mathsf{lam}}(t)\times C"] \arrow[urr, "\lambda t\times C" description] \arrow[rrr, "t" description] \arrow[urr, bend left=60, "(q_{c,c'}\times C)\circ(\underline{\mathsf{lam}}(t)\times C)"] \arrow[rrr, bend left=70, "(\mathrm{eval}_{C,C'}\circ(q_{c,c'}\times C))\circ(\underline{\mathsf{lam}}(t)\times C)"] \arrow[d, "r\times c"] \arrow[dd, bend right=60, "\overset{\times}{\mathrm{q}}_{Q,B}\circ(r\times c)"'] & & & C' \arrow[dd, "c'"] \\
\mathfrak{J}Q\times\mathfrak{J}B \arrow[d, "\overset{\times}{\mathrm{q}}_{Q,B}"'] & & \overset{\alpha}{\Leftarrow} & \\
\mathfrak{J}(Q\times B) \arrow[rrr, "\mathfrak{J}s"'] & & & \mathfrak{J}B'
\end{tikzcd}$$

Hence $\underline{\varepsilon_{\underline{t}}} := (\underline{\mathsf{e}}_t, \varepsilon_s)$ is a 2-cell in gl($\mathfrak{J}$).

Proof. Unfolding the first diagram, one sees that it is equal to the composite

$$\begin{tikzcd}
c' \circ ((\mathrm{eval}_{C,C'} \circ (q_{c,c'} \times C)) \circ (\underline{\mathsf{lam}}(t) \times C)) \arrow[r] \arrow[d, "(*)"'] & \mathfrak{J}s \circ \left(\mathrm{q}^{\times}_{Q,B} \circ (r \times c)\right) \\
\mathrm{eval}_{\mathfrak{J}B,\mathfrak{J}B'} \circ (\mathrm{m}_{B,B'} \circ (p_{c,c'} \circ \underline{\mathsf{lam}}(t))) \times c \arrow[d, "\mathrm{eval}_{\mathfrak{J}B,\mathfrak{J}B'} \circ (\mathrm{m}_{B,B'} \circ \Gamma_{c,c'}) \times C"'] & \mathrm{eval}_{C,\mathfrak{J}B} \circ ((\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}) \circ (\mathfrak{J}(\lambda s) \circ r)) \times C \arrow[u, "\mathrm{U}_\alpha"'] \\
\mathrm{eval}_{\mathfrak{J}B,\mathfrak{J}B'} \circ (\mathrm{m}_{B,B'} \circ (\mathfrak{J}(\lambda s) \circ r)) \times c \arrow[dr, bend right, "\cong"'] & (\mathrm{eval}_{C,\mathfrak{J}B} \circ (\lambda\widetilde{c} \times C)) \circ (\mathrm{m}_{B,B'} \circ (\mathfrak{J}(\lambda s) \circ r)) \times C \arrow[u, "\cong"'] \\
 & (\mathrm{eval}_{B,B'} \circ \widetilde{c}) \circ (\mathrm{m}_{B,B'} \circ (\mathfrak{J}(\lambda s) \circ r)) \times C \arrow[u, "\varepsilon^{-1}_{\widetilde{c}} \circ (\mathrm{m}_{B,B'} \circ \mathfrak{J}(\lambda s) \circ r) \times C"']
\end{tikzcd}$$

where the arrow labelled $(*)$ arises by composing the following with structural isomorphisms and Φ:

$$\begin{tikzcd}
c' \circ (\mathrm{eval}_{C,C'} \circ (q_{c,c'} \times C)) \arrow[r] \arrow[d, "\cong"'] & \mathrm{eval}_{\mathfrak{J}B,\mathfrak{J}B'} \circ ((\mathrm{m}_{B,B'} \circ p_{c,c'}) \times c) \\
c' \circ \left(\mathrm{eval}_{C,C'} \circ (q_{c,c'} \times C)\right) \arrow[d, "\mathrm{E}_{C,C'}"'] & (\mathrm{eval}_{\mathfrak{J}B,\mathfrak{J}B'} \circ (\mathrm{m}_{B,B'} \times \mathfrak{J}B)) \circ (p_{c,c'} \times c) \arrow[u, "\cong"'] \\
\mathfrak{J}(\mathrm{eval}_{B,B'}) \circ \left(\mathrm{q}^{\times}_{(B \Rightarrow B',B)} \circ (p_{c,c'} \times c)\right) \arrow[r, "\cong"'] & \left(\mathfrak{J}(\mathrm{eval}_{B,B'}) \circ \mathrm{q}^{\times}_{(B \Rightarrow B',B)}\right) \circ (p_{c,c'} \times c) \arrow[u, "\varepsilon^{-1}_{\mathrm{eval} \circ (\mathrm{m} \times \mathfrak{J}B)} \circ (p_{c,c'} \times c)"']
\end{tikzcd}$$

Applying the coherence condition (7.16), the first diagram in the claim reduces further to

$$\begin{tikzcd}
c' \circ \left(\left(\mathrm{eval}_{C,C'} \circ (q_{c,c'} \times C)\right) \circ (\underline{\mathsf{lam}}(t) \times C)\right) \arrow[r] \arrow[d, "\cong"'] & \mathfrak{J}s \circ \left(\mathrm{q}^{\times}_{Q,B} \circ (r \times c)\right) \\
c' \circ \left(\mathrm{eval}_{C,C'} \circ \left((q_{c,c'} \circ \underline{\mathsf{lam}}(t)\right) \times C\right) \arrow[d, "c' \circ \mathrm{eval}_{C,C'} \circ (\Delta_{C,C'} \times C)"'] & \\
c' \circ \left(\mathrm{eval}_{C,C'} \circ (\lambda t \times C)\right) \arrow[d, "\cong"'] & \mathrm{eval}_{\mathfrak{J}B,C'} \circ \left(\left(\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}\right) \circ (\mathfrak{J}(\lambda s) \circ r)\right) \times C \arrow[uu, "\mathrm{U}_\alpha"'] \\
\left(c' \circ \mathrm{eval}_{C,C'}\right) \circ (\lambda t \times C) \arrow[d, "\varepsilon^{-1}_{(c' \circ \mathrm{eval})} \circ (\lambda t \times C)"'] & \\
\left(\mathrm{eval}_{\mathfrak{J}B,C'} \circ (\lambda(c' \circ \mathrm{eval}_{C,C'}) \times C)\right) \circ (\lambda t \times C) \arrow[r, "\cong"'] & \mathrm{eval}_{\mathfrak{J}B,C'} \circ \left(\lambda(c' \circ \mathrm{eval}_{C,C'}) \circ \lambda t\right) \times C \arrow[uu, "\mathrm{eval}_{\mathfrak{J}B,C'} \circ (\mathrm{L}_\alpha \times C)"']
\end{tikzcd} \tag{7.18}$$

Next, by the definition of L_α and the triangle law relating η and ε, one sees that

$$\begin{tikzcd}
 & \mathrm{eval}_{\mathfrak{J}B,C} \circ (\lambda h \times C) \arrow[r, "\varepsilon_h"] \arrow[dr, "\mathrm{eval}_{\mathfrak{J}B,C} \circ \lambda(\eta_h^{-1} \times C)" description] & h \arrow[d, equal, bend left] \\
\mathrm{eval}_{\mathfrak{J}B,C} \circ \left(\lambda(c' \circ \mathrm{eval}_{C,C'}) \circ \lambda t\right) \times C \arrow[ur, "\mathrm{eval}_{\mathfrak{J}B,C} \circ (\mathrm{e}^{\dagger}(\mathrm{K}_\alpha) \times C)" description] \arrow[urr, bend left=40, "\mathrm{K}_\alpha"] \arrow[rr, "\mathrm{eval}_{\mathfrak{J}B,C'} \circ (\mathrm{L}_\alpha \times C)"'] & & h
\end{tikzcd}$$

for $h := \mathrm{eval}_{C,\mathfrak{J}B} \circ ((\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}) \circ (\mathfrak{J}(\lambda s) \circ r)) \times C$. Hence, the composite (7.18) is equal to the anti-clockwise route around the diagram below, in which (†) abbreviates

$$(c' \circ \mathrm{eval}_{C,C'}) \circ (\lambda t \times C) \overset{\cong}{\Rightarrow} c' \circ (\mathrm{eval}_{C,C'} \circ (\lambda t \times C)) \xRightarrow{c' \circ \varepsilon_t} c' \circ t$$

and the bottom two shapes commute by definition:

$$\begin{tikzcd}
c' \circ ((\mathrm{eval}_{C,C'} \circ (q_{c,c'} \times C)) \circ (\underline{\mathsf{lam}}(t) \times C)) \arrow[d, "\cong"'] & & \\
(c' \circ \mathrm{eval}_{C,C'}) \circ \big(q_{c,c'} \circ \underline{\mathsf{lam}}(t)\big) \times C \arrow[d, "c' \circ \mathrm{eval}_{C,C'} \circ (\Delta_{c,c'} \times C)"'] & & \\
(c' \circ \mathrm{eval}_{C,C'}) \circ (\lambda t \times C) \arrow[d, "\varepsilon^{-1}_{(c' \circ \mathrm{eval})} \circ (\lambda t \times C)"'] \arrow[dr, equal, bend left] & & \\
(\mathrm{eval}_{\mathfrak{J}B,C'} \circ (\lambda(c' \circ \mathrm{eval}_{C,C'}) \times C)) \circ (\lambda t \times C) \arrow[r, "\varepsilon_{(c' \circ \mathrm{eval})} \circ (\lambda t \times C)"'] \arrow[d, "\cong"'] & (c' \circ \mathrm{eval}_{C,C'}) \circ (\lambda t \times C) \arrow[r, "(\dagger)"] & c' \circ t \arrow[dd, "\alpha"] \\
\mathrm{eval}_{\mathfrak{J}B,C'} \circ \big(\lambda(c' \circ \mathrm{eval}_{C,C'}) \circ \lambda t\big) \times C \arrow[d, "\mathrm{K}_\alpha"'] \arrow[urr, "\mathrm{T}_\alpha" description] & & \\
\mathrm{eval}_{\mathfrak{J}B,C'} \circ ((\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}) \circ (\mathfrak{J}(\lambda s) \circ r)) \times C \arrow[rr, "\mathrm{U}_\alpha"'] & & \mathfrak{J}s \circ \Big(\mathrm{q}^{\times}_{Q,B} \circ (r \times c)\Big)
\end{tikzcd}$$

The clockwise route around this diagram is equal to the 2-cell given by the second diagram in the claim, so the proof is complete. □

We have now constructed all the data we shall require. It remains to show that, together, it defines an adjoint equivalence

$$\underline{\lambda} : \mathrm{gl}(\mathfrak{J})\big(\underline{R} \times \underline{C}, \underline{C'}\big) \leftrightarrows \mathrm{gl}(\mathfrak{J})\big(\underline{R}, \underline{C} \Rightarrow \underline{C'}\big) : \underline{\mathrm{eval}}_{\underline{C},\underline{C'}} \circ (- \times \underline{C})$$

Thus, we need to check that for every pair of 1-cells $\underline{g} : \underline{R} \to (\underline{C} \Rightarrow \underline{C'})$ and $\underline{t} : \underline{R} \times \underline{C} \to \underline{C'}$ related by a 2-cell $\underline{\tau} := (\tau, \sigma) : \underline{\mathrm{eval}}_{\underline{C},\underline{C'}} \circ (\underline{g} \times \underline{C}) \Rightarrow \underline{t}$, there exists a 2-cell $\mathsf{e}^{\dagger}(\underline{\tau}) : \underline{g} \Rightarrow \underline{\lambda t}$, unique such that

$$\begin{tikzcd}
\underline{\mathrm{eval}}_{\underline{C},\underline{C'}} \circ (\underline{g} \times \underline{C}) \arrow[rr, "\underline{\mathrm{eval}}_{\underline{C},\underline{C'}} \circ (\mathsf{e}^{\dagger}(\underline{\tau}) \times \underline{C})"] \arrow[dr, "\underline{\tau}"'] & & \underline{\mathrm{eval}}_{\underline{C},\underline{C'}} \circ (\underline{\lambda t} \times \underline{C}) \arrow[dl, "\underline{\varepsilon}_t"] \\
& \underline{t} &
\end{tikzcd} \tag{7.19}$$

We turn to this next.

Universality of $\underline{\varepsilon} = (\underline{\mathsf{e}}, \varepsilon)$. We begin with the existence part of the claim. Let $\underline{g} := (g, \gamma, f) : (R, r, Q) \to (C \supset C', p_{c,c'}, B \Rightarrow\!\!\!> B')$ and $\underline{t} := (t, \alpha, s) : (R \times C, \mathrm{q}^{\times}_{Q,B} \circ (r \times c), Q \times B) \to (C', c', B')$ be 1-cells and suppose that $\underline{\tau} := (\tau, \sigma) : \underline{\mathrm{eval}}_{\underline{C},\underline{C'}} \circ (\underline{g} \times \underline{C}) \Rightarrow \underline{t}$. Thus, τ and σ have type

$$\tau : (\mathrm{eval}_{C,C'} \circ (q_{c,c'} \times C)) \circ (g \times C) \Rightarrow t$$
$$\sigma : \mathrm{eval}_{B,B'} \circ (f \times B) \Rightarrow s$$

and we are required to provide 2-cells $\tau^{\sharp}$ and $\sigma^{\sharp}$ of type

$$\tau^{\sharp} : g \Rightarrow \underline{\mathsf{lam}}(t)$$
$$\sigma^{\sharp} : f \Rightarrow \lambda s$$

satisfying the cylinder condition. For the second component we can simply take $\mathsf{e}^{\dagger}(\sigma)$. For the first component we use the universal property of pullbacks. We aim to construct a pair of 2-cells

$$p_{c,c'} \circ g \Rightarrow \mathfrak{J}(\lambda s) \circ r$$
$$q_{c,c'} \circ g \Rightarrow \lambda t$$

such that the coherence condition (7.16) holds. We claim that the following 2-cells suffice

$$\begin{aligned} \Sigma_1 &:= p_{c,c'} \circ g \overset{\gamma}{\Rightarrow} \mathfrak{J}(f) \circ r \xRightarrow{\mathfrak{J}(\mathsf{e}^{\dagger}(\sigma)) \circ r} \mathfrak{J}(\lambda s) \circ r \\ \Sigma_2 &:= q_{c,c'} \circ g \xRightarrow{\mathsf{e}^{\dagger}(\chi)} \lambda t \end{aligned} \tag{7.20}$$

where $\chi := \mathrm{eval}_{C,C'} \circ ((q_{c,c'} \circ g) \times C) \overset{\cong}{\Rightarrow} (\mathrm{eval}_{C,C'} \circ (q_{c,c'} \times C)) \circ (g \times c) \overset{\tau}{\Rightarrow} \lambda t$. The required coherence condition is the subject of the following lemma.

Lemma 7.3.13. Consider a pair of 1-cells

$$\underline{g} := (g, \gamma, f) : (R, r, Q) \to (C \supset C', p_{c,c'}, B \Rightarrow\!\!\!> B')$$
$$\underline{t} := (t, \alpha, s) : (R \times C, \mathrm{q}^{\times}_{Q,B} \circ (r \times c), Q \times B) \to (C', c', B')$$

in $\mathrm{gl}(\mathfrak{J})$ related by a 2-cell $\underline{\tau} := (\tau, \sigma) : \underline{\mathrm{eval}}_{\underline{C},\underline{C'}} \circ (\underline{g} \times \underline{C}) \Rightarrow \underline{t}$. Then, where Σ_1 and Σ_2 are defined in (7.20), the following diagram commutes:

$$\begin{tikzcd}
(\lambda(c' \circ \mathrm{eval}_{C,C'}) \circ q_{c,c'}) \circ g \arrow[r, "\cong"] \arrow[d, "\omega_{c,c'} \circ g"'] & \lambda(c' \circ \mathrm{eval}_{C,C'}) \circ (q_{c,c'} \circ g) \arrow[r, "\lambda(c' \circ \mathrm{eval}_{C,C'}) \circ \Sigma_2"] & \lambda(c' \circ \mathrm{eval}_{C,C'}) \circ \lambda t \arrow[d, "\mathrm{L}_{\alpha}"] \\
((\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}) \circ p_{c,c'}) \circ g \arrow[r, "\cong"'] & (\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}) \circ (p_{c,c'} \circ g) \arrow[r, "\lambda\widetilde{c} \circ \mathrm{m}_{B,B'} \circ \Sigma_1"'] & (\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}) \circ (\mathfrak{J}(\lambda s) \circ r)
\end{tikzcd}$$

Proof. Straightforward manipulations and an application of the cylinder condition on $\underline{\tau}$ unfolds the clockwise route to the following composite:

$$\begin{tikzcd}
\big(\lambda(c' \circ \mathrm{eval}_{C,C'}) \circ q_{c,c'}\big) \circ g \arrow[r] \arrow[dd, "\eta"'] & \big(\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}\big) \circ \big(\mathfrak{J}(\lambda s) \circ r\big) \\
& \lambda\big(\mathrm{eval}_{C,\mathfrak{J}B} \circ \big((\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}) \circ (\mathfrak{J}(\lambda s) \circ r)\big) \times C\big) \arrow[u, "\eta^{-1}"'] \\
\lambda\left(\mathrm{eval}_{C,\mathfrak{J}B'} \circ \big(\big(\lambda(c' \circ \mathrm{eval}_{C,C'}) \circ q_{c,c'}\big) \circ g\big) \times C\right) \arrow[r, "\lambda\zeta"'] & \lambda\Big(\mathfrak{J}s \circ \Big(\mathrm{q}^{\times}_{Q,B} \circ (r \times c)\Big)\Big) \arrow[u, "\lambda \mathrm{U}_{\alpha}^{-1}"']
\end{tikzcd} \tag{7.21}$$

Here $\zeta : \mathrm{eval}_{C,\mathfrak{J}B'} \circ ((\lambda(c' \circ \mathrm{eval}_{C,C'}) \circ q_{c,c'}) \circ g) \times C \to \mathfrak{J}s \circ \Big(\mathrm{q}^{\times}_{Q,B} \circ (r \times c)\Big)$ is the composite defined by commutativity of the following diagram:

$$\begin{tikzcd}
\mathrm{eval}_{C,\mathfrak{J}B'} \circ ((\lambda(c' \circ \mathrm{eval}_{C,C'}) \circ q_{c,c'}) \circ g) \times C \arrow[r, "\zeta"] \arrow[d, "\cong"'] & \mathfrak{J}s \circ \Big(\mathrm{q}^{\times}_{Q,B} \circ (r \times c)\Big) \\
(\mathrm{eval}_{C,\mathfrak{J}B'} \circ (\lambda(c' \circ \mathrm{eval}_{C,C'}) \times C)) \circ ((q_{c,c'} \circ g) \times C) \arrow[d, "\varepsilon_{(c' \circ \mathrm{eval})} \circ (qg \times C)"'] & \mathfrak{J}\left(\mathrm{eval}_{B,B'} \circ (f \times B)\right) \circ \Big(\mathrm{q}^{\times}_{Q,B} \circ (r \times c)\Big) \arrow[u, "\mathfrak{J}(\sigma) \circ \mathrm{q}^{\times} \circ (r \times c)"'] \\
(c' \circ \mathrm{eval}_{C,C'}) \circ ((q_{c,c'} \circ g) \times C) \arrow[d, "\cong"'] & (\mathfrak{J}(\mathrm{eval}_{B,B'}) \circ \mathfrak{J}(f \times B)) \circ \Big(\mathrm{q}^{\times}_{Q,B} \circ (r \times c)\Big) \arrow[u, "\phi^{\mathfrak{J}}_{\mathrm{eval}, f \times B} \circ \mathrm{q}^{\times} \circ (r \times c)"'] \\
(c' \circ (\mathrm{eval}_{C,C'} \circ (q_{c,c'} \times C))) \circ (g \times C) \arrow[d, "\mathrm{E}_{\underline{C},\underline{C'}} \circ (g \times C)"'] & \Big(\mathfrak{J}(\mathrm{eval}_{B,B'}) \circ \Big(\mathfrak{J}(f \times B) \circ \mathrm{q}^{\times}_{Q,B}\Big)\Big) \circ (r \times c) \arrow[u, "\cong"'] \\
\big(\mathfrak{J}(\mathrm{eval}_{B,B'}) \circ \big(\mathrm{q}^{\times}_{(B \Rightarrow B',B)} \circ (p_{c,c'} \times c)\big)\big) \circ (g \times C) \arrow[d, "\cong"'] & \\
\big(\mathfrak{J}(\mathrm{eval}_{B,B'}) \circ \mathrm{q}^{\times}_{(B \Rightarrow B',B)}\big) \circ \big((p_{c,c'} \circ g) \times c\big) \arrow[d, "\mathfrak{J}(\mathrm{eval}) \circ \mathrm{q}^{\times} \circ (\gamma \times c)"'] & \big(\mathfrak{J}(\mathrm{eval}_{B,B'}) \circ \big(\mathrm{q}^{\times}_{(B \Rightarrow B',B)} \circ (\mathfrak{J}f \times \mathfrak{J}\mathrm{Id}_B)\big)\big) \circ (r \times c) \arrow[uu, "\mathfrak{J}(\mathrm{eval}) \circ \mathsf{nat}_{f,\mathrm{Id}} \circ (r \times c)"'] \\
\big(\mathfrak{J}(\mathrm{eval}_{B,B'}) \circ \mathrm{q}^{\times}_{(B \Rightarrow B',B)}\big) \circ \big((\mathfrak{J}f \circ r) \times c\big) \arrow[r, "\cong"'] & \big(\mathfrak{J}(\mathrm{eval}_{B,B'}) \circ \big(\mathrm{q}^{\times}_{(B \Rightarrow B',B)} \circ (\mathfrak{J}f \times \mathfrak{J}B)\big)\big) \circ (r \times c) \arrow[u, "\mathfrak{J}(\mathrm{eval}) \circ \mathrm{q}^{\times} \circ (\mathfrak{J}f \times \psi^{\mathfrak{J}}_{B}) \circ (r \times c)"']
\end{tikzcd}$$

A short calculation shows that the following also commutes:

$$\begin{tikzcd}
\mathrm{eval}_{C,\mathfrak{J}B'} \circ ((\lambda(c' \circ \mathrm{eval}_{C,C'}) \circ q_{c,c'}) \circ g) \times C \arrow[r, "\zeta"] \arrow[d, "\mathrm{eval}\circ(\omega_{c,c'}\circ g)\times C"'] & \mathfrak{J}s \circ \left(\mathrm{q}^{\times}_{Q,B} \circ (r \times c)\right) \\
\mathrm{eval}_{C,\mathfrak{J}B'} \circ (((\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}) \circ p_{c,c'}) \circ g) \times C \arrow[d, "\cong"'] & \\
\mathrm{eval}_{C,\mathfrak{J}B'} \circ ((\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}) \circ (p_{c,c'} \circ g)) \times C \arrow[r, "\mathrm{eval}\circ(\lambda\widetilde{c}\circ\mathrm{m}\circ\Sigma_1)\times C"'] & \mathrm{eval}_{C,\mathfrak{J}B'} \circ ((\lambda\widetilde{c} \circ \mathrm{m}_{B,B'}) \circ (\mathfrak{J}(\lambda s) \circ r)) \times C \arrow[uu, "\mathrm{U}_\alpha"']
\end{tikzcd}$$

Substituting this back into (7.21) and applying the naturality of η, one obtains the anticlockwise route around the claim, as required. □

It follows that (g, Σ_1, Σ_2) is a fill-in. By the universality of the fill-in $(\underline{\mathsf{lam}}(t), \Gamma, \Delta)$, therefore, one obtains a 2-cell $\Sigma^\dagger : g \Rightarrow \underline{\mathsf{lam}}(t)$, unique such that the following two diagrams commute $\big($*c.f.* (7.10)$\big)$:

$$\begin{tikzcd}
p_{c,c'} \circ g \arrow[r, "\gamma"] \arrow[d, "p_{c,c'}\circ\Sigma^\dagger"'] & \mathfrak{J}(f) \circ r \arrow[d, "\mathfrak{J}(\mathsf{e}^\dagger(\sigma))\circ r"] & & q_{c,c'} \circ g \arrow[d, "q_{c,c'}\circ\Sigma^\dagger"'] \arrow[dr, "\mathsf{e}^\dagger(\chi)"] & \\
p_{c,c'} \circ \underline{\mathsf{lam}}(t) \arrow[r, "\Gamma_{c,c'}"'] & \mathfrak{J}(\lambda s) \circ r & & q_{c,c'} \circ \underline{\mathsf{lam}}(t) \arrow[r, "\Delta_{c,c'}"'] & \lambda t
\end{tikzcd} \tag{7.22}$$

We therefore define the components of $\mathsf{e}^\dagger(\underline{\tau})$ as follows:

$$\begin{aligned}
\tau^\sharp &:= \Sigma^\dagger : g \Rightarrow \underline{\mathsf{lam}}(t) \\
\sigma^\sharp &:= \mathsf{e}^\dagger(\sigma) : f \Rightarrow \lambda s
\end{aligned} \tag{7.23}$$

Note that the left-hand diagram of (7.22) establishes this pair is a 2-cell in $\mathrm{gl}(\mathfrak{J})$. We need to show that this 2-cell makes (7.19) commute. For the second component, this holds by assumption. For the first component, we observe that $\underline{\mathsf{e}}_t$ is the right-hand leg of the following diagram:

$$\begin{tikzcd}
(\mathrm{eval}_{C,C'} \circ (q_{c,c'} \times C)) \circ (g \times C) \arrow[rr, "\mathrm{eval}_{C,C'}\circ(q_{c,c'}\times C)\circ(\Sigma^\dagger\times C)"] \arrow[dr, "\cong"] \arrow[dddrr, bend right=40, "\tau"'] & \overset{\text{nat.}}{=} & (\mathrm{eval}_{C,C'} \circ (q_{c,c'} \times C)) \circ (\underline{\mathsf{lam}}(t) \times C) \arrow[d, "\cong"] \\
& \mathrm{eval}_{C,C'} \circ \big((q_{c,c'} \circ g\big) \times C) \arrow[r, "\mathrm{eval}_{C,C'}\circ(\Sigma^\dagger\times C)"] \arrow[dr, bend right] \arrow[ddr, bend right=30, "\chi"'] & \mathrm{eval}_{C,C'} \circ \big((q_{c,c'} \circ \underline{\mathsf{lam}}(t)\big) \times C) \arrow[d, "\mathrm{eval}_{C,C'}\circ(\Delta_{c,c'}\times C)"] \\
& \overset{\text{def.}}{=} & \mathrm{eval}_{C,C'} \circ (\lambda t \times C) \arrow[d, "\varepsilon_t"] \\
& \overset{\text{UMP}}{=} & t
\end{tikzcd}$$

The unlabelled inner arrow is $\mathrm{eval}_{C,C'} \circ (\mathsf{e}^\dagger(\chi) \times C)$ (where χ is defined just after (7.20)), so the triangular shape commutes by (7.22). This completes the existence part of the universality claim; we record our progress so far in the following lemma.

Lemma 7.3.14. For any triple of 1- and 2-cells as in Lemma 7.3.13, the pair $\mathsf{e}^\dagger(\underline{\tau}) := (\Sigma^\dagger, \mathsf{e}^\dagger(\sigma))$ defined in (7.23) is a 2-cell $\underline{g} \Rightarrow \underline{\lambda}\,\underline{t}$ in $\mathrm{gl}(\mathfrak{J})$ satisfying (7.19). □

It remains to show uniqueness. Suppose given a 2-cell $\underline{\theta} : \underline{g} \Rightarrow \underline{\lambda}\,\underline{t}$ in $\mathrm{gl}(\mathfrak{J})$ with components

$$\theta : g \Rightarrow \underline{\mathsf{lam}}(t)$$
$$\vartheta : f \Rightarrow \lambda s$$

such that $\underline{\theta}$ fills (7.19). Examining the second component, it is immediate from the universal property of $\mathsf{e}^\dagger(\sigma)$ that $\mathsf{e}^\dagger(\sigma) = \vartheta$. For the first component, we show that $\theta = \Sigma^\dagger$ by showing that θ satisfies the two diagrams of (7.22). For the left-hand diagram, the cylinder condition on $\underline{\theta}$ requires that

$$\begin{tikzcd}
p_{c,c'} \circ g \arrow[r, "\gamma"] \arrow[d, "p_{c,c'} \circ \theta"'] & \mathfrak{J}(f) \circ r \arrow[d, "\mathfrak{J}(\vartheta) \circ r"] \\
p_{c,c'} \circ \underline{\mathsf{lam}}(t) \arrow[r, "\Gamma_{c,c'}"'] & \mathfrak{J}(\lambda s) \circ r
\end{tikzcd}$$

But we already know that $\vartheta = \mathsf{e}^\dagger(\sigma)$, so the required diagram commutes. For the right-hand diagram, it follows from (7.19) and the definition of $\underline{\mathsf{e}}_t$ that the following commutes:

$$\begin{tikzcd}
\mathrm{eval}_{C,C'} \circ ((q_{c,c'} \circ g) \times C) \arrow[rr, "\cong"] \arrow[d, "\mathrm{eval}_{C,C'} \circ (q_{c,c'} \circ \theta) \times C"'] & & (\mathrm{eval}_{C,C'} \circ (q_{c,c'} \times C)) \circ (g \times C) \arrow[d, "\tau"] \\
\mathrm{eval}_{C,C'} \circ ((q_{c,c'} \circ \underline{\mathsf{lam}}(t)) \times C) \arrow[r, "\mathrm{eval}_{C,C'} \circ (\Delta_{c,c'} \times C)"'] & \mathrm{eval}_{C,C'} \circ (\lambda t \times C) \arrow[r, "\varepsilon_t"'] & t
\end{tikzcd}$$

The claim then holds by the universal property of $\mathsf{e}^\dagger(\vartheta)$. Thus:

Lemma 7.3.15. For any triple of 1- and 2-cells as in Lemma 7.3.13, the pair $\mathsf{e}^\dagger(\underline{\tau}) := (\Sigma^\dagger, \mathsf{e}^\dagger(\sigma))$ defined in (7.23) is the unique 2-cell $\underline{g} \Rightarrow \underline{\lambda}\,\underline{t}$ in $\mathrm{gl}(\mathfrak{J})$ satisfying (7.19). □

This completes the proof that for any $\underline{R}, \underline{C}$ and $\underline{C'}$ in $\mathrm{gl}(\mathfrak{J})$ the diagram

$$\underline{\lambda} : \mathrm{gl}(\mathfrak{J})\big(\underline{R} \times \underline{C}, \underline{C'}\big) \leftrightarrows \mathrm{gl}(\mathfrak{J})\big(\underline{R}, \underline{C} \Rightarrow \underline{C'}\big) : \underline{\mathrm{eval}}_{\underline{C},\underline{C'}} \circ (- \times \underline{C})$$

is an adjoint equivalence, and hence the proof of Theorem 7.3.9.

Chapter 8

Normalisation-by-evaluation for $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$

We now turn to the main result of this thesis, namely the coherence result for cartesian closed bicategories. Our strategy is to employ a bicategorical treatment of the *normalisation-by-evaluation* proof technique. It is well-known that the naïve strategy for proving strong normalisation of the simply-typed lambda calculus—by a straightforward structural induction on terms—fails because an application $\mathsf{app}(t, u)$ may contain redexes that do not occur in either t or u. One classical solution, originally due to Tait [Tai67], is to strengthen the inductive hypothesis using *reducibility predicates*. This approach was refined by Girard [Gir72], who introduced the notion of *neutral terms*. These can be viewed as the obstructions to the normalisation proof: they are the terms whose introduction rules may introduce new β-redexes.

Normalisation-by-evaluation provides an alternative strategy: as a slogan, one 'inverts the evaluation functional' to construct a mapping from neutral to normal terms. Loosely speaking, one constructs a model with enough intensional information to pass back and forth between semantics and syntax. One *quotes* a morphism f to a (normal) term in the syntax, and *unquotes* a term t to a morphism in the semantics (these operations are also known as *reify* and *reflect*).

The intuition is—very roughly—as follows. Consider a semantics $[\![-]\!]$ for the simply-typed lambda calculus, determined by a choice of cartesian closed category and an interpretation of the base types, and suppose that one has constructed mappings quote and $\mathsf{unquote}$ between the syntax and semantics, as indicated above. For a term $(x : A \vdash t : B)$ one has an interpretation $[\![t]\!] : [\![A]\!] \to [\![B]\!]$. Now, where x is a generic fresh variable, $\mathsf{unquote}(x) : [\![A]\!]$. So one may evaluate $[\![t]\!]$ at $\mathsf{unquote}(x)$ to obtain a normal term $\mathsf{quote}\left([\![t]\!]\left(\mathsf{unquote}(x)\right)\right)$ of type B. The normal form of $\lambda x.t$ is then $\lambda x.\,\mathsf{quote}\big([\![t]\!]\left(\mathsf{unquote}(x)\right)\big)$.

First introduced by Berger & Schwichtenberg [BS91] for the simply-typed lambda calculus, normalisation-by-evaluation has become a standard tool for tackling normalisation problems. It has been extended to a number of richer calculi, including the simply-typed lambda calculus with sum types [ADHS01], versions of Martin-Löf type theory (*e.g.* [ACD07, AK16, AK17]), and even to type theories with algebraic effects [Sta13]. Moreover, the normalisation algorithm one extracts from normalisation-by-evaluation is generally highly efficient, which has led to significant study for applications in interactive proof systems (see *e.g.* [BES98]).

Here we follow in the vein of *categorical* reconstructions of the normalisation-by-evaluation argument (*e.g.* [AHS95, CD97, CD98, Fio02]). In particular, the argument we present closely follows [Fio02]; the reliance on categorical properties there lends itself especially to bicategorical translation.

The chapter is arranged as follows. We begin in Section 8.1 by briefly recapitulating the argument of [Fio02]. In Sections 8.2–8.3 we show how the crucial elements of this argument can be lifted to the bicategorical setting. Section 8.4 presents the main result of this thesis: $\Lambda_{\mathrm{ps}}^{\times,\to}$ is locally coherent.

8.1 Fiore's categorical normalisation-by-evaluation proof

We extract the bare bones of Fiore's argument [Fio02]. The intention is not to provide the reader with the full proof, but to waypoint the key steps in the bicategorical argument we present thereafter.

Syntax as presheaves. For any set of base types $\mathfrak{B}$, let $\mathrm{Con}_{\widetilde{\mathfrak{B}}}$ denote the free strict cocartesian category on the set $\widetilde{\mathfrak{B}}$ generated by the grammar

$$X_1, \dots, X_n, Y, Z ::= B \mid \textstyle\prod_n(X_1, \dots, X_n) \mid Y \Rightarrow Z \qquad (B \in \mathfrak{B}, n \in \mathbb{N})$$

Explicitly, this is the comma category $(\mathbb{F} \downarrow \widetilde{\mathfrak{B}})$, where $\mathbb{F}$ is a skeleton of the category of finite sets and all set-theoretic functions. For our purposes, however, we identify it with the *category of contexts*, in which the objects are contexts (defined by Figure 8.1, below) and the morphisms are context renamings. Note that we index from 0 to avoid awkward off-by-one manipulations.

$$\frac{}{\diamond\ \mathrm{ctx}} \qquad\qquad \frac{\Gamma\ \mathrm{ctx} \qquad |\Gamma| = n}{\Gamma, x_n : A\ \mathrm{ctx}}\ (A \in \widetilde{\mathfrak{B}})$$

Figure 8.1: Rules for contexts

To ensure that that $\mathrm{Con}_{\widetilde{\mathfrak{B}}}$ is strict cocartesian, we stipulate that variables are named in order according to a fixed enumeration. However, following our standing abuse (Notation 3.2.12), we shall freely employ more indicative variable names, such as using f to denote a variable of exponential type.

An object $\gamma : [n] \to \widetilde{\mathfrak{B}}$ (for $[n] = \{0, \dots, n-1\} \in \mathbb{F}$) in $(\mathbb{F} \downarrow \widetilde{\mathfrak{B}})$ corresponds to the context $(x_i : \gamma(i))_{i=1,\dots,n}$. A morphism $h : \gamma \to \delta$, namely a set map $[n] \to [m]$ such that the diagram below commutes, corresponds to the context renaming $x_i \mapsto x_{hi}$.

$$\begin{tikzcd}
{[n]} \arrow[rr, "h"] \arrow[dr, "\gamma"'] & & {[m]} \arrow[dl, "\delta"] \\
& \widetilde{\mathfrak{B}} &
\end{tikzcd}$$

The coproduct $\Gamma + \Delta$ is the concatenated context $\Gamma \mathbin{@} \Delta$.

We denote the universal embedding of $\widetilde{\mathfrak{B}}$ into $\mathrm{Con}_{\widetilde{\mathfrak{B}}}$ by $[-]$; thus, $[A]$ *coerces* the type A into the unary context $(x_1 : A)$, and the coproduct $\Gamma + [A]$ is the weakening of Γ by a variable of type A. The notation is chosen to suggest a list of length one.

In the tradition of *algebraic type theory* (*e.g.* [FPT99, Fio11]), the category $\mathcal{P}({\mathrm{Con}_{\widetilde{\mathfrak{B}}}}^{\mathrm{op}})$ of covariant presheaves $\mathrm{Con}_{\widetilde{\mathfrak{B}}} \to \mathrm{Set}$ provides a semantic universe for the study of abstract syntax. For example, for the simply-typed lambda calculus $\Lambda^{\times,\to}(\mathfrak{B})$ over $\mathfrak{B}$, the set of terms-in-context of a given type B (modulo α-equivalence) define a presheaf $L(-;B)$ by $L(\Gamma;B) := \{t \mid \Gamma \vdash t : B\} /{=_\alpha}$. The functorial action is given by context renamings: for contexts $\Gamma := (x_i : A_i)_{i=1,\dots,n}$ and $\Delta := (y_j : B_j)_{j=1,\dots,m}$ and a context renaming $r : \Gamma \to \Delta$, one obtains a mapping

$$\begin{aligned} L(\Gamma;B) &\to L(\Delta;B) \\ t &\mapsto t[r(x_i)/x_i] \end{aligned}$$

by the admissibility of the rule

$$\frac{\Gamma \vdash t : B \qquad r : \Delta \to \Gamma}{\Delta \vdash t[r(x_i)/x_i] : B}$$

The Yoneda embedding y yields a *presheaf of variables*: for any type $A \in \widetilde{\mathfrak{B}}$ and context Γ, $\mathrm{y}([A])(\Gamma) = \mathrm{y}(x : A)(\Gamma) = \mathrm{Con}_{\widetilde{\mathfrak{B}}}((x : A), \Gamma)$ corresponds to the set of inclusions of contexts $(x : A) \hookrightarrow \Gamma$. This determines a presheaf $V(-;A)$ defined by $V(\Gamma;A) = \{x \mid \Gamma \vdash x : A\}$. The well-known fact that $[\mathrm{y}X, P] \cong P(- \times X)$ in any presheaf category over a cartesian category corresponds to the observation that the exponential presheaf $[\mathrm{y}A, L(-;B)]$ consists of terms of type B in the extended context $\Gamma + [A]$ (note that, since $\mathrm{Con}_{\widetilde{\mathfrak{B}}}$ is strict cocartesian, its opposite category is strict cartesian).

Intensional Kripke relations We extend the *Kripke logical relations of varying arity* of [JT93, Ali95] to a category of *intensional Kripke relations.* Encoding this extra intensional information allows one to extract a normalisation algorithm from the proof. Abstractly, the key to this construction is the *relative hom-functor* (also known as the *nerve* functor). For any functor $\mathfrak{J} : \mathbb{B} \to \mathbb{X}$ the left Kan extension $\langle \mathfrak{J} \rangle := \mathrm{lan}_{\mathfrak{J}}(\mathrm{y})$ exists as in the following diagram, in which $\mathcal{P}(\mathbb{B})$

denotes the presheaf category:

$$\begin{tikzcd}
\mathbb{B} \arrow[rr, hook, "\mathrm{y}"] \arrow[dr, "\mathfrak{J}"'] & \Downarrow \mathrm{lan} & \mathcal{P}(\mathbb{B}) \\
& \mathbb{X} \arrow[ur, "\langle \mathfrak{J} \rangle"'] &
\end{tikzcd} \tag{8.1}$$

Explicitly, $\langle \mathfrak{J} \rangle(X) := \mathbb{X}\big(\mathfrak{J}(-), X\big) : \mathbb{B}^{\mathrm{op}} \to \mathrm{Set}$ and $\mathrm{lan}_B : \mathbb{B}(-, B) \Rightarrow \mathbb{X}\big(\mathfrak{J}(-), \mathfrak{J}B\big)$ is just the functorial action of $\mathfrak{J}$. This construction is particularly well-known in the context of profunctors (distributors), since $\mathbb{B}\big(\mathfrak{J}(-), X\big)$ and $\mathbb{B}\big(X, \mathfrak{J}(-)\big)$ provide canonical (indeed, adjoint) profunctors $\mathbb{X} \nrightarrow \mathbb{B}$ for every functor $\mathfrak{J} : \mathbb{B} \to \mathbb{X}$ (*e.g.* [Bor94, Example 7.8.3]).

Definition 8.1.1.

1. For $\mathfrak{J} : \mathbb{B} \to \mathbb{X}$ a functor, the *relative hom-functor* is the functor $\langle \mathfrak{J} \rangle : \mathbb{X} \to \mathcal{P}(\mathbb{B})$ defined above.

2. For a category $\mathbb{B}$ and a functor $\mathfrak{J} : \mathbb{B} \to \mathbb{X}$, the category of $\mathbb{B}$-*intensional Kripke relations of arity* $\mathfrak{J}$ is the glueing category $\mathrm{gl}(\langle \mathfrak{J} \rangle)$ associated to the relative hom-functor. ◂

The relative hom-functor preserves limits so, when $\mathbb{X}$ is cartesian closed, the glueing category $\mathrm{gl}(\langle \mathfrak{J} \rangle)$ is cartesian closed and the forgetful functor to $\mathbb{X}$ strictly preserves products and exponentials. Moreover, the Yoneda embedding extends to an embedding $\underline{\mathrm{y}} : \mathbb{B} \to \mathrm{gl}(\langle \mathfrak{J} \rangle)$ by $\underline{\mathrm{y}}(B) := \Big(\mathrm{y}(B), \mathrm{y}(B) \xRightarrow{\mathrm{lan}_B} \langle \mathfrak{J} \rangle(\mathfrak{J}B), \mathfrak{J}B\Big)$.

Consider now the following situation. Fix a set of base types $\mathfrak{B}$ and an interpretation $h : \mathfrak{B} \to \mathbb{X}$ in a cartesian closed category $\mathbb{X}$. By the cartesian closed structure, this extends to a map $\widetilde{\mathfrak{B}} \to \mathbb{X}$ we also denote by h. Applying the universal property, h extends in turn to a cartesian functor $\underline{h} : {\mathrm{Con}_{\widetilde{\mathfrak{B}}}}^{\mathrm{op}} \to \mathbb{X}$ interpreting all contexts within $\mathbb{X}$. Moreover, writing $\mathcal{F}(\widetilde{\mathfrak{B}})$ for the free cartesian closed category on $\widetilde{\mathfrak{B}}$, namely the syntactic model of the simply-typed lambda calculus $\Lambda^{\times,\to}(\mathfrak{B})$, the coercion $[-] : \widetilde{\mathfrak{B}} \hookrightarrow \mathrm{Con}_{\widetilde{\mathfrak{B}}}$ extends to a cartesian functor $\mathrm{Con}_{\widetilde{\mathfrak{B}}} \to \mathcal{F}(\widetilde{\mathfrak{B}})$. By the various uniqueness properties, this factors the semantic interpretation $h[\![-]\!] : \mathcal{F}(\widetilde{\mathfrak{B}}) \to \mathbb{X}$ extending h. The situation is summarised in the following diagram.

$$\begin{tikzcd}
& \mathcal{F}(\widetilde{\mathfrak{B}}) \arrow[dr, dashed, "h[\![-]\!]"] & \\
{\mathrm{Con}_{\widetilde{\mathfrak{B}}}}^{\mathrm{op}} \arrow[ur, dashed] \arrow[rr, dashed, "\underline{h}"'] & & \mathbb{X} \\
\widetilde{\mathfrak{B}} \arrow[u, hook, "{[-]}"] \arrow[urr] & & \\
\mathfrak{B} \arrow[u, hook] \arrow[uurr, "h"'] & &
\end{tikzcd} \tag{8.2}$$

Note in particular that $\underline{h}\Gamma = h[\![\Gamma]\!]$ for every context $\Gamma \in \mathrm{Con}_{\widetilde{\mathfrak{B}}}$, and that for any type $A \in \widetilde{\mathfrak{B}}$ the interpretation $h[\![A]\!]$ is equal to $\underline{h}[A]$. (Here we use the assumption that $\prod_1(X) = X$ to identify $h[\![x : A]\!]$ with $h[\![A]\!]$.)

An object in the category $\mathrm{gl}(\langle \underline{h} \rangle)$ of $\mathrm{Con}_{\widetilde{\mathfrak{B}}}$-intensional Kripke relations of arity $\underline{h}$ then consists of a presheaf $P : \mathrm{Con}_{\widetilde{\mathfrak{B}}} \to \mathrm{Set}$ (which one might think of as *syntactic* intensional information), an object $X \in \mathbb{X}$, and a natural transformation $\pi : P \Rightarrow \mathbb{X}(\underline{h}(-), X)$ (which one might think of as *semantic* information). One may think of this category as internalising the relationship between syntax and semantics required for the normalisation-by-evaluation argument.

Neutral and normal terms as glued objects. The definitions of neutral and (long-$\beta\eta$) normal terms for the simply-typed lambda calculus, given in Figure 8.2 below, are standard (*e.g.* [GTL89, Chapter 4]). We define a family of judgements $\Gamma \vdash_M t : B$ and $\Gamma \vdash_N t : B$ characterising neutral and normal terms, respectively, by mutual induction.

$$\frac{}{x_1 : A_1, \ldots, x_n : A_n \vdash_M x_i : A_i}\ \text{var}$$

$$\frac{\Gamma \vdash_M t : \prod_n(A_1, \ldots, A_n)}{\Gamma \vdash_M \pi_k(t) : A_k}\ \text{proj } (k = 1, \ldots, n) \qquad \frac{\Gamma \vdash_M t : A \Rightarrow B \qquad \Gamma \vdash_N u : A}{\Gamma \vdash_M \mathsf{app}(t, u) : B}\ \text{app}$$

$$\frac{\Gamma \vdash_N t_i : A_i \quad (i = 1, \ldots, n)}{\Gamma \vdash_N \langle t_1, \ldots, t_n \rangle : \prod_n(A_1, \ldots, A_n)}\ \text{tuple} \qquad \frac{\Gamma, x : A \vdash_N t : B}{\Gamma \vdash_N \lambda x.t : A \Rightarrow B}\ \text{lam}$$

$$\frac{\Gamma \vdash_M t : B}{\Gamma \vdash_N t : B}\ \text{inc } (B \text{ a base type})$$

Figure 8.2: Neutral terms and normal terms in the simply-typed lambda calculus

Crucially, the sets of neutral and normal terms are invariant under renaming, so for every type $A \in \widetilde{\mathfrak{B}}$ one now obtains four presheaves $\mathrm{Con}_{\widetilde{\mathfrak{B}}} \to \mathrm{Set}$, defined at $\Gamma \in \mathrm{Con}_{\widetilde{\mathfrak{B}}}$ as follows:

$$\begin{aligned} V(\Gamma; A) &:= \mathrm{y}[A] = \{x \mid \Gamma \vdash x : A\} \,/ =_\alpha \\ M(\Gamma; A) &:= \{t \mid \Gamma \vdash_M t : A\} \,/ =_\alpha \\ N(\Gamma; A) &:= \{t \mid \Gamma \vdash_N t : A\} \,/ =_\alpha \\ L(\Gamma; A) &:= \{t \mid \Gamma \vdash t : A\} \,/ =_\alpha \end{aligned} \tag{8.3}$$

Each rule of Figure 8.2 defines a morphism on these indexed families of presheaves. For the lambda abstraction case we employ the coproduct structure on $\mathrm{Con}_{\widetilde{\mathfrak{B}}}$.

Lemma 8.1.2. The rules of Figure 8.2 give rise to natural transformations, as follows:

$$
\begin{aligned}
\mathsf{var}(-;A_i) : V(-;A_i) &\Rightarrow M(-;A_i) \\
\mathsf{inc}\big(-;B\big) : M(-;B) &\Rightarrow N(-;B) && (B \text{ a base type}) \\
\mathsf{proj}_k(-;A_\bullet) : M(-;\textstyle\prod_n(A_1,\ldots,A_n)) &\Rightarrow M(-;A_k) && (k = 1,\ldots,n) \\
\mathsf{app}(-;A,B) : M(-;A \Rightarrow B) \times N(-;A) &\Rightarrow M(-;B) \\
\mathsf{tuple}(-;A_\bullet) : \textstyle\prod_{i=1}^{n} N(-;A_i) &\Rightarrow N(-;\textstyle\prod_n(A_1,\ldots,A_n)) \\
\mathsf{lam}(-;A \Rightarrow B) : N\big(-+[A];B\big) &\Rightarrow N(-;A \Rightarrow B)
\end{aligned}
$$

Proof. The mappings are just the operations on terms. In each case naturality follows from the definition of the meta-operation of capture-avoiding substitution, in particular the fact that substitution passes through the various constructors, and that it respects α-equivalence. □

Returning to the development described by the diagram (8.2), and noting that $\langle\underline{h}\rangle(\underline{h}[A]) = \mathbb{X}(\underline{h}(-),\underline{h}[A]) = \mathbb{X}(h[\![-]\!],h[\![A]\!])$ for every type A, one obtains the following glued objects in $\mathrm{gl}(\langle\underline{h}\rangle)$ for every $A \in \widetilde{\mathfrak{B}}$:

$$
\begin{aligned}
\underline{V}_A &:= (V(-;A), V(-;A) \Rightarrow \langle\underline{h}\rangle(h[\![A]\!]), h[\![A]\!]) = \underline{\mathrm{y}}([A]) \\
\underline{M}_A &:= (M(-;A), M(-;A) \Rightarrow \langle\underline{h}\rangle(h[\![A]\!]), h[\![A]\!]) \\
\underline{N}_A &:= (N(-;A), N(-;A) \Rightarrow \langle\underline{h}\rangle(h[\![A]\!]), h[\![A]\!]) \\
\underline{L}_A &:= (L(-;A), L(-;A) \Rightarrow \langle\underline{h}\rangle(h[\![A]\!]), h[\![A]\!])
\end{aligned}
\tag{8.4}
$$

In each case, the natural transformation is the canonical interpretation of $\Lambda^{\times,\rightarrow}(\mathfrak{B})$-terms in $\mathbb{X}$. Moreover, extending the natural transformations induced from the rules of Figure 8.2 in a similar fashion, one obtains a morphism in $\mathrm{gl}(\langle\underline{h}\rangle)$ for each rule.

Normalisation-by-evaluation. We paste together the various elements seen thus far. Since $\mathrm{gl}(\langle\underline{h}\rangle)$ is cartesian closed, one may consider the interpretation $B \mapsto \underline{M}_B$ of base types in $\mathrm{gl}(\langle\underline{h}\rangle)$. This extends to an interpretation $\overline{h}[\![-]\!] : \mathcal{F}(\widetilde{\mathfrak{B}}) \to \mathrm{gl}(\langle\underline{h}\rangle)$. Write $\overline{h}[\![A]\!] := (G_A, \gamma_A, h[\![A]\!])$ and $\overline{h}[\![\Gamma \vdash t : A]\!] := (h'[\![\Gamma \vdash t : A]\!], h[\![\Gamma \vdash t : A]\!])$. Since the forgetful functor $\pi_{\mathrm{dom}} : \mathrm{gl}(\langle\underline{h}\rangle) \to \mathbb{X}$ is strictly cartesian closed, the final component in each case is exactly the interpretation in $\mathbb{X}$ extending h.

One then employs the cartesian closed structure of $\mathrm{gl}(\langle\underline{h}\rangle)$, and the 1-cells in $\mathrm{gl}(\langle\underline{h}\rangle)$ induced from the rules of Figure 8.2, to inductively define *quote* and *unquote* as $\widetilde{\mathfrak{B}}$-indexed maps of the following type:

$$
\begin{aligned}
\mathsf{unquote}_A &: \underline{M}_A \to \overline{h}[\![A]\!] \\
\mathsf{quote}_A &: \overline{h}[\![A]\!] \to \underline{N}_A
\end{aligned}
$$

For every $\Lambda^{\times,\rightarrow}(\mathfrak{B})$-term $\Gamma \vdash t : A$ (where $\Gamma := (x_i : A_i)_{i=1,\dots,n}$), one thereby obtains the following commutative diagram in $\mathcal{P}(\mathrm{Con}_{\widetilde{\mathfrak{B}}}{}^{\mathrm{op}})$, in which the unlabelled arrows are the canonical interpretations of terms inside $\mathbb{X}$:

$$\begin{tikzcd}
\prod_{i=1}^{n} M(-;A_i) \arrow[r, "\prod_{i=1}^{n} \mathsf{unquote}_{A_i}"] \arrow[dr, bend right] & \prod_{i=1} G_{A_i} \arrow[r, "h'[\![\Gamma \vdash t:A]\!]"] \arrow[d, "\prod_{i=1}^{n} \gamma_{A_i}"'] & G_A \arrow[r, "\mathsf{quote}_A"] \arrow[dd, "\gamma_A"] & N(-;A) \arrow[ddl] \\
& \prod_{i=1}^{n} \mathbb{X}(h[\![-]\!], h[\![A_i]\!]) \arrow[d, "\cong"'] & & \\
& \mathbb{X}(h[\![-]\!], h[\![\Gamma]\!]) \arrow[r, "h[\![\Gamma \vdash t:A]\!] \circ (-)"'] & \mathbb{X}(h[\![-]\!], h[\![A]\!]) &
\end{tikzcd} \tag{8.5}$$

Chasing the n-ary variable-projection tuple $(\Gamma \vdash x_i : A_i)_{i=1,\dots,n}$ through this diagram, one obtains a normal term $\mathsf{nf}(t)$ for which the semantic interpretation $h[\![\mathsf{nf}(t)]\!]$ is equal to $h[\![t]\!]$. Moreover, for every type A the projections $\pi_{\mathrm{dom}}(\mathsf{quote}_A)$ and $\pi_{\mathrm{dom}}(\mathsf{unquote}_A)$ are both the identity. It follows that, for $\mathbb{X} = \mathcal{F}(\widetilde{\mathfrak{B}})$ the syntactic model of $\Lambda^{\times,\rightarrow}(\mathfrak{B})$, one obtains a normal form $\mathsf{nf}(t)$ for t such that $t =_{\beta\eta} \mathsf{nf}(t)$. Hence, every $\Lambda^{\times,\rightarrow}(\mathfrak{B})$-term has a long-$\beta\eta$ normal form, which can be explicitly calculated. This yields a normalisation algorithm.

Our aim in what follows is to leverage as much of this proof as possible as we lift it to the bicategorical setting. We follow the strategy just outlined stage-by-stage, with the aim of building up a version of (8.5) in which each of the commuting shapes is filled by a witnessing 2-cell. Throughout we shall assume that $\mathfrak{B}$ is a fixed set of base types.

8.2 Syntax as pseudofunctors

The locally discrete 2-category of contexts. The notion of context in $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$ is the same as that in the simply-typed lambda calculus. We therefore require the same categorical structure on the category of contexts $\mathrm{Con}_{\widetilde{\mathfrak{B}}}$, which we now wish to treat as a degenerate 2-category. Keeping track of such degeneracies will help identify instances where we can apply the 1-categorical theory.

Notation 8.2.1.

1. For S a set, write ∂S for the discrete category with objects the elements of S. Similarly, write ∂f for the discrete functor $\partial S \to \partial S'$ induced by the set map $f : S \to S'$.

2. a) For $\mathbb{C}$ a category, write $\mathrm{d}\mathbb{C}$ for the *locally discrete* 2-category with objects those of $\mathbb{C}$ and hom-categories $(\mathrm{d}\mathbb{C})(X,Y) := \partial(\mathbb{C}(X,Y))$.

   b) Write $\mathrm{d}F$ for the *locally discrete* 2-functor $\mathrm{d}\mathbb{C} \to \mathrm{d}\mathbb{D}$ induced from the functor $F : \mathbb{C} \to \mathbb{D}$ by setting $(\mathrm{d}F)X := FX$ and $(\mathrm{d}F)_{X,Y} := \partial(F_{X,Y})$.

c) Write $\mathrm{d}\mu$ for the *locally discrete* 2-natural transformation $\mathrm{d}F \Rightarrow \mathrm{d}G$ induced from the natural transformation $\mu : F \Rightarrow G : \mathbb{C} \to \mathbb{D}$ by setting $(\mathrm{d}\mu)_C := \mu_C$ for every $C \in \mathbb{C}$. ◂

The $\partial(-)$ and $\mathrm{d}(-)$ constructions will be our main technical tool for constructing (degenerate) bicategorical structure from 1-categorical data. The next lemma collects together some of their important properties. The proofs are not especially difficult, but stating all the details precisely requires some care. Since we employ the notation $- \Rightarrow\!\!\!\triangleright =$ for exponentials in $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$ we denote the usual categorical functor category by $\mathrm{Fun}(\mathbb{C}, \mathbb{D})$.

Lemma 8.2.2. Let $\mathbb{C}$ and $\mathbb{D}$ be 1-categories. Then:

1. $(\mathrm{d}\mathbb{C})^{\mathrm{op}} = \mathrm{d}(\mathbb{C}^{\mathrm{op}})$.
2. There exists an isomorphism of 2-categories $\mathrm{d}\big(\mathrm{Fun}(\mathbb{C}, \mathbb{D})\big) \cong \mathrm{Hom}(\mathrm{d}\mathbb{C}, \mathrm{d}\mathbb{D})$.
3. There exists an injective-on-objects, locally isomorphic 2-functor $\iota : \mathrm{dFun}(\mathbb{C}, \mathrm{Set}) \hookrightarrow \mathrm{Hom}(\mathrm{d}\mathbb{C}, \mathbf{Cat})$, which induces a commutative diagram

$$\begin{tikzcd}
\mathrm{d}\big(\mathrm{Fun}(\mathbb{C}, \mathrm{Set})\big) \arrow[r, hook, "\iota"] & \mathrm{Hom}(\mathrm{d}\mathbb{C}, \mathbf{Cat}) \\
\mathrm{d}\mathbb{C} \arrow[u, "\mathrm{dy}"] \arrow[ur, "\mathrm{Y}"'] &
\end{tikzcd} \tag{8.6}$$

In particular, $\mathrm{Y}(C) = (\mathrm{dy})C$ for all $C \in \mathbb{C}$.

4. If $\mathbb{C}$ is cartesian (resp. cartesian closed) as a 1-category, then $\mathrm{d}\mathbb{C}$ has finite products (resp. is cartesian closed) as a 2-category.
5. Let $P, Q : \mathbb{C} \to \mathrm{Set}$. The exponential $[\iota P, \iota Q]$ in $\mathrm{Hom}(\mathrm{d}\mathbb{C}, \mathbf{Cat})$ is given up to equivalence by $\iota\big(\mathrm{Fun}(\mathbb{C}, \mathrm{Set})\big(\mathrm{y}(-) \times P, Q\big)\big)$, for $\mathrm{y} : \mathbb{C} \to \mathrm{Fun}(\mathbb{C}, \mathrm{Set})$ the 1-categorical Yoneda embedding.

Proof. (1) is immediate from the definitions.

For (2), consider the mapping $\mathrm{d}(-) : \mathrm{d}\big(\mathrm{Fun}(\mathbb{C}, \mathbb{D})\big) \to \mathrm{Hom}(\mathrm{d}\mathbb{C}, \mathrm{d}\mathbb{D})$ taking $F : \mathbb{C} \to \mathbb{D}$ to the locally discrete 2-functor $\mathrm{d}F$ and $\mu : F \to G$ to the locally discrete pseudonatural transformation $\mathrm{d}\mu$. Since $\mathrm{d}\big(\mathrm{Fun}(\mathbb{C}, \mathbb{D})\big)$ is locally discrete, this extends canonically to a 2-functor.

Now suppose that $F : \mathrm{d}\mathbb{C} \to \mathrm{d}\mathbb{D}$ is a pseudofunctor. By definition, this is a set map $F : \mathit{ob}(\mathrm{d}\mathbb{C}) \to \mathit{ob}(\mathrm{d}\mathbb{D})$ with functors $F_{X,Y} : (\mathrm{d}\mathbb{C})(X, Y) \to (\mathrm{d}\mathbb{D})(FX, FY)$ for every $X, Y \in \mathrm{d}\mathbb{C}$. Since every $(\mathrm{d}\mathbb{C})(X, Y)$ is a discrete category, every $F_{X,Y}$ is discrete, and so $F = \mathrm{d}H$ for a unique functor $H : \mathbb{C} \to \mathbb{D}$. So $\mathrm{d}(-)$ is bijective on objects.

Next fix functors $F, G : \mathbb{C} \to \mathbb{D}$ and consider the hom-category $\mathrm{Hom}(\mathrm{d}\mathbb{C}, \mathrm{d}\mathbb{D})(\mathrm{d}F, \mathrm{d}G)$. A pseudonatural transformation $(\mathsf{k}, \overline{\mathsf{k}}) : \mathrm{d}F \Rightarrow \mathrm{d}G$ consists of a family of 1-cells $\mathsf{k}_X : FX \to GX$ ($X \in \mathrm{d}\mathbb{C}$), together with a 2-cell $\overline{\mathsf{k}}_f : \mathsf{k}_Y \circ Ff \Rightarrow Gf \circ \mathsf{k}_X$ in $\mathrm{d}\mathbb{D}$ for every $f : X \to Y$ in $\mathrm{d}\mathbb{C}$. Since $\mathrm{d}\mathbb{D}$ is locally discrete, the only choice of such a 2-cell is the identity. So $(\mathsf{k}, \overline{\mathsf{k}})$ is a 2-natural transformation, and is of the form $\mathrm{d}\mu$ for a unique natural transformation $\mu : F \Rightarrow G$. Similarly, every modification

$\Xi : (\mathsf{k}, \overline{\mathsf{k}}) \to (\mathsf{j}, \bar{\mathsf{j}}) : \mathrm{d}F \Rightarrow \mathrm{d}G$ consists of a family of 2-cells, and must therefore be the identity. It follows that $\mathrm{d}(-)_{F,G} : \mathrm{d}\big(\mathrm{Fun}(\mathbb{C}, \mathbb{D})\big)(F, G) \to \mathrm{Hom}(\mathrm{d}\mathbb{C}, \mathrm{d}\mathbb{D})(\mathrm{d}F, \mathrm{d}G)$ is an isomorphism for every F and G, as required.

For (3), we define ι by setting ιP to be the composite $\mathbb{C} \xrightarrow{P} \mathrm{Set} \xrightarrow{\partial(-)} \mathbf{Cat}$, so that $\iota P := \lambda C^{\mathbb{C}} . \partial(PC)$ and $(\iota\mu)_C := \partial(\mu_C)$ for every $\mu : P \Rightarrow Q$ and $C \in \mathbb{C}$. It is clear that ι is injective on objects. To see that $\iota_{P,Q} : \mathrm{d}\big(\mathrm{Fun}(\mathbb{C}, \mathrm{Set})\big)(P, Q) \to \mathrm{Hom}(\mathrm{d}\mathbb{C}, \mathbf{Cat})(\iota P, \iota Q)$ is an isomorphism for every P and Q, one reasons as above: since $(\iota P)C$ is a discrete category for every $C \in \mathbb{C}$, every pseudonatural transformation $\iota P \Rightarrow \iota Q$ must be of the form $\iota(\mu)$ for a unique natural transformation $\mu : P \Rightarrow Q$, and there can be no non-identity modifications between such transformations.

To relate the 1-categorical and bicategorical Yoneda embeddings, one notes that

$$\begin{aligned}(\iota \circ \mathrm{dy})(C) &= \iota\big(\mathbb{C}(C, -)\big)\\ &= \lambda X^{\mathbb{C}} . \partial(\mathbb{C}(C, X))\\ &= \lambda X^{\mathbb{C}} .(\mathrm{d}\mathbb{C})(C, X)\\ &= \mathrm{Y}C\end{aligned}$$

as claimed.

For (4), one simply observes that the natural isomorphisms $\mathbb{C}(X, \prod_{i=1}^n A_i) \cong \prod_{i=1}^n \mathbb{C}(X, A_i)$ immediately provide 2-natural isomorphisms of hom-categories $(\mathrm{d}\mathbb{C})(X, \prod_{i=1}^n A_i) \cong \prod_{i=1}^n (\mathrm{d}\mathbb{C})(X, A_i)$, and similarly for exponentials.

For (5), recall from Theorem 6.1.10 that for pseudofunctors $G, H : \mathrm{d}\mathbb{C} \to \mathbf{Cat}$, the exponential $[G, H]$ may be given by the pseudofunctor $\mathrm{Hom}(\mathrm{d}\mathbb{C}, \mathbf{Cat})(\mathrm{Y}(-) \times G, H) : \mathrm{d}\mathbb{C} \to \mathbf{Cat}$. Next observe that the embedding ι of (3) preserves products:

$$\begin{aligned}\big(\iota(P \times Q)\big)C &= \partial\big((P \times Q)(C)\big)\\ &= \partial(PC \times QC)\\ &= \partial(PC) \times \partial(QC)\\ &= (\partial P \times \partial Q)C\\ &= \big(\iota(P) \times \iota(Q)\big)C\end{aligned}$$

Hence:

$$\begin{aligned}\mathrm{Hom}(\mathrm{d}\mathbb{C}, \mathbf{Cat})(\mathrm{Y}X \times \iota P, \iota Q) &= \mathrm{Hom}(\mathrm{d}\mathbb{C}, \mathbf{Cat})((\iota \circ \mathrm{dy})X \times \mathrm{d}P, \mathrm{d}Q) && \text{by diagram (8.6)}\\ &= \mathrm{Hom}(\mathrm{d}\mathbb{C}, \mathbf{Cat})(\iota(\mathrm{y}X) \times \iota(P), \iota(Q))\\ &= \mathrm{Hom}(\mathrm{d}\mathbb{C}, \mathbf{Cat})(\iota(\mathrm{y}X \times P), \iota(Q))\\ &\cong (\mathrm{dFun}(\mathbb{C}, \mathrm{Set}))\,(\mathrm{y}X \times P, Q) && \text{by (3)}\\ &= \partial(\mathrm{Fun}(\mathbb{C}, \mathrm{Set})(\mathrm{y}X \times P, Q)) && \text{by definition of d}(-)\end{aligned}$$

completing the proof. □

The preceding lemma provides a framework for treating the category of contexts $\mathrm{Con}_{\widetilde{\mathfrak{B}}}$ as a 2-category. Next we show how to extend from an interpretation of (base) types to an interpretation of all contexts, that is, to an fp-pseudofunctor out of ${\mathrm{dCon}_{\widetilde{\mathfrak{B}}}}^{\mathrm{op}}$. In the categorical setting, one merely uses the fact that ${\mathrm{Con}_{\widetilde{\mathfrak{B}}}}^{\mathrm{op}}$ is the free strict cartesian category on $\widetilde{\mathfrak{B}}$. The pseudo nature of bicategorical products and exponentials entails a little more work, but the construction is essentially the same.

Note that any interpretation $s : \mathfrak{B} \to \mathcal{X}$ of base types in a cc-bicategory $(\mathcal{X}, \Pi_n(-), {=\!\!\Rightarrow})$ extends canonically to an interpretation $\widetilde{\mathfrak{B}} \to \mathcal{X}$ by the cartesian closed structure, which we also denote by s.

Lemma 8.2.3. For any set of base types $\mathfrak{B}$, cc-bicategory $(\mathcal{X}, \Pi_n(-), {=\!\!\Rightarrow})$, and set map $s : \mathfrak{B} \to \mathcal{X}$, there exists an fp-pseudofunctor $\underline{s} : {\mathrm{dCon}_{\widetilde{\mathfrak{B}}}}^{\mathrm{op}} \to \mathcal{X}$ making the following diagram commute:

$$\begin{tikzcd}
 & {\mathrm{dCon}_{\widetilde{\mathfrak{B}}}}^{\mathrm{op}} \arrow[dr, "\underline{s}"] & \\
\widetilde{\mathfrak{B}} \arrow[ur, "{[-]}"] \arrow[rr] & & \mathcal{X} \\
\mathfrak{B} \arrow[u, hook] \arrow[urr, "s"'] & &
\end{tikzcd}$$

Proof. We define $\underline{s}$ on types by $\underline{s}A := sA$ and extend to contexts in the usual manner: $\underline{s}\big((x_i : A_i)_{i=1,\dots,n}\big) := \prod_{i=1}^{n} \underline{s}A_i$ and $\underline{s}(\diamond) := \prod_0()$. In particular, for a unary context $(x : A)$ we define $\underline{s}(x : A) = sA$, so that $\underline{s}[A] = sA$.

The action on 1-cells is the following. For contexts $\Gamma := (x_i : A_i)_{i=1,\dots,n}$ and $\Delta := (y_j : B_j)_{j=1,\dots,m}$ and a context renaming $r : \Gamma \to \Delta$, we define $\underline{s}r : \prod_{j=1}^{m}\underline{s}B_j \to \prod_{i=1}^{n}\underline{s}A_i$ to be $\big\langle \pi_{r(1)}, \dots, \pi_{r(n)} \big\rangle$, where we write $r(i)$ to indicate the index of $r(x_i)$ within $(y_1, \dots, y_m)$. The action on 2-cells is trivial since ${\mathrm{dCon}_{\widetilde{\mathfrak{B}}}}^{\mathrm{op}}$ is locally discrete.

For the 2-cell $\psi_{\Gamma}^{\underline{s}} : \mathrm{Id}_{\underline{s}\Gamma} \Rightarrow \underline{s}(\mathrm{Id}_{\Gamma})$ we take

$$\widehat{\varsigma}_{\mathrm{Id}_{\underline{s}\Gamma}} := \mathrm{Id}_{\underline{s}\Gamma} \xRightarrow{\varsigma_{\mathrm{Id}_{\underline{s}\Gamma}}} \langle \pi_1 \circ \mathrm{Id}_{\underline{s}\Gamma}, \dots, \pi_n \circ \mathrm{Id}_{\underline{s}\Gamma} \rangle \overset{\cong}{\Rightarrow} \langle \pi_1, \dots, \pi_n \rangle$$

For a composable pair of context renamings $\Sigma \xrightarrow{r} \Gamma \xrightarrow{r'} \Delta$, we define $\phi_{r',r}^{\underline{s}}$ to be the composite

$$\begin{tikzcd}
\big\langle \pi_{r(1)}, \dots, \pi_{r(n)} \big\rangle \circ \big\langle \pi_{r'(1)}, \dots, \pi_{r'(m)} \big\rangle \arrow[d, "\mathsf{post}"'] \arrow[dr, bend left, "\phi_{r',r}^{\underline{s}}"] & \\
\big\langle \pi_{r(1)} \circ \big\langle \pi_{r'(\bullet)} \big\rangle, \dots, \pi_{r(n)} \circ \big\langle \pi_{r'(\bullet)} \big\rangle \big\rangle \arrow[r, "{\langle \varpi^{(r(1))}, \dots, \varpi^{(r(n))} \rangle}"'] & \big\langle \pi_{r'r(1)}, \dots, \pi_{r'r(n)} \big\rangle
\end{tikzcd}$$

The three axioms to check are diagram chases using the product structure, along with the properties of Lemma 4.1.7. For the associativity law one uses naturality and the commutativity of the following diagram, in which we abbreviate $\langle \pi_{r(1)}, \dots, \pi_{r(n)} \rangle$ by $\langle \pi_r \rangle$:

$$\begin{tikzcd}
\langle \pi_r \rangle \circ \langle \pi_{r'} \rangle \circ \langle \pi_{r''} \rangle \arrow[d, "\mathsf{post} \circ \langle \pi_{r''} \rangle"'] \arrow[dr, bend left, "\mathsf{post}"] & \\
\langle \pi_r \circ \langle \pi_{r'} \rangle \rangle \circ \langle \pi_{r''} \rangle \arrow[r, "\mathsf{post}"'] & \langle \pi_r \circ \langle \pi_{r'} \rangle \circ \langle \pi_{r''} \rangle \rangle
\end{tikzcd}$$

For the left and right unit laws, one respectively uses the diagrams on the left and right below:

$$\begin{tikzcd}
\mathrm{Id}_{\underline{s}\Sigma} \circ \langle \pi_r \rangle \arrow[d, "\varsigma_{\mathrm{Id}} \circ \langle \pi_r \rangle"'] \arrow[dr, bend left, "\varsigma_{\mathrm{Id} \circ \langle \pi_r \rangle}"] & \\
\langle \pi_\bullet \circ \mathrm{Id}_{\underline{s}\Sigma} \rangle \circ \langle \pi_r \rangle \arrow[r, "\mathsf{post}"'] & \langle \pi_\bullet \circ \mathrm{Id}_{\underline{s}\Sigma} \circ \langle \pi_r \rangle \rangle
\end{tikzcd}
\qquad
\begin{tikzcd}
\langle \pi_r \rangle \circ \mathrm{Id}_{\underline{s}\Gamma} \arrow[d, "\mathsf{post}"'] \arrow[dr, bend left, "\cong"] & \\
\langle \pi_{r(1)} \circ \mathrm{Id}_{\underline{s}\Gamma}, \ldots, \pi_{r(n)} \circ \mathrm{Id}_{\underline{s}\Gamma} \rangle \arrow[r, "\cong"'] & \langle \pi_r \rangle
\end{tikzcd}$$

It remains to show that $\underline{s}$ preserves products. For n contexts $\Gamma_1, \ldots, \Gamma_n$ $(n \in \mathbb{N})$ of the form $\Gamma_i := (x_j^{(i)} : A_j^{(i)})_{j=1,\ldots,|\Gamma_i|}$, note that

$$\underline{s}(\textstyle\prod_{i=1}^n \Gamma_i) = \underline{s}(\Gamma_1 \mathbin{@} \cdots \mathbin{@} \Gamma_n) = \prod_{\substack{j_i=1,\ldots,|\Gamma_i| \\ i=1,\ldots,n}} s(A_i)$$

$$\textstyle\prod_{i=1}^n \underline{s}(\Gamma_i) = \prod_{i=1}^n \prod_{j=1}^{|\Gamma_i|} s(A_j^{(i)})$$

and that $\underline{s}(\pi_k) = \underline{s}(\Gamma_k \hookrightarrow \Gamma_1 \mathbin{@} \cdots \mathbin{@} \Gamma_n)$ is the 1-cell $\langle \pi_{1+\sum_{i=1}^{k-1}|\Gamma_i|}, \ldots, \pi_{\sum_{i=1}^{k}|\Gamma_i|} \rangle$. One therefore obtains the required equivalence $\prod_{i=1}^n \prod_{j=1}^{|\Gamma_i|} s(A_i^{(j)}) \simeq \prod_{\substack{j=1,\ldots,|\Gamma_i| \\ i=1,\ldots,n}} s(A_i^{(j)})$ by taking $\mathrm{q}^{\times}_{\Gamma_\bullet}$ to be the 1-cell $\prod_{i=1}^n \prod_{j=1}^{|\Gamma_i|} s(A_j^{(i)}) \to \prod_{\substack{j=1,\ldots,|\Gamma_i| \\ i=1,\ldots,n}} s(A_j^{(i)})$ given by

$$\langle \pi_1 \circ \pi_1, \ldots, \pi_{|\Gamma_1|} \circ \pi_1, \ldots, \pi_1 \circ \pi_k, \ldots, \pi_{|\Gamma_k|} \circ \pi_k, \ldots, \pi_1 \circ \pi_n, \ldots, \pi_{|\Gamma_n|} \circ \pi_n \rangle \tag{8.7}$$

This defines an equivalence with witnessing 2-cells defined by the commutativity of the following two diagrams:

$$\begin{tikzcd}
\langle \pi_1 \circ \pi_1, \ldots, \pi_{|\Gamma_n|} \circ \pi_n \rangle \circ \langle \underline{s}\pi_\bullet \rangle \arrow[r] \arrow[d, "\mathsf{post}"'] & \mathrm{Id}_{\underline{s}(\prod_i \Gamma_i)} \\
\left\langle \ldots, \pi_1 \circ \pi_k \circ \langle \underline{s}\pi_\bullet \rangle, \ldots, \pi_{|\Gamma_k|} \circ \pi_k \circ \langle \underline{s}\pi_\bullet \rangle, \ldots \right\rangle \arrow[d, "\langle \ldots, \pi_1 \circ \varpi^{(k)}, \ldots, \pi_{|\Gamma_k|} \circ \varpi^{(k)}, \ldots \rangle"'] & \langle \pi_1, \ldots, \pi_{\sum_{i=1}^n \sum_{j=1}^{|\Gamma_i|} j} \rangle \arrow[u, "\widehat{\varsigma}^{-1}_{\mathrm{Id}_{\underline{s}(\prod_i \Gamma_i)}}"'] \\
\left\langle \ldots, \pi_j \circ \langle \pi_{1+\sum_{i=1}^{k-1}|\Gamma_i|}, \ldots, \pi_{\sum_{i=1}^{k}|\Gamma_i|} \rangle, \ldots \right\rangle \arrow[r, "\langle \ldots, \varpi^{(j)}, \ldots \rangle"'] & \left\langle \ldots, \pi_{j+\sum_{i=1}^{k-1}|\Gamma_i|}, \ldots \right\rangle \arrow[u, equal]
\end{tikzcd}$$

$$\begin{tikzcd}
\langle \underline{s}(\pi_\bullet) \rangle \circ \langle \pi_1 \circ \pi_1, \ldots, \pi_{|\Gamma_n|} \circ \pi_n \rangle \arrow[r] \arrow[d, "\mathsf{post}"'] & \mathrm{Id}_{(\prod_i \underline{s}\Gamma_i)} \\
\left\langle \ldots, \underline{s}(\pi_k) \circ \langle \pi_1 \circ \pi_1, \ldots, \pi_{|\Gamma_n|} \circ \pi_n \rangle, \ldots \right\rangle \arrow[d, "\cong"'] & \langle \pi_1, \ldots, \pi_n \rangle \arrow[u, "\widehat{\varsigma}^{-1}_{\mathrm{Id}_{(\prod_i \underline{s}\Gamma_i)}}"'] \\
\left\langle \ldots, \langle \pi_1, \ldots, \pi_{|\Gamma_k|} \rangle \circ \pi_k, \ldots \right\rangle \arrow[r, "\langle \widehat{\varsigma}^{-1}_{\mathrm{Id}_{(\underline{s}\Gamma_n)}} \circ \pi_\bullet \rangle"'] & \left\langle \mathrm{Id}_{(\underline{s}\Gamma_1)} \circ \pi_1, \ldots, \mathrm{Id}_{(\underline{s}\Gamma_n)} \circ \pi_n \right\rangle \arrow[u, "\cong"']
\end{tikzcd}$$

The downwards arrow labelled $\cong$ is the n-ary tupling of

$$\begin{tikzcd}
\big\langle \pi_{1+\sum_{i=1}^{k-1}|\Gamma_i|}, \dots, \pi_{\sum_{i=1}^{k}|\Gamma_i|} \big\rangle \circ \langle \pi_1 \circ \pi_1, \dots, \pi_{|\Gamma_n|} \circ \pi_n \rangle \arrow[r] \arrow[d, "\mathsf{post}"'] & \langle \pi_1, \dots, \pi_{|\Gamma_k|} \rangle \circ \pi_k \\
\big\langle \dots, \pi_{j+\sum_{i=1}^{k-1}|\Gamma_i|} \circ \langle \pi_1 \circ \pi_1, \dots, \pi_{|\Gamma_n|} \circ \pi_n \rangle, \dots \big\rangle_{j=1,\dots,|\Gamma_k|} \arrow[r, "{\langle \dots, \varpi^{(j+\sum_{i=1}^{k-1})}, \dots \rangle}"'] & \langle \dots, \pi_j \circ \pi_k, \dots \rangle_{j=1,\dots,|\Gamma_k|} \arrow[u, "\mathsf{post}^{-1}"']
\end{tikzcd}$$

for $k = 1, \dots, n$. Hence $\underline{s}$ is an fp-pseudofunctor, as claimed. □

Remark 8.2.4. We shall need the following special case of the fact that the pseudofunctor $\underline{s}$ preserves products. For a context $\Gamma = (x_i : A_i)_{i=1,\dots,n}$ and type A, the 1-cell (8.7) becomes simply $\langle \pi_1 \circ \pi_1, \dots, \pi_n \circ \pi_1, \pi_2 \rangle : \underline{s}\Gamma \times \underline{s}[A] \to \underline{s}(\Gamma \mathbin{@} [A])$. ◂

One also obtains the following version of Proposition 5.3.22 by taking the *context extension* product structure of the syntactic model instead of the type-theoretic product structure (recall Section 4.3.3).

Proposition 8.2.5. For any $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature $\mathcal{S}$, cc-bicategory $(\mathcal{X}, \Pi_n(-), \Rightarrow)$, and $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature homomorphism $s : \mathcal{S} \to \mathcal{X}$, there exists a cc-pseudofunctor $s[\![-]\!] : \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\mathcal{S}) \to \mathcal{X}$ with respect to the context extension product structure, such that $s[\![-]\!] \circ \iota = s$, for $\iota : \mathcal{S} \hookrightarrow \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\mathcal{S})$ the inclusion.

Proof. Define $s[\![-]\!]$ as in Proposition 5.3.22, except that for preservation of products one takes $\mathrm{q}^{\times}$ as in the preceding lemma. Preservation of exponentials then takes the following form. For $\Gamma := (x_i : A_i)_{i=1,\dots,n}$ and $\Delta := (y_j : B_j)_{j=1,\dots,m}$, the evaluation map is the m-tuple with components

$$f : \textstyle\prod_n A_\bullet \Rightarrow \prod_m B_\bullet, x_1 : A_1, \dots, x_n : A_n \vdash \pi_j\{\mathsf{eval}\{f, \mathsf{tup}(x_1, \dots, x_n)\}\} : B_j$$

for $j = 1, \dots, m$. One then obtains the following chain of natural isomorphisms:

$$\begin{aligned}
&s[\![\mathrm{eval}_{\Gamma,\Delta}]\!] \circ \mathrm{q}^{\times}_{\Gamma \Rightarrow \Delta, \Gamma} \\
&\qquad = \big\langle \pi_\bullet \circ \mathrm{eval}_{s[\![\prod_n A_\bullet]\!], s[\![\prod_m B_\bullet]\!]} \circ \langle \pi_1, \langle \pi_2, \dots, \pi_{n+1} \rangle \rangle \big\rangle \circ \langle \pi_1, \pi_1 \circ \pi_2, \dots, \pi_n \circ \pi_2 \rangle \\
&\qquad \cong \big\langle \pi_\bullet \circ \mathrm{eval}_{s[\![\prod_n A_\bullet]\!], s[\![\prod_m B_\bullet]\!]} \circ \langle \pi_1, \langle \pi_1 \circ \pi_2, \dots, \pi_n \circ \pi_2 \rangle \rangle \big\rangle \\
&\qquad \cong \big\langle \pi_\bullet \circ \mathrm{eval}_{s[\![\prod_n A_\bullet]\!], s[\![\prod_m B_\bullet]\!]} \circ \langle \pi_1, \langle \pi_1, \dots, \pi_n \rangle \circ \pi_2 \rangle \big\rangle \\
&\qquad \cong \big\langle \pi_\bullet \circ \mathrm{eval}_{s[\![\prod_n A_\bullet]\!], s[\![\prod_m B_\bullet]\!]} \circ \langle \pi_1, \pi_2 \rangle \big\rangle \\
&\qquad \cong \big\langle \pi_\bullet \circ \mathrm{eval}_{s[\![\prod_n A_\bullet]\!], s[\![\prod_m B_\bullet]\!]} \big\rangle \\
&\qquad \cong \langle \pi_1, \dots, \pi_m \rangle \circ \mathrm{eval}_{s[\![\prod_n A_\bullet]\!], s[\![\prod_m B_\bullet]\!]} \\
&\qquad \cong \mathrm{eval}_{s[\![\prod_n A_\bullet]\!], s[\![\prod_m B_\bullet]\!]}
\end{aligned}$$

It follows that $\mathrm{m}_{\Gamma,\Delta} = \lambda(s[\![\mathrm{eval}_{\Gamma,\Delta}]\!]) \cong \lambda\big(\mathrm{eval}_{s[\![\prod_n A_\bullet]\!], s[\![\prod_m B_\bullet]\!]}\big) \cong \mathrm{id}_{s[\![\Gamma \Rightarrow \Delta]\!]}$, so $s[\![-]\!]$ preserves exponentials. □

While the interpretation of Proposition 5.3.22 is useful for proving uniqueness properties, the interpretation of the preceding proposition is the natural choice when working with the (2-)category of contexts. Of course, the two pseudofunctors are canonically equivalent. Throughout this chapter, we shall work with the version just defined.

For any interpretation of base types $s : \mathfrak{B} \to \mathcal{X}$ in a cc-bicategory $(\mathcal{X}, \Pi_n(-), \Rightarrow)$, one therefore obtains the following diagram lifting (8.2) to the bicategorical setting:

$$\begin{tikzcd}
 & \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}}) \arrow[dr, "s[\![-]\!]"] & \\
\mathrm{dCon}_{\widetilde{\mathfrak{B}}}{}^{\mathrm{op}} \arrow[ur, "\underline{\iota}"] \arrow[rr, "\underline{s}"'] & & \mathcal{X} \\
\widetilde{\mathfrak{B}} \arrow[u, "{[-]}"] \arrow[urr] & & \\
\mathfrak{B} \arrow[u] \arrow[uurr, "s"] & &
\end{tikzcd}$$

Note in particular that, just as in the 1-categorical case, the equality $s[\![\Gamma]\!] = \underline{s}\Gamma$ holds for every context Γ.

Syntactic presheaves for $\Lambda_{\mathrm{ps}}^{\times,\to}$. Lemma 8.2.3 provides a way to interpret contexts whenever one has an interpretation of base types, while Lemma 8.2.2 guarantees that, in order to interpret the syntax of $\Lambda_{\mathrm{ps}}^{\times,\to}$ as a pseudofunctor $\mathrm{dCon}_{\widetilde{\mathfrak{B}}} \to \mathbf{Cat}$, it suffices to a define a presheaf $\mathrm{Con}_{\widetilde{\mathfrak{B}}} \to \mathrm{Set}$ on the underlying category. There remains the question of what it means to be a neutral or normal term in $\Lambda_{\mathrm{ps}}^{\times,\to}$. The answer is provided by the embedding of $\Lambda^{\times,\to}$ into $\Lambda_{\mathrm{ps}}^{\times,\to}$ constructed in Section 5.4. Thus, for every $A \in \widetilde{\mathfrak{B}}$ we define four presheaves $\mathcal{V}(-;A), \mathcal{M}(-;A), \mathcal{N}(-;A), \mathcal{L}(-;A) : \mathrm{Con}_{\widetilde{\mathfrak{B}}} \to \mathrm{Set}$ by setting

$$\begin{aligned}
\mathcal{V}(\Gamma; A) &:= \{(\!| t |\!) \mid t \in V(\Gamma; A)\} \\
\mathcal{M}(\Gamma; A) &:= \{(\!| t |\!) \mid t \in M(\Gamma; A)\} \\
\mathcal{N}(\Gamma; A) &:= \{(\!| t |\!) \mid t \in N(\Gamma; A)\} \\
\mathcal{L}(\Gamma; A) &:= \{(\!| t |\!) \mid t \in L(\Gamma; A)\}
\end{aligned} \tag{8.8}$$

where $(\!| - |\!)$ is defined in Construction 5.4.3 on page 191 and the presheaves $V(;A), M(-;A), N(-;A)$ and $L(-;A)$ are defined in (8.3) on page 257. Since $(\!| - |\!)$ respects α-equivalence (Lemma 5.4.4), these definitions are well-defined on α-equivalence classes. To see that these definitions are invariant under variable renamings, recall from Construction 5.4.6 that the following rule is admissible in $\Lambda_{\mathrm{ps}}^{\times,\to}$:

$$\dfrac{\Gamma \vdash (\!| t |\!) : B \qquad r : \Gamma \to \Delta}{\Delta \vdash \mathsf{cont}(t;r) : (\!| t |\!)\{x_i \mapsto r(x_i)\} \Rightarrow (\!| t[r(x_i)/x_i] |\!) : B}$$

Since a rewrite $\tau : t \Rightarrow t'$ is typeable in context Γ only if both t and t' are also typeable in Γ, it follows that the following rule is admissible:

$$\frac{\Gamma \vdash (\!| t |\!) : B \qquad r : \Gamma \to \Delta}{\Delta \vdash (\!| t[r(x_i)/x_i] |\!) : B}$$

Since the presheaves (8.3) are invariant under renamings, it follows that those of (8.8) are too, as required.

The functorial action is the unique choice such that the following diagram commutes, where $K(-;A) \in \{V(-;A), M(-;A), N(-;A)\}$ and $\mathcal{K}(-;A)$ denotes the image of $K(-;A)$ under $(\!| - |\!)$:

$$\begin{tikzcd} K(\Gamma;A) \arrow[r, "K(r;A)"] \arrow[d, "(\!| - |\!)_A^{\Gamma}"'] & K(\Delta;A) \arrow[d, "(\!| - |\!)_A^{\Delta}"] \\ \mathcal{K}(\Gamma;A) \arrow[r, "\mathcal{K}(r;A)"'] & \mathcal{K}(\Delta;A) \end{tikzcd} \tag{8.9}$$

Explicitly, for a context renaming $r : \Gamma \to \Delta$ we define $\mathcal{K}(-;A)(r)((\!| t |\!)_A^{\Gamma}) := (\!| t[r(x_i)/x_i] |\!)_A^{\Delta}$.

This formulation is particularly convenient as it allows one to make use of standard facts about the simply-typed lambda calculus. Moreover, we can employ many of the details of Fiore's proof via the following observation.

Lemma 8.2.6. For any type $A \in \widetilde{\mathfrak{B}}$, let $K(-;A) \in \{V(-;A), M(-;A), N(-;A), L(-;A)\}$ and let $\mathcal{K}(-;A) \in \{\mathcal{V}(-;A), \mathcal{M}(-;A), \mathcal{N}(-;A), \mathcal{L}(-;A)\}$ denote the image of K_A under $(\!| - |\!)$. Then the mappings $(\!| - |\!)_A^{(=)} : K_A \Rightarrow \mathcal{K}_A$ form a natural isomorphism.

Proof. Since $(\!| - |\!)_A^{(=)}$ respects the typings, it is clear from the definition that it is an injection, hence a bijection onto its image. Naturality is exactly (8.9). □

For example, one may immediately extend the natural transformations of Lemma 8.1.2 to $\Lambda_{\mathrm{ps}}^{\times,\to}$. One therefore obtains the following natural transformations:

$$\begin{aligned} \mathsf{var}(-;A_i) &: \mathcal{V}(-;A_i) \Rightarrow \mathcal{M}(-;A_i) \\ \mathsf{inc}(-;B) &: \mathcal{M}(-;B) \Rightarrow \mathcal{N}(-;B) && (B \text{ a base type}) \\ \mathsf{proj}_k(-;A_\bullet) &: \mathcal{M}(-;\textstyle\prod_n(A_1,\ldots,A_n)) \Rightarrow \mathcal{M}(-;A_k) && (k = 1,\ldots,n) \\ \mathsf{app}(-;A,B) &: \mathcal{M}(-;A \Rightarrow B) \times \mathcal{N}(-;A) \Rightarrow \mathcal{M}(-;B) \\ \mathsf{tuple}(-;A_\bullet) &: \textstyle\prod_{i=1}^n \mathcal{N}(-;A_i) \Rightarrow \mathcal{N}(-;\prod_n(A_1,\ldots,A_n)) \\ \mathsf{lam}(-;A,B) &: \mathcal{N}\big(-+[A];B\big) \Rightarrow \mathcal{N}(-;A \Rightarrow B) \end{aligned} \tag{8.10}$$

Explicitly, the action on terms is the following:

$$\begin{aligned} x_k &\mapsto x_k \\ (\!| t |\!) &\mapsto (\!| t |\!) \end{aligned}$$

$$\begin{aligned}
(\!|\, t \,|\!) &\mapsto (\!|\, \pi_k(t) \,|\!) = \pi_k\{(\!|\, t \,|\!)\} && (k = 1, \ldots, n) \\
((\!|\, t \,|\!), (\!|\, u \,|\!)) &\mapsto (\!|\, \mathsf{app}(t, u) \,|\!) = \mathsf{eval}\{(\!|\, t \,|\!), (\!|\, u \,|\!)\} \\
((\!|\, t_1 \,|\!), \ldots, (\!|\, t_n \,|\!)) &\mapsto (\!|\, \langle t_1, \ldots, t_n \rangle \,|\!) = \mathsf{tup}((\!|\, t_1 \,|\!), \ldots, (\!|\, t_n \,|\!)) \\
(\!|\, t \,|\!) &\mapsto (\!|\, \lambda x.t \,|\!) = \lambda x.(\!|\, t \,|\!)
\end{aligned}$$

The presheaves (8.8) and natural transformations (8.10)—viewed as locally discrete pseudofunctors and locally discrete pseudonatural transformations—describe the syntax of $\Lambda_{\mathrm{ps}}^{\times,\rightarrow}$ within $\mathrm{Hom}(\mathrm{dCon}_{\widetilde{\mathfrak{B}}}{}^{\mathrm{op}}, \mathbf{Cat})$. As we saw in Chapter 6, this bicategory shares many of the important features of the presheaf category $\mathcal{P}(\mathrm{Con}_{\widetilde{\mathfrak{B}}}{}^{\mathrm{op}})$. Our next task, therefore, is to construct the bicategorical correlate to the category of intensional Kripke relations.

8.2.1 Bicategorical intensional Kripke relations

The relative hom-pseudofunctor. We start by constructing the pseudo correlate of the relative hom-functor and establishing its key properties. Precisely, we show that diagram (8.1) on page 256 lifts to the bicategorical setting, and that the relative hom-pseudofunctor preserves bilimits.

The construction is the natural bicategorification of Definition 8.1.1.

Construction 8.2.7. For any pseudofunctor $\mathfrak{J} : \mathcal{B} \to \mathcal{X}$ one obtains a *relative hom-pseudofunctor* $\langle\mathfrak{J}\rangle : \mathcal{X} \to \mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$ as follows.

On objects, we set $\langle\mathfrak{J}\rangle X := \mathcal{X}(\mathfrak{J}(-), X)$. On morphisms, we define a pseudonatural transformation $\langle\mathfrak{J}\rangle f : \langle\mathfrak{J}\rangle X \Rightarrow \langle\mathfrak{J}\rangle X'$ for every $f : X \to X'$ in $\mathcal{X}$. The 1-cell components are

$$(\langle\mathfrak{J}\rangle f)_B := \mathcal{X}(\mathfrak{J}B, X) \xrightarrow{f \circ (-)} \mathcal{X}(\mathfrak{J}B, X')$$

and for $g : B' \to B$ in $\mathcal{B}$ the witnessing 2-cell $\overline{(\langle\mathfrak{J}\rangle f)}_g$ filling

$$\begin{tikzcd}
\mathcal{X}(\mathfrak{J}B, X) \arrow[r, "(\langle\mathfrak{J}\rangle X)(g)"] \arrow[d, "f\circ(-)"'] \arrow[dr, phantom, "\overset{\overline{(\langle\mathfrak{J}\rangle f)}_g}{\Leftarrow}"] & \mathcal{X}(\mathfrak{J}B', X) \arrow[d, "f\circ(-)"] \\
\mathcal{X}(\mathfrak{J}B, X') \arrow[r, "(\langle\mathfrak{J}\rangle X')(g)"'] & \mathcal{X}(\mathfrak{J}B', X')
\end{tikzcd}$$

is the structural isomorphism $\lambda h^{\mathcal{X}(\mathfrak{J}B,X)} . \mathsf{a}^{-1}_{f,h,\mathfrak{J}g}$. Finally, for a 2-cell $\tau : f \Rightarrow f'$ in $\mathcal{X}$, we define a modification $\langle\mathfrak{J}\rangle f \to \langle\mathfrak{J}\rangle f'$ by setting $\langle\mathfrak{J}\rangle\tau := \tau \circ (-)$. The modification axiom holds by the naturality of the associator a.

It remains to give the extra data witnessing preservation of units and composition. For $\psi_X^{\langle\mathfrak{J}\rangle} : \mathrm{Id}_{\langle\mathfrak{J}\rangle X} \Rightarrow \langle\mathfrak{J}\rangle(\mathrm{Id}_X)$ we take the modification with components given by the structural isomorphisms $\mathrm{id}_{\mathcal{X}(\mathfrak{J}B,X)} \overset{\cong}{\Rightarrow} \mathrm{Id}_X \circ (-)$. Similarly, for a composable pair $X \xrightarrow{g} X' \xrightarrow{f} X''$ in $\mathbb{X}$, the modification $\phi_{f,g}^{\langle\mathfrak{J}\rangle} : \langle\mathfrak{J}\rangle(f) \circ \langle\mathfrak{J}\rangle(g) \Rightarrow \langle\mathfrak{J}\rangle(f \circ g)$ has components $f \circ (g \circ (-)) \overset{\cong}{\Rightarrow} (f \circ g) \circ (-)$. ◂

The preceding construction leads us to the following definition (*c.f.* Definition 8.1.1).

Definition 8.2.8. For a category $\mathcal{B}$ and pseudofunctor $\mathfrak{J} : \mathcal{B} \to \mathcal{X}$, the bicategory of $\mathcal{B}$-*intensional Kripke relations of arity* $\mathfrak{J}$ is the glueing bicategory $\mathrm{gl}(\langle\mathfrak{J}\rangle)$ associated to the relative hom-pseudofunctor. ◂

To bicategorify (8.1) we employ the canonical equivalences $\mathrm{Hom}(\mathcal{C} \times \mathcal{B}, \mathcal{V}) \simeq \mathrm{Hom}(\mathcal{B} \times \mathcal{C}, \mathcal{V}) \simeq \mathrm{Hom}\big(\mathcal{B}, \mathrm{Hom}(\mathcal{C}, \mathcal{V})\big)$ of [Str80, §1.34].

Lemma 8.2.9. For any pseudofunctor $\mathfrak{J} : \mathcal{B} \to \mathcal{X}$ there exists a pseudonatural transformation $(l, \bar{l})$ as in the diagram

$$\begin{tikzcd}
\mathcal{B}^{\mathrm{op}} \times \mathcal{B} \arrow[rr, "\mathrm{Hom}(-,=)"] \arrow[dr, "\mathfrak{J}^{\mathrm{op}}\times\mathfrak{J}"'] & {} \arrow[d, phantom, "\Downarrow (l,\bar{l})"] & \mathbf{Cat} \\
& \mathcal{X}^{\mathrm{op}} \times \mathcal{X} \arrow[ur, "\mathrm{Hom}(-,=)"'] &
\end{tikzcd} \tag{8.11}$$

where

$$\mathfrak{J}^{\mathrm{op}} := \mathfrak{J} : ob(\mathcal{B}^{\mathrm{op}}) \to ob(\mathcal{X}^{\mathrm{op}})$$

$$(\mathfrak{J}_{B,C})^{\mathrm{op}} := \mathcal{B}^{\mathrm{op}}(B, C) = \mathcal{B}(C, B) \xrightarrow{\mathfrak{J}_{C,B}} \mathcal{X}(C, B) = \mathcal{X}^{\mathrm{op}}(C, B)$$

Proof. For the functors $l_{(B,C)} : \mathcal{B}(B, C) \to \mathcal{X}(\mathfrak{J}B, \mathfrak{J}C)$ we take $\mathfrak{J}_{B,C}$. For $f : B' \to B$ and $g : C \to C'$, the witnessing isomorphism $\bar{l}_{(f,g)}$ in the diagram below

$$\begin{tikzcd}
\mathcal{B}(B, C) \arrow[rr, "\mathcal{B}(f,g)"] \arrow[d, "\mathfrak{J}_{B,C}"'] & \arrow[d, phantom, "\overset{\bar{l}_{(f,g)}}{\Leftarrow}"] & \mathcal{B}(B', C') \arrow[d, "\mathfrak{J}_{B',C'}"] \\
\mathcal{X}(\mathfrak{J}B, \mathfrak{J}C) \arrow[rr, "\mathcal{X}(\mathfrak{J}^{\mathrm{op}}f, \mathfrak{J}g)"'] & {} & \mathcal{X}(\mathfrak{J}B', \mathfrak{J}C')
\end{tikzcd}$$

is defined to be the composite natural isomorphism

$$\mathfrak{J}\big(g \circ (h \circ f)\big) \xRightarrow{(\phi^{\mathfrak{J}}_{g,h\circ f})^{-1}} \mathfrak{J}(g) \circ \mathfrak{J}(h \circ f) \xRightarrow{\mathfrak{J}(g)\circ(\phi^{\mathfrak{J}}_{h,f})^{-1}} \mathfrak{J}(g) \circ \big(\mathfrak{J}h \circ \mathfrak{J}f\big) \tag{8.12}$$

This composite is natural in g and f; the unit and associativity laws follow from the corresponding laws of a pseudofunctor. □

Corollary 8.2.10. For any pseudofunctor $\mathfrak{J} : \mathcal{B} \to \mathcal{X}$ there exists a pseudonatural transformation $(l, \bar{l}) : \mathrm{Y} \Rightarrow \langle\mathfrak{J}\rangle \circ \mathfrak{J} : \mathcal{B} \to \mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$, which is given by the functorial action of $\mathfrak{J}$ on hom-categories.

Proof. Passing (8.11) through the equivalences $\mathrm{Hom}(\mathcal{B}^{\mathrm{op}} \times \mathcal{B}, \mathbf{Cat}) \simeq \mathrm{Hom}(\mathcal{B} \times \mathcal{B}^{\mathrm{op}}, \mathbf{Cat}) \simeq \mathrm{Hom}\big(\mathcal{B}, \mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})\big)$ at an arbitrary $P : \mathcal{B}^{\mathrm{op}} \times \mathcal{B} \to \mathbf{Cat}$ yields the following:

$$\lambda(B, C)^{\mathcal{B}^{\mathrm{op}}\times\mathcal{B}} . P(B, C) \mapsto \lambda(C, B)^{\mathcal{B}\times\mathcal{B}^{\mathrm{op}}} . P(B, C) \mapsto \lambda C^{\mathcal{B}} . \lambda B^{\mathcal{B}^{\mathrm{op}}} . P(B, C)$$

so that $\mathrm{Hom}(-,=) \mapsto \lambda C^{\mathcal{B}} . \mathrm{Y}C$ and $\mathrm{Hom}(\mathfrak{J}(-),\mathfrak{J}(=)) \mapsto \lambda C^{\mathcal{B}} . \langle\mathfrak{J}\rangle(C)$. By the preceding lemma, these are related by the pseudonatural transformation with components $l_C := \mathfrak{J}_{(-),C} : \mathcal{B}(-,C) \to \mathcal{X}\big(\mathfrak{J}(-),\mathfrak{J}C\big)$ and witnessing 2-cells given as in (8.12). □

We may now extend the Yoneda pseudofunctor Y to its glued counterpart $\underline{\mathrm{Y}}$.

Construction 8.2.11. For any pseudofunctor $\mathfrak{J} : \mathcal{B} \to \mathcal{X}$, define the *extended Yoneda pseudofunctor* $\underline{\mathrm{Y}} : \mathcal{B} \to \mathrm{gl}(\langle\mathfrak{J}\rangle)$ as follows.

On objects, we set

$$\underline{\mathrm{Y}}B := \big(\mathrm{Y}B, (l,\bar{l})_{(-,B)}, \mathfrak{J}B\big) \tag{8.13}$$

where $(l,\bar{l})_{(-,B)}$ is pseudonatural since $(l,\bar{l})$ is pseudonatural in both arguments.

For a 1-cell $f : B \to B'$ in $\mathcal{B}$, we define $\underline{\mathrm{Y}}f$ to be the 1-cell $(\mathrm{Y}f, (\phi^{\mathfrak{J}}_{-,f})^{-1}, \mathfrak{J}f)$ as in the diagram

$$\begin{tikzcd}
\mathcal{B}(-,B) \arrow[r, "f\circ(-)"] \arrow[d, "\mathfrak{J}_{-,B}"'] & \mathcal{B}(-,B') \arrow[d, "\mathfrak{J}_{-,B'}"] \arrow[dl, phantom, "\overset{(\phi^{\mathfrak{J}}_{-,f})^{-1}}{\Leftarrow}"] \\
\mathcal{X}\big(\mathfrak{J}(-),\mathfrak{J}B\big) \arrow[r, "\mathfrak{J}(f)\circ(-)"'] & \mathcal{X}\big(\mathfrak{J}(-),\mathfrak{J}B'\big)
\end{tikzcd}$$

On 2-cells, we set $\underline{\mathrm{Y}}(\tau : f \Rightarrow f' : B \to B')$ to be the pair $(\mathrm{Y}\tau, \mathfrak{J}\tau)$, which satisfies the cylinder condition by the naturality of $\phi^{\mathfrak{J}}$.

Finally we need to define $\psi^{\underline{\mathrm{Y}}}$ and $\phi^{\underline{\mathrm{Y}}}$. Since $\mathrm{Y}\mathrm{Id}_X = (\mathrm{Y}\mathrm{Id}_X, \mathfrak{J}\mathrm{Id}_X)$, we may take simply $\psi^{\underline{\mathrm{Y}}} := (\psi^{\mathrm{Y}}, \psi^{\mathfrak{J}})$. This forms a 2-cell in $\mathrm{gl}(\langle\mathfrak{J}\rangle)$ by the unit law on $(l,\bar{l})$. Similarly, for $\phi^{\underline{\mathrm{Y}}}$ we take $(\phi^{\mathrm{Y}}, \phi^{\mathfrak{J}})$, which satisfies the cylinder condition by the associativity law on $(l,\bar{l})$. The three axioms to check then hold pointwise. ◂

In the next section we shall provide an explicit presentation of exponentials $\underline{\mathrm{Y}}B \Rightarrow \underline{X}$ in the glueing bicategory, which will provide a bicategorical, glued correlate of the identification $[\mathrm{y}B, P] \cong P(- \times X)$ for presheaves. First, however, we finish our examination of the relative hom-pseudofunctor by showing that it preserves bilimits.

Lemma 8.2.12. For any pseudofunctor $\mathfrak{J} : \mathcal{B} \to \mathcal{X}$ the relative hom-pseudofunctor $\langle\mathfrak{J}\rangle : \mathcal{X} \to \mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$ preserves all bilimits that exist in $\mathcal{X}$.

Proof. Let $H : \mathcal{J} \to \mathcal{X}$ be a pseudofunctor and suppose the bilimit $(\mathrm{bilim}_{j\in\mathcal{J}} Hj, \lambda_j)$ exists in $\mathcal{X}$. By Proposition 6.0.1, the bilimit $\mathrm{bilim}(\langle\mathfrak{J}\rangle \circ H)$ exists in $\mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$ and is given pointwise.

Now, since representable pseudofunctors preserve bilimits (Lemma 2.3.4), the canonical map $e_B : \mathrm{bilim}_{j\in\mathcal{J}} \mathcal{X}(\mathfrak{J}B, Hj) \to \mathcal{X}(\mathfrak{J}B, \mathrm{bilim}_{j\in\mathcal{J}} Hj)$ is an equivalence for every $B \in \mathcal{B}$. These extend canonically to a pseudonatural transformation, yielding the required equivalence $\mathrm{bilim}(\langle\mathfrak{J}\rangle \circ H) \overset{\simeq}{\Rightarrow} \langle\mathfrak{J}\rangle\,(\mathrm{bilim}\, H)$. □

It will be useful to have an explicit description of how $\langle\mathfrak{J}\rangle$ preserves products. For this we rely on the post 2-cells.

Lemma 8.2.13. For any fp-bicategory $(\mathcal{B}, \Pi_n(-))$, the n-ary tupling operation and 2-cells post together form a pseudonatural transformation $\prod_{i=1}^n \mathcal{B}(-, B_i) \Rightarrow \mathcal{B}(-, \prod_{i=1}^n B_i)$, and hence an equivalence of pseudofunctors $\prod_{i=1}^n \mathcal{B}(-, B_i) \simeq \mathcal{B}(-, \prod_{i=1}^n B_i)$ in $\mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$.

Proof. For every $X \in \mathcal{B}$ the n-ary tupling operation defines a functor $\langle -, \ldots, = \rangle : \prod_{i=1}^n \mathcal{B}(X, B_i) \to \mathcal{B}\left(X, \prod_{i=1}^n B_i\right)$ which, by the definition of an fp-bicategory (Definition 4.1.1), is an equivalence in $\mathbf{Cat}$. For these functors to be the components of a pseudonatural transformation, we need to provide an invertible 2-cell filling the diagram below for every $f : Y \to X$:

$$\begin{tikzcd}
\prod_{i=1}^n \mathcal{B}(X, B_i) \arrow[r, "\prod_{i=1}^n \mathcal{B}(f, B_i)"] \arrow[d, "\langle -, \ldots, = \rangle"'] \arrow[dr, phantom, "\Leftarrow"] & \prod_{i=1}^n \mathcal{B}(Y, B_i) \arrow[d, "\langle -, \ldots, = \rangle"] \\
\mathcal{B}(X, \prod_{i=1}^n B_i) \arrow[r, "\mathcal{B}(f, \prod_{i=1}^n B_i)"'] & \mathcal{B}(Y, \prod_{i=1}^n B_i)
\end{tikzcd}$$

Thus, we require a natural isomorphism $\langle h_1 \circ f, \ldots, h_n \circ f \rangle \Rightarrow \langle h_1, \ldots, h_n \rangle \circ f$, for which we take $\mathsf{post}(h_\bullet, f)^{-1}$. The two axioms are exercises in using Lemma 4.1.7. □

Corollary 8.2.14. For any pseudofunctor $\mathfrak{J} : \mathcal{B} \to \mathcal{X}$, the relative hom-pseudofunctor $\langle\mathfrak{J}\rangle$ extends to an fp-pseudofunctor $(\langle\mathfrak{J}\rangle, \mathrm{q}^\times)$ with $\mathrm{q}^\times_{X_\bullet}$ given by the pseudonatural transformation $(\langle -, \ldots, = \rangle, \mathsf{post})$ defined in the preceding lemma. □

Remark 8.2.15. From the perspective of biuniversal arrows, Lemma 8.2.13 is an instance of Lemma 2.4.4. ◂

8.2.2 Exponentiating by glued representables

In order to emulate Fiore's construction of the 1-cells quote and $\mathsf{unquote}$ in the glueing bicategory, we require a correlate of the following categorical fact:

Lemma 8.2.16 ([Fio02])**.** For any cartesian category $\mathbb{B}$, cartesian closed category $\mathbb{X}$ and cartesian functor $\mathfrak{J} : \mathbb{B} \to \mathbb{X}$, the exponential $\left[\underline{\mathrm{y}}B, (P, p, X)\right]$ in $\mathrm{gl}(\langle\mathfrak{J}\rangle)$ may be described explicitly as

$$[\mathrm{y}B, P] \xrightarrow{[\mathrm{y}B, p]} [\mathrm{y}B, \langle\mathfrak{J}\rangle(X)] \xrightarrow{\cong} \langle\mathfrak{J}\rangle\,(\mathfrak{J}B \Rightarrow X)$$

Here the unlabelled isomorphism is the composite

$$[\mathrm{y}B, \langle\mathfrak{J}\rangle(X)] \xrightarrow{\cong} \mathbb{X}\,(\mathfrak{J}(- \times B), X) \xrightarrow{\cong} \mathbb{X}\,(\mathfrak{J}(-) \times \mathfrak{J}B, X) \xrightarrow{\cong} \mathbb{X}\,(\mathfrak{J}(-), \mathfrak{J}B \Rightarrow X)$$

arising from the canonical isomorphism $[\mathrm{y}B, P] \cong P(- \times X)$, the product-preservation of $\mathfrak{J}$, and the cartesian closed structure on $\mathbb{X}$. □

For the bicategorical version of this lemma we note that, since products in **Cat** are strict, one obtains $\mathrm{id}_P \times \mathrm{id}_Q = \mathrm{id}_{P \times Q}$ for every $P, Q : \mathcal{B}^{\mathrm{op}} \to \mathbf{Cat}$, so that $\big[\mathrm{id}_P, (\mathsf{k}, \overline{\mathsf{k}})\big] : [P, Q] \Rightarrow [P, Q']$ is equal to $\Lambda\big((\mathsf{k}, \overline{\mathsf{k}}) \circ (e, \overline{e})\big)$ (recall from Section 6.1 that $(e, \overline{e})$ denotes the evaluation 1-cell in $\mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$). With our (locally discrete) use-case in mind, we shall simplify what follows by assuming the bicategory $\mathcal{B}$ to be a 2-category.

Proposition 8.2.17. For any 2-category $\mathcal{B}$ with pseudo-products, cc-bicategory $(\mathcal{X}, \Pi_n(-), \Rightarrow)$ and fp-pseudofunctor $(\mathfrak{J}, \mathrm{q}^{\times}) : (\mathcal{B}, \Pi_n(-)) \to (\mathcal{X}, \Pi_n(-))$, the exponential $\underline{\mathrm{Y}}B \Rightarrow \big(K, (\mathsf{k}, \overline{\mathsf{k}}), X\big)$ in $\mathrm{gl}(\langle \mathfrak{J} \rangle)$ may be given explicitly by the following composite in $\mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$:

$$[\mathrm{Y}B, K] \xrightarrow{[\mathrm{Y}B, (\mathsf{k}, \overline{\mathsf{k}})]} [\mathrm{Y}B, \langle \mathfrak{J} \rangle X] \xrightarrow{u_{B,X}} \langle \mathfrak{J} \rangle (\mathfrak{J}B \Rightarrow X) \tag{8.14}$$

where $u_{B,X}$ is the composite of equivalences

$$[\mathrm{Y}B, \langle \mathfrak{J} \rangle X] \xrightarrow{\overset{(1)}{\simeq}} \mathcal{X}\big(\mathfrak{J}(- \times B), X\big) \xrightarrow{\overset{(2)}{\simeq}} \mathcal{X}\big(\mathfrak{J}(-) \times \mathfrak{J}B, X\big) \xrightarrow{\overset{(3)}{\simeq}} \mathcal{X}\big(\mathfrak{J}(-), \mathfrak{J}B \Rightarrow X\big) \tag{8.15}$$

arising from the following, respectively:

1. The canonical equivalence arising from the identification of $(\langle \mathfrak{J} \rangle X)(- \times B)$ as $[\mathrm{Y}B, \langle \mathfrak{J} \rangle X]$ (Theorem 6.2.7),
2. The fact that $\mathfrak{J}$ preserves products,
3. The definition of exponentials in $\mathcal{X}$. □

Our strategy is to show that the composite (8.14) is the left-hand leg of a pullback diagram in $\mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$; by Lemma 7.3.8, this is sufficient to prove an equivalence in the glueing bicategory. We prove this using the following fact, which generalises the 1-categorical situation.

Lemma 8.2.18. Let $\mathcal{B}$ be a bicategory and $e : B \leftrightarrows C : f$ be any adjoint equivalence in $\mathcal{B}$, with witnessing invertible 2-cells $\mathsf{v} : \mathrm{Id}_C \overset{\cong}{\Rightarrow} e \circ f$ and $\mathsf{w} : f \circ e \overset{\cong}{\Rightarrow} \mathrm{Id}_B$. Then for any $r : A \to C$ the pullback of the cospan $(B \xrightarrow{e} C \xleftarrow{r} A)$ exists and is given by

$$\begin{tikzcd}
A \arrow[r, "\mathrm{Id}_A"] \arrow[d, "r"'] \arrow[dd, bend right=60, "f \circ r"'] & A \arrow[dd, "r"] \\
C \arrow[d, "f"'] \arrow[dr, bend left, "\mathrm{Id}_C"] \arrow[ur, phantom, "\cong"] & \\
B \arrow[r, "e"'] \arrow[ur, phantom, "\overset{\mathsf{v}}{\cong}"] & C
\end{tikzcd} \tag{8.16}$$

where the top isomorphism is a composite of structural isomorphisms.

Proof. Suppose given any other iso-commuting square

$$\begin{tikzcd}
X \arrow[r, "p"] \arrow[d, "q"'] \arrow[dr, phantom, "\overset{\rho}{\cong}"] & A \arrow[d, "r"] \\
B \arrow[r, "e"'] & C
\end{tikzcd}$$

We take the mediating map $X \to A$ to be p. For the 2-cells we take $\Gamma := \mathrm{Id}_A \circ p \overset{\cong}{\Rightarrow} p$ and Δ to be defined by the following diagram:

$$\begin{tikzcd}
(f \circ r) \circ p \arrow[r, "\Delta"] \arrow[d, "\cong"'] & q \\
f \circ (r \circ p) \arrow[d, "f \circ \rho"'] & \mathrm{Id}_B \circ q \arrow[u, "\cong"'] \\
f \circ (e \circ q) \arrow[r, "\cong"'] & (f \circ e) \circ q \arrow[u, "\mathsf{w} \circ q"']
\end{tikzcd}$$

A short diagram chase using the triangle law relating v and w shows this is a fill-in.

Next we claim that (p, Γ, Δ) is universal. To this end, let (v, Σ_1, Σ_2) be any other fill-in, so that the following diagram commutes:

$$\begin{tikzcd}
(r \circ \mathrm{Id}_A) \circ v \arrow[r, "\cong"] \arrow[d] & r \circ (\mathrm{Id}_A \circ v) \arrow[r, "r \circ \Sigma_1"] & r \circ p \arrow[d, "\rho"] \\
(e \circ (f \circ r)) \circ v \arrow[r, "\cong"'] & e \circ ((f \circ r) \circ v) \arrow[r, "e \circ \Sigma_2"'] & e \circ q
\end{tikzcd} \tag{8.17}$$

The unlabelled arrow is the composite (8.16) given in the claim.

We define $\Sigma^{\dagger} := v \overset{\cong}{\Rightarrow} \mathrm{Id}_A \circ v \overset{\Sigma_1}{\Longrightarrow} p$, and claim that both the following equations hold:

$$\begin{tikzcd}
\mathrm{Id}_A \circ v \arrow[rr, "\mathrm{Id}_A \circ \Sigma^{\dagger}"] \arrow[dr, "\Sigma_1"'] & & \mathrm{Id}_A \circ p \arrow[dl, "\Gamma"] & (f \circ r) \circ v \arrow[rr, "(f \circ r) \circ \Sigma^{\dagger}"] \arrow[dr, "\Sigma_2"'] & & (f \circ r) \circ p \arrow[dl, "\Delta"] \\
& p & & & q &
\end{tikzcd} \tag{8.18}$$

The right-hand diagram is an relatively easy check. The left-hand diagram follows by naturality, the triangle law relating v and w, and the assumption (8.17).

It remains to check the uniqueness condition for $\Sigma^{\dagger}$. For any other $\Theta : v \Rightarrow p$ satisfying the two diagrams of (8.18), one sees that

$$\begin{tikzcd}
v \arrow[r, "\Theta"] \arrow[d, "\cong"'] \arrow[dr, phantom, "\overset{\text{nat.}}{=}"] & p \arrow[d, dashed, "\cong"] \arrow[dd, equal, bend left=60] \\
\mathrm{Id}_A \circ v \arrow[r, "\mathrm{Id}_A \circ \Theta"] \arrow[dr, "\Sigma_1"'] & \mathrm{Id}_A \circ p \arrow[d, dashed, "\cong"] \\
& p
\end{tikzcd}$$

where the bottom triangle commutes by the right-hand diagram of (8.18), and the left-hand leg is exactly the definition of $\Sigma^\dagger$. Hence $\Theta = \Sigma^\dagger$ as required. Finally we observe that $\mathrm{id}^\dagger$ is certainly invertible. □

The requirement for an adjoint equivalence in the preceding lemma is, by the usual argument, no stronger than requiring just an equivalence (*e.g.* [Lei04, Proposition 1.5.7]). Importantly, the adjoint equivalence one constructs from an equivalence has the same 1-cells.

In the light of the lemma, if we can show that the equivalence $u_{B,X}$ defined in (8.15) has a pseudo-inverse given by the composite $\big[(l,\bar{l})_{(-,B)}, \langle\mathfrak{J}\rangle X\big] \circ \mathrm{m}_{\mathfrak{J}B,X}$, then the following is a pullback diagram:

$$\begin{tikzcd}
{[\mathrm{Y}B, K]} \arrow[rr, "\mathrm{Id}_{[\mathrm{Y}B,K]}"] \arrow[d, "{[\mathrm{Y}B,(\mathsf{k},\bar{\mathsf{k}})]}"'] & \cong & {[\mathrm{Y}B, K]} \arrow[dd, "\Lambda((\mathsf{k},\bar{\mathsf{k}})\circ(e,\bar{e}))"] \\
{[\mathrm{Y}B, \langle\mathfrak{J}\rangle X]} \arrow[d, "u_{B,X}"'] \arrow[rrd, "\mathrm{Id}_{\mathrm{Y}B \Rightarrow \langle\mathfrak{J}\rangle X}"] & \cong & \\
\langle\mathfrak{J}\rangle(\mathfrak{J}B \Rightarrow X) \arrow[r, "\mathrm{m}_{\mathfrak{J}B,X}"'] & {[\langle\mathfrak{J}\rangle(\mathfrak{J}B), \langle\mathfrak{J}\rangle X]} \arrow[r, "\Lambda((e,\bar{e})\circ([\langle\mathfrak{J}\rangle(\mathfrak{J}B),\langle\mathfrak{J}\rangle X]\times(l,\bar{l})))"'] & {[\mathrm{Y}B, \langle\mathfrak{J}\rangle X]}
\end{tikzcd}$$

It will then follow that for any $\underline{K} := \big(K, (\mathsf{k}, \bar{\mathsf{k}}), X\big)$ the composite (8.14)—the left-hand leg of the above diagram—is an explicit description of the exponential $(\underline{\mathrm{Y}}X \Rightarrow \underline{K})$. The difficulty, therefore, is not in showing that $u_{B,X}$ is an equivalence, but in checking whether it has a pseudo-inverse of the form we require. We turn to this next. (The cartesian closed structures we employ are summarised in Appendix B).

The equivalence $[\mathrm{Y}B, \langle\mathfrak{J}\rangle X] \simeq \langle\mathfrak{J}\rangle(\mathfrak{J}B \Rightarrow X)$: calculating the 1-cells

In this section we shall calculate the action of the maps $u_{B,X}$ and $\big[(l,\bar{l})_{(-,B)}, \langle\mathfrak{J}\rangle X\big] \circ \mathrm{m}_{\mathfrak{J}B,X}$; in the next section we shall show these form an equivalence. To shorten notation, let us introduce the following abbreviation:

$$[\underline{w}]_{B,X} := \big[(l,\bar{l})_{(-,B)}, \langle\mathfrak{J}\rangle X\big] \circ \mathrm{m}_{\mathfrak{J}B,X}$$

Our first task is to unfold each of the equivalences in the definition of $u_{B,X}$ to determine the action of the whole composite.

Calculating the composite $u_{B,X}$. If $[X, Y]$ and $X \Rightarrow Y$ are both the exponential of X and Y in a bicategory $\mathcal{B}$, with associated currying operation and evaluation maps $\lambda, \mathrm{eval}_{X,Y}$ and $\widehat{\lambda}, \widehat{\mathrm{eval}}_{X,Y}$, respectively, then $\widehat{\lambda}\Big(([X,Y]) \times X \xrightarrow{\mathrm{eval}_{X,Y}} Y\Big) : [X,Y] \to (X \Rightarrow Y)$ is canonically an equivalence.

Now let $(\mathcal{B}, \Pi_n(-))$ be a 2-category with pseudo-products, $B \in \mathcal{B}$, and $P : \mathcal{B}^{\mathrm{op}} \to \mathbf{Cat}$ be any pseudofunctor. We calculate the equivalence $[\mathrm{Y}B, P] = \mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})\,(\mathrm{Y}(-) \times \mathrm{Y}B, P) \xrightarrow{\simeq}$

$P(- \times B)$ arising from Theorem 6.2.7. The evaluation 1-cell $\mathrm{eval}_{\mathrm{Y}B,P} : [\mathrm{Y}B, P] \times \mathrm{Y}B \to P$ is the pseudonatural transformation $(e, \overline{e})$ with components

$$\begin{aligned} \mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})(\mathrm{Y}C \times \mathrm{Y}B, P) \times \mathcal{B}(C, B) &\xrightarrow{e_C} PC \\ \big((\mathsf{k}, \overline{\mathsf{k}}), h\big) &\mapsto \mathsf{k}_C(\mathrm{Id}_C, h) \end{aligned}$$

On the other hand, the currying operation

$$\widehat{\Lambda} : \mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})(R \times \mathrm{Y}B, P) \to \mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})\big(R, P(- \times B)\big)$$

witnessing $P(- \times X)$ as an exponential takes a pseudonatural transformation $(\mathsf{j}, \bar{\mathsf{j}})$ to the pseudonatural transformation with components $RC \xrightarrow{R\pi_1} R(C \times B) \xrightarrow{\mathsf{j}_{C \times B}(-,\pi_2)} P(C \times B)$. Using the assumption that $\mathcal{B}$ is a 2-category, the component of the canonical equivalence $[\mathrm{Y}B, P] \xrightarrow{\simeq} P(- \times B)$ at $C \in \mathcal{B}$ is therefore

$$\begin{aligned} \mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})\big(\mathrm{Y}C \times \mathrm{Y}B, P\big) &\to P(C \times B) \\ (\mathsf{k}, \overline{\mathsf{k}}) &\mapsto \mathsf{k}_{C \times B}(\pi_1, \pi_2) \end{aligned} \tag{8.19}$$

It follows that $u_{B,X}(C)$ is the following composite:

$$\begin{array}{ccccccc} [\mathrm{Y}B, \langle\mathfrak{J}\rangle X](C) & \xrightarrow{\simeq} & \mathcal{X}\big(\mathfrak{J}(C \times B), X\big) & \xrightarrow{\simeq} & \mathcal{X}\big(\mathfrak{J}C \times \mathfrak{J}B, X\big) & \xrightarrow{\simeq} & \mathcal{X}\big(\mathfrak{J}C, \mathfrak{J}B \Rightarrow X\big) \\ (\mathsf{k}, \overline{\mathsf{k}}) & \mapsto & \mathsf{k}_{C \times B}(\pi_1, \pi_2) & \mapsto & \mathsf{k}_{C \times B}(\pi_1, \pi_2) \circ \mathrm{q}^{\times}_{C,B} & \mapsto & \lambda\big(\mathsf{k}_{C \times B}(\pi_1, \pi_2) \circ \mathrm{q}^{\times}_{C,B}\big) \end{array} \tag{8.20}$$

Next we turn to calculating $[\underline{w}]_{B,X} := \big[(l, \overline{l})_{(-,B)}, \langle\mathfrak{J}\rangle X\big] \circ \mathrm{m}_{\mathfrak{J}B,X}$.

Calculating $\big[(l, \overline{l})_{(-,B)}, \langle\mathfrak{J}\rangle X\big]$. We begin by calculating the composite

$$[\langle\mathfrak{J}\rangle(\mathfrak{J}B), \langle\mathfrak{J}\rangle(X)] \times \mathrm{Y}B \xrightarrow{[\langle\mathfrak{J}\rangle(\mathfrak{J}B),\langle\mathfrak{J}\rangle(X)] \times (l,\overline{l})_{(-,B)}} [\langle\mathfrak{J}\rangle(\mathfrak{J}B), \langle\mathfrak{J}\rangle(X)] \times \langle\mathfrak{J}\rangle\mathfrak{J}B \xrightarrow{(e,\overline{e})} \langle\mathfrak{J}\rangle(X) \tag{8.21}$$

Applying the definition of $(e, \overline{e})$ again, the component of the composite (8.21) at $C \in \mathcal{B}$ is

$$\begin{aligned} \mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})\big(\mathcal{B}(-, C) \times \mathcal{X}(\mathfrak{J}(-), \mathfrak{J}B), \mathcal{X}(\mathfrak{J}(-), X)\big) \times \mathcal{B}(C, B) &\to \mathcal{X}(\mathfrak{J}C, X) \\ \big((\mathsf{k}, \overline{\mathsf{k}}), h\big) &\mapsto \mathsf{k}(C, \mathrm{Id}_C, \mathfrak{J}h) \end{aligned}$$

Naturality in C is witnessed by the following 2-cell, where $r : C' \to C$ is any 1-cell in $\mathcal{B}$:

$$\begin{tikzcd} \mathsf{k}\big(C', \mathrm{Id}_{C'} \circ r, \mathfrak{J}(h \circ r)\big) \arrow[r] \arrow[d, "\mathsf{k}\left(C', \mathrm{Id}_{C'} \circ r, (\phi^{\mathfrak{J}}_{h,r})^{-1}\right)"'] & \mathsf{k}(C, \mathrm{Id}_C, \mathfrak{J}h) \circ \mathfrak{J}r \\ \mathsf{k}(C', \mathrm{Id}_{C'} \circ r, \mathfrak{J}h \circ \mathfrak{J}r) \arrow[r, equal] & \mathsf{k}(C', r \circ \mathrm{Id}_C, \mathfrak{J}h \circ \mathfrak{J}r) \arrow[u, "\overline{\mathsf{k}}(r, \mathrm{Id}_C, \mathfrak{J}h)"'] \end{tikzcd}$$

Instantiating this with the cartesian closed structure constructed in Section 6.1, one may identify $\big[(l, \overline{l})_{(-,B)}, \langle\mathfrak{J}\rangle X\big] : [\langle\mathfrak{J}\rangle(\mathfrak{J}B), \langle\mathfrak{J}\rangle(X)] \to [\mathrm{Y}B, \langle\mathfrak{J}\rangle(X)]$ as in the following lemma.

Lemma 8.2.19. For any 2-category with pseudo-products $(\mathcal{B}, \Pi_n(-))$, cc-bicategory $(\mathcal{X}, \Pi_n(-), \mathbin{=\!\rhd})$, and fp-pseudofunctor $(\mathfrak{J}, \mathrm{q}^{\times}) : (\mathcal{B}, \Pi_n(-)) \to (\mathcal{X}, \Pi_n(-))$, the pseudonatural transformation $\big[(l, \bar{l})_{(-,B)}, \langle\mathfrak{J}\rangle X\big] : [\langle\mathfrak{J}\rangle(\mathfrak{J}B), \langle\mathfrak{J}\rangle(X)] \Rightarrow [\mathrm{Y}B, \langle\mathfrak{J}\rangle(X)]$ (where $B \in \mathcal{B}$ and $X \in \mathcal{X}$) has functorial components

$$\begin{aligned} [\langle\mathfrak{J}\rangle(\mathfrak{J}B), \langle\mathfrak{J}\rangle(X)](C) &\xrightarrow{[(l,\bar{l})_{(-,B)}, \langle\mathfrak{J}\rangle X](C)} [\mathrm{Y}B, \langle\mathfrak{J}\rangle(X)](C) \\ (\mathsf{k}, \bar{\mathsf{k}}) &\mapsto \lambda A^{\mathcal{B}} \,.\, \lambda h^{A \to C} \,.\, \lambda p^{A \to B} \,.\, \mathsf{k}(A, h, \mathfrak{J}p) \end{aligned}$$

For $s : A' \to A$, the witnessing 2-cell of $\big[(l, \bar{l})_{(-,B)}, \langle\mathfrak{J}\rangle X\big](C)\big((\mathsf{k}, \bar{\mathsf{k}})\big)$ as in the diagram

$$\begin{tikzcd}
\mathcal{B}(A, C) \times \mathcal{B}(A, B) \arrow[r, "\mathcal{B}(s,C) \times \mathcal{B}(s,B)"] \arrow[d, "\mathsf{k}(A,-,\mathfrak{J}(=))"'] \arrow[dr, phantom, "\overset{\cong}{\Leftarrow}"] & \mathcal{B}(A', C) \times \mathcal{B}(A', B) \arrow[d, "\mathsf{k}(A',-,\mathfrak{J}(=))"] \\
\mathcal{X}(\mathfrak{J}A, X) \arrow[r, "\mathcal{X}(\mathfrak{J}s,X)"'] & \mathcal{X}(\mathfrak{J}A', X)
\end{tikzcd}$$

is given by

$$\mathsf{k}\big(A', (-) \circ s, \mathfrak{J}(= \circ\, s)\big) \xRightarrow{\mathsf{k}(A',(-)\circ s,(\phi^{\mathfrak{J}}_{(=),s})^{-1})} \mathsf{k}\big(A', (-) \circ s, \mathfrak{J}(=) \circ \mathfrak{J}s\big) \xRightarrow{\bar{\mathsf{k}}(s,-,\mathfrak{J}(=))} \mathsf{k}(A, -, \mathfrak{J}(=)) \circ \mathfrak{J}s$$

□

Calculating $\mathrm{m}_{\mathfrak{J}B,X}$**.** By Lemma 8.2.13, the pseudonatural transformation $\langle\mathfrak{J}\rangle(\mathrm{eval}_{\mathfrak{J}B,X}) \circ \mathrm{q}^{\times}_{\mathfrak{J}B,X}$ has components defined by $\lambda C^{\mathcal{B}} \,.\, \lambda h^{\mathfrak{J}C \to (\mathfrak{J}B \mathbin{=\!\rhd} X)} \,.\, \lambda g^{\mathfrak{J}C \to \mathfrak{J}B} \,.\, \mathrm{eval}_{\mathfrak{J}B,X} \circ \langle h, g\rangle$ and witnessing 2-cells of the form

$$\begin{tikzcd}
\mathcal{X}\big(\mathfrak{J}C, \mathfrak{J}B \mathbin{=\!\rhd} X\big) \times \mathcal{X}\big(\mathfrak{J}C, \mathfrak{J}B\big) \arrow[r, "\mathcal{X}(\mathfrak{J}f,\mathfrak{J}B \mathbin{=\!\rhd} X) \times \mathcal{X}(\mathfrak{J}f,\mathfrak{J}B)"] \arrow[d, "\mathrm{eval}_{\mathfrak{J}B,X} \circ \langle -,= \rangle"'] \arrow[dr, phantom, "\overset{\cong}{\Leftarrow}"] & \mathcal{X}\big(\mathfrak{J}C', \mathfrak{J}B \mathbin{=\!\rhd} X\big) \times \mathcal{X}\big(\mathfrak{J}C', \mathfrak{J}B\big) \arrow[d, "\mathrm{eval}_{\mathfrak{J}B,X} \circ \langle -,= \rangle"] \\
\mathcal{X}\big(\mathfrak{J}C, X\big) \arrow[r, "\mathcal{X}(\mathfrak{J}f,X)"'] & \mathcal{X}\big(\mathfrak{J}C', X\big)
\end{tikzcd}$$

given by

$$\mathrm{eval}_{\mathfrak{J}B,X} \circ \langle h \circ \mathfrak{J}f, g \circ \mathfrak{J}f\rangle \xRightarrow{\mathrm{eval}_{\mathfrak{J}B,X} \circ \mathsf{post}^{-1}} \mathrm{eval}_{\mathfrak{J}B,X} \circ (\langle h, g\rangle \circ \mathfrak{J}f) \overset{\cong}{\Rightarrow} (\mathrm{eval}_{\mathfrak{J}B,X} \circ \langle h, g\rangle) \circ \mathfrak{J}f$$

for every $f : C' \to C$ in $\mathcal{B}$. Applying the currying operation defined in Section 6.1, one obtains the following characterisation of $\mathrm{m}_{\mathfrak{J}B,X}$.

Lemma 8.2.20. For any 2-category with pseudo-products $(\mathcal{B}, \Pi_n(-))$, cc-bicategory $(\mathcal{X}, \Pi_n(-), \Rightarrow)$, and fp-pseudofunctor $(\mathfrak{J}, \mathrm{q}^{\times}) : (\mathcal{B}, \Pi_n(-)) \to (\mathcal{X}, \Pi_n(-))$, the pseudonatural transformation $\mathrm{m}_{\mathfrak{J}B,X}$ has components $\mathrm{m}_{\mathfrak{J}B,X}(C)$ given by the functors

$$\begin{aligned}
&\mathcal{X}(\mathfrak{J}C, \mathfrak{J}B \Rightarrow X) \to \mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})\big(\mathrm{Y}C \times \langle\mathfrak{J}\rangle(\mathfrak{J}B), \langle\mathfrak{J}\rangle X\big) \\
&\qquad\qquad f \mapsto \lambda A^{\mathcal{B}} \,.\, \lambda(h^{A\to C}, g^{\mathfrak{J}A\to\mathfrak{J}B}) \,.\, \big(\mathfrak{J}A \xrightarrow{\langle f\circ\mathfrak{J}h, g\rangle} (\mathfrak{J}B \Rightarrow X) \times \mathfrak{J}B \xrightarrow{\mathrm{eval}_{\mathfrak{J}B,X}} X\big)
\end{aligned}$$

Moreover, for every $r : A' \to A$ the pseudonatural transformation $\mathrm{m}_{\mathfrak{J}B,X}(C)(f)$ has witnessing 2-cell

$$\begin{tikzcd}
\mathcal{B}(A, C) \times \mathcal{X}(\mathfrak{J}A, \mathfrak{J}B) \arrow[r, "\mathcal{B}(r,C)\times\mathcal{X}(\mathfrak{J}r,\mathfrak{J}B)"] \arrow[d, "\mathrm{eval}_{\mathfrak{J}B,X}\circ\langle f\circ\mathfrak{J}(-),=\rangle"'] \arrow[dr, phantom, "\overset{\overline{\mathrm{m}_{\mathfrak{J}B,X}(C)(f)}_r}{\Leftarrow}"] & \mathcal{B}(A', C) \times \mathcal{X}(\mathfrak{J}A', \mathfrak{J}B) \arrow[d, "\mathrm{eval}_{\mathfrak{J}B,X}\circ\langle f\circ\mathfrak{J}(-),=\rangle"] \\
\mathcal{X}(\mathfrak{J}A, X) \arrow[r, "\mathcal{B}(\mathfrak{J}r,X)"'] & \mathcal{X}(\mathfrak{J}A', X)
\end{tikzcd}$$

defined by

$$\begin{tikzcd}
\mathrm{eval}_{\mathfrak{J}B,X} \circ \langle f \circ \mathfrak{J}(h \circ r), g \circ \mathfrak{J}r\rangle \arrow[rr, "\overline{\mathrm{m}_{\mathfrak{J}B,X}(C)(f)}_r"] \arrow[d, "\mathrm{eval}_{\mathfrak{J}B,X}\circ\left\langle f\circ(\phi^{\mathfrak{J}}_{h,r})^{-1}, g\circ\mathfrak{J}r\right\rangle"'] & & (\mathrm{eval}_{\mathfrak{J}B,X} \circ \langle f \circ \mathfrak{J}h, g\rangle) \circ \mathfrak{J}r \\
\mathrm{eval}_{\mathfrak{J}B,X} \circ \langle f \circ (\mathfrak{J}h \circ \mathfrak{J}r), g \circ \mathfrak{J}r\rangle \arrow[dr, "\cong"'] & & \mathrm{eval}_{\mathfrak{J}B,X} \circ (\langle f \circ \mathfrak{J}h, g\rangle \circ \mathfrak{J}r) \arrow[u, "\cong"'] \\
& \mathrm{eval}_{\mathfrak{J}B,X} \circ \langle (f \circ \mathfrak{J}h) \circ \mathfrak{J}r, g \circ \mathfrak{J}r\rangle \arrow[ur, "\mathrm{eval}_{\mathfrak{J}B,X}\circ\mathsf{post}^{-1}"'] &
\end{tikzcd}$$

□

Calculating $[\underline{w}]_{B,X}$. Combining Lemma 8.2.19 with Lemma 8.2.20, one obtains the following identification of $[\underline{w}]_{B,X}$.

Lemma 8.2.21. For any 2-category with pseudo-products $(\mathcal{B}, \Pi_n(-))$, cc-bicategory $(\mathcal{X}, \Pi_n(-), \Rightarrow)$, and fp-pseudofunctor $(\mathfrak{J}, \mathrm{q}^{\times}) : (\mathcal{B}, \Pi_n(-)) \to (\mathcal{X}, \Pi_n(-))$, the composite pseudonatural transformation $[\underline{w}]_{B,X} : \langle\mathfrak{J}\rangle(\mathfrak{J}B \Rightarrow X) \to [\mathrm{Y}B, \langle\mathfrak{J}\rangle X]$ has components

$$\begin{aligned}
&\mathcal{X}(\mathfrak{J}C, \mathfrak{J}B \Rightarrow X) \xrightarrow{[\underline{w}]_{B,X}(C)} \mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})\big(\mathrm{Y}C \times \mathrm{Y}B, \mathcal{X}(\mathfrak{J}(-), X)\big) \\
&\qquad\qquad f \mapsto \lambda A^{\mathcal{B}} \,.\, \lambda h^{A\to C} \,.\, \lambda p^{A\to B} \,.\, \big(\mathfrak{J}A \xrightarrow{\langle f\circ\mathfrak{J}h, \mathfrak{J}p\rangle} (\mathfrak{J}B \Rightarrow X) \times \mathfrak{J}B \xrightarrow{\mathrm{eval}_{\mathfrak{J}B,X}} X\big)
\end{aligned}$$

The witnessing 2-cells for the pseudonatural transformation $[\underline{w}]_{B,X}(C)(f)$ are defined by the following commutative diagram, where $r : A' \to A$ is any 1-cell:

$$\begin{tikzcd}
\mathrm{eval}_{\mathfrak{J}B,X} \circ \langle f \circ \mathfrak{J}(h \circ r), \mathfrak{J}(p \circ r) \rangle \arrow[rr, "\overline{[\underline{w}]_{B,X}(C)(f)}_{r}"] \arrow[d, "\mathrm{eval}_{\mathfrak{J}B,X} \circ \langle f \circ (\phi^{\mathfrak{J}}_{h,r})^{-1}{,} (\phi^{\mathfrak{J}}_{p,r})^{-1} \rangle"'] & & \mathrm{eval}_{\mathfrak{J}B,X} \circ \langle f \circ \mathfrak{J}h, \mathfrak{J}p \rangle \circ \mathfrak{J}r \\
\mathrm{eval}_{\mathfrak{J}B,X} \circ \langle f \circ (\mathfrak{J}h \circ \mathfrak{J}r), \mathfrak{J}p \circ \mathfrak{J}r \rangle \arrow[dr, "\cong"'] & & \mathrm{eval}_{\mathfrak{J}B,X} \circ (\langle f \circ \mathfrak{J}h, \mathfrak{J}p \rangle \circ \mathfrak{J}r) \arrow[u, "\cong"'] \\
& \mathrm{eval}_{\mathfrak{J}B,X} \circ \langle (f \circ \mathfrak{J}h) \circ \mathfrak{J}r, \mathfrak{J}p \circ \mathfrak{J}r \rangle \arrow[ur, "\mathrm{eval}_{\mathfrak{J}B,X} \circ \mathsf{post}^{-1}"'] &
\end{tikzcd} \tag{8.22}$$

□

The equivalence $[\mathrm{Y}B, \langle\mathfrak{J}\rangle X] \simeq \langle\mathfrak{J}\rangle(\mathfrak{J}B \Rightarrow X)$

We are finally in a position to prove that $u_X : [\mathrm{Y}B, \langle\mathfrak{J}\rangle X] \leftrightarrows \langle\mathfrak{J}\rangle(\mathfrak{J}B \Rightarrow X) : [\underline{w}]_{B,X}$ defines an equivalence of pseudofunctors in $\mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$. By Lemma 2.1.16 it suffices to construct an equivalence of categories $u_{B,X}(C) : [\mathrm{Y}B, \langle\mathfrak{J}\rangle X](C) \leftrightarrows \langle\mathfrak{J}\rangle(\mathfrak{J}B \Rightarrow X)(C) : [\underline{w}]_{B,X}(C)$ for each $C \in \mathcal{B}$. We deal with this in the following lemma.

Lemma 8.2.22. For any 2-category with pseudo-products $(\mathcal{B}, \Pi_n(-))$, cc-bicategory $(\mathcal{X}, \Pi_n(-), \Rightarrow)$, and fp-pseudofunctor $(\mathfrak{J}, \mathrm{q}^{\times}) : (\mathcal{B}, \Pi_n(-)) \to (\mathcal{X}, \Pi_n(-))$, the following composites are naturally isomorphic to the identity functor for every $B, C \in \mathcal{B}$ and $X \in \mathcal{X}$:

1.
$$\mathcal{X}(\mathfrak{J}C, \mathfrak{J}B \Rightarrow X) \xrightarrow{[\underline{w}]_{B,X}(C)} \mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})(\mathrm{Y}C \times \mathrm{Y}B, \langle\mathfrak{J}\rangle X) \xrightarrow{u_{B,X}(C)} \mathcal{X}(C, \mathfrak{J}B \Rightarrow X)$$

2.
$$\begin{tikzcd}
\mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})(\mathrm{Y}C \times \mathrm{Y}B, \langle\mathfrak{J}\rangle X) \arrow[rr] \arrow[dr, "u_{B,X}(C)"'] & & \mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})(\mathrm{Y}C \times \mathrm{Y}B, \langle\mathfrak{J}\rangle X) \\
& \mathcal{X}(\mathfrak{J}C, \mathfrak{J}B \Rightarrow X) \arrow[ur, "{[\underline{w}]_{B,X}(C)}"'] &
\end{tikzcd}$$

Hence, $[\underline{w}]_{B,X}$ is pseudo-inverse to $u_{B,X} : [\mathrm{Y}B, \langle\mathfrak{J}\rangle X] \to \langle\mathfrak{J}\rangle(\mathfrak{J}B \Rightarrow X)$ in $\mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$.

Proof. For (1), we begin by calculating

$$\begin{aligned}
\big(u_{B,X}(C) \circ [\underline{w}]_{B,X}(C)\big)(f) &= u_{B,X}(C)\big(\lambda A^{\mathcal{B}} . \lambda h^{A \to C} . \lambda p^{A \to B} . \mathrm{eval}_{\mathfrak{J}B,X} \circ \langle f \circ \mathfrak{J}h, \mathfrak{J}p \rangle\big) \\
&= \lambda\big(\mathfrak{J}C \times \mathfrak{J}B \xrightarrow{\mathrm{q}^{\times}_{C,B}} \mathfrak{J}(C \times B) \xrightarrow{\mathrm{eval}_{\mathfrak{J}B,X} \circ \langle f \circ \mathfrak{J}\pi_1, \mathfrak{J}\pi_2 \rangle} X\big)
\end{aligned}$$

for $f : \mathfrak{J}C \to (\mathfrak{J}B \Rightarrow X)$. For each such f, one obtains an invertible 2-cell $\big(u_{B,X} \circ [\underline{w}]_{B,X}(C)\big)(f) \overset{\cong}{\Rightarrow} f$ as the composite

$$\begin{tikzcd}
\lambda\big((\mathrm{eval}_{\mathfrak{J}B,X} \circ \langle f \circ \mathfrak{J}\pi_1, \mathfrak{J}\pi_2\rangle) \circ \mathrm{q}^{\times}_{C,B}\big) \arrow[rr] \arrow[d, "\lambda(\mathrm{eval}_{\mathfrak{J}B,X} \circ \mathsf{fuse}^{-1} \circ \mathrm{q}^{\times}_{C,B})"'] & & f \\
\lambda\big((\mathrm{eval}_{\mathfrak{J}B,X} \circ ((f \times \mathfrak{J}B) \circ \langle \mathfrak{J}\pi_1, \mathfrak{J}\pi_2\rangle)) \circ \mathrm{q}^{\times}_{C,B}\big) \arrow[d, "\cong"'] & & \lambda\big(\mathrm{eval}_{\mathfrak{J}B,X} \circ (f \times \mathfrak{J}B)\big) \arrow[u, "\eta_f^{-1}"'] \\
\lambda\big((\mathrm{eval}_{\mathfrak{J}B,X} \circ (f \times \mathfrak{J}B)) \circ (\langle \mathfrak{J}\pi_1, \mathfrak{J}\pi_2\rangle) \circ \mathrm{q}^{\times}_{C,B}\big) \arrow[rr, "\lambda(\mathrm{eval}_{\mathfrak{J}B,X} \circ (f \times \mathfrak{J}B) \circ (\mathrm{u}^{\times}_{C,B})^{-1})"'] & & \lambda\big((\mathrm{eval}_{\mathfrak{J}B,X} \circ (f \times \mathfrak{J}B)) \circ \mathrm{Id}_{\mathfrak{J}B \times \mathfrak{J}C}\big) \arrow[u, "\cong"']
\end{tikzcd}$$

where the bottom isomorphism arises from the equivalence $\langle \mathfrak{J}\pi_1, \mathfrak{J}\pi_2\rangle : \mathfrak{J}(B \times C) \leftrightarrows \mathfrak{J}B \times \mathfrak{J}C : \mathrm{q}^{\times}_{C,B}$ witnessing $(\mathfrak{J}, \mathrm{q}^{\times})$ as an fp-pseudofunctor. This composite is clearly natural in f, so one obtains the required natural isomorphism.

For (2) one must work a little harder. We are required to construct an invertible modification $\Xi^{(\mathsf{k},\overline{\mathsf{k}})} : \big([\underline{w}]_{B,X}(C) \circ u_{B,X}(C)\big)\big((\mathsf{k},\overline{\mathsf{k}})\big) \xrightarrow{\cong} (\mathsf{k},\overline{\mathsf{k}})$ for every pseudonatural transformation $(\mathsf{k},\overline{\mathsf{k}}) : \mathrm{Y}C \times \mathrm{Y}B \Rightarrow \mathscr{X}\big(\mathfrak{J}(-), X\big)$, and this family which must be natural in the sense that, for any modification $\Psi : (\mathsf{k},\overline{\mathsf{k}}) \to (\mathsf{j},\overline{\mathsf{j}})$, the following diagram commutes:

$$\begin{tikzcd}
\big([\underline{w}]_{B,X}(C) \circ u_{B,X}(C)\big)\big((\mathsf{k},\overline{\mathsf{k}})\big) \arrow[r, "([\underline{w}]_{B,X}(C) \circ u_{B,X}(C))(\Psi)"] \arrow[d, "\cong"'] & \big([\underline{w}]_{B,X}(C) \circ u_{B,X}(C)\big)\big((\mathsf{j},\overline{\mathsf{j}})\big) \arrow[d, "\cong"] \\
(\mathsf{k},\overline{\mathsf{k}}) \arrow[r, "\Psi"'] & (\mathsf{j},\overline{\mathsf{j}})
\end{tikzcd} \tag{8.23}$$

To this end, let us first unwind the data we are given. Applying the work of the preceding section, one sees that for $(\mathsf{k},\overline{\mathsf{k}}) : \mathrm{Y}C \times \mathrm{Y}B \to \mathscr{X}\big(\mathfrak{J}(-), X\big)$ one has

$$\begin{aligned}
&\big([\underline{w}]_{B,X}(C) \circ u_{B,X}(C)\big)\big((\mathsf{k},\overline{\mathsf{k}})\big) \\
&\qquad = [\underline{w}]_{B,X}(C)\Big(\lambda\big(\mathsf{k}_{C \times B}(\pi_1, \pi_2) \circ \mathrm{q}^{\times}_{C,B}\big)\Big) \\
&\qquad = \lambda A^{\mathcal{B}} \,.\, \lambda h^{A \to C} \,.\, \lambda p^{A \to B} \,.\, \mathrm{eval}_{\mathfrak{J}B,X} \circ \Big\langle \lambda\big(\mathsf{k}_{C \times B}(\pi_1, \pi_2) \circ \mathrm{q}^{\times}_{C,B}\big) \circ \mathfrak{J}h, \mathfrak{J}p \Big\rangle
\end{aligned}$$

Moreover, writing $L := \mathsf{k}_{C \times B}(\pi_1, \pi_2) \circ \mathrm{q}^{\times}_{C,B}$, the 2-cell required for the diagram below (in which $r : A' \to A$) is the composite defined in (8.22) with $f := \lambda L$:

$$\begin{tikzcd}
\mathcal{B}(A, C) \times \mathcal{B}(A, B) \arrow[r, "\mathcal{B}(r,C) \times \mathcal{B}(r,B)"] \arrow[d, "\mathrm{eval}_{\mathfrak{J}B,X} \circ \langle \lambda L \circ \mathfrak{J}(-), \mathfrak{J}(=)\rangle"'] \arrow[dr, phantom, "\overset{\overline{([\underline{w}]_{B,X}(C) \circ u_{B,X}(C))((\mathsf{k},\overline{\mathsf{k}}))_r}}{\Leftarrow}"] & \mathcal{B}(A', C) \times \mathcal{B}(A', B) \arrow[d, "\mathrm{eval}_{\mathfrak{J}B,X} \circ \langle \lambda L \circ \mathfrak{J}(-), \mathfrak{J}(=)\rangle"] \\
\mathscr{X}(\mathfrak{J}A, X) \arrow[r, "\mathscr{X}(\mathfrak{J}r, X)"'] & \mathscr{X}(\mathfrak{J}A', X)
\end{tikzcd}$$

We now turn to defining the modification $\Xi^{(\mathsf{k},\overline{\mathsf{k}})}$. For $A \in \mathcal{B}$ and $(h, p) \in \mathcal{B}(A, C) \times \mathcal{B}(A, B)$ there exists an evident choice of isomorphism $\Xi^{(\mathsf{k},\overline{\mathsf{k}})}(A, h, p) : \big([\underline{w}]_{B,X}(C) \circ u_{B,X}(C)\big)\big((\mathsf{k},\overline{\mathsf{k}})\big)(A, h, p) \Rightarrow$

$\mathsf{k}(A, h, p)$, namely

$$\begin{tikzcd}
\mathrm{eval}_{\mathfrak{J}B,X} \circ \langle \lambda L \circ \mathfrak{J}h, \mathfrak{J}p \rangle \arrow[r, "\Xi^{(\mathsf{k},\bar{\mathsf{k}})}_A(h,p)"] \arrow[d, "\cong"'] & \mathsf{k}_A(h,p) \\
\mathrm{eval}_{\mathfrak{J}B,X} \circ \langle \lambda L \circ \mathfrak{J}h, \mathrm{Id}_{\mathfrak{J}B} \circ \mathfrak{J}p \rangle \arrow[d, "\mathrm{eval}_{\mathfrak{J}B,X} \circ \mathsf{fuse}^{-1}"'] & \mathsf{k}_A(\pi_1\langle p,q\rangle, \pi_2\langle p,q\rangle) \arrow[u, "\mathsf{k}_A(\varpi^{(1)}_{p,q}, \varpi^{(2)}_{p,q})"'] \\
\mathrm{eval}_{\mathfrak{J}B,X} \circ ((\lambda L \times \mathfrak{J}\mathrm{Id}_B) \circ \langle \mathfrak{J}h, \mathfrak{J}p \rangle) \arrow[d, "\mathrm{eval}_{\mathfrak{J}B,X} \circ (\lambda L \times (\psi^{\mathfrak{J}}_B)^{-1}) \circ \langle \mathfrak{J}h, \mathfrak{J}p \rangle"'] & \mathsf{k}_{C \times B}(\pi_1, \pi_2) \circ \mathfrak{J}\langle h, p \rangle \arrow[u, "\bar{\mathsf{k}}^{-1}_{\langle p,h\rangle}(\pi_1,\pi_2)"'] \\
\mathrm{eval}_{\mathfrak{J}B,X} \circ ((\lambda L \times \mathfrak{J}B) \circ \langle \mathfrak{J}h, \mathfrak{J}p \rangle) \arrow[d, "\cong"'] & (\mathsf{k}_{C \times B}(\pi_1, \pi_2) \circ \mathrm{Id}_{\mathfrak{J}C \times \mathfrak{J}B}) \circ \mathfrak{J}\langle h, p \rangle \arrow[u, "\cong"'] \\
(\mathrm{eval}_{\mathfrak{J}B,X} \circ (\lambda L \times \mathfrak{J}B)) \circ \langle \mathfrak{J}h, \mathfrak{J}p \rangle \arrow[d, "\varepsilon_L \circ \langle \mathfrak{J}h, \mathfrak{J}p \rangle"'] & \Big(\mathsf{k}_{C \times B}(\pi_1, \pi_2) \circ \Big(\mathrm{q}^{\times}_{C,B} \circ \langle \mathfrak{J}\pi_1, \mathfrak{J}\pi_2 \rangle\Big)\Big) \circ \mathfrak{J}\langle h, p \rangle \arrow[u, "\mathsf{k}_{C \times B}(\pi_1,\pi_2) \circ \mathrm{c}^{\times}_{C,B} \circ \mathfrak{J}\langle h,p \rangle"'] \\
\Big(\mathsf{k}_{C \times B}(\pi_1, \pi_2) \circ \mathrm{q}^{\times}_{C,B}\Big) \circ \langle \mathfrak{J}h, \mathfrak{J}p \rangle \arrow[r, "\mathsf{k}_{C \times B}(\pi_1,\pi_2) \circ \mathrm{q}^{\times}_{C,B} \circ \mathsf{unpack}^{-1}"'] & \Big(\mathsf{k}_{C \times B}(\pi_1, \pi_2) \circ \mathrm{q}^{\times}_{C,B}\Big) \circ (\langle \mathfrak{J}\pi_1, \mathfrak{J}\pi_2 \rangle \circ \mathfrak{J}\langle h, p \rangle) \arrow[u, "\cong"']
\end{tikzcd}$$

It is clear from the definition that $\Xi^{(\mathsf{k},\bar{\mathsf{k}})}_A := \Xi^{(\mathsf{k},\bar{\mathsf{k}})}(A, -, =)$ is natural in its two arguments and so a 2-cell $\big([\underline{w}]_{B,X}(C) \circ u_{B,X}(C)\big)\big((\mathsf{k}, \bar{\mathsf{k}})\big)(A, -, =) \Rightarrow \mathsf{k}(A, -, =)$ in **Cat**. Moreover, the naturality condition (8.23) holds by naturality of each of the components defining $\Xi^{(\mathsf{k},\bar{\mathsf{k}})}$ and the modification axiom on $\Psi : (\mathsf{k}, \bar{\mathsf{k}}) \to (\mathsf{j}, \bar{\mathsf{j}})$, which requires that the following diagram commutes for every $r : A' \to A$ in $\mathcal{B}$ and $(p, h) \in \mathcal{B}(A, C) \times \mathcal{B}(A, B)$:

$$\begin{tikzcd}
\mathsf{k}_{A'}(pr, hr) \arrow[r, "\bar{\mathsf{k}}_r(p,h)"] \arrow[d, "\Psi'_A(pr,hr)"'] & \mathsf{k}_A(p, h) \circ \mathfrak{J}r \arrow[d, "\Psi_A(p,h) \circ \mathfrak{J}r"] \\
\mathsf{j}_{A'}(pr, hr) \arrow[r, "\bar{\mathsf{j}}_r(p,h)"'] & \mathsf{j}_A(p, h) \circ \mathfrak{J}r
\end{tikzcd}$$

It therefore remains to show that the family of 2-cells $(\Xi^{(\mathsf{k},\bar{\mathsf{k}})}_A)_{A \in \mathcal{B}}$ satisfies the following instance of the modification axiom for every $r : A' \to A$ in $\mathcal{B}$:

$$\begin{tikzcd}
\big([\underline{w}]_{B,X}(C) \circ u_{B,X}(C)\big)\big((\mathsf{k}, \bar{\mathsf{k}})\big)(A, pr, hr) \arrow[r, "\Xi^{(\mathsf{k},\bar{\mathsf{k}})}(A,pr,hr)"] \arrow[d, "\overline{([\underline{w}]_{B,X}(C) \circ u_{B,X}(C))((\mathsf{k},\bar{\mathsf{k}}))}_r"'] & \mathsf{k}(A, pr, hr) \arrow[d, "\bar{\mathsf{k}}_r(p,h)"] \\
\big([\underline{w}]_{B,X}(C) \circ u_{B,X}(C)\big)\big((\mathsf{k}, \bar{\mathsf{k}})\big)(A, p, h) \circ \mathfrak{J}r \arrow[r, "\Xi^{(\mathsf{k},\bar{\mathsf{k}})}(A,p,h) \circ \mathfrak{J}r"'] & \mathsf{k}(A, p, h) \circ \mathfrak{J}r
\end{tikzcd}$$

Unfolding the definitions around the anticlockwise composite and applying the lemma relating fuse and post (Lemma 4.1.7), the problem reduces to the following two lemmas:

$$\begin{tikzcd}
 & \mathsf{k}_A\left(\pi_1\langle p,h\rangle, \pi_2\langle p,h\rangle\right) \circ \mathfrak{J}r \arrow[rd, "\mathsf{k}_A(\varpi^{(1)}_{p,h}{,}\varpi^{(1)}_{p,h})\circ\mathfrak{J}r"] & \\
(\mathsf{k}_{B\times C}(\pi_1,\pi_2) \circ \mathfrak{J}\langle p,h\rangle) \circ \mathfrak{J}r \arrow[ru, "\overline{\mathsf{k}}^{-1}_{\langle p,h\rangle}(\pi_1{,}\pi_2)\circ\mathfrak{J}r"] \arrow[d, "\cong"'] & & \mathsf{k}_A(p,h) \circ \mathfrak{J}r \\
\mathsf{k}_{B\times C}(\pi_1,\pi_2) \circ (\mathfrak{J}\langle p,h\rangle \circ \mathfrak{J}r) \arrow[d, "\mathsf{k}_{B\times C}(\pi_1{,}\pi_2)\circ\phi^{\mathfrak{J}}_{\langle p,h\rangle,r}"'] & & \\
\mathsf{k}_{B\times C}(\pi_1,\pi_2) \circ \mathfrak{J}(\langle p,h\rangle \circ r) \arrow[d, "\mathsf{k}_{B\times C}(\pi_1{,}\pi_2)\circ\mathfrak{J}\mathsf{post}"'] & & \mathsf{k}_{A'}(pr,hr) \arrow[uu, "\overline{\mathsf{k}}_r(p{,}h)"'] \\
\mathsf{k}_{B\times C}(\pi_1,\pi_2) \circ \mathfrak{J}\langle pr,hr\rangle \arrow[rr, "\overline{\mathsf{k}}^{-1}_{\langle pr,hr\rangle}(\pi_1{,}\pi_2)"'] & & \mathsf{k}_{A'}\left(\pi_1\langle pr,hr\rangle, \pi_2\langle pr,hr\rangle\right) \arrow[u, "\mathsf{k}_{A'}(\varpi^{(1)}_{pr,hr}{,}\varpi^{(2)}_{pr,hr})"']
\end{tikzcd} \tag{8.24}$$

and

$$\begin{tikzcd}
 & \mathrm{q}^{\times}_{C,B} \circ \left((\langle \mathfrak{J}\pi_1, \mathfrak{J}\pi_2\rangle \circ \mathfrak{J}\langle p,h\rangle) \circ \mathfrak{J}r\right) \arrow[ld, "\mathrm{q}^{\times}_{C,B}\circ\mathsf{unpack}\circ\mathfrak{J}r"'] \arrow[rd, "\cong"] & \\
\mathrm{q}^{\times}_{C,B} \circ \langle \mathfrak{J}p, \mathfrak{J}h\rangle \circ \mathfrak{J}r \arrow[d, "\mathrm{q}^{\times}_{C,B}\circ\mathsf{post}"'] & & \mathfrak{J}\langle p,h\rangle \circ \mathfrak{J}r \arrow[d, "\phi^{\mathfrak{J}}_{\langle p,h\rangle,r}"] \\
\mathrm{q}^{\times}_{C,B} \circ \langle \mathfrak{J}p \circ \mathfrak{J}r, \mathfrak{J}h \circ \mathfrak{J}r\rangle \arrow[d, "\mathrm{q}^{\times}_{C,B}\circ\left\langle\phi^{\mathfrak{J}}_{p,r}{,}\phi^{\mathfrak{J}}_{h,r}\right\rangle"'] & & \mathfrak{J}(\langle p,h\rangle \circ r) \arrow[d, "\mathfrak{J}\mathsf{post}"] \\
\mathrm{q}^{\times}_{C,B} \circ \langle \mathfrak{J}(pr), \mathfrak{J}(hr)\rangle \arrow[d, "\mathrm{q}^{\times}_{C,B}\circ\mathsf{unpack}^{-1}"'] & & \mathfrak{J}\langle pr,hr\rangle \\
\mathrm{q}^{\times}_{C,B} \circ (\langle \mathfrak{J}\pi_1, \mathfrak{J}\pi_2\rangle \circ \mathfrak{J}\langle pr,hr\rangle) \arrow[rr, "\cong"'] & & \left(\mathrm{q}^{\times}_{C,B} \circ \langle \mathfrak{J}\pi_1, \mathfrak{J}\pi_2\rangle\right) \circ \mathfrak{J}\langle pr,hr\rangle \arrow[u, "\mathsf{c}^{\times}_{C,B}\circ h"']
\end{tikzcd} \tag{8.25}$$

Here the top unlabelled isomorphism is the composite

$$\begin{tikzcd}
\mathrm{q}^{\times}_{C,B} \circ \left((\langle \mathfrak{J}\pi_1, \mathfrak{J}\pi_2\rangle \circ \mathfrak{J}\langle p,h\rangle) \circ \mathfrak{J}r\right) \arrow[r] \arrow[d, "\cong"'] & \mathfrak{J}\langle p,h\rangle \circ \mathfrak{J}r \\
\left(\mathrm{q}^{\times}_{C,B} \circ \langle \mathfrak{J}\pi_1, \mathfrak{J}\pi_2\rangle\right) \circ (\mathfrak{J}\langle p,h\rangle \circ \mathfrak{J}r) \arrow[r, "\mathsf{c}^{\times}_{C,B}\circ\mathfrak{J}\langle p{,}h\rangle\circ\mathfrak{J}r"'] & \mathrm{Id}_{\mathfrak{J}(B\times C)} \circ (\mathfrak{J}\langle p,h\rangle \circ \mathfrak{J}r) \arrow[u, "\cong"']
\end{tikzcd}$$

applying the isomorphism $\mathsf{c}^{\times}_{C,B}$ witnessing that $\mathrm{q}^{\times}_{C,B} : \mathfrak{J}C \times \mathfrak{J}B \leftrightarrows \mathfrak{J}(C \times B) : \langle \mathfrak{J}\pi_1, \mathfrak{J}\pi_2\rangle$ forms an equivalence.

For (8.24), one applies the associativity law for $(\mathsf{k}, \overline{\mathsf{k}})$ along with the definition of post as part of a short diagram chase. For (8.25), one unwinds the definition of unpack in each of the two given composites and repeatedly applies naturality. □

This lemma, together with Lemma 8.2.18, completes the proof of Proposition 8.2.17.

8.3 Glueing syntax and semantics

Our aim now is to show how the structure of $\Lambda_{\mathrm{ps}}^{\times,\to}$, together with the identification of neutral and normal terms in Section 8.2, determines data in the bicategory of intensional Kripke relations (*c.f.* (8.4) on page 258). Fix a cc-bicategory $(\mathcal{X}, \Pi_n(-), \Rightarrow)$ and consider an interpretation $\mathfrak{B} \to \mathcal{X}$ of base types in $\mathcal{X}$ with canonical extension $s : \widetilde{\mathfrak{B}} \to \mathcal{X}$. We show that the terms of $\Lambda_{\mathrm{ps}}^{\times,\to}$ determine objects in the glueing bicategory, and that the typing rules determine 1-cells.

From terms to glued objects. On neutral and normal terms, the key observation is that the interpretation of $\Lambda_{\mathrm{ps}}^{\times,\to}$-terms in $\mathcal{X}$ is pseudonatural.

Construction 8.3.1. Let $\mathfrak{B}$ be a set of base types, $(\mathcal{X}, \Pi_n(-), \Rightarrow)$ be a cc-bicategory, and $s : \widetilde{\mathfrak{B}} \to \mathcal{X}$ the canonical extension of a set map $\mathfrak{B} \to \mathcal{X}$. By Proposition 5.3.22 there exists a cc-pseudofunctor $s\llbracket - \rrbracket : \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}}) \to \mathcal{X}$ interpreting $\Lambda_{\mathrm{ps}}^{\times,\to}(\widetilde{\mathfrak{B}})$ in $\mathcal{X}$ (see Construction C.2.2 for the full definition). We define a pseudonatural transformation $(s\llbracket - \rrbracket, \overline{s\llbracket - \rrbracket}) : \mathrm{d}\mathcal{L}(-; A) \Rightarrow \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket A \rrbracket\big) : \mathrm{dCon}_{\widetilde{\mathfrak{B}}} \to \mathbf{Cat}$ for every $A \in \widetilde{\mathfrak{B}}$.

For the component at $\Gamma \in \mathrm{Con}_{\widetilde{\mathfrak{B}}}$ we take the functor

$$\mathrm{d}\mathcal{L}(\Gamma; A) \xrightarrow{s\llbracket - \rrbracket_{\Gamma,A}} \mathcal{X}(s\llbracket \Gamma \rrbracket, s\llbracket A \rrbracket)$$
$$\llparenthesis\, t \,\rrparenthesis \mapsto s\llbracket \Gamma \vdash \llparenthesis\, t \,\rrparenthesis : A \rrbracket$$

Next, for every context renaming $r : \Gamma \to \Delta$ we need to provide a 2-cell—*i.e.* natural isomorphism—as in

$$\begin{tikzcd}
\mathrm{d}\mathcal{L}(\Gamma; A) \arrow[r, "\mathrm{d}\mathcal{L}(r;A)"] \arrow[d, "s\llbracket - \rrbracket"'] \arrow[dr, phantom, "\overset{(\overline{s\llbracket - \rrbracket})_r}{\Leftarrow}"] & \mathrm{d}\mathcal{L}(\Delta; A) \arrow[d, "s\llbracket - \rrbracket"] \\
\mathcal{X}(s\llbracket \Gamma \rrbracket, s\llbracket A \rrbracket) \arrow[r, "\mathcal{X}(s\llbracket r \rrbracket, s\llbracket A \rrbracket)"'] & \mathcal{X}(s\llbracket \Delta \rrbracket, s\llbracket A \rrbracket)
\end{tikzcd}$$

Thus, for every $\llparenthesis\, t \,\rrparenthesis \in L(\Gamma; A)$ we need to provide an isomorphism in $\mathcal{X}$ of type $s\llbracket \Delta \vdash \llparenthesis\, t[r(x_i)/x_i] \,\rrparenthesis : A \rrbracket \to s\llbracket \Gamma \vdash \llparenthesis\, t \,\rrparenthesis : A \rrbracket \circ s\llbracket r \rrbracket$. Calculating, one sees that

$$\begin{aligned} s[\![\llparenthesis\, \Gamma \vdash \llparenthesis\, t \,\rrparenthesis : A \,\rrparenthesis]\!] \circ s[\![r]\!] &= s[\![\llparenthesis\, \Gamma \vdash \llparenthesis\, t \,\rrparenthesis : A \,\rrparenthesis]\!] \circ \big\langle \pi_{r(1)}, \ldots , \pi_{r(n)} \big\rangle \\ &= s[\![\llparenthesis\, \Gamma \vdash \llparenthesis\, t \,\rrparenthesis : A \,\rrparenthesis]\!] \circ \big\langle s[\![(\Delta \vdash x_{r(i)} : A_{r(i)}]\!]) \big\rangle_i \\ &= s[\![\llparenthesis\, \Gamma \vdash \llparenthesis\, t \,\rrparenthesis : A \,\rrparenthesis]\!] \circ s\big[\!\big[(\Delta \vdash x_{r(i)} : A_{r(i)})_{i=1,\ldots,n}\big]\!\big] \\ &= s[\![\Delta \vdash \llparenthesis\, t \,\rrparenthesis\{r\} : A]\!] \end{aligned}$$

Now recall from Construction 5.4.6 that we have already constructed a rewrite typed by the rule

$$\frac{\Gamma \vdash \llparenthesis\, t \,\rrparenthesis : A \qquad r : \Gamma \to \Delta}{\Delta \vdash \mathsf{cont}(t; r) : \llparenthesis\, t \,\rrparenthesis\{x_i \mapsto r(x_i)\} \Rightarrow \llparenthesis\, t[r(x_i)/x_i] \,\rrparenthesis : A}$$

We therefore define $(\overline{s[\![-]\!]})_r$ to be the interpretation of cont:

$$(\overline{s[\![-]\!]})_r(t) := s[\![\Delta \vdash \mathsf{cont}(t; r) : \llparenthesis\, t \,\rrparenthesis\{x_i \mapsto r(x_i)\} \Rightarrow \llparenthesis\, t[r(x_i)/x_i] \,\rrparenthesis : A]\!]$$

To see that this is a pseudonatural transformation, observe first that it is certainly natural: there are no non-trivial 2-cells in $\mathrm{d}L(\Gamma; A)$. For the unit law, we need to show that

$$\begin{tikzcd} s[\![\Gamma \vdash \llparenthesis\, t \,\rrparenthesis : A]\!] \circ \mathrm{Id}_{s[\![\Gamma]\!]} \arrow[r, "\cong"] \arrow[d, "s[\![\Gamma\vdash\llparenthesis\, t \,\rrparenthesis:A]\!]\circ\widehat{\varsigma}_{\mathrm{Id}_{s[\![\Gamma]\!]}}"'] & s[\![\Delta \vdash \llparenthesis\, t[x_i/x_i] \,\rrparenthesis : A]\!] \arrow[d, equal] \\ s[\![\Gamma \vdash \llparenthesis\, t \,\rrparenthesis : A]\!] \circ \langle \pi_1, \ldots , \pi_n \rangle \arrow[r, "s[\![\Gamma\vdash\mathsf{cont}(t;\mathrm{id}_\Gamma):t\{x_i\mapsto x_i\}\Rightarrow\llparenthesis\, t[x_i/x_i] \,\rrparenthesis:A]\!]"'] & s[\![\Delta \vdash \llparenthesis\, t \,\rrparenthesis : A]\!] \end{tikzcd} \tag{8.26}$$

where $\widehat{\varsigma}_{\mathrm{Id}_{s[\![\Gamma]\!]}} := \mathrm{Id}_{s[\![\Gamma]\!]} \xRightarrow{\varsigma_{\mathrm{Id}_{s[\![\Gamma]\!]}}} \big\langle \pi_1 \circ \mathrm{Id}_{s[\![\Gamma]\!]}, \ldots , \pi_n \circ \mathrm{Id}_{s[\![\Gamma]\!]} \big\rangle \overset{\cong}{\Rightarrow} \langle \pi_1, \ldots , \pi_n \rangle$. To see this commutes, note that $s[\![\Gamma \vdash \iota_{\llparenthesis\, t \,\rrparenthesis} : \llparenthesis\, t \,\rrparenthesis \Rightarrow \llparenthesis\, t \,\rrparenthesis\{x_i \mapsto x_i\} : A]\!]$ is, by definition, the composite

$$s[\![\Gamma \vdash \llparenthesis\, t \,\rrparenthesis : A]\!] \overset{\cong}{\Rightarrow} s[\![\Gamma \vdash \llparenthesis\, t \,\rrparenthesis : A]\!] \circ \mathrm{Id}_{s[\![\Gamma]\!]} \xRightarrow{s[\![\Gamma\vdash\llparenthesis\, t \,\rrparenthesis:A]\!]\circ\widehat{\varsigma}_{\mathrm{Id}_{s[\![\Gamma]\!]}}} s[\![\Gamma \vdash \llparenthesis\, t \,\rrparenthesis : A]\!] \circ \langle \pi_1, \ldots , \pi_n \rangle$$

Hence (8.26) commutes by Lemma 5.4.8 and Lemma 5.4.9(1).

For the associativity law we need to show that, for any contexts $\Gamma := (x_i : A_i)_{i=1,\ldots,n}$ and $\Delta := (y_j : A_j)_{j=1,\ldots,m}$, and any context renamings $\Gamma \xrightarrow{r} \Delta \xrightarrow{r'} \Sigma$, the following diagram commutes:

$$\begin{tikzcd} s[\![\Gamma \vdash \llparenthesis\, t \,\rrparenthesis : A]\!] \circ (\langle \pi_r \rangle \circ \langle \pi_{r'} \rangle) \arrow[rr, "s[\![\Gamma\vdash\llparenthesis\, t \,\rrparenthesis:A]\!]\circ\mathsf{post}"] & & s[\![\Gamma \vdash \llparenthesis\, t \,\rrparenthesis : A]\!] \circ \langle \pi_r \circ \langle \pi_{r'} \rangle \rangle \arrow[d, "s[\![\Gamma\vdash\llparenthesis\, t \,\rrparenthesis:A]\!]\circ\langle\varpi^{(r)}\rangle"] \\ (s[\![\Gamma \vdash \llparenthesis\, t \,\rrparenthesis : A]\!] \circ \langle \pi_r \rangle) \circ \langle \pi_{r'} \rangle \arrow[u, "\cong"] \arrow[d, "s[\![\mathsf{cont}(\llparenthesis\, t \,\rrparenthesis;r)]\!]\circ\langle\pi_{r'}\rangle"'] & & s[\![\Gamma \vdash \llparenthesis\, t \,\rrparenthesis : A]\!] \circ \langle \pi_{r'r} \rangle \arrow[d, "s[\![\mathsf{cont}(\llparenthesis\, t \,\rrparenthesis;r'r)]\!]"] \\ s[\![\Delta \vdash \llparenthesis\, t[r(x_i)/x_i] \,\rrparenthesis : A]\!] \circ \langle \pi_{r'} \rangle \arrow[dr, "s[\![\mathsf{cont}(\llparenthesis\, t[r(x_i)/x_i] \,\rrparenthesis;r')]\!]"'] & & s[\![\Sigma \vdash \llparenthesis\, t[r'r(x_i)/x_i] \,\rrparenthesis : A]\!] \arrow[dl, equal] \\ & s[\![\Sigma \vdash \llparenthesis\, t[r(x_i)/x_i][r'(y_j)/y_j] \,\rrparenthesis : A]\!] & \end{tikzcd}$$

We suppress the full typing judgement in the vertical arrows for reasons of space. By Lemma 5.4.8, this diagram is exactly the image of Lemma 5.4.9(3) under $s[\![-]\!]$, and so it commutes. ◂

The preceding construction restricts to neutral and normal terms, giving pseudonatural transformations

$$\mathrm{d}\mathcal{M}(-;A) \xRightarrow{(s[\![-]\!],\overline{s[\![-]\!]})\big|_M} \mathcal{X}\big(s[\![-]\!], s[\![A]\!]\big)$$
$$\mathrm{d}\mathcal{N}(-;A) \xRightarrow{(s[\![-]\!],\overline{s[\![-]\!]})\big|_N} \mathcal{X}\big(s[\![-]\!], s[\![A]\!]\big)$$

One thereby obtains the following glued objects for every type $A \in \widetilde{\mathfrak{B}}$:

$$\begin{aligned} \mu_A &:= \big(\mathrm{d}\mathcal{M}(-;A), (s[\![-]\!], \overline{s[\![-]\!]})\big|_M, s[\![A]\!]\big) \\ \eta_A &:= \big(\mathrm{d}\mathcal{N}(-;A), (s[\![-]\!], \overline{s[\![-]\!]})\big|_N, s[\![A]\!]\big) \end{aligned} \tag{8.27}$$

Finally, for variables, we take

$$\nu_A := \underline{\mathrm{Y}}([A]) = \big(\mathrm{dCon}_{\widetilde{\mathfrak{B}}}(-;A), (l,\bar{l})_{(-,A)}, s[\![A]\!]\big)$$

where $(l,\bar{l})_{(-,A)}$ is the pseudonatural transformation of Corollary 8.2.10.

From typing rules to glued 1-cells. We also lift the natural transformations of (8.10)—viewed as locally discrete pseudonatural transformations—to morphisms in $\mathrm{gl}(\langle \underline{s} \rangle)$.

For the lambda abstraction case we will use the following observation. For types $A, B \in \widetilde{\mathfrak{B}}$ the exponential $[\mathrm{d}\mathcal{V}(-;A), \mathrm{d}\mathcal{N}(-;B)] = [\mathrm{d}(\mathrm{y}[A]), \mathrm{d}\mathcal{N}(-;B)] = [\mathrm{Y}[A], \mathrm{d}\mathcal{N}(-;B)]$ in $\mathrm{Hom}(\mathrm{dCon}_{\widetilde{\mathfrak{B}}}, \mathbf{Cat})$ is, by Theorem 6.2.7, equivalent to $\mathrm{d}\mathcal{N}(- @ [A]; B)$. One thereby obtains a composite

$$[\mathrm{d}\mathcal{V}(-;A), \mathrm{d}\mathcal{N}(-;B)] \xrightarrow{\simeq} \mathrm{d}\mathcal{N}(- @ [A]; B) \xrightarrow{\mathsf{dlam}(-;A,B)} \mathrm{d}\mathcal{N}(-; A \Rightarrow B) \tag{8.28}$$

We put this to work in the next result, which is the bicategorical version of Fiore's [Fio02, Proposition 7 and Proposition 8].

Remark 8.3.2. Examining the equivalence $[\mathrm{d}\mathcal{V}(-;A), \mathrm{d}\mathcal{N}(-;B)] \simeq \mathrm{d}\mathcal{N}(- @ [A]; B)$, one sees that it is in fact an isomorphism. Since $\mathcal{N}(\Gamma @ [A]; B)$ is a set for every context Γ, the composite $\mathcal{N}(\Gamma @ [A]; B) \to [\mathrm{d}\mathcal{V}(-;A), \mathrm{d}\mathcal{N}(-;B)](\Gamma) \to \mathcal{N}(\Gamma @ [A]; B)$ must be equal to the identity. On the other hand, by Lemma 8.2.2(5), the exponential $[\mathrm{d}\mathcal{V}(-;A), \mathrm{d}\mathcal{N}(-;B)]$ may be given by $\mathrm{d}(\mathrm{Fun}(\mathbb{C}, \mathrm{Set})\,(\mathrm{y}(-) \times \mathcal{V}(=;A), \mathcal{N}(=;B)))$. But $\mathbf{Cat}(\mathrm{d}\mathbb{C}, \mathrm{Set})\,(\mathrm{y}\Gamma \times \mathcal{V}(=;A), \mathcal{N}(=;B))$ is also a set for every context Γ. Hence, the composite $[\mathrm{d}\mathcal{V}(-;A), \mathrm{d}\mathcal{N}(-;B)] \to [\mathrm{d}\mathcal{V}(-;A), \mathrm{d}\mathcal{N}(-;B)]$ must also be the identity. ◂

Proposition 8.3.3. For every set of base types $\mathfrak{B}$, cc-bicategory $(\mathcal{X}, \Pi_n(-), \Rightarrow)$, and set map $s : \widetilde{\mathfrak{B}} \to \mathcal{X}$ canonically induced from an interpretation of base types $\mathfrak{B} \to \mathcal{X}$,

1. For every type $A_i \in \widetilde{\mathfrak{B}}$, the triple $\underline{\mathsf{var}} := \big(\mathsf{dvar}(-; A_i), \cong, \mathrm{Id}_{s[\![A_i]\!]}\big)$ is a 1-cell $\nu_{A_i} \to \mu_{A_i}$ in $\mathrm{gl}(\langle \underline{s} \rangle)$, where the 2-cell $\cong$ filling

$$\begin{tikzcd}
\mathrm{d}\mathcal{V}(-; A_i) \arrow[r, "\mathsf{dvar}(-;A_i)"] \arrow[d, "s[\![-]\!]"'] \arrow[dr, phantom, "\overset{\cong}{\Leftarrow}"] & \mathrm{d}\mathcal{M}(-; A_i) \arrow[d, "s[\![-]\!]"] \\
\mathcal{X}\big(s[\![-]\!], s[\![A_i]\!]\big) \arrow[r, "\mathcal{X}\left(s[\![-]\!], \mathrm{Id}_{s[\![A_i]\!]}\right)"'] & \mathcal{X}\big(s[\![-]\!], s[\![A_i]\!]\big)
\end{tikzcd}$$

is the structural isomorphism $s[\![\Gamma \vdash x_i : A_i]\!] \overset{\cong}{\Rightarrow} \mathrm{Id}_{s[\![A_i]\!]} \circ s[\![\Gamma \vdash x_i : A_i]\!]$.

2. For any base type $B \in \mathfrak{B}$, the triple $\underline{\mathsf{inc}} := \big(\mathsf{inc}(-; B), \cong, \mathrm{Id}_{s[\![B]\!]}\big)$, in which $\cong$ is a structural isomorphism, is an isomorphism $\mu_B \xrightarrow{\cong} \eta_B$ in $\mathrm{gl}(\langle \underline{s} \rangle)$.

3. For every sequence of types $A_1, \ldots, A_n \in \widetilde{\mathfrak{B}}$ $(n \in \mathbb{N})$, the triple $\underline{\mathsf{proj}_k} := \big(\mathsf{dproj}_k(-; A_\bullet), \mathrm{id}, \pi_k\big)$ is a 1-cell $\mu_{\prod_n(A_1, \ldots, A_n)} \to \mu_{A_k}$ in $\mathrm{gl}(\langle \underline{s} \rangle)$ for $k = 1, \ldots, n$.

4. For every pair of types $A, B \in \widetilde{\mathfrak{B}}$, the triple $\underline{\mathsf{app}} := \big(\mathsf{dapp}(-; A, B), \mathrm{id}, \mathrm{eval}_{s[\![A]\!], s[\![B]\!]}\big)$ is a 1-cell $\mu_{A \Rightarrow B} \times \eta_A \to \mu_B$ in $\mathrm{gl}(\langle \underline{s} \rangle)$.

5. For every sequence of types $A_1, \ldots, A_n \in \widetilde{\mathfrak{B}}$ $(n \in \mathbb{N})$, the triple $\underline{\mathsf{tuple}} := \big(\mathsf{dtuple}(-; A_\bullet), \cong, \mathrm{Id}_{s[\![\prod_n A_\bullet]\!]}\big)$ is a 1-cell $\prod_{i=1}^n \eta_{A_i} \to \eta_{\prod_n(A_1, \ldots, A_n)}$ in $\mathrm{gl}(\langle \underline{s} \rangle)$, where the isomorphism filling

$$\begin{tikzcd}
\prod_{i=1}^n \mathrm{d}\mathcal{N}(-; A_i) \arrow[r, "\mathsf{dtuple}(-;A_\bullet)"] \arrow[d, "\prod_{i=1}^n s[\![-]\!]"'] & \mathrm{d}\mathcal{N}\big(-; \prod_n(A_1, \ldots, A_n)\big) \arrow[dd, "s[\![-]\!]"] \\
\prod_{i=1}^n \mathcal{X}\big(s[\![-]\!], s[\![A_i]\!]\big) \arrow[d, "\langle -, \ldots, = \rangle"'] \arrow[r, phantom, "\overset{\cong}{\Leftarrow}"] & \\
\mathcal{X}(s[\![-]\!], \prod_{i=1}^n s[\![A_i]\!]) \arrow[r, "\mathcal{X}\left(s[\![-]\!], \mathrm{Id}_{s[\![\prod_n A_\bullet]\!]}\right)"'] & \mathcal{X}(s[\![-]\!], s[\![\prod_n(A_1, \ldots, A_n)]\!])
\end{tikzcd}$$

is the structural isomorphism

$$s[\![\Gamma \vdash \mathsf{tup}((\!| t_1 |\!), \ldots, (\!| t_n |\!)) : \textstyle\prod_n A_\bullet]\!] = \langle s[\![\Gamma \vdash (\!| t_\bullet |\!) : A_\bullet]\!] \rangle \overset{\cong}{\Rightarrow} \mathrm{Id}_{(\prod_i sA_i)} \circ \langle s[\![\Gamma \vdash (\!| t_\bullet |\!) : A_\bullet]\!] \rangle$$

6. For any pair of types $A, B \in \widetilde{\mathfrak{B}}$, write $\mathsf{L}_{A,B}$ for the composite

$$[\mathrm{d}\mathcal{V}(-; A), \mathrm{d}\mathcal{N}(-; B)] \xrightarrow{\cong} \mathrm{d}\mathcal{N}\big(- + [A], B\big) \xrightarrow{\mathsf{dlam}(-;A,B)} \mathrm{d}\mathcal{N}\big(-, A \Rightarrow B\big)$$

of (8.28). Then, where $\cong$ denotes a structural isomorphism, $\underline{\mathsf{lam}} := (\mathsf{L}_{A,B}, \cong, \mathrm{Id}_{s[\![A]\!] \Rightarrow s[\![B]\!]})$ is a 1-cell $(\nu_A \Rightarrow \eta_B) \xrightarrow{\cong} \eta_{A \Rightarrow B}$ in $\mathrm{gl}(\langle \underline{s} \rangle)$.

Proof. (1) is immediate. For (2), observe first that the only way to construct normal terms of base type is via the `inc` rule. Hence the natural transformation inc is a natural isomorphism. Next consider the diagram

$$\begin{tikzcd}
\mathrm{d}\mathcal{M}(-;B) \arrow[r, "\mathsf{inc}(-;B)"] \arrow[d, "s\llbracket - \rrbracket"'] \arrow[dr, phantom, "\overset{\cong}{\Leftarrow}"] & \mathrm{d}\mathcal{N}(-;B) \arrow[d, "s\llbracket - \rrbracket"] \\
\mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket B \rrbracket\big) \arrow[r, "\mathcal{X}(s\llbracket - \rrbracket, \mathrm{Id}_{s\llbracket B \rrbracket})"'] & \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket B \rrbracket\big)
\end{tikzcd}$$

For a context Γ and term $t \in \mathcal{M}(\Gamma; B)$, the clockwise route returns $s\llbracket \Gamma \vdash t : B \rrbracket$ while the anticlockwise route returns $\mathrm{Id}_{s\llbracket B \rrbracket} \circ s\llbracket \Gamma \vdash t : B \rrbracket$. Hence the diagram is filled by a structural isomorphism, and $\big(\mathsf{inc}(-;B), \cong, \mathrm{Id}_{s\llbracket B \rrbracket}\big)$ is a 1-cell in $\mathrm{gl}(\langle \underline{s} \rangle)$. To see that it is an isomorphism in $\mathrm{gl}(\langle \underline{s} \rangle)$, observe that the diagram

$$\begin{tikzcd}
\mathrm{d}\mathcal{N}(-;B) \arrow[r, "\mathsf{inc}(-;B)^{-1}"] \arrow[d, "s\llbracket - \rrbracket"'] \arrow[dr, phantom, "\overset{\cong}{\Leftarrow}"] & \mathrm{d}\mathcal{M}(-;B) \arrow[d, "s\llbracket - \rrbracket"] \\
\mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket B \rrbracket\big) \arrow[r, "\mathcal{X}(s\llbracket - \rrbracket, \mathrm{Id}_{s\llbracket B \rrbracket})"'] & \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket B \rrbracket\big)
\end{tikzcd}$$

is also filled by a structural isomorphism, giving a 1-cell $\big(\mathsf{inc}(-;B)^{-1}, \cong, \mathrm{Id}_{s\llbracket B \rrbracket}\big)$. Then, by the coherence theorem for bicategories, the composite

$$\begin{tikzcd}
\mathrm{d}\mathcal{M}(-;B) \arrow[rr, bend left, "\mathrm{Id}_{\mathrm{d}\mathcal{M}(-;B)}"] \arrow[r, "\mathsf{inc}(-;B)"] \arrow[d, "s\llbracket - \rrbracket"'] \arrow[dr, phantom, "\overset{\cong}{\Leftarrow}"] & \mathrm{d}\mathcal{N}(-;B) \arrow[u, phantom, "="] \arrow[r, "\mathsf{inc}(-;B)^{-1}"] \arrow[d, "s\llbracket - \rrbracket"] \arrow[dr, phantom, "\overset{\cong}{\Leftarrow}"] & \mathrm{d}\mathcal{M}(-;B) \arrow[d, "s\llbracket - \rrbracket"] \\
\mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket B \rrbracket\big) \arrow[r, "\mathcal{X}(s\llbracket - \rrbracket, \mathrm{Id}_{s\llbracket B \rrbracket})"'] \arrow[rr, bend right, "\mathcal{X}(s\llbracket - \rrbracket, \mathrm{Id}_{s\llbracket B \rrbracket})"'] & \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket B \rrbracket\big) \arrow[d, phantom, "\Downarrow \cong"] \arrow[r, "\mathcal{X}(s\llbracket - \rrbracket, \mathrm{Id}_{s\llbracket B \rrbracket})"'] & \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket B \rrbracket\big) \\
 & &
\end{tikzcd}$$

is equal to the identity 1-cell Id_{μ_B} in $\mathrm{gl}(\langle \underline{s} \rangle)$, and similarly for the other composite.

For (3) one needs to check that the following diagram commutes on the nose:

$$\begin{tikzcd}
\mathrm{d}\mathcal{M}\left(-; \prod_n(A_1, \ldots, A_n)\right) \arrow[r, "\mathsf{dproj}_k(-;A_\bullet)"] \arrow[d, "s\llbracket - \rrbracket"'] & \mathrm{d}\mathcal{M}(-;A_k) \arrow[d, "s\llbracket - \rrbracket"] \\
\mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket \prod_n(A_1, \ldots, A_n) \rrbracket\big) \arrow[r, "\mathcal{X}(s\llbracket - \rrbracket, \pi_k)"'] & \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket A_k \rrbracket\big)
\end{tikzcd}$$

For a fixed context Γ and term $\llparenthesis\, t\, \rrparenthesis \in \mathcal{M}(\Gamma; B)$,

$$s\llbracket \mathsf{proj}_k(\Gamma; A_\bullet)(t) \rrbracket = s\llbracket \llparenthesis\, \pi_k(t)\, \rrparenthesis \rrbracket = s\llbracket \pi_k\{\llparenthesis\, t\, \rrparenthesis\} \rrbracket = \pi_k \circ s\llbracket \Gamma \vdash \llparenthesis\, t\, \rrparenthesis : \textstyle\prod_n(A_1, \ldots, A_n) \rrbracket$$

as required.

For (4) one observes that the product $\mu_{A \Rightarrowtriangle B} \times \eta_A$ in $\mathrm{gl}(\langle \underline{s} \rangle)$ is the pseudonatural transformation $\kappa_{A,B}$ defined by the diagram below.

$$\begin{tikzcd}
 & \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket A \Rightarrowtriangle B \rrbracket\big) \times \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket A \rrbracket\big) \arrow[dr, "{\langle -,= \rangle}"] & \\
\mathrm{d}\mathcal{M}(-; A \Rightarrowtriangle B) \times \mathrm{d}\mathcal{N}(-; A) \arrow[ur, "{s\llbracket - \rrbracket \times s\llbracket - \rrbracket}"] \arrow[rr, "\kappa_{A,B}"'] & & \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket A \Rightarrowtriangle B \rrbracket \times s\llbracket A \rrbracket\big)
\end{tikzcd}$$

Hence, the composite $\mathcal{X}\big(s\llbracket - \rrbracket, \mathrm{eval}_{sA,sB}\big) \circ \kappa_{A,B}$ instantiated at a context Γ and a pair of terms $(\llparenthesis\, t \,\rrparenthesis, \llparenthesis\, u \,\rrparenthesis)$ returns

$$\begin{aligned}
\mathrm{eval}_{sA,sB} \circ \langle s\llbracket \Gamma \vdash \llparenthesis\, t \,\rrparenthesis : A \Rightarrowtriangle B \rrbracket, s\llbracket \Gamma \vdash \llparenthesis\, u \,\rrparenthesis : A \rrbracket \rangle &= s\llbracket \mathsf{eval}\{\llparenthesis\, t \,\rrparenthesis, \llparenthesis\, u \,\rrparenthesis\} \rrbracket \\
&= s\llbracket \mathsf{dapp}(\Gamma; A, B)(\llparenthesis\, t \,\rrparenthesis, \llparenthesis\, u \,\rrparenthesis) \rrbracket
\end{aligned}$$

as required. The calculation for (5) is similar.

For (6) some calculations are required. Since $\nu_A = \mathrm{Y}[A]$, the exponential $\nu_A \Rightarrowtriangle \eta_B$ may, by Proposition 8.2.17, be given by the composite

$$[\mathrm{Y}[A], \mathrm{d}\mathcal{N}(-; B)] \xrightarrow{[\mathrm{Y}[A], (s\llbracket - \rrbracket, \overline{s\llbracket - \rrbracket})]} [\mathrm{Y}[A], \mathcal{X}(s\llbracket - \rrbracket, s\llbracket B \rrbracket)] \xrightarrow{u_{[A], s\llbracket B \rrbracket}} \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket A \rrbracket \Rightarrowtriangle s\llbracket B \rrbracket\big)$$

We therefore calculate the two routes around the diagram

$$\begin{tikzcd}
{[\mathrm{Y}[A], \mathrm{d}\mathcal{N}(-; B)]} \arrow[r, "\simeq"] \arrow[d, "{[\mathrm{Y}[A], (s\llbracket - \rrbracket, \overline{s\llbracket - \rrbracket})]}"'] & \mathrm{d}\mathcal{N}(- + [A]; B) \arrow[r, "{\mathsf{dlam}(-; A, B)}"] & \mathrm{d}\mathcal{N}(-; A \Rightarrowtriangle B) \arrow[dd, "{s\llbracket - \rrbracket}"] \\
\big[\mathrm{Y}[A], \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket B \rrbracket\big)\big] \arrow[d, "{u_{[A], s\llbracket B \rrbracket}}"'] & & \\
\mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket A \rrbracket \Rightarrowtriangle s\llbracket B \rrbracket\big) \arrow[rr, "{\mathcal{X}(s\llbracket - \rrbracket, \mathrm{Id}_{s\llbracket A \rrbracket \Rightarrowtriangle s\llbracket B \rrbracket})}"'] & & \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket A \rrbracket \Rightarrowtriangle s\llbracket B \rrbracket\big)
\end{tikzcd}$$

We begin with the anticlockwise route, instantiated at a context Γ. For $(\mathsf{j}, \bar{\mathsf{j}}) : \mathrm{Y}\Gamma \times \mathrm{Y}[A] \Rightarrow \mathrm{d}\mathcal{N}(-; B)$ the pseudonatural transformation $[\mathrm{Y}[A], (s\llbracket - \rrbracket, \overline{s\llbracket - \rrbracket})](\mathsf{j}, \bar{\mathsf{j}})$ is simply the composite

$$\mathrm{Y}\Gamma \times \mathrm{Y}[A] \overset{(\mathsf{j}, \bar{\mathsf{j}})}{\Longrightarrow} \mathrm{d}\mathcal{N}(-; B) \overset{(s\llbracket - \rrbracket, \overline{s\llbracket - \rrbracket})}{\Longrightarrow} \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket B \rrbracket\big) \tag{8.29}$$

Moreover, from (8.20) on page 274 we know that, at Γ, the equivalence $u_{s\llbracket A \rrbracket, s\llbracket B \rrbracket}$ takes a pseudonatural transformation $(\mathsf{k}, \bar{\mathsf{k}}) : \mathrm{Y}\Gamma \times \mathrm{Y}[A] \Rightarrow \mathcal{X}(s\llbracket - \rrbracket, s\llbracket B \rrbracket)$ to the 1-cell

$$\lambda\big(s\llbracket \Gamma \rrbracket \times s\llbracket A \rrbracket \xrightarrow{\mathsf{q}^{\times}_{\Gamma, [A]}} s\llbracket \Gamma \mathbin{@} [A] \rrbracket \xrightarrow{\mathsf{k}_{\Gamma \mathbin{@} [A]}(\iota_1, \iota_2)} s\llbracket B \rrbracket\big)$$

in $\mathcal{X}$, where ι_1 and ι_2 denote the two inclusions $\Gamma \hookrightarrow \Gamma + [A]$ and $[A] \hookrightarrow \Gamma + [A]$. Instantiating in the case where $(\mathsf{k}, \overline{\mathsf{k}})$ is given by (8.29), one obtains

$$\big(u_{[A],s[\![B]\!]} \circ [\mathrm{Y}[A], s[\![-]\!]]\big)(\mathsf{j}, \bar{\mathsf{j}}) = \lambda\big(s[\![\mathsf{j}_{\Gamma \,@\, [A]}(\iota_1, \iota_2)]\!]\big) \circ \mathrm{q}^{\times}_{\Gamma,[A]}$$

It follows that the value of the whole anticlockwise route is $\mathrm{Id}_{sA \Rightarrow sB} \circ \lambda(s[\![\mathsf{j}_{\Gamma+[A]}(\iota_1, \iota_2)]\!] \circ \mathrm{q}^{\times}_{\Gamma,[A]})$.

Next we calculate the clockwise route. For a context Γ and pseudonatural transformation $(\mathsf{j}, \bar{\mathsf{j}})$ as above, the unlabelled equivalence returns the 1-cell $\mathsf{j}_{\Gamma \,@\, [A]}(\iota_1, \iota_2)$ (recall (8.19) on page 274). This is a normal term of type B in context $\Gamma \,@\, [A] = (\Gamma, x_{|\Gamma|+1} : A)$; let us write j for this term. The clockwise composite therefore returns

$$\begin{aligned} s[\![\Gamma \vdash \lambda x.\mathsf{j} : A \Rightarrow B]\!] &= \lambda\big(s[\![\Gamma, x_{|\Gamma|+1} : A \vdash \mathsf{j} : B]\!] \circ \langle \pi_1 \circ \pi_1, \ldots, \pi_n \circ \pi_1, \pi_2\rangle\big) \\ &= \lambda\big(s[\![\mathsf{j}_{\Gamma+[A]}(\iota_1, \iota_2)]\!] \circ \langle \pi_1 \circ \pi_1, \ldots, \pi_n \circ \pi_1, \pi_2\rangle\big) \end{aligned}$$

Since the tupling of projections on the right is exactly $\mathrm{q}^{\times}_{\Gamma,[A]}$ (Remark 8.2.4), the required 2-cell is a structural isomorphism:

$$\begin{aligned} \mathrm{Id}_{sA \Rightarrow sB} \circ \lambda(s[\![\mathsf{j}_{\Gamma \,@\, [A]}(\iota_1, \iota_2)]\!] \circ \mathrm{q}^{\times}_{\Gamma,[A]}) &\cong \lambda(s[\![\mathsf{j}_{\Gamma \,@\, [A]}(\iota_1, \iota_2)]\!] \circ \mathrm{q}^{\times}_{\Gamma,[A]}) \\ &= \lambda\left(s[\![\mathsf{j}_{\Gamma \,@\, [A]}(\iota_1, \iota_2)]\!] \circ \langle \pi_1 \circ \pi_1, \ldots, \pi_n \circ \pi_1, \pi_2\rangle\right) \end{aligned}$$

□

8.4 $\Lambda_{\mathrm{ps}}^{\times,\to}$ is locally coherent

We are finally in a position to prove the main result. To this end, let $\mathfrak{B}$ be a set of base types, $(\mathcal{X}, \Pi_n(-), \Rightarrow)$ be a cc-bicategory, and $s : \widetilde{\mathfrak{B}} \to \mathcal{X}$ be the canonical extension of a set map $\mathfrak{B} \to \mathcal{X}$. This extends in turn to an interpretation $s[\![-]\!] : \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}}) \to \mathcal{X}$. From this interpretation one obtains the glued objects of (8.27) (page 283) and hence a set map $\mathfrak{B} \to \mathrm{gl}(\langle \underline{s} \rangle)$ sending $B \mapsto \mu_B$. This extends via the cartesian closed structure of $\mathrm{gl}(\langle \underline{s} \rangle)$ to an interpretation $\overline{s}[\![-]\!] : \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}}) \to \mathrm{gl}(\langle \underline{s} \rangle)$. Since the forgetful functor $\mathrm{gl}(\langle \underline{s} \rangle) \to \mathcal{X}$ strictly preserves the cc-bicategorical structure, we may write $\overline{s}[\![A]\!] := (G_A, \gamma_B, s[\![A]\!])$ for every type $A \in \widetilde{\mathfrak{B}}$. Moreover, for every context $\Gamma := (x_i : A_i)_{i=1,\ldots,n}$ and term $\Gamma \vdash t : B$ in $\Lambda_{\mathrm{ps}}^{\times,\to}(\widetilde{\mathfrak{B}})$, one obtains a 1-cell $\overline{s}[\![\Gamma \vdash t : B]\!] = \prod_{i=1}^{n} \overline{s}[\![A_i]\!] \to \overline{s}[\![B]\!]$. Write $(\overline{s}'[\![\Gamma \vdash t : B]\!], \overline{\sigma}[\![\Gamma \vdash t : B]\!], s[\![\Gamma \vdash t : B]\!])$ for this 1-cell, which is described pictorially by the following pseudo-commutative diagram in $\mathrm{Hom}(\mathrm{dCon}_{\widetilde{\mathfrak{B}}}, \mathbf{Cat})$

(note that, since $\underline{s}$ is contravariant on $\mathrm{Con}_{\widetilde{\mathfrak{B}}}$, the composite $\mathcal{X}(\underline{s}(-), X) = \mathcal{X}(s\llbracket - \rrbracket, X)$ is covariant):

$$\begin{tikzcd}
\prod_{i=1}^n G_{A_i} \arrow[rr, "\overline{s}'\llbracket \Gamma \vdash t : B \rrbracket"] \arrow[d, "\prod_{i=1}^n \gamma_{A_i}"'] & & G_B \arrow[dd, "\gamma_B"] \\
\prod_{i=1}^n \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket A_i \rrbracket\big) \arrow[d, "\langle -, \dots, = \rangle"'] \arrow[r, phantom, "\overset{\overline{\sigma}\llbracket \Gamma \vdash t : B \rrbracket}{\Leftarrow}\cong"] & \, & \\
\mathcal{X}\big(s\llbracket - \rrbracket, \prod_{i=1}^n s\llbracket A_i \rrbracket\big) \arrow[rr, "s\llbracket \Gamma \vdash t : B \rrbracket \circ (-)"'] & & \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket B \rrbracket\big)
\end{tikzcd} \tag{8.30}$$

Finally, for every rewrite $\Gamma \vdash \tau : t \Rightarrow t' : B$ one obtains a pair of 2-cells

$$\overline{s}'\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket : \overline{s}'\llbracket \Gamma \vdash t : B \rrbracket \Rightarrow \overline{s}'\llbracket \Gamma \vdash t' : B \rrbracket$$
$$s\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket : s\llbracket \Gamma \vdash t : B \rrbracket \Rightarrow s\llbracket \Gamma \vdash t' : B \rrbracket$$

which, by the cylinder condition, satisfy the diagram below. Since $\mathrm{Hom}(\mathrm{dCon}_{\widetilde{\mathfrak{B}}}, \mathbf{Cat})$ is a 2-category, there is no need to distinguish between bracketings.

$$\begin{tikzcd}
\gamma_B \circ \overline{s}'\llbracket \Gamma \vdash t : B \rrbracket \arrow[r, "\gamma_B \circ \overline{s}'\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket"] \arrow[d, "\overline{\sigma}\llbracket \Gamma \vdash t : B \rrbracket"'] & \gamma_B \circ \overline{s}'\llbracket \Gamma \vdash t' : B \rrbracket \arrow[d, "\overline{\sigma}\llbracket \Gamma \vdash t' : B \rrbracket"] \\
s\llbracket \Gamma \vdash t : B \rrbracket \circ \langle -, \dots, = \rangle \circ \prod_{i=1}^n \gamma_{A_i} \arrow[r, "s\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket \circ \langle -, \dots, = \rangle \circ \prod_{i=1}^n \gamma_{A_i}"'] & s\llbracket \Gamma \vdash t' : B \rrbracket \circ \langle -, \dots, = \rangle \circ \prod_{i=1}^n \gamma_{A_i}
\end{tikzcd} \tag{8.31}$$

We now use Proposition 8.3.3 to define 1-cells $\mathsf{unquote}_A : \mu_A \to \overline{s}\llbracket A \rrbracket$ and $\mathsf{quote}_A : \overline{s}\llbracket A \rrbracket \to \eta_A$ by induction on types. On base types B, we take

$$\begin{aligned} \mathsf{unquote}_B &:= \mathrm{Id}_{\mu_B} : \mu_B \to \mu_B = \overline{s}\llbracket B \rrbracket \\ \mathsf{quote}_B &:= (\mathsf{inc}(-; B)^{-1}, \cong, \mathrm{Id}_{sB}) : \overline{s}\llbracket B \rrbracket \to \eta_B \end{aligned}$$

where $(\mathsf{dinc}(-; B)^{-1}, \cong, \mathrm{Id}_{sB})$ is defined in Proposition 8.3.3(2).

On product types $\prod_n(A_1, \ldots, A_n)$, the 1-cell $\mathsf{unquote}_{(\prod_n A_\bullet)} : \mu_{(\prod_n A_\bullet)} \to \prod_{i=1}^n \overline{s}\llbracket A_i \rrbracket$ is the n-ary tupling of the composite

$$\mu_{(\prod_n A_\bullet)} \xrightarrow{(\mathsf{dproj}_k, \mathrm{id}, \pi_k)} \mu_{A_k} \xrightarrow{\mathsf{unquote}_{A_k}} \overline{s}\llbracket A_k \rrbracket$$

for $k = 1, \ldots, n$, where the first 1-cell is defined in Proposition 8.3.3(3). For $\mathsf{quote}_{(\prod_n A_\bullet)}$, we define

$$\mathsf{quote}_{(\prod_n A_\bullet)} := \prod_{i=1}^n \overline{s}\llbracket A_i \rrbracket \xrightarrow{\prod_{i=1}^n \mathsf{quote}_{A_i}} \prod_{i=1}^n \eta_{A_i} \xrightarrow{(\mathrm{dtuple}, \cong, \mathrm{Id}_{s\llbracket \prod_n A_\bullet \rrbracket})} \eta_{(\prod_n A_\bullet)}$$

where the second 1-cell is defined in Proposition 8.3.3(5).

Finally, for exponential types we define $\mathsf{unquote}_{A \Rightarrow B}$ to be the currying of $\Big(\mathsf{unquote}_B \circ \underline{\mathsf{app}}\Big) \circ (\mu_{A \Rightarrow B} \times \mathsf{quote}_A)$, thus:

$$\lambda\Bigg(\mu_{A \Rightarrow B} \times \overline{s}\llbracket A \rrbracket \xrightarrow{\mu_{A \Rightarrow B} \times \mathsf{quote}_A} \big(\mu_{A \Rightarrow B}\big) \times \eta_A \xrightarrow{(\mathrm{dapp}(-;A,B), \mathrm{id}, \mathrm{eval}_{s\llbracket A \rrbracket, s\llbracket B \rrbracket})} \mu_B \xrightarrow{\mathsf{unquote}_B} \overline{s}\llbracket B \rrbracket\Bigg)$$

where we use Proposition 8.3.3(4) for the second arrow. For $\mathsf{quote}_{A \Rightarrow B}$ we define

$$\mathsf{quote}_{A \Rightarrow B} := (\overline{s}\llbracket A \rrbracket \Rightarrow \overline{s}\llbracket B \rrbracket) \to (\nu_A \Rightarrow \eta_B) \xrightarrow{(\mathsf{L}_{A,B}, \cong, \mathrm{Id}_{s\llbracket A \rrbracket \Rightarrow s\llbracket B \rrbracket})} \eta_{A \Rightarrow B}$$

where the second arrow is defined in Proposition 8.3.3(6) and the first arrow is the currying of $\big(\mathsf{quote}_B \circ \mathrm{eval}_{\overline{s}\llbracket A \rrbracket, \overline{s}\llbracket B \rrbracket}\big) \circ \big(((\overline{s}\llbracket A \rrbracket \Rightarrow \overline{s}\llbracket B \rrbracket) \times \mathsf{unquote}_A) \circ ((\overline{s}\llbracket A \rrbracket \Rightarrow \overline{s}\llbracket B \rrbracket) \times \underline{\mathsf{var}})\big)$; that is, the currying of the following composite:

$$\begin{tikzcd}
(\overline{s}\llbracket A \rrbracket \Rightarrow \overline{s}\llbracket B \rrbracket) \times \nu_A \arrow[d, "(\overline{s}\llbracket A \rrbracket \Rightarrow \overline{s}\llbracket B \rrbracket) \times \underline{\mathsf{var}}"'] \arrow[ddrr, bend left] \arrow[dd, bend right=60] & & \\
(\overline{s}\llbracket A \rrbracket \Rightarrow \overline{s}\llbracket B \rrbracket) \times \mu_A \arrow[d, "(\overline{s}\llbracket A \rrbracket \Rightarrow \overline{s}\llbracket B \rrbracket) \times \mathsf{unquote}_A"'] & & \\
(\overline{s}\llbracket A \rrbracket \Rightarrow \overline{s}\llbracket B \rrbracket) \times \overline{s}\llbracket A \rrbracket \arrow[r, "\underline{\mathrm{eval}}_{\overline{s}\llbracket A \rrbracket, \overline{s}\llbracket B \rrbracket}"'] \arrow[rr, bend right, "\mathsf{quote}_B \circ \underline{\mathrm{eval}}_{\overline{s}\llbracket A \rrbracket, \overline{s}\llbracket B \rrbracket}"'] & \overline{s}\llbracket B \rrbracket \arrow[r, "\mathsf{quote}_B"'] & \eta_B
\end{tikzcd}$$

The morphism $\underline{\mathsf{var}} := \big(\mathsf{dvar}(-; A_i), \cong, \mathrm{Id}_{s\llbracket A_i \rrbracket}\big)$ is defined in Proposition 8.3.3(1). Let us denote $\mathsf{unquote}_B := (\widehat{u}_B, \overline{u}_B, u_B)$ and $\mathsf{quote}_B := (\widehat{q}_B, \overline{q}_B, q_B)$, so that $\pi_{\mathrm{dom}}(\mathsf{unquote}_B) = u_B$ and $\pi_{\mathrm{dom}}(\mathsf{quote}_B) = q_B$.

Lemma 8.4.1. For every type $B \in \widetilde{\mathfrak{B}}$, there exist natural isomorphisms $\pi_{\mathrm{dom}}(\mathsf{unquote}_B) \cong \mathrm{Id}_{s[\![B]\!]}$ and $\pi_{\mathrm{dom}}(\mathsf{quote}_B) \cong \mathrm{Id}_{s[\![B]\!]}$.

Proof. We proceed inductively. On base types the claim holds trivially. For product types, we observe that, where $A_1, \ldots, A_n \in \widetilde{\mathfrak{B}}$ $(n \in \mathbb{N})$:

$$\begin{aligned}
\pi_{\mathrm{dom}}(\mathsf{unquote}_{(\prod_n A_\bullet)}) &= \langle u_{A_1} \circ \pi_1, \ldots, u_{A_n} \circ \pi_n \rangle \\
&\cong \left(\textstyle\prod_{i=1}^n u_{A_i}\right) \circ \langle \pi_1, \ldots, \pi_n \rangle \\
&\overset{\mathrm{IH}}{\cong} \left(\textstyle\prod_{i=1}^n \mathrm{Id}_{A_i}\right) \circ \langle \pi_1, \ldots, \pi_n \rangle \\
&\cong \mathrm{Id}_{s[\![\prod_n A_\bullet]\!]} \\
\pi_{\mathrm{dom}}(\mathsf{quote}_{(\prod_n A_\bullet)}) &= \mathrm{Id}_{s[\![\prod_n A_\bullet]\!]} \circ \textstyle\prod_{i=1}^n q_{A_i} \\
&\cong \textstyle\prod_{i=1}^n q_{A_i} \\
&\overset{\mathrm{IH}}{\cong} \textstyle\prod_{i=1}^n \mathrm{Id}_{s[\![A_i]\!]} \\
&\cong \mathrm{Id}_{s[\![\prod_n A_\bullet]\!]}
\end{aligned}$$

Finally, for exponentials, one sees that

$$\begin{aligned}
\pi_{\mathrm{dom}}(\mathsf{unquote}_{A \Rightarrow B}) &= \lambda\Big(\big(u_B \circ \mathrm{eval}_{s[\![A]\!],s[\![B]\!]}\big) \circ (\mathrm{Id}_{s[\![A \Rightarrow B]\!]} \times q_A)\Big) \\
&\overset{\mathrm{IH}}{\cong} \lambda\Big(\big(\mathrm{Id}_{s[\![B]\!]} \circ \mathrm{eval}_{s[\![A]\!],s[\![B]\!]}\big) \circ (\mathrm{Id}_{s[\![A \Rightarrow B]\!]} \times \mathrm{Id}_{s[\![A]\!]})\Big) \\
&\cong \lambda\big(\mathrm{eval}_{s[\![A]\!],s[\![B]\!]} \circ (\mathrm{Id}_{s[\![A \Rightarrow B]\!]} \times \mathrm{Id}_{s[\![A]\!]})\big) \\
&\overset{\eta}{\cong} \mathrm{Id}_{s[\![A \Rightarrow B]\!]} \\
\pi_{\mathrm{dom}}(\mathsf{quote}_{A \Rightarrow B}) &\cong \lambda\Big(\big(q_B \circ \mathrm{eval}_{s[\![A]\!],[\![B]\!]}\big) \circ \Big((\mathrm{Id}_{s[\![A \Rightarrow B]\!]} \times u_A) \circ (\mathrm{Id}_{s[\![A \Rightarrow B]\!]} \times \mathrm{Id}_{s[\![A]\!]})\Big)\Big) \\
&\overset{\mathrm{IH}}{\cong} \lambda\Big(\big(\mathrm{Id}_{s[\![B]\!]} \circ \mathrm{eval}_{s[\![A]\!],[\![B]\!]}\big) \circ \Big((\mathrm{Id}_{s[\![A \Rightarrow B]\!]} \times u_A) \circ (\mathrm{Id}_{s[\![A \Rightarrow B]\!]} \times \mathrm{Id}_{s[\![A]\!]})\Big)\Big) \\
&\cong \lambda\Big(\big(\mathrm{Id}_{s[\![B]\!]} \circ \mathrm{eval}_{s[\![A]\!],[\![B]\!]}\big) \circ \Big((\mathrm{Id}_{s[\![A \Rightarrow B]\!]} \times \mathrm{Id}_{s[\![A]\!]})\Big)\Big) \\
&\cong \lambda\Big(\mathrm{eval}_{s[\![A]\!],[\![B]\!]} \circ \big(\mathrm{Id}_{s[\![A \Rightarrow B]\!]} \times \mathrm{Id}_{s[\![A]\!]}\big)\Big) \\
&\overset{\eta}{\cong} \mathrm{Id}_{s[\![A \Rightarrow B]\!]}
\end{aligned}$$

In each case the isomorphisms are composites of structural isomorphisms or canonical isomorphisms for the cartesian closed structure, hence natural. □

The definitions of $\mathsf{unquote}$ and quote, together with the preceding lemma and the 2-cells $\psi_X^{s[\![-]\!]}$, give rise to diagrams of the following form for every type $B \in \widetilde{\mathfrak{B}}$:

$$\begin{tikzcd}
\mathrm{d}\mathcal{M}(-;B) \arrow[r, "\widehat{u}_B"] \arrow[d, "s\llbracket - \rrbracket"'] \arrow[dr, phantom, "\overset{\overline{u}_B}{\Leftarrow}"] & G_B \arrow[d, "\gamma_B"] \\
\mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket B \rrbracket\big) \arrow[r, "\mathcal{X}(s\llbracket - \rrbracket, u_B)", "\cong"'] \arrow[r, bend right, "\mathrm{Id}_{\mathcal{X}(s\llbracket - \rrbracket, s\llbracket B \rrbracket)}"'] & \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket B \rrbracket\big)
\end{tikzcd}
\qquad
\begin{tikzcd}
G_B \arrow[r, "\widehat{q}_B"] \arrow[d, "\gamma_B"'] \arrow[dr, phantom, "\overset{\overline{q}_B}{\Leftarrow}"] & \mathrm{d}\mathcal{N}(-;B) \arrow[d, "s\llbracket - \rrbracket"] \\
\mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket B \rrbracket\big) \arrow[r, "\mathcal{X}(s\llbracket - \rrbracket, q_B)", "\cong"'] \arrow[r, bend right, "\mathrm{Id}_{\mathcal{X}(s\llbracket - \rrbracket, s\llbracket B \rrbracket)}"'] & \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket B \rrbracket\big)
\end{tikzcd}$$

Thus, for any sequence of types $A_1, \ldots, A_n \in \widetilde{\mathfrak{B}}$ $(n \in \mathbb{N})$, one obtains a diagram of shape

$$\begin{tikzcd}
\prod_{i=1}^n \mathrm{d}\mathcal{M}(-;A_i) \arrow[r, "\prod_{i=1}^n \widehat{u}_{A_i}"] \arrow[d, "\prod_{i=1}^n s\llbracket - \rrbracket"'] \arrow[dr, phantom, "\overset{\cong}{\Leftarrow}"] & \prod_{i=1}^n G_{A_i} \arrow[d, "\prod_{i=1}^n \gamma_{A_i}"] \\
\prod_{i=1}^n \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket A_i \rrbracket\big) \arrow[r, "\prod_{i=1}^n \mathcal{X}(s\llbracket - \rrbracket, u_{A_i})", "\cong"'] \arrow[r, bend right, "\mathrm{Id}_{\prod_i \mathcal{X}(s\llbracket - \rrbracket, s\llbracket A_i \rrbracket)}"'] & \prod_{i=1}^n \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket A_i \rrbracket\big)
\end{tikzcd}$$

by composing with the fuse 2-cells. Pasting these diagrams together with (8.30), one obtains the following diagram in $\mathrm{Hom}(\mathrm{dCon}_{\widetilde{\mathfrak{B}}}, \mathbf{Cat})$ for every rewrite $(\Gamma \vdash \tau : t \Rightarrow t' : B)$ in $\Lambda_{\mathrm{ps}}^{\times,\to}(\widetilde{\mathfrak{B}})$. We write $\overline{s}'\llbracket \tau \rrbracket$ for $\overline{s}'\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket$ and $s\llbracket \tau \rrbracket$ for $s\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket$. Since there are no constants in $\Lambda_{\mathrm{ps}}^{\times,\to}(\widetilde{\mathfrak{B}})$, these rewrites are necessarily invertible.

$$\begin{tikzcd}
\prod_{i=1}^n \mathrm{d}\mathcal{M}(-;A_i) \arrow[r, "\prod_{i=1}^n \widehat{u}_{A_i}"] \arrow[d, "\prod_{i=1}^n s\llbracket - \rrbracket"'] \arrow[dr, phantom, "\overset{\cong}{\Leftarrow}"] & \prod_{i=1}^n G_{A_i} \arrow[d, "\prod_{i=1}^n \gamma_{A_i}"'] \arrow[r, bend left, "\overline{s}'\llbracket \Gamma \vdash t' : B \rrbracket"] \arrow[r, bend right, "\overline{s}'\llbracket \Gamma \vdash t : B \rrbracket"'] \arrow[r, phantom, "\overset{\overline{s}'\llbracket \tau \rrbracket}{\Uparrow \cong}"] & G_B \arrow[r, "\widehat{q}_B"] \arrow[dd, "\gamma_B"] \arrow[ddr, phantom, "\overset{\overline{q}_B}{\underset{\cong}{\Leftarrow}}"] & \mathrm{d}\mathcal{N}(-;B) \arrow[dd, "s\llbracket - \rrbracket"] \\
\prod_{i=1}^n \mathcal{X}(s\llbracket - \rrbracket, s\llbracket A_i \rrbracket) \arrow[r, "\prod_{i=1}^n \mathcal{X}(s\llbracket - \rrbracket, u_{A_i})", "\cong"'] \arrow[r, bend right, "\mathrm{Id}_{\prod_i \mathcal{X}(s\llbracket - \rrbracket, s\llbracket A_i \rrbracket)}"'] & \prod_{i=1}^n \mathcal{X}\big(s\llbracket - \rrbracket, s\llbracket A_i \rrbracket\big) \arrow[d, "\langle -, \ldots, = \rangle"] \arrow[r, phantom, "\overset{\overline{\sigma}\llbracket \Gamma \vdash t : B \rrbracket}{\underset{\cong}{\Leftarrow}}"] & & \\
& \mathcal{X}(s\llbracket - \rrbracket, \prod_{i=1}^n s\llbracket A_i \rrbracket) \arrow[r, bend left, "s\llbracket \Gamma \vdash t : B \rrbracket \circ (-)"] \arrow[r, bend right, "s\llbracket \Gamma \vdash t' : B \rrbracket \circ (-)"'] \arrow[r, phantom, "\overset{s\llbracket \tau \rrbracket \circ (-)}{\Downarrow \cong}"] & \mathcal{X}(s\llbracket - \rrbracket, s\llbracket B \rrbracket) \arrow[r, "\mathcal{X}(s\llbracket - \rrbracket, q_B)", "\cong"'] \arrow[r, bend right, "\mathrm{Id}_{\mathcal{X}(s\llbracket - \rrbracket, s\llbracket B \rrbracket)}"'] & \mathcal{X}(s\llbracket - \rrbracket, s\llbracket B \rrbracket)
\end{tikzcd} \tag{8.32}$$

The proof now hinges on two facts. Firstly, since $\mathcal{N}(-;B)$ is a set, the composite 2-cell obtained by whiskering across the top row of the diagram above must be the identity.

Secondly, the middle part of the diagram satisfies the cylinder condition. Precisely, writing tup for $\langle -, \ldots, = \rangle$, let κ_t be the invertible 2-cell obtained from the front face:

$$\begin{tikzcd}
s\llbracket - \rrbracket \circ \widehat{q}_B \circ \overline{s}'\llbracket \Gamma \vdash t : B \rrbracket \circ \prod_{i=1}^n \widehat{u}_{A_i} \arrow[r, "\kappa_t"] \arrow[d, "\overline{q}_B \circ \overline{s}'\llbracket \Gamma \vdash t : B \rrbracket \circ \prod_{i=1}^n \widehat{u}_{A_i}"', "\cong"] & s\llbracket \Gamma \vdash t : B \rrbracket \circ \mathrm{tup} \circ \prod_{i=1}^n s\llbracket - \rrbracket \\
\mathcal{X}(s\llbracket - \rrbracket, q_B) \circ \gamma_B \circ \overline{s}'\llbracket \Gamma \vdash t : B \rrbracket \circ \prod_{i=1}^n \widehat{u}_{A_i} \arrow[d, "\cong"'] & s\llbracket \Gamma \vdash t : B \rrbracket \circ \mathrm{tup} \circ \mathrm{Id}_{\mathcal{X}(s\llbracket - \rrbracket, u_{A_i})} \circ \prod_{i=1}^n s\llbracket - \rrbracket \arrow[u, "\cong"'] \\
\mathrm{Id}_{\mathcal{X}(s\llbracket - \rrbracket, s\llbracket B \rrbracket)} \circ \gamma_B \circ \overline{s}'\llbracket \Gamma \vdash t : B \rrbracket \circ \prod_{i=1}^n \widehat{u}_{A_i} \arrow[d, "\cong"'] & s\llbracket \Gamma \vdash t : B \rrbracket \circ \mathrm{tup} \circ \prod_{i=1}^n \mathcal{X}(s\llbracket - \rrbracket, u_{A_i}) \circ \prod_{i=1}^n s\llbracket - \rrbracket \arrow[u, "\cong"'] \\
\gamma_B \circ \overline{s}'\llbracket \Gamma \vdash t : B \rrbracket \circ \prod_{i=1}^n \widehat{u}_{A_i} \arrow[d, "\overline{\sigma}\llbracket \Gamma \vdash t : B \rrbracket \circ \prod_{i=1}^n \widehat{u}_{A_i}"', "\cong"] & s\llbracket \Gamma \vdash t : B \rrbracket \circ \mathrm{tup} \circ \prod_{i=1}^n \left(\mathcal{X}(s\llbracket - \rrbracket, u_{A_i}) \circ s\llbracket - \rrbracket \right) \arrow[u, "\cong", "s\llbracket \Gamma \vdash t : B \rrbracket \circ \mathrm{tup} \circ \mathsf{fuse}^{-1}"'] \\
s\llbracket \Gamma \vdash t : B \rrbracket \circ \mathrm{tup} \circ \prod_{i=1}^n \gamma_{A_i} \circ \prod_{i=1}^n \widehat{u}_{A_i} \arrow[r, "s\llbracket \Gamma \vdash t : B \rrbracket \circ \mathrm{tup} \circ \mathsf{fuse}"'] & s\llbracket \Gamma \vdash t : B \rrbracket \circ \mathrm{tup} \circ \prod_{i=1}^n (\gamma_{A_i} \circ \widehat{u}_{A_i}) \arrow[u, "\cong", "s\llbracket \Gamma \vdash t : B \rrbracket \circ \mathrm{tup} \circ \prod_{i=1}^n \overline{u}_{A_i}"']
\end{tikzcd} \tag{8.33}$$

The cylinder condition (8.31) and the functorality of horizontal composition imply that κ_t satisfies the following property in $\mathrm{Hom}(\mathrm{dCon}_{\widetilde{\mathfrak{B}}}, \mathbf{Cat})$:

$$\begin{tikzcd}
s\llbracket - \rrbracket \circ \widehat{q}_B \circ \overline{s}'\llbracket \Gamma \vdash t : B \rrbracket \circ \prod_{i=1}^n \widehat{u}_{A_i} \arrow[r, "s\llbracket - \rrbracket \circ \widehat{q}_B \circ \overline{s}'\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket \circ \prod_{i=1}^n \widehat{u}_{A_i}"] \arrow[d, "\kappa_t"', "\cong"] & s\llbracket - \rrbracket \circ \widehat{q}_B \circ \overline{s}'\llbracket \Gamma \vdash t' : B \rrbracket \circ \prod_{i=1}^n \widehat{u}_{A_i} \arrow[d, leftrightarrow, "\cong"', "\kappa_{t'}"] \\
s\llbracket \Gamma \vdash t : B \rrbracket \circ \mathrm{tup} \circ \prod_{i=1}^n s\llbracket - \rrbracket \arrow[r, "s\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket \circ \mathrm{tup} \circ \prod_{i=1}^n s\llbracket - \rrbracket"'] & s\llbracket \Gamma \vdash t' : B \rrbracket \circ \mathrm{tup} \circ \prod_{i=1}^n s\llbracket - \rrbracket
\end{tikzcd}$$

Applying the first fact, this diagram degenerates to the following:

$$\begin{tikzcd}
s\llbracket - \rrbracket \circ \widehat{q}_B \circ \overline{s}'\llbracket \Gamma \vdash t : B \rrbracket \circ \prod_{i=1}^n \widehat{u}_{A_i} \arrow[r, equal] \arrow[d, "\kappa_t"', "\cong"] & s\llbracket - \rrbracket \circ \widehat{q}_B \circ \overline{s}'\llbracket \Gamma \vdash t' : B \rrbracket \circ \prod_{i=1}^n \widehat{u}_{A_i} \arrow[d, leftrightarrow, "\cong"', "\kappa_{t'}"] \\
s\llbracket \Gamma \vdash t : B \rrbracket \circ \mathrm{tup} \circ \prod_{i=1}^n s\llbracket - \rrbracket \arrow[r, "s\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket \circ \mathrm{tup} \circ \prod_{i=1}^n s\llbracket - \rrbracket"'] & s\llbracket \Gamma \vdash t' : B \rrbracket \circ \mathrm{tup} \circ \prod_{i=1}^n s\llbracket - \rrbracket
\end{tikzcd} \tag{8.34}$$

Instantiating the bottom row of this diagram at the context $\Gamma := (x_i : A_i)_{i=1,\ldots,n}$ and the n-tuple of terms $(\Gamma \vdash x_i : A_i)_{i=1,\ldots,n}$, one sees that

$$\begin{aligned}
\left(s\llbracket \Gamma \vdash t : B \rrbracket \circ \mathrm{tup} \circ \textstyle\prod_{i=1}^n s\llbracket - \rrbracket \right) (\Gamma \vdash x_i : A_i)_{i=1,\ldots,n} &= s\llbracket \Gamma \vdash t : B \rrbracket \circ \langle s\llbracket \Gamma \vdash x_i : A_i \rrbracket \rangle_i \\
&= s\llbracket \Gamma \vdash t : B \rrbracket \circ \langle \pi_1, \ldots, \pi_n \rangle
\end{aligned}$$

We may now extend (8.34) downwards. Writing $T_t := s\llbracket - \rrbracket \circ \widehat{q}_B \circ \overline{s}'\llbracket \Gamma \vdash t : B \rrbracket \circ \prod_{i=1}^n \widehat{u}_{A_i}$ and instantiating at $(\Gamma \vdash x_i : A_i)_{i=1,\dots,n}$, one obtains the following diagram.

$$\begin{tikzcd}
T_t(\Gamma \vdash x_i : A_i)_{i=1,\dots,n} \arrow[r, equal] \arrow[d, "\kappa_t"', "\cong"] & T_{t'}(\Gamma \vdash x_i : A_i)_{i=1,\dots,n} \arrow[d, "\cong"', "\kappa_{t'}"] \\
s\llbracket \Gamma \vdash t : B \rrbracket \circ \langle \pi_1, \dots, \pi_n \rangle \arrow[r, "s\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket \circ \langle \pi_1, \dots, \pi_n \rangle"'] \arrow[d, "\widehat{\varsigma}^{-1}_{\mathrm{Id}_{s\llbracket \Gamma \rrbracket}}"', "\cong"] & s\llbracket \Gamma \vdash t' : B \rrbracket \circ \langle \pi_1, \dots, \pi_n \rangle \arrow[d, "\cong"', "\widehat{\varsigma}^{-1}_{\mathrm{Id}_{s\llbracket \Gamma \rrbracket}}"] \\
s\llbracket \Gamma \vdash t : B \rrbracket \circ \mathrm{Id}_{s\llbracket \Gamma \rrbracket} \arrow[r, "s\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket \circ \mathrm{Id}_{s\llbracket \Gamma \rrbracket}"'] \arrow[d, "\cong"'] & s\llbracket \Gamma \vdash t' : B \rrbracket \circ \mathrm{Id}_{s\llbracket \Gamma \rrbracket} \arrow[d, "\cong"] \\
s\llbracket \Gamma \vdash t : B \rrbracket \arrow[r, "s\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket"'] & s\llbracket \Gamma \vdash t' : B \rrbracket
\end{tikzcd} \tag{8.35}$$

The bottom two squares commute by naturality. Hence, since each component is invertible, it must be the case that $s\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket$ is equal to the clockwise composite around this diagram. We record this result as the following proposition.

Proposition 8.4.2. For any set of base types $\mathfrak{B}$, cc-bicategory $(\mathscr{X}, \Pi_n(-), \Rightarrow)$ and interpretation $s : \mathfrak{B} \to \mathscr{X}$, the induced interpretation $s\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket$ of any rewrite $(\Gamma \vdash \tau : t \Rightarrow t' : B)$ in $\mathscr{X}$ is equal to the 2-cell obtained by composing clockwise around (8.35). Moreover, this 2-cell depends only on the context Γ, the type B, and the terms t and t'. □

Hence, any pair of parallel rewrites $(\Gamma \vdash \tau : t \Rightarrow t' : B)$ and $(\Gamma \vdash \tau' : t \Rightarrow t' : B)$ must be interpreted by the same 2-cell, namely the 2-cell obtained by composing clockwise around (8.35).

Theorem 8.4.3. For any parallel pair of rewrites $\Gamma \vdash \tau : t \Rightarrow t' : B$ and $\Gamma \vdash \tau' : t \Rightarrow t' : B$ in $\Lambda^{\times,\rightarrow}_{\mathrm{ps}}(\widetilde{\mathfrak{B}})$, the interpretations $s\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket$ and $s\llbracket \Gamma \vdash \tau' : t \Rightarrow t' : B \rrbracket$ are equal. □

We wish to instantiate this theorem in the syntactic bicategory to see that any parallel pair of rewrites must be equal in the equational theory of $\Lambda^{\times,\rightarrow}_{\mathrm{ps}}$. However, the cc-pseudofunctor $\iota\llbracket - \rrbracket : \mathcal{T}^{@,\times,\rightarrow}_{\mathrm{ps}}(\widetilde{\mathfrak{B}}) \to \mathcal{T}^{@,\times,\rightarrow}_{\mathrm{ps}}(\widetilde{\mathfrak{B}})$ extending the inclusion $\iota : \mathfrak{B} \hookrightarrow \mathcal{T}^{@,\times,\rightarrow}_{\mathrm{ps}}(\widetilde{\mathfrak{B}})$ is not the identity: the definition for lambda abstractions requires an extra equivalence. Nonetheless, one can leverage the universal property to show that $\iota\llbracket - \rrbracket$ is equivalent to the identity (*c.f.* Corollary 5.3.30).

Lemma 8.4.4. For any set of base types $\mathfrak{B}$, the cc-pseudofunctor $\iota\llbracket - \rrbracket : \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}}) \to \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}})$ extending the inclusion $\iota : \widetilde{\mathfrak{B}} \hookrightarrow \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}})$ is equivalent to the identity. Hence, $\iota\llbracket - \rrbracket$ is a biequivalence.

Proof. By Proposition 5.3.28, the canonical cc-pseudofunctor $\iota^{\#}(-) : \mathcal{FB}ct^{\times,\to}(\widetilde{\mathfrak{B}}) \to \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}})$ (defined in Lemma 5.2.19) is part of a biequivalence; write V_ι for its pseudo-inverse. Moreover, considering the diagram

$$\begin{tikzcd}
\mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}}) \arrow[r, "\iota\llbracket - \rrbracket"] & \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}}) \\
\mathcal{FB}ct^{\times,\to}(\widetilde{\mathfrak{B}}) \arrow[u, "\iota^{\#}(-)"] \arrow[ur, "\iota^{\#}(-)"']
\end{tikzcd}$$

and applying Lemma 5.2.20, one sees that there exists an equivalence $\iota\llbracket - \rrbracket \circ \iota^{\#}(-) \simeq \iota^{\#}(-)$. One therefore obtains a chain of equivalences

$$\begin{aligned}
\mathrm{id}_{\mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}})} &\simeq \iota^{\#}(-) \circ V_\iota \\
&\simeq (\iota\llbracket - \rrbracket \circ \iota^{\#}(-)) \circ V_\iota \\
&\simeq \iota\llbracket - \rrbracket \circ \mathrm{id}_{\mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}})} \\
&\simeq \iota\llbracket - \rrbracket
\end{aligned}$$

as required. □

We can finally prove our theorem.

Theorem 8.4.5. For any set of base types $\mathfrak{B}$ and any rewrites $(\Gamma \vdash \tau : t \Rightarrow t' : B)$ and $(\Gamma \vdash \tau' : t \Rightarrow t' : B)$ in $\Lambda_{\mathrm{ps}}^{\times,\to}(\widetilde{\mathfrak{B}})$, the judgement $(\Gamma \vdash \tau \equiv \tau' : t \Rightarrow t' : B)$ is derivable in $\Lambda_{\mathrm{ps}}^{\times,\to}(\widetilde{\mathfrak{B}})$. Hence, $\Lambda_{\mathrm{ps}}^{\times,\to}(\widetilde{\mathfrak{B}})$ is locally coherent.

Proof. Consider the interpretation in the syntactic model $\iota\llbracket - \rrbracket : \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}}) \to \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}})$ extending the inclusion of base types. Instantiating Proposition 8.4.2, one sees that $\iota\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket = \iota\llbracket \Gamma \vdash \tau' : t \Rightarrow t' : B \rrbracket$ for every parallel pair of rewrites τ and τ'. But biequivalences are locally fully faithful, so by the preceding lemma $\iota\llbracket \Gamma \vdash \tau : t \Rightarrow t' : B \rrbracket = \iota\llbracket \Gamma \vdash \tau' : t \Rightarrow t' : B \rrbracket$ holds if and only if τ and τ' are equal 2-cells in $\mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}})$; that is, $(\Gamma \vdash \tau \equiv \tau' : t \Rightarrow t' : B)$. □

Theorem 8.4.6. Let $\mathfrak{B}$ be any set and $\tau, \sigma : t \Rightarrow t'$ be a parallel pair of 2-cells in the free cc-bicategory on $\mathfrak{B}$. Then $\tau \equiv \sigma$.

Proof. By Proposition 5.3.25, the syntactic bicategory $\mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}})$ is biequivalent to $\mathcal{FB}ct^{\times,\to}(\widetilde{\mathfrak{B}})$, the free cc-bicategory on $\mathfrak{B}$. By the preceding theorem, the images of the 2-cells τ and σ in $\mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}})$ must be equal. Since biequivalences are locally fully faithful, it follows that $\tau \equiv \sigma$. □

We can express this informally as follows. For any cc-bicategory $(\mathcal{B}, \Pi_n(-), \Rightarrow)$ and pair of parallel 2-cells $\sigma, \tau : f \Rightarrow g$ in $\mathcal{B}$, if σ and τ are constructed from the cartesian closed structure using solely structural isomorphisms and the operations of vertical composition and horizontal composition, then $\sigma = \tau$. As a slogan: *all pasting diagrams in the free cc-bicategory commute.*

8.4.1 Evaluating the proof

It is worth examining where the proof of Theorem 8.4.5 would fail if $\Lambda_{\mathrm{ps}}^{\times,\to}$ were not locally coherent. Our reasoning here is only informal, but it should provide a measure of confidence that the many pages of proof do not contain a fatal error, as well as throwing light on what makes the argument work.

The normalisation-by-evaluation proof hinges crucially on two facts: (1) that any interpretation of $\Lambda_{\mathrm{ps}}^{\times,\to}$ induces an interpretation in the glueing bicategory, and (2) that the canonical interpretation of $\Lambda_{\mathrm{ps}}^{\times,\to}$ in the syntactic model is biequivalent to the identity. The first fact entails that, whenever τ and σ are parallel rewrites of type $t \Rightarrow t'$, their interpretations $s[\![\tau]\!]$ and $s[\![\sigma]\!]$ must coincide in every model. Then, writing J for the inverse to $((\iota[\![-]\!])_{\Gamma,A})_{t,t'} : \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}})(\Gamma; A)(t, t') \to \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}})(\Gamma; A)(\iota[\![t]\!], \iota[\![t']\!])$, the second fact allows one to construct the chain of equalities

$$\sigma \equiv \mathsf{J}(\iota[\![\sigma]\!]) \equiv \mathsf{J}(\iota[\![\tau]\!]) \equiv \tau$$

witnessing local coherence. We give a small example showing how (1) fails if one adds extra structure that is not locally coherent.

Consider the $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature $\mathcal{S}$ consisting of a set of base types and a single constant rewrite $x : B \vdash \kappa : x \Rightarrow x : B$ at a base type B. Since we add no extra equations, $\Lambda_{\mathrm{ps}}^{\times,\to}(\mathcal{S})$ is clearly not locally coherent. Now let $(\mathcal{X}, \Pi_n(-), \Rightarrow)$ be any cc-bicategory and $s : \mathfrak{B} \to \mathcal{X}$ an interpretation of base types. Since variables are normal terms, the interpretation of our additional rewrite in the glueing bicategory as in (8.30) on page 288 yields the diagram below, for which we use the fact that the interpretation of the judgement $(x : B \vdash x : B)$ is the identity:

$$\begin{tikzcd}
\mathrm{d}\mathcal{M}(-;B) \arrow[r, bend left, "\mathrm{id}_{\mathrm{d}\mathcal{M}(-;B)}"] \arrow[r, bend right, "\mathrm{id}_{\mathrm{d}\mathcal{M}(-;B)}"'] \arrow[r, phantom, "\Downarrow \overline{s}'[\![\kappa]\!]"] \arrow[d, "s[\![-]\!]"'] \arrow[dr, phantom, "\overset{\cong}{\Leftarrow}"] & \mathrm{d}\mathcal{M}(-;B) \arrow[d, "s[\![-]\!]"] \\
\mathcal{X}\big(s[\![-]\!], s[\![B]\!]\big) \arrow[r, bend left, "s[\![x:B\vdash x:B]\!]\circ(-)"] \arrow[r, bend right, "s[\![x:B\vdash x:B]\!]\circ(-)"'] \arrow[r, phantom, "\Downarrow s[\![\kappa]\!]\circ(-)"] & \mathcal{X}\big(s[\![-]\!], s[\![B]\!]\big)
\end{tikzcd}$$

Since $\mathrm{d}\mathcal{M}(-;B)$ is locally discrete, the 2-cell $\overline{s}'[\![x : B \vdash \kappa : x \Rightarrow x : B]\!]$ can only be the identity. Now consider a context Γ and evaluate at a neutral term $(\!| t |\!) \in M(\Gamma; B)$. The isomorphism filling the central shape is the structural isomorphism $s[\![\Gamma \vdash t : B]\!] \overset{\mathfrak{l}_{s[\![t]\!]}}{\cong} \mathrm{Id}_{s[\![B]\!]} \circ s[\![\Gamma \vdash t : B]\!]$, so the cylinder condition requires that

$$s[\![x : B \vdash \kappa : x \Rightarrow x : B]\!] = \mathfrak{l}_{s[\![t]\!]} \bullet \mathrm{id}_{\mathrm{id}_{\mathrm{d}\mathcal{M}(-;B)}} \bullet \mathfrak{l}_{s[\![t]\!]}^{-1} = \mathrm{id}_{s[\![x]\!]} = s[\![x : B \vdash \mathrm{id}_x : x \Rightarrow x : B]\!]$$

Now, following the argument employed to prove Theorem 8.4.5, one sees that this equation is satisfied for the interpretation extending $\iota : \mathfrak{B} \hookrightarrow \mathcal{T}_{\mathrm{ps}}^{@,\times,\rightarrow}(\widetilde{\mathfrak{B}})$ if and only if the judgement $(x : B \vdash \kappa \equiv \mathrm{id}_x : x \Rightarrow x : B)$ is derivable. Since we assumed this not to be the case, *the cylinder condition cannot hold.* Thus, the constant rewrite κ may not be soundly interpreted in every glueing bicategory $\mathrm{gl}(\langle s \rangle)$, so one cannot rerun the normalisation-by-evaluation proof.

8.5 Another Yoneda-style proof of coherence

Proposition 5.1.10 proved a form of coherence for cc-bicategories. It turns out that this can be extended to an alternative proof of the main result just presented. The strategy is similar to that presented in Section 8.4, but only relies on the universal property of the free cc-bicategory $\mathcal{FB}ct^{\times,\rightarrow}(\widetilde{\mathfrak{B}})$ (defined in Construction 5.2.18). Nonetheless, the development highlights the core of the normalisation-by-evaluation argument as just described.

Fix a set of base types $\mathfrak{B}$ and an interpretation $h : \mathfrak{B} \to \mathcal{X}$ in a cc-bicategory $(\mathcal{X}, \Pi_n(-), \Rightarrow)$. This extends to an interpretation $\widetilde{\mathfrak{B}} \to \mathcal{X}$ we also denote by h. Now let $(\mathcal{C}, \Pi_n(-), \Rightarrow)$ be a 2-category with strict products and exponentials and $(F, \mathrm{q}^{\times}, \mathrm{q}^{\Rightarrow}) : (\mathcal{X}, \Pi_n(-), \Rightarrow) \to (\mathcal{C}, \Pi_n(-), \Rightarrow)$ be any cc-pseudofunctor. Writing F_0 for the underlying set map $ob(\mathcal{X}) \to ob(\mathcal{C})$, one obtains an interpretation $F_0 \circ h : \mathfrak{B} \to \mathcal{C}$. One thereby obtains a *weak* interpretation in $\mathcal{X}$ and a *strict* interpretation in $\mathcal{C}$. The situation is described by the following commutative diagram:

$$\begin{tikzcd}
 & & & & \mathcal{C} & \\
 & & & & \mathcal{X} \arrow[u, "F"] & \simeq \\
\mathfrak{B} \arrow[r, hook] & \widetilde{\mathfrak{B}} \arrow[r, hook] \arrow[uurrr, bend left, "F_0\circ h"] \arrow[urrr, "h"] \arrow[rrr, hook, bend right] & \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}}) \arrow[rr, "\iota[\![-]\!]"'] & & \mathcal{FB}ct^{\times,\to}(\widetilde{\mathfrak{B}}) \arrow[u, "h^{\#}"'] \arrow[uu, bend right=60, "(F\circ h)^{\#}"'] &
\end{tikzcd}$$

Now, the composite $F \circ h^{\#}$ is a cc-pseudofunctor, so by Lemma 5.2.20 there exists an equivalence $(F_0 \circ h)^{\#} \simeq F \circ h^{\#} : \mathcal{FB}ct^{\times,\to}(\widetilde{\mathfrak{B}}) \to \mathcal{C}$. Denote this by $(\mathsf{k}, \overline{\mathsf{k}}) : F \circ h^{\#} \Rightarrow (F_0 \circ h)^{\#}$. For any 1-cell $t : \Gamma \to A$ in $\mathcal{FB}ct^{\times,\to}(\widetilde{\mathfrak{B}})$, one therefore obtains an iso-commuting square

$$\begin{tikzcd}
(F \circ h^{\#})\Gamma \arrow[r, "(F\circ h^{\#})t"] \arrow[d, "\mathsf{k}_\Gamma"'] \arrow[dr, phantom, "\overset{\overline{\mathsf{k}}_t}{\cong}"] & (F \circ h^{\#})A \arrow[d, "\mathsf{k}_A"] \\
(F_0 \circ h)^{\#}\Gamma \arrow[r, "(F_0\circ h)^{\#}t"'] & (F_0 \circ h)^{\#}A
\end{tikzcd}$$

Moreover, the naturality condition on $\overline{\mathsf{k}}_t$ requires that, for any 2-cell $\tau : t \Rightarrow t' : \Gamma \to A$ in $\mathcal{FB}ct^{\times,\to}(\widetilde{\mathfrak{B}})$, the following commutes:

$$\begin{tikzcd}
\mathsf{k}_A \circ (F \circ h^{\#})(t) \arrow[r, "\mathsf{k}_A\circ(F\circ h^{\#})(\tau)"] \arrow[d, "\overline{\mathsf{k}}_t"'] & \mathsf{k}_A \circ (F \circ h^{\#})(t') \arrow[d, "\overline{\mathsf{k}}_{t'}"] \\
(F_0 \circ h)^{\#}(t) \circ \mathsf{k}_\Gamma \arrow[r, "(F_0\circ h)^{\#}(\tau)\circ\mathsf{k}_\Gamma"'] & (F_0 \circ h)^{\#}(t') \circ \mathsf{k}_\Gamma
\end{tikzcd} \tag{8.36}$$

But the cartesian closed structure of $\mathcal{C}$ is strict and the definition of the pseudofunctor $(F_0 \circ h)^{\#}$ only employs the canonical 2-cells of the cc-bicategory structure, so $(F_0 \circ h)^{\#}(\tau)$ is the identity for every 2-cell τ. To see this, one argues by induction on the definition of the cc-pseudofunctor $k^{\#}$ extending a map k interpreting base types (Lemma 5.2.19). It follows that (8.36) degenerates to the following:

$$\begin{tikzcd}
\mathsf{k}_A \circ (F \circ h^{\#})(t) \arrow[r, "\mathsf{k}_A\circ(F\circ h^{\#})(\tau)"] \arrow[d, "\overline{\mathsf{k}}_t"'] & \mathsf{k}_A \circ (F \circ h^{\#})(t') \arrow[d, "\overline{\mathsf{k}}_{t'}"] \\
(F_0 \circ h)^{\#}(t) \circ \mathsf{k}_\Gamma \arrow[r, equal] & (F_0 \circ h)^{\#}(t') \circ \mathsf{k}_\Gamma
\end{tikzcd} \tag{8.37}$$

Now, since $(\mathsf{k}, \overline{\mathsf{k}})$ is an equivalence, every component k_X has a pseudoinverse. Let us denote this by $\mathsf{k}^{\star}_X$. From (8.37), one sees that the following commutes:

$$\begin{tikzcd}
(F \circ h^{\#})(t) \arrow[r, "(F \circ h^{\#})(\tau)"] \arrow[d, "\cong"'] & (F \circ h^{\#})(t') \arrow[d, "\cong"] \\
(\mathsf{k}_A^{\star} \circ \mathsf{k}_A) \circ (F \circ h^{\#})(t) \arrow[r, "(\mathsf{k}_A^{\star} \circ \mathsf{k}_A) \circ (F \circ h^{\#})(\tau)"'] \arrow[d, "\cong"'] & (\mathsf{k}_A^{\star} \circ \mathsf{k}_A) \circ (F \circ h^{\#})(t) \arrow[d, "\cong"] \\
\mathsf{k}_A^{\star} \circ \big(\mathsf{k}_A \circ (F \circ h^{\#})(t)\big) \arrow[r, "\mathsf{k}_A^{\star} \circ \big(\mathsf{k}_A \circ (F \circ h^{\#})(\tau)\big)"'] \arrow[d, "\mathsf{k}_A^{\star} \circ \overline{\mathsf{k}}_t"'] & \mathsf{k}_A^{\star} \circ \big(\mathsf{k}_A \circ (F \circ h^{\#})(t')\big) \arrow[d, "\mathsf{k}_A^{\star} \circ \overline{\mathsf{k}}_{t'}"] \\
\mathsf{k}_A^{\star} \circ \Big((F_0 \circ h)^{\#}(t) \circ \mathsf{k}_\Gamma\Big) \arrow[r, equal] & \mathsf{k}_A^{\star} \circ \Big((F_0 \circ h)^{\#}(t') \circ \mathsf{k}_\Gamma\Big)
\end{tikzcd}$$

One thereby sees that $(F \circ h^{\#})\tau$ is completely determined by a composite of 2-cells, none of which depend on τ.

Proposition 8.5.1. Let $(\mathcal{X}, \Pi_n(-), \Rightarrow\!\!\!\!\rhd)$ be a cc-bicategory , $(\mathcal{C}, \Pi_n(-), \Rightarrow\!\!\!\!\rhd)$ be a 2-category with strict products and exponentials, and $(F, \mathrm{q}^{\times}, \mathrm{q}^{\Rightarrow\!\!\!\rhd}) : (\mathcal{X}, \Pi_n(-), \Rightarrow\!\!\!\!\rhd) \to (\mathcal{C}, \Pi_n(-), \Rightarrow\!\!\!\!\rhd)$ be any cc-pseudofunctor. Then if $h : \widetilde{\mathfrak{B}} \to \mathcal{X}$ is the canonical extension of an interpretation $\mathfrak{B} \to \mathcal{X}$ and $\tau : t \Rightarrow t'$ is any 2-cell in $\mathcal{FB}ct^{\times,\to}(\widetilde{\mathfrak{B}})$, the 2-cell $(F \circ h^{\#})(\tau)$ in $\mathcal{C}$ is completely determined by t and t'. Hence, for any parallel pair of 2-cells $\tau, \sigma : t \Rightarrow t'$ in $\mathcal{FB}ct^{\times,\to}(\widetilde{\mathfrak{B}})$, one has the equality $(F \circ h^{\#})(\tau) = (F \circ h^{\#})(\sigma)$. □

Together with Proposition 5.1.10, one obtains the local coherence of $\mathcal{FB}ct^{\times,\to}(\widetilde{\mathfrak{B}})$, which completes our alternative proof of Theorem 8.4.6.

Theorem 8.5.2. For any set of base types $\mathfrak{B}$ and any pair of parallel 2-cells $\tau, \sigma : t \Rightarrow t'$ in $\mathcal{FB}ct^{\times,\to}(\widetilde{\mathfrak{B}})$, the equality $\tau \equiv \sigma$ holds.

Proof. Instantiate the preceding proposition with $h := \iota : \widetilde{\mathfrak{B}} \hookrightarrow \mathcal{FB}ct^{\times,\to}(\widetilde{\mathfrak{B}})$ the inclusion and F the biequivalence between a cc-bicategory and a 2-category with strict products and exponentials arising from Proposition 5.1.10. Note that $\iota^{\#} \simeq \mathrm{id}_{\mathcal{FB}ct^{\times,\to}(\widetilde{\mathfrak{B}})}$ by Lemma 5.2.20, so that $F \circ \iota^{\#}$ is a biequivalence. Then $F \circ \iota^{\#}$ is locally fully faithful, so $(F \circ \iota^{\#})(\tau) = (F \circ \iota^{\#})(\sigma)$ if and only if $\tau \equiv \sigma$. The result then follows from the preceding proposition. □

Since $\mathcal{FB}ct^{\times,\to}(\widetilde{\mathfrak{B}}) \simeq \mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\widetilde{\mathfrak{B}})$, this entails the local coherence of $\mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\mathcal{S})$. One therefore recovers Theorem 8.4.5.

We end with some comments on the argument just presented. First, as it stands it is not constructive. We make use of the coherence theorem for fp-bicategories (Proposition 4.1.8), for which one chooses a pseudoinverse to the inclusion of a bicategory into its image under the Yoneda embedding. This choice is only determined up to equivalence, so one does not obtain an explicit witness for the product structure. Second, the argument relies crucially on the interplay between weak and strict structure. We use the strictness of $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$ to obtain a strict cc-bicategory biequivalent to our original one, and then we use the strictness of this bicategory to degenerate (8.36) into (8.37). It is, therefore, a strategy that is only available in the higher-categorical setting.

Chapter 9

Conclusions

We leave a full investigation of the applications of the development in this thesis for future work. We do note, however, that the problem we posed in the introduction now disappears.

Consider a structure definable in any cartesian closed category. Examples include the canonical comonoid structure on any object, or the monoid structure on any endo-exponential. This definition is witnessed by a $\Lambda^{\times,\rightarrow}$-term up to $\beta\eta$-equality, and hence—by Proposition 5.4.14—by a $\Lambda^{\times,\rightarrow}_{\mathrm{ps}}$-term over the same signature, with $\beta\eta$-equalities replaced by rewrites. (Since we explicitly construct the correspondence between $\Lambda^{\times,\rightarrow}$-terms and $\Lambda^{\times,\rightarrow}_{\mathrm{ps}}$-terms, this construction can be done via a terminating decision procedure.) These rewrites will provide the data required to define a bicategorical version of the structure under consideration. Theorem 8.4.5 then entails that the required coherence axioms must hold. One thereby obtains the following principle.

Principle 9.1. To show that a *pseudo* structure may be constructed in any cartesian closed bicategory, it suffices to show that its strict version—that is, the image of the corresponding $\Lambda^{\times,\rightarrow}_{\mathrm{ps}}$-term in $\Lambda^{\times,\rightarrow}$—may be constructed in any cartesian closed category. ◂

Applying this principle immediately entails the following results.

Definition 9.2. For any cc-bicategory,

1. Every object has a canonical commutative pseudo-comonoid structure, and
2. Every endo-exponential has a canonical pseudomonoid structure. □

Further work

There are many interesting avenues for further work; we mention a few here.

Extensions to $\Lambda_{\mathrm{ps}}^{\times,\to}$. It is natural to consider incorporating further type-theoretic constructions into $\Lambda_{\mathrm{ps}}^{\times,\to}$. One example would be sum types, corresponding to bicategorical coproducts. Extending the local coherence proof to this type theory would likely require a bicategorical development of *Groethendieck logical relations* [FS99], with possible connections to the theory of stacks. A more ambitious development would be the inclusion of Martin-Löf style dependent types [ML84]. This would be particularly intriguing as the interpretation of these constructions in locally cartesian closed categories is, properly speaking, bicategorical [CD14].

From a different perspective, Pitts has suggested considering the theory of fixpoints. In an unpublished manuscript [Pit87], Pitts considered a calculus for *initial fixpoint categories* (*IFP-categories*): 2-categories equipped with finite products and a notion of 'initial algebra' on every endomorphism of the form $A \xrightarrow{\langle \mathrm{id}_A, a\rangle} A \times B \xrightarrow{f} B$, representing a formal fixpoint construction. Other important examples in a similar vein include *algebraically complete* categories [Fre91], or *iteration (2-)theories* [É99, BÉLM01]. The fact that bicategories represent a natural setting for 'formal category theory' suggests considering constructions of type-theoretic interest (such as fixpoints) as well as constructions of category-theoretic interest (such as monads) as particular constructions within $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$.

An orthogonal line of development would be towards higher levels of categorical structure. One might, for example, extend to tricategories; restricting to unary contexts would recover a type theory for monoidal bicategories. (An alternative approach to the same result would be to introduce a *linear* version of $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$). It may even be possible to inductively generate higher levels of structure to recover some form of ∞-category. For these developments to be principled, the first consideration ought to be the appropriate correlate of biclones.

Applications to higher category theory. Each extension to the type theory raises the question of its coherence. As outlined in the introduction to Chapter 8, there is a wealth of literature studying various forms of normalisation-by-evaluation for extensions to the simply-typed lambda calculus. It is plausible that their bicategorical correlates would lift to extensions of $\Lambda_{\mathrm{ps}}^{\times,\to}$. More speculatively, one might hope that by constructing higher-dimensional type theories and examining their relationship to well-understood classical type theories (in the style of Section 5.4, for instance), one may gain a better understanding of where coherence can be expected and—in the cases it cannot—why it fails.

This thesis also lays the groundwork for bicategorifying further category theoretic results. For instance, the conservative extension result of [FDCB02, §3] shares many tools with the normalisation-by-evaluation argument of [Fio02], such as glueing and the relative hom-functor. It should be possible, therefore, to extend the bicategorical theory presented here to show that cc-bicategories are a conservative extension of fp-bicategories.

Higher-dimensional universal algebra. Moving away from type-theoretic concerns, there remains the question of the universal algebra associated to (mono-sorted) biclones. In the classical setting, it is well-known that the three components of the monad–Lawvere theory–clone triad are all equivalent. Biclones appear to represent one corner of the bicategorical version of this triad: whether pseudomonads and some bicategorical notion of Lawvere theory complete the picture remains to be seen.

Part III

Appendices

Appendix A

An index of free structures and syntactic models

We summarise the various bicategorical free constructions and syntactic models employed throughout this thesis. As a rule of thumb, we use Syn to denote biclones (and their nuclei, *i.e.*restrictions to unary contexts) and $\mathcal{T}_{\mathrm{ps}}$ to denote bicategories.

Chapter 3			
$\mathcal{FC}l(\mathcal{G})$	free biclone on a 2-multigraph	Construction 3.1.16	p. 45
$\mathcal{FB}ct(\mathcal{G})$	free bicategory on a 2-graph	Lemma 3.1.18	p. 47
$\mathrm{Syn}(\mathcal{G})$	syntactic biclone of $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$ on a 2-multigraph	Construction 3.2.11	p. 60
$\mathrm{Syn}(\mathcal{G})\big\|_1$	syntactic bicategory of $\Lambda_{\mathrm{ps}}^{\mathrm{bicat}}$ on a 2-graph	Construction 3.2.15	p. 62
$\mathcal{H}(\mathcal{G})$	syntactic biclone of H^{cl} on a 2-multigraph	Construction 3.3.7	p. 69
Chapter 4			
$\mathcal{FC}l^{\times}(\mathcal{S})$	free cartesian biclone on a $\Lambda_{\mathrm{ps}}^{\times}$-signature	Construction 4.2.58	p. 124
$\mathcal{FB}ct^{\times}(\mathcal{S})$	free fp-bicategory on a unary $\Lambda_{\mathrm{ps}}^{\times}$-signature	Lemma 4.2.62	p. 126
$\mathrm{Syn}^{\times}(\mathcal{S})$	syntactic biclone of $\Lambda_{\mathrm{ps}}^{\times}$ on a $\Lambda_{\mathrm{ps}}^{\times}$-signature	Construction 4.3.6	p. 129
$\mathrm{Syn}^{\times}(\mathcal{S})\big\|_1$	syntactic model of type theory obtained by restricting $\Lambda_{\mathrm{ps}}^{\times}$ to unary contexts	Theorem 4.3.10	p. 131
$\mathcal{T}_{\mathrm{ps}}^{@,\times}(\mathcal{S})$	extension of $\mathrm{Syn}^{\times}(\mathcal{S})\big\|_1$ with context extension product structure	Construction 4.3.15	p. 137
Chapter 5			
$\mathcal{FC}l^{\times,\to}(\mathcal{S})$	free cartesian closed biclone on a $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature	Construction 5.2.16	p. 158
$\mathcal{FB}ct^{\times,\to}(\mathcal{S})$	free cc-bicategory on a $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature	Construction 5.2.18	p. 160
$\mathrm{Syn}^{\times,\to}(\mathcal{S})$	syntactic biclone of $\Lambda_{\mathrm{ps}}^{\times,\to}$ on a $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature	Construction 5.3.8	p. 171
$\overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})}$	nucleus of $\mathrm{Syn}^{\times,\to}(\mathcal{S})$	Construction 5.3.11	p. 172
$\mathcal{T}_{\mathrm{ps}}^{@,\times,\to}(\mathcal{S})$	extension of $\overline{\mathrm{Syn}^{\times,\to}(\mathcal{S})}$ with context extension product structure	Construction 5.3.20	p. 179

Table A.1: An index of free constructions and syntactic models

Appendix B

Cartesian closed structures

We summarise the cartesian closed structures of $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$ and $\mathrm{gl}(F)$.

Cartesian closed structure on $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$**.** Let $\mathcal{B}$ be any 2-category. Then the 2-category $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$ has finite products given pointwise and exponentials given as in the following table:

Exponential $[P, Q]$	$\lambda X^{\mathcal{B}} \,.\, \mathrm{Hom}(\mathcal{B}, \mathbf{Cat})(\mathrm{Y}X \times P, Q)$
Evaluation 1-cell $\underline{\mathrm{eval}}_{P,Q}$	$\lambda X^{\mathcal{B}} \,.\, \lambda(\mathsf{k}, \overline{\mathsf{k}})^{\mathrm{Y}X \times P \Rightarrow Q} \,.\, \lambda p^{PX} \,.\, \mathsf{k}(X, \mathrm{Id}_X, p)$
$\Lambda(\mathsf{j}, \bar{\mathsf{j}})^{R \times P \Rightarrow Q}$	$\lambda X^{\mathcal{B}} \,.\, \lambda r^{RX} \,.\, \lambda A^{\mathcal{B}} \,.\, \lambda(h, p)^{\mathrm{Y}(X,A) \times PA} \,.\, \mathsf{j}\big(X, (Rh)(r), p\big)$ with naturality witnessed by by Lemmas 6.1.4 and 6.1.5
Counit $\mathrm{E}_{P,Q}(\mathsf{j}, \bar{\mathsf{j}})$	$\lambda X^{\mathcal{B}} \,.\, \lambda(r, p)^{RX \times PX} \,.\, \mathsf{j}\big(X, (\psi^R)^{-1}(r), p\big)$
$\mathsf{e}^{\dagger}(\Xi)$	defined by diagram (6.9)

Table B.1: Exponential structure in $\mathrm{Hom}(\mathcal{B}, \mathbf{Cat})$, from Section 6.1

Moreover, for a pseudofunctor $P : \mathcal{B}^{\mathrm{op}} \to \mathbf{Cat}$ and object $X \in \mathcal{B}$ the exponential $[\mathrm{Y}X, P]$ in $\mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$ is given by $P(- \times X)$, with structure summarised in Table B.2.

Cartesian closed structure on $\mathrm{gl}(\mathfrak{J})$**.** Let $(\mathfrak{J}, \mathrm{q}^{\times}) : (\mathcal{B}, \Pi_n(-)) \to (\mathcal{C}, \Pi_n(-))$ be an fp-pseudofunctor between cc-bicategories and suppose that $\mathcal{C}$ has all pullbacks. Then $\mathrm{gl}(\mathfrak{J})$ is cartesian closed, with structure given as in Tables B.3–B.4.

Evaluation 1-cell $\underline{\mathrm{eval}}_{P,Q}$	$\lambda B^{\mathcal{B}} \,.\, \lambda(p,h)^{P(B \times X) \times \mathcal{B}(B,X)} \,.\, P\big(\langle \mathrm{Id}_B, h\rangle\big)(p)$ with naturality witnessed by Lemma 6.2.1
$\Lambda(\mathsf{k}, \overline{\mathsf{k}})^{R \times \mathrm{Y}X \Rightarrow P}$	$\lambda B^{\mathcal{B}} \,.\, \lambda r^{RB} \,.\, \mathsf{k}_{B \times X}\big(R(\pi_1)(r), \pi_2\big)$ with naturality witnessed by Corollary 6.2.3
Counit $\mathrm{E}(\mathsf{k}, \overline{\mathsf{k}})$	defined by diagram (6.15)
$\mathsf{e}^{\dagger}(\Xi)$	defined by diagram (6.17)

Table B.2: Exponential structure in Hom($\mathcal{B}$, **Cat**), from Section 6.2

Product $\prod_i (C_i, c_i, B_i)_i$	$\big(\prod_i C_i, \mathrm{q}^{\times} \circ \prod_i c_i, \prod_i B_i\big)$
Projection 1-cells $\underline{\pi}_k$	(π_k, μ_k, π_k) for μ_k defined in (7.5)
n-ary tupling $\langle \underline{t}_1, \ldots, \underline{t}_n \rangle$ for $\underline{t}_i := (t_i, \alpha_i, s_i)$	$(\langle t_\bullet \rangle, \{\alpha_\bullet\}, \langle s_\bullet \rangle)$ for $\{\alpha_\bullet\}$ defined in (7.6)
Counit $\underline{\varpi}$	kth component is $(\varpi^{(k)}_{f_\bullet}, \varpi^{(k)}_{g_\bullet})$
$\mathsf{p}^{\dagger}(\underline{\tau}_1, \ldots, \underline{\tau}_n)$ for $\underline{\tau}_i := (\tau_i, \sigma_i) : \underline{\pi}_k \circ \underline{u} \Rightarrow \underline{t}_i$ $(i = 1, \ldots, n)$	$\big(\mathsf{p}^{\dagger}(\tau_1, \ldots, \tau_n), \mathsf{p}^{\dagger}(\sigma_1, \ldots, \sigma_n)\big)$

Table B.3: Product structure in gl($\mathfrak{J}$), from Section 7.3.1

Exponential $(C, c, B) \Rightarrow (C', c', B')$	$(C \supset C', p_{c,c'}, B \Rightarrow B')$ defined by the pullback (7.11)
Evaluation 1-cell $\underline{\mathrm{eval}}_{\underline{C},\underline{C'}}$	$(\mathrm{eval}_{C,C'} \circ (q_{c,c'} \times C), \underline{E}_{\underline{C},\underline{C'}}, \mathrm{eval}_{B,B'})$ for $\mathrm{E}_{\underline{C},\underline{C'}}$ defined in (7.12) and (7.13)
$\underline{\lambda}(t, \alpha, s)$	$(\underline{\mathsf{lam}}(t), \Gamma_{c,c'}, \lambda s)$ for $\underline{\mathsf{lam}}(t)$ and $\Gamma_{c,c'}$ defined by UMP of pullback applied to L_α (7.15)
Counit $\underline{\varepsilon}$	$(\underline{\mathsf{e}}, \varepsilon)$ for $\underline{\mathsf{e}}$ defined in (7.17)
$\mathsf{e}^{\dagger}(\underline{\tau})$ for $\underline{\tau} := (\tau, \sigma)$	$\big(\tau^{\sharp}, \mathsf{e}^{\dagger}(\sigma)\big)$ for $\tau^{\sharp}$ defined by UMP of pullback applied to fill-in defined in (7.20)

Table B.4: Exponential structure in gl($\mathfrak{J}$), from Section 7.3.2

Appendix C

The type theory and its semantic interpretation

C.1 The type theory $\Lambda_{\mathrm{ps}}^{\times,\to}$

Fix a $\Lambda_{\mathrm{ps}}^{\times,\to}$-signature $\mathcal{S} = (\mathfrak{B}, \mathcal{G})$ (Definition 5.2.13 on page 157). We give the rules for the full type theory $\Lambda_{\mathrm{ps}}^{\times,\to}$. The type theories $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$ and $\Lambda_{\mathrm{ps}}^{\times}$ are fragments of $\Lambda_{\mathrm{ps}}^{\times,\to}$, and the type theories $\Lambda_{\mathrm{ps}}^{\mathrm{bicat}}$ and $\Lambda_{\mathrm{ps}}^{\times}\big|_1$ are respectively obtained by restricting $\Lambda_{\mathrm{ps}}^{\mathrm{bicl}}$ and $\Lambda_{\mathrm{ps}}^{\times}$ to unary contexts.

$$\frac{}{\diamond\ \mathrm{ctx}} \qquad\qquad \frac{\Gamma\ \mathrm{ctx} \qquad x \notin \mathrm{dom}(\Gamma)}{\Gamma, x : A\ \mathrm{ctx}}\ (A \in \widetilde{\mathfrak{B}})$$

Figure C.1: Rules for contexts

$$\frac{}{x_1 : A_1, \ldots, x_n : A_n \vdash x_k : A_k}\;\text{var } (1 \leqslant k \leqslant n)$$

$$\frac{c \in \mathcal{G}(A_1, \ldots, A_n; B)}{x_1 : A_1, \ldots, x_n : A_n \vdash c(x_1, \ldots, x_n) : B}\;\text{const}$$

$$\frac{x_1 : A_1, \ldots, x_n : A_n \vdash t : B \qquad (\Delta \vdash u_i : A_i)_{i=1,\ldots,n}}{\Delta \vdash t\{x_1 \mapsto u_1, \ldots, x_n \mapsto u_n\} : B}\;\text{horiz-comp}$$

$$\frac{}{p : \prod_n(A_1, \ldots, A_n) \vdash \pi_k(p) : A_k}\;k\text{-proj } (1 \leqslant k \leqslant n)$$

$$\frac{\Gamma \vdash t_1 : A_1 \quad \ldots \quad \Gamma \vdash t_n : A_n}{\Gamma \vdash \mathsf{tup}(t_1, \ldots, t_n) : \prod_n(A_1, \ldots, A_n)}\;n\text{-tuple}$$

$$\frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x.t : A \mathrel{=\!\!\rhd} B}\;\text{lam} \qquad\qquad \frac{}{f : A \mathrel{=\!\!\rhd} B, x : A \vdash \mathsf{eval}(f, x) : B}\;\text{eval}$$

Figure C.2: Introduction rules for terms

$$\frac{x_1 : A_1, \ldots, x_n : A_n \vdash t : B}{\begin{array}{c} x_1 : A_1, \ldots, x_n : A_n \vdash \iota_t : t \Rightarrow t\{x_i \mapsto x_i\} : B \\ x_1 : A_1, \ldots, x_n : A_n \vdash \iota_t^{-1} : t\{x_i \mapsto x_i\} \Rightarrow t : B \end{array}}\;\iota\text{-intro}$$

$$\frac{x_1 : A_1, \ldots, x_n : A_n \vdash x_k : A_k \qquad (\Delta \vdash u_i : A_i)_{i=1,\ldots,n}}{\begin{array}{c} \Delta \vdash \varrho^{(k)}_{u_1,\ldots,u_n} : x_k\{x_i \mapsto u_i\} \Rightarrow u_k : A_k \\ \Delta \vdash \varrho^{(-k)}_{u_1,\ldots,u_n} : u_k \Rightarrow x_k\{x_i \mapsto u_i\} : A_k \end{array}}\;\varrho^{(k)}\text{-intro } (1 \leqslant k \leqslant n)$$

$$\frac{\begin{array}{c} (\Delta \vdash u_j : A_j)_{j=1,\ldots m} \\ (x_1 : A_1, \ldots, x_m : A_m \vdash v_i : B_i)_{i=1,\ldots,n} \\ y_1 : B_1, \ldots, y_n : B_n \vdash t : C \end{array}}{\begin{array}{c} \Delta \vdash \mathsf{assoc}_{t,v_\bullet,u_\bullet} : t\{y_i \mapsto v_i\}\{x_j \mapsto u_j\} \Rightarrow t\{y_i \mapsto v_i\{x_j \mapsto u_j\}\} : C \\ \Delta \vdash \mathsf{assoc}^{-1}_{t,v_\bullet,u_\bullet} : t\{y_i \mapsto v_i\{x_j \mapsto u_j\}\} \Rightarrow t\{y_i \mapsto v_i\}\{x_j \mapsto u_j\} : C \end{array}}\;\text{assoc-intro}$$

Figure C.3: Introduction rules for structural rewrites

$$\frac{\Gamma \vdash t : A}{\Gamma \vdash \mathrm{id}_t : t \Rightarrow t : A}\ \text{id-intro}$$

$$\frac{\kappa \in \mathcal{G}(A_1, \ldots, A_n; B)(c, c')}{x_1 : A_1, \ldots, x_n : A_n \vdash \kappa(x_1, \ldots, x_n) : c(x_1, \ldots, x_n) \Rightarrow c'(x_1, \ldots, x_n) : B}\ \text{2-const}$$

$$\frac{\Gamma \vdash t_1 : A_1 \qquad \ldots \qquad \Gamma \vdash t_n : A_n}{\Gamma \vdash \varpi^{(k)}_{t_1, \ldots, t_n} : \pi_k\{\mathsf{tup}(t_1, \ldots, t_n)\} \Rightarrow t_k : A_k}\ \varpi^{(k)}\text{-intro } (1 \leqslant k \leqslant n)$$

$$\frac{\Gamma \vdash u : \textstyle\prod_n(A_1, \ldots, A_n) \qquad (\Gamma \vdash \alpha_i : \pi_i\{u\} \Rightarrow t_i : A_i)_{i=1,\ldots,n}}{\Gamma \vdash \mathsf{p}^{\dagger}(\alpha_1, \ldots, \alpha_n) : u \Rightarrow \mathsf{tup}(t_1, \ldots, t_n) : \textstyle\prod_n(A_1, \ldots, A_n)}\ \mathsf{p}^{\dagger}(\alpha_1, \ldots, \alpha_n)\text{-intro}$$

$$\frac{\Gamma, x : A \vdash t : B}{\Gamma, x : A \vdash \varepsilon_t : \mathsf{eval}\{(\lambda x.t)\{\mathrm{inc}_x\}, x\} \Rightarrow t : B}\ \varepsilon\text{-intro}$$

$$\frac{\begin{array}{c}\Gamma, x : A \vdash t : B \qquad \Gamma \vdash u : A \mathrel{=\!\!\rhd} B \\ \Gamma, x : A \vdash \alpha : \mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} \Rightarrow t : B\end{array}}{\Gamma \vdash \mathsf{e}^{\dagger}(x \,.\, \alpha) : u \Rightarrow \lambda x.t : A \mathrel{=\!\!\rhd} B}\ \mathsf{e}^{\dagger}(x \,.\, \alpha)\text{-intro}$$

Figure C.4: Introduction rules for basic rewrites

$$\frac{\Gamma \vdash \tau : t \Rightarrow t' : A \qquad \Gamma \vdash \tau' : t' \Rightarrow t'' : A}{\Gamma \vdash \tau' \bullet \tau : t \Rightarrow t'' : A}\ \text{vert-comp}$$

$$\frac{x_1 : A_1, \ldots, x_n : A_n \vdash \tau : t \Rightarrow t' : B \qquad (\Delta \vdash \sigma_i : u_i \Rightarrow u_i' : A_i)_{i=1,\ldots,n}}{\Delta \vdash \tau\{x_i \mapsto \sigma_i\} : t\{x_i \mapsto u_i\} \Rightarrow t'\{x_i \mapsto u_i'\} : B}\ \text{horiz-comp}$$

Figure C.5: Composition operations for rewrites

$$\frac{\Gamma \vdash t_1 : A_1 \qquad \ldots \qquad \Gamma \vdash t_n : A_n}{\Gamma \vdash \varpi^{(-k)}_{t_1, \ldots, t_n} : t_k \Rightarrow \pi_k\{\mathsf{tup}(t_1, \ldots, t_n)\} : A_k}\ \varpi^{(-k)}\text{-intro } (1 \leqslant k \leqslant n)$$

$$\frac{\Gamma \vdash t : \textstyle\prod_n(A_1, \ldots, A_n)}{\Gamma \vdash \varsigma_t^{-1} : \mathsf{tup}(\pi_1\{t\}, \ldots, \pi_n\{t\}) \Rightarrow t : \textstyle\prod_n(A_1, \ldots, A_n)}\ \varsigma^{-1}\text{-intro}$$

$$\frac{\Gamma \vdash u : A \mathrel{=\!\!\rhd} B}{\Gamma \vdash \eta_u^{-1} : \lambda x.\mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} \Rightarrow u : A \mathrel{=\!\!\rhd} B}\ \eta^{-1}\text{-intro}$$

$$\frac{\Gamma, x : A \vdash t : B}{\Gamma, x : A \vdash \varepsilon_t^{-1} : t \Rightarrow \mathsf{eval}\{(\lambda x.t)\{\mathrm{inc}_x\}, x\} : B}\ \varepsilon^{-1}\text{-intro}$$

Figure C.6: Introduction rules for pseudo cartesian closed structure

$$\frac{\Gamma \vdash \tau : t \Rightarrow t' : A}{\Gamma \vdash \tau \bullet \mathrm{id}_t \equiv \tau : t \Rightarrow t' : A}\ \bullet\text{-right-unit} \qquad \frac{\Gamma \vdash \tau : t \Rightarrow t' : A}{\Gamma \vdash \tau \equiv \mathrm{id}_{t'} \bullet \tau : t \Rightarrow t' : A}\ \bullet\text{-left-unit}$$

$$\frac{\Gamma \vdash \tau'' : t'' \Rightarrow t''' : A \qquad \Gamma \vdash \tau' : t' \Rightarrow t'' : A \qquad \Gamma \vdash \tau : t \Rightarrow t' : A}{\Gamma \vdash (\tau'' \bullet \tau') \bullet \tau \equiv \tau'' \bullet (\tau' \bullet \tau) : t \Rightarrow t''' : A}\ \bullet\text{-assoc}$$

Figure C.7: Categorical structure of vertical composition

$$\frac{x_1 : A_1, \ldots, x_n : A_n \vdash t : B \qquad (\Delta \vdash u_i : A_i)_{i=1,\ldots,n}}{\Delta \vdash \mathrm{id}_t\{x_i \mapsto u_i\} \equiv \mathrm{id}_{t\{x_i \mapsto u_i\}} : t\{x_i \mapsto u_i\} \Rightarrow t\{x_i \mapsto u_i\} : B}\ \text{id-preservation}$$

$$\frac{\begin{array}{cc} x_1 : A_1, \ldots, x_n : A_n \vdash \tau : t \Rightarrow t' : B & (\Delta \vdash \sigma_i : u_i \Rightarrow u'_i : A_i)_{i=1,\ldots,n} \\ x_1 : A_1, \ldots, x_n : A_n \vdash \tau' : t' \Rightarrow t'' : B & (\Delta \vdash \sigma'_i : u'_i \Rightarrow u''_i : A_i)_{i=1,\ldots,n} \end{array}}{\Delta \vdash \tau'\{x_i \mapsto \sigma'_i\} \bullet \tau\{x_i \mapsto \sigma_i\} \equiv (\tau' \bullet \tau)\{x_i \mapsto \sigma'_i \bullet \sigma_i\} : t\{x_i \mapsto u_i\} \Rightarrow t''\{x_i \mapsto u''_i\} : B}\ \text{interchange}$$

Figure C.8: Preservation rules

$$\frac{(\Delta \vdash \sigma_i : u_i \Rightarrow u'_i : A_i)_{i=1,\ldots,n}}{\Delta \vdash \varrho^{(k)}_{u'_1,\ldots,u'_n} \bullet x_k\{x_i \mapsto \sigma_i\} \equiv \sigma_k \bullet \varrho^{(k)}_{u_1,\ldots,u_n} : x_k\{x_i \mapsto u_i\} \Rightarrow u'_k : A_k}\ (1 \leqslant k \leqslant n)$$

$$\frac{x_1 : A_1, \ldots, x_n : A_n \vdash \tau : t \Rightarrow t' : B}{x_1 : A_1, \ldots, x_n : A_n \vdash \iota_{t'} \bullet \tau \equiv \tau\{x_i \mapsto x_i\} \bullet \iota_t : t \Rightarrow t'\{x_i \mapsto x_i\} : B}$$

$$\frac{\begin{array}{c} (\Delta \vdash \mu_j : u_j \Rightarrow u'_j : A_j)_{j=1,\ldots m} \\ (x_1 : A_1, \ldots, x_m : A_m \vdash \sigma_i : v_i \Rightarrow v'_i : B_i)_{i=1,\ldots,n} \\ y_1 : B_1, \ldots, y_n : B_n \vdash \tau : t \Rightarrow t' : C \end{array}}{\begin{array}{r} \Delta \vdash \mathsf{assoc}_{t',v_\bullet,u_\bullet} \bullet \tau\{y_i \mapsto \sigma_i\}\{x_j \mapsto \mu_j\} \equiv \tau\{y_i \mapsto \sigma_i\{x_j \mapsto \mu_j\}\} \bullet \mathsf{assoc}_{t,v_\bullet,u_\bullet} \\ : t\{y_i \mapsto v_i\}\{x_j \mapsto u_j\} \Rightarrow t'\{y_i \mapsto v'_i\{x_j \mapsto u'_j\}\} : C \end{array}}$$

Figure C.9: Naturality rules for structural rewrites

$$\frac{x_1 : A_1, \dots, x_n : A_n \vdash t : B \qquad (\Delta \vdash u_i : A_i)_{i=1,\dots,n}}{\Delta \vdash t\{x_i \mapsto \varrho^{(i)}_{u_\bullet}\} \bullet \mathsf{assoc}_{t,x_\bullet,u_\bullet} \bullet \iota_t\{x_i \mapsto u_i\} \equiv \mathrm{id}_{t\{x_i \mapsto u_i\}} : t\{x_i \mapsto u_i\} \Rightarrow t\{x_i \mapsto u_i\} : B}$$

$$\frac{\begin{array}{cc} (\Delta \vdash u_j : A_j)_{j=1,\dots m} & (y_1 : B_1, \dots, y_n : B_n \vdash w_j : C_k)_{k=1,\dots,l} \\ (x_1 : A_1, \dots, x_m : A_m \vdash v_i : B_i)_{i=1,\dots,n} & z_1 : C_1, \dots, z_l : C_l \vdash t : D \end{array}}{\begin{array}{l} \Delta \vdash t\{z_k \mapsto \mathsf{assoc}_{w_k,v_\bullet,u_\bullet}\} \bullet \mathsf{assoc}_{t,w_\bullet\{y_j \mapsto v_j\},u_\bullet} \bullet \mathsf{assoc}_{t,w_\bullet,v_\bullet}\{x_j \mapsto u_j\} \\ \qquad \equiv \mathsf{assoc}_{t,w_\bullet,v_\bullet\{x_j \mapsto u_i\}} \bullet \mathsf{assoc}_{t\{z_k \mapsto w_k\},v_\bullet,u_\bullet} \\ \qquad\quad : t\{z_k \mapsto w_k\}\{y_i \mapsto v_i\}\{x_j \mapsto u_j\} \Rightarrow t\{z_k \mapsto w_k\{y_i \mapsto v_i\{x_j \mapsto u_j\}\}\} : D \end{array}}$$

Figure C.10: Biclone laws

$$\frac{\Gamma \vdash \alpha_1 : \pi_1\{u\} \Rightarrow t_1 : A_1 \quad \dots \quad \Gamma \vdash \alpha_n : \pi_n\{u\} \Rightarrow t_n : A_n}{\Gamma \vdash \alpha_k \equiv \varpi^{(k)}_{t_1,\dots,t_n} \bullet \pi_k\big\{\mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n)\big\} : \pi_k\{u\} \Rightarrow t_k : A_k} \text{ U1 } (1 \leqslant k \leqslant n)$$

$$\frac{\Gamma \vdash \gamma : u \Rightarrow \mathsf{tup}(t_1, \dots, t_n) : \prod_n(A_1, \dots, A_n)}{\Gamma \vdash \gamma \equiv \mathsf{p}^\dagger(\varpi^{(1)}_{t_\bullet} \bullet \pi_1\{\gamma\}, \dots, \varpi^{(n)}_{t_\bullet} \bullet \pi_n\{\gamma\}) : u \Rightarrow \mathsf{tup}(t_1, \dots, t_n) : \prod_n(A_1, \dots, A_n)} \text{ U2}$$

$$\frac{\big(\Gamma \vdash \alpha_i \equiv \alpha'_i : \pi_i\{u\} \Rightarrow t_i : A_i\big)_{i=1,\dots,n}}{\Gamma \vdash \mathsf{p}^\dagger(\alpha_1, \dots, \alpha_n) \equiv \mathsf{p}^\dagger(\alpha'_1, \dots, \alpha'_n) : u \Rightarrow \mathsf{tup}(t_1, \dots, t_n) : \prod_n(A_1, \dots, A_n)} \text{ cong}$$

Figure C.11: Universal property of $\mathsf{p}^\dagger(\alpha)$

$$\frac{\Gamma, x : A \vdash \alpha : \mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} \Rightarrow t : B}{\Gamma, x : A \vdash \alpha \equiv \varepsilon_t \bullet \mathsf{eval}\big\{\mathsf{e}^\dagger(x \,.\, \alpha)\{\mathrm{inc}_x\}, x\big\} : \mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} \Rightarrow t : B} \text{ U1}$$

$$\frac{\Gamma \vdash \gamma : u \Rightarrow \lambda x.t : A \Rightarrow B}{\Gamma \vdash \gamma \equiv \mathsf{e}^\dagger(x \,.\, \varepsilon_t \bullet \mathsf{eval}\{\gamma\{\mathrm{inc}_x\}, x\}) : u \Rightarrow \lambda x.t : A \Rightarrow B} \text{ U2}$$

$$\frac{\Gamma, x : A \vdash \alpha \equiv \alpha' : \mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} \Rightarrow t : B}{\Gamma \vdash \mathsf{e}^\dagger(x \,.\, \alpha) \equiv \mathsf{e}^\dagger(x \,.\, \alpha') : u \Rightarrow \lambda x.t : A \Rightarrow B} \text{ cong}$$

Figure C.12: Universal property of $\mathsf{e}^\dagger(\alpha)$

$$\frac{\Gamma \vdash t_1 : A_1 \qquad \ldots \qquad \Gamma \vdash t_n : A_n}{\Gamma \vdash \varpi^{(-k)}_{t_1,\ldots,t_n} \bullet \varpi^{(k)}_{t_1,\ldots,t_n} \equiv \mathrm{id}_{\pi_k\{\mathsf{tup}(t_1,\ldots,t_n)\}} : \pi_k\{\mathsf{tup}(t_1,\ldots,t_n)\} \Rightarrow \pi_k\{\mathsf{tup}(t_1,\ldots,t_n)\} : A_k}$$

$$\frac{\Gamma \vdash t_1 : A_1 \qquad \ldots \qquad \Gamma \vdash t_n : A_n}{\Gamma \vdash \varpi^{(k)}_{t_1,\ldots,t_n} \bullet \varpi^{(-k)}_{t_1,\ldots,t_n} \equiv \mathrm{id}_{t_k} : t_k \Rightarrow t_k : A_k}$$

$$\frac{\Gamma \vdash t : \prod_n(A_1,\ldots,A_n)}{\Gamma \vdash \varsigma_t^{-1} \bullet \varsigma_t \equiv \mathrm{id}_t : t \Rightarrow t : \prod_n(A_1,\ldots,A_n)}$$

$$\frac{\Gamma \vdash t : \prod_n(A_1,\ldots,A_n)}{\Gamma \vdash \varsigma_t \bullet \varsigma_t^{-1} \equiv \mathrm{id}_{\mathsf{tup}(\pi_1\{t\},\ldots,\pi_n\{t\})} : \mathsf{tup}(\pi_\bullet\{t\}) \Rightarrow \mathsf{tup}(\pi_\bullet\{t\}) : \prod_n(A_1,\ldots,A_n)}$$

$$\frac{\Gamma \vdash u : A \mathbin{=\!\!\rhd} B}{\Gamma \vdash \eta_u \bullet \eta_u^{-1} \equiv \mathrm{id}_{\lambda x.\mathsf{eval}\{u\{\mathrm{inc}_x\},x\}} : \lambda x.\mathsf{eval}\{u\{\mathrm{inc}_x\},x\} \Rightarrow \lambda x.\mathsf{eval}\{u\{\mathrm{inc}_x\},x\} : A \mathbin{=\!\!\rhd} B}$$

$$\frac{\Gamma \vdash u : A \mathbin{=\!\!\rhd} B}{\Gamma \vdash \eta_u^{-1} \bullet \eta_u \equiv \mathrm{id}_u : u \Rightarrow u : A \mathbin{=\!\!\rhd} B} \qquad \frac{\Gamma, x : A \vdash t : B}{\Gamma, x : A \vdash \varepsilon_t \bullet \varepsilon_t^{-1} \equiv \mathrm{id}_t : t \Rightarrow t : B}$$

$$\frac{\Gamma, x : A \vdash t : B}{\Gamma, x : A \vdash \varepsilon_t^{-1} \bullet \varepsilon_t \equiv \mathrm{id}_{\mathsf{eval}\{(\lambda x.t)\{\mathrm{inc}_x\},x\}} : \mathsf{eval}\{(\lambda x.t)\{\mathrm{inc}_x\},x\} \Rightarrow \mathsf{eval}\{(\lambda x.t)\{\mathrm{inc}_x\},x\} : B}$$

Figure C.13: Invertibility rules for pseudo cartesian closed structure

$$\frac{\Gamma \vdash t : B}{\Gamma \vdash \iota_t^{-1} \bullet \iota_t \equiv \mathrm{id}_t : t \Rightarrow t : B}$$

$$\frac{x_1 : A_1, \dots, x_n : A_n \vdash t : B}{x_1 : A_1, \dots, x_n : A_n \vdash \iota_t \bullet \iota_t^{-1} \equiv \mathrm{id}_t : t\{x_i \mapsto x_i\} \Rightarrow t\{x_i \mapsto x_i\} : B}$$

$$\frac{x_1 : A_1, \dots, x_n : A_n \vdash u_1 : A_1 \quad \dots \quad x_1 : A_1, \dots, x_n : A_n \vdash u_n : A_n}{x_1 : A_1, \dots, x_n : A_n \vdash \varrho_{u_\bullet}^{(-k)} \bullet \varrho_{u_\bullet}^{(k)} \equiv \mathrm{id}_{x_k\{x_i \mapsto u_i\}} : x_k\{x_i \mapsto u_i\} \Rightarrow x_k\{x_i \mapsto u_i\} : A_k}\ (1 \leqslant k \leqslant n)$$

$$\frac{x_1 : A_1, \dots, x_n : A_n \vdash u : B}{x_1 : A_1, \dots, x_n : A_n \vdash \varrho_{u_\bullet}^{(k)} \bullet \varrho_{u_\bullet}^{(-k)} \equiv \mathrm{id}_u : u \Rightarrow u : A}\ (1 \leqslant k \leqslant n)$$

$$\frac{\begin{array}{l}(\Delta \vdash u_j : A_j)_{j=1,\dots m} \\ (x_1 : A_1, \dots, x_m : A_m \vdash v_i : B_i)_{i=1,\dots,n} \qquad y_1 : B_1, \dots, y_n : B_n \vdash t : C\end{array}}{\Delta \vdash \mathsf{assoc}_{t,v_\bullet,u_\bullet}^{-1} \bullet \mathsf{assoc}_{t,v_\bullet,u_\bullet} \equiv \mathrm{id}_{t\{v_i\}\{u_j\}} : t\{y_i \mapsto v_i\}\{x_j \mapsto u_j\} \Rightarrow t\{y_i \mapsto v_i\}\{x_j \mapsto u_j\} : C}$$

$$\frac{\begin{array}{l}(\Delta \vdash u_j : A_j)_{j=1,\dots m} \\ (x_1 : A_1, \dots, x_m : A_m \vdash v_i : B_i)_{i=1,\dots,n} \qquad y_1 : B_1, \dots, y_n : B_n \vdash t : C\end{array}}{\Delta \vdash \mathsf{assoc}_{t,v_\bullet,u_\bullet} \bullet \mathsf{assoc}_{t,v_\bullet,u_\bullet}^{-1} \equiv \mathrm{id}_{t\{v_i\{u_j\}\}} : t\{y_i \mapsto v_i\{x_j \mapsto u_j\}\} \Rightarrow t\{y_i \mapsto v_i\{x_j \mapsto u_j\}\} : C}$$

Figure C.14: Invertibility of structural rewrites

$$\frac{\Gamma \vdash \tau : t \Rightarrow t' : A}{\Gamma \vdash \tau \equiv \tau : t \Rightarrow t' : A}\ \text{refl} \qquad \frac{\Gamma \vdash \tau \equiv \tau' : t \Rightarrow t' : A}{\Gamma \vdash \tau' \equiv \tau : t \Rightarrow t' : A}\ \text{symm}$$

$$\frac{\Gamma \vdash \tau' \equiv \tau'' : t \Rightarrow t' : A \qquad \Gamma \vdash \tau \equiv \tau' : t \Rightarrow t' : A}{\Gamma \vdash \tau \equiv \tau'' : t \Rightarrow t' : A}\ \text{trans}$$

$$\frac{\Gamma \vdash \tau' \equiv \sigma' : t' \Rightarrow t'' : A \qquad \Gamma \vdash \tau \equiv \sigma : t \Rightarrow t' : A}{\Gamma \vdash (\tau' \bullet \tau) \equiv (\sigma' \bullet \sigma) : t \Rightarrow t'' : A}$$

$$\frac{x_1 : A_1, \dots, x_n : A_n \vdash \tau \equiv \tau' : t \Rightarrow t' : B \qquad (\Delta \vdash \sigma_i \equiv \sigma_i' : u_i \Rightarrow u_i' : A_i)_{i=1,\dots,n}}{\Delta \vdash \tau\{x_i \mapsto \sigma_i\} \equiv \tau'\{x_i \mapsto \sigma_i'\} : t\{x_i \mapsto u_i\} \Rightarrow t'\{x_i \mapsto u_i'\} : B}$$

Figure C.15: Congruence rules

C.2 The semantic interpretation of $\Lambda^{\times,\to}_{\mathrm{ps}}$

We employ the same notation as Example 5.2.12 (page 154).

Notation C.2.1. For any $A_1, \dots, A_n, B \in \mathcal{B}$ $(n \in \mathbb{N})$ in an fp-bicategory $(\mathcal{B}, \Pi_n(-))$ there exists a canonical equivalence

$$e_{A_\bullet,B} : \prod\nolimits_{n+1}(A_1, \dots, A_n, B) \leftrightarrows \prod\nolimits_2\left(\prod\nolimits_n(A_1, \dots, A_n), B\right) : e^\star_{A_\bullet,B}$$

where $e_{A_\bullet,B} := \langle\langle\pi_1, \dots, \pi_n\rangle, \pi_{n+1}\rangle$ and $e^\star_{A_\bullet,B} := \langle\pi_1 \circ \pi_1, \dots, \pi_n \circ \pi_1, \pi_2\rangle$. We denote the witnessing 2-cells by

$$\begin{aligned} \mathsf{v}_{A_\bullet,B} &: \mathrm{Id}_{\prod_n(A_1,\dots,A_n)\times B} \Rightarrow e_{A_\bullet,B} \circ e^\star_{A_\bullet,B} \\ \mathsf{w}_{A_\bullet,B} &: e^\star_{A_\bullet,B} \circ e_{A_\bullet,B} \Rightarrow \mathrm{Id}_{\prod_{n+1}(A_1,\dots,A_n,B)} \end{aligned}$$

◂

Construction C.2.2 (Semantic interpretation of $\Lambda^{\times,\to}_{\mathrm{ps}}$)**.** For any unary $\Lambda^{\times,\to}_{\mathrm{ps}}$-signature $\mathcal{S}$, cc-bicategory $(\mathcal{B}, \Pi_n(-), \mathrel{=\!\!\rhd})$ and $\Lambda^{\times,\to}_{\mathrm{ps}}$-signature morphism $h : \mathcal{S} \to \mathcal{B}$, the *interpretation* $h[\![-]\!]$ of the syntax of $\Lambda^{\times,\to}_{\mathrm{ps}}(\mathcal{S})$ is defined by induction.
$\boxed{\text{Types.}}$

$$\begin{aligned} h[\![B]\!] &:= hB & &\text{for } B \text{ a base type} \\ h[\![\prod\nolimits_n(A_1, \dots, A_n)]\!] &:= \prod\nolimits_n\big(h[\![A_1]\!], \dots, h[\![A_n]\!]\big) \\ h[\![A \mathrel{=\!\!\rhd} B]\!] &:= (h[\![A]\!] \mathrel{=\!\!\rhd} h[\![B]\!]) \end{aligned}$$

On contexts, we set $h[\![x_1 : A_1, \dots, x_n : A_n]\!] := \prod_n \big(h[\![A_1]\!], \dots, h[\![A_n]\!]\big)$.
$\boxed{\text{Terms.}}$ Let $\Gamma := (x_i : A_i)_{i=1,\dots,n}$ be any context.

$$\begin{aligned} h[\![\Gamma \vdash x_i : A_i]\!] &:= \pi_i \\ h[\![\Gamma \vdash c(x_1, \dots, x_n) : B]\!] &:= h(c) \\ h[\![p : \prod\nolimits_m(B_1, \dots, B_m) \vdash \pi_i(p) : B_i]\!] &:= \pi_i \\ h[\![\Gamma \vdash \mathsf{tup}(t_1, \dots, t_m) : \prod\nolimits_m(B_1, \dots, B_m)]\!] &:= \langle h[\![\Gamma \vdash t_1 : B_1]\!], \dots, h[\![\Gamma \vdash t_m : B_m]\!]\rangle \\ h[\![f : (A \mathrel{=\!\!\rhd} B), x : A \vdash \mathsf{eval}(f, x) : B]\!] &:= \mathrm{eval}_{h[\![A]\!],h[\![B]\!]} \\ h[\![\Gamma \vdash \lambda x.t : B \mathrel{=\!\!\rhd} C]\!] &:= \lambda\big(h[\![\Gamma, x : B \vdash t : C]\!] \circ e^\star_{A_\bullet,B}\big) \\ h[\![\Delta \vdash t\{x_i \mapsto u_i\} : B]\!] &:= h[\![\Gamma \vdash t : B]\!] \circ \langle h[\![\Delta \vdash u_i : A_i]\!]\rangle_i \end{aligned}$$

We omit easily-recovered typing information for the purpose of readability.

$\boxed{\text{Rewrites.}}$ For composition, constants and products the definition is direct:

$$\begin{aligned}
h\llbracket \Gamma \vdash \mathrm{id}_t : t \Rightarrow t : B \rrbracket &:= \mathrm{id}_{h\llbracket t \rrbracket} \\
h\llbracket \Gamma \vdash \tau' \bullet \tau : t \Rightarrow t'' : B \rrbracket &:= h\llbracket \tau' \rrbracket \bullet h\llbracket \tau \rrbracket \\
h\llbracket \Delta \vdash \tau\{x_i \mapsto \sigma_i\} : t\{x_i \mapsto u_i\} \Rightarrow t'\{x_i \mapsto u_i'\} : B \rrbracket &:= h\llbracket \tau \rrbracket \circ \langle h\llbracket \sigma_i \rrbracket \rangle_i \\
h\llbracket \Gamma \vdash \kappa : c(x_\bullet) \Rightarrow c'(x_\bullet) : B \rrbracket &:= h(\kappa) \\
h\left\llbracket \Gamma \vdash \varpi^{(k)}_{t_1, \dots, t_m} : \pi_k\{\mathsf{tup}(t_1, \dots, t_m)\} \Rightarrow t_k : B_k \right\rrbracket &:= \varpi^{(k)}_{h\llbracket t_1 \rrbracket, \dots, h\llbracket t_m \rrbracket} \\
h\llbracket \Gamma \vdash \mathsf{p}^\dagger(\alpha_1, \dots, \alpha_m) : u \Rightarrow \mathsf{tup}(t_1, \dots, t_m) : \textstyle\prod_m(B_1, \dots, B_m) \rrbracket &:= \mathsf{p}^\dagger(h\llbracket \alpha_1 \rrbracket, \dots, h\llbracket \alpha_m \rrbracket)
\end{aligned}$$

The structural rewrites are interpreted by composites of structural isomorphisms. For $\varrho^{(k)}$ and ι one has:

$$h\llbracket \varrho^{(k)}_{u_1, \dots, u_n} \rrbracket := \pi_k \circ \langle h\llbracket u_i \rrbracket \rangle_i \xRightarrow{\varpi^{(k)}_{h\llbracket u_\bullet \rrbracket}} h\llbracket u_k \rrbracket$$

$$h\llbracket \iota_t \rrbracket := h\llbracket t \rrbracket \overset{\cong}{\Rightarrow} h\llbracket t \rrbracket \circ \mathrm{Id}_{h\llbracket \Gamma \rrbracket} \xRightarrow{h\llbracket t \rrbracket \circ \varsigma_{\mathrm{Id}}} h\llbracket t \rrbracket \circ \langle \pi_\bullet \circ h\llbracket \Gamma \rrbracket \rangle \overset{\cong}{\Rightarrow} h\llbracket t \rrbracket \circ \langle \pi_\bullet \rangle$$

For assoc one has

$$\begin{tikzcd}
h\llbracket t\{u_i\}\{v_j\} \rrbracket \arrow[rr, "h\llbracket \mathsf{assoc}_{t, u_\bullet, v_\bullet} \rrbracket"] \arrow[d, equal] & & h\llbracket t\{u_i\{v_\bullet\}\} \rrbracket \arrow[d, equal] \\
(h\llbracket t \rrbracket \circ \langle h\llbracket u_i \rrbracket \rangle_i) \circ \langle h\llbracket v_j \rrbracket \rangle_j \arrow[r, "\cong"'] & h\llbracket t \rrbracket \circ \big(\langle h\llbracket u_i \rrbracket \rangle_i \circ \langle h\llbracket v_j \rrbracket \rangle_j \big) \arrow[r, "h\llbracket t \rrbracket \circ \mathsf{post}"'] & h\llbracket t \rrbracket \circ \langle h\llbracket u_i \rrbracket_i \circ \langle h\llbracket v_\bullet \rrbracket \rangle \rangle_i
\end{tikzcd}$$

Finally we come to the exponential rewrites ε_t and $\mathsf{e}^\dagger(x \,.\, \alpha)$. Suppose that $\Gamma \vdash u : B \Rightarrow C$. Then

$$\begin{aligned}
h\llbracket \Gamma, x : B \vdash \mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} : C \rrbracket &= \mathrm{eval}_{h\llbracket B \rrbracket, h\llbracket C \rrbracket} \circ \langle h\llbracket \Gamma, x : B \vdash u\{\mathrm{inc}_x\} : B \Rightarrow C \rrbracket, \pi_{n+1} \rangle \\
&= \mathrm{eval}_{h\llbracket B \rrbracket, h\llbracket C \rrbracket} \circ \langle h\llbracket \Gamma \vdash u : B \Rightarrow C \rrbracket \circ \langle \pi_1, \dots, \pi_n \rangle, \pi_{n+1} \rangle
\end{aligned}$$

The interpretation $h\llbracket \Gamma, x : B \vdash \varepsilon_t : \mathsf{eval}\{(\lambda x.t)\{\mathrm{inc}_x\}, x\} \Rightarrow t : C \rrbracket$ is the following composite, in which we abbreviate $h\llbracket\Gamma, x : B \vdash t : C\rrbracket$ by $h\llbracket t\rrbracket^{\Gamma,x:B}$:

$$\begin{tikzcd}
\mathrm{eval}_{h\llbracket B\rrbracket,h\llbracket C\rrbracket} \circ \left\langle \lambda(h\llbracket t\rrbracket^{\Gamma,x:B} \circ e^{\star}_{h\llbracket A_\bullet\rrbracket,h\llbracket B\rrbracket}) \circ \langle \pi_1, \ldots, \pi_n\rangle, \pi_{n+1} \right\rangle \arrow[r] \arrow[dd, "\cong"'] & h\llbracket t\rrbracket^{\Gamma,x:B} \\
& h\llbracket t\rrbracket^{\Gamma,x:B} \circ \mathrm{Id}_{\prod(h\llbracket A_\bullet\rrbracket)\times h\llbracket B\rrbracket} \arrow[u, "\cong"'] \\
\mathrm{eval}_{h\llbracket B\rrbracket,h\llbracket C\rrbracket} \circ \left\langle \lambda(h\llbracket t\rrbracket^{\Gamma,x:A} \circ e^{\star}_{h\llbracket A_\bullet\rrbracket,h\llbracket B\rrbracket}) \circ \langle \pi_1, \ldots, \pi_n\rangle, \mathrm{Id}_{h\llbracket B\rrbracket} \circ \pi_{n+1} \right\rangle \arrow[dd, "\mathsf{eval}\circ\mathsf{fuse}^{-1}"'] & \\
& h\llbracket t\rrbracket^{\Gamma,x:B} \circ \left(e^{\star}_{h\llbracket A_\bullet\rrbracket,h\llbracket B\rrbracket} \circ e_{h\llbracket A_\bullet\rrbracket,h\llbracket B\rrbracket} \right) \arrow[uu, "h\llbracket t\rrbracket^{\Gamma,x:B}\circ \mathsf{w}_{h\llbracket A_\bullet\rrbracket,h\llbracket B\rrbracket}"] \\
\mathrm{eval}_{h\llbracket B\rrbracket,h\llbracket C\rrbracket} \circ \left(\left(\lambda(h\llbracket t\rrbracket^{\Gamma,x:B} \circ e^{\star}_{h\llbracket A_\bullet\rrbracket,h\llbracket B\rrbracket}) \times h\llbracket B\rrbracket \right) \circ e_{h\llbracket A_\bullet\rrbracket,h\llbracket B\rrbracket} \right) \arrow[dd, "\cong"'] & \\
& \left(h\llbracket t\rrbracket^{\Gamma,x:B} \circ e^{\star}_{h\llbracket A_\bullet\rrbracket,h\llbracket B\rrbracket} \right) \circ e_{h\llbracket A_\bullet\rrbracket,h\llbracket B\rrbracket} \arrow[uu, "\cong"] \\
\left(\mathrm{eval}_{h\llbracket B\rrbracket,h\llbracket C\rrbracket} \circ \left(\lambda(h\llbracket t\rrbracket^{\Gamma,x:B} \circ e^{\star}_{h\llbracket A_\bullet\rrbracket,h\llbracket B\rrbracket}) \times h\llbracket B\rrbracket \right) \right) \circ e_{h\llbracket A_\bullet\rrbracket,h\llbracket B\rrbracket} \arrow[ur, bend right, "\varepsilon_{(h\llbracket t\rrbracket\circ e^{\star})}\circ e"'] &
\end{tikzcd}$$

On the other hand, for a judgement $(\Gamma, x : B \vdash \alpha : \mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} \Rightarrow t : C)$, the interpretation of α has type

$$\mathrm{eval}_{h\llbracket B\rrbracket,h\llbracket C\rrbracket} \circ \langle h\llbracket\Gamma \vdash u : B \Rightarrow C\rrbracket \circ \langle \pi_1, \ldots, \pi_n\rangle, \pi_{n+1}\rangle \Rightarrow h\llbracket\Gamma, x : B \vdash t : C\rrbracket \tag{C.1}$$

To interpret $(\Gamma \vdash \mathsf{e}^{\dagger}(x \,.\, \alpha) : u \Rightarrow \lambda x.t : A \Rightarrow B)$ using the universal property of exponentials, we distort (C.1) into a composite $h\llbracket\alpha\rrbracket^{\circ}$ as in the diagram below. We suppress the subscripts on $e_{A_\bullet,B}$

and $e^{\star}_{A_{\bullet},B}$ to fit the diagram better onto the page.

$$\begin{tikzcd}
\mathrm{eval}_{h\llbracket B\rrbracket,h\llbracket C\rrbracket} \circ (h\llbracket u\rrbracket^{\Gamma} \times h\llbracket B\rrbracket) \arrow[r, "h\llbracket\alpha\rrbracket^{\circ}"] \arrow[d, "\cong"'] & h\llbracket t\rrbracket^{\Gamma,x:B} \circ e^{\star} \\
\big(\mathrm{eval}_{h\llbracket B\rrbracket,h\llbracket C\rrbracket} \circ (h\llbracket u\rrbracket^{\Gamma} \times h\llbracket B\rrbracket)\big) \circ \mathrm{Id}_{\prod_2((\prod_n h\llbracket A_{\bullet}\rrbracket),h\llbracket B\rrbracket)} \arrow[d, "\mathrm{eval}\circ(h\llbracket u\rrbracket^{\Gamma}\times h\llbracket B\rrbracket)\circ\mathsf{v}_{\prod_2((\prod_n h\llbracket A_{\bullet}\rrbracket),h\llbracket B\rrbracket))}"'] & \\
\big(\mathrm{eval}_{h\llbracket B\rrbracket,h\llbracket C\rrbracket} \circ (h\llbracket u\rrbracket^{\Gamma} \times h\llbracket B\rrbracket)\big) \circ (e \circ e^{\star}) \arrow[d, "\cong"'] & \\
\big(\mathrm{eval}_{h\llbracket B\rrbracket,h\llbracket C\rrbracket} \circ \big((h\llbracket u\rrbracket^{\Gamma} \times h\llbracket B\rrbracket)\big) \circ e\big) \circ e^{\star} \arrow[d, "\mathsf{eval}\circ\mathsf{fuse}\circ e^{\star}"'] & \\
\big(\mathrm{eval}_{h\llbracket B\rrbracket,h\llbracket C\rrbracket} \circ \big\langle h\llbracket u\rrbracket^{\Gamma} \circ \langle \pi_1, \dots, \pi_n\rangle, \mathrm{Id}_{h\llbracket B\rrbracket} \circ \pi_{n+1}\big\rangle\big) \circ e^{\star} \arrow[r, "\cong"'] & \big(\mathrm{eval}_{h\llbracket B\rrbracket,h\llbracket C\rrbracket} \circ \big\langle h\llbracket u\rrbracket^{\Gamma} \circ \langle \pi_{\bullet}\rangle, \pi_{n+1}\big\rangle\big) \circ e^{\star} \arrow[uuuu, "h\llbracket\alpha\rrbracket^{\Gamma,x:B}\circ e^{\star}"']
\end{tikzcd}$$

The unlabelled arrow is $\mathrm{eval}_{h\llbracket B\rrbracket,h\llbracket C\rrbracket} \circ \langle h\llbracket u\rrbracket^{\Gamma} \circ \varpi^{(1)}, \mathrm{Id}_{h\llbracket B\rrbracket} \circ \varpi^{(2)}\rangle \circ e^{\star}_{h\llbracket A_{\bullet}\rrbracket,h\llbracket B\rrbracket}$. Finally, then, one has

$$h\llbracket \Gamma \vdash \mathsf{e}^{\dagger}(x \,.\, \alpha) : u \Rightarrow \lambda x.t : B \Rightarrow C\rrbracket := \mathsf{e}^{\dagger}(h\llbracket \Gamma, x : B \vdash \alpha : \mathsf{eval}\{u\{\mathrm{inc}_x\}, x\} \Rightarrow t : C\rrbracket^{\circ})$$

◂

Appendix D

The universal property of a bicategorical pullback

Recall the following definition of a pullback (Definition 7.3.5 on page 237).

Definition D.1. Let C (for 'cospan') denote the category $(1 \xrightarrow{h_1} 0 \xleftarrow{h_2} 2)$ and $\mathcal{B}$ be any bicategory. A *pullback* of the cospan $(X_1 \xrightarrow{f_1} X_0 \xleftarrow{f_2} X_2)$ in $\mathcal{B}$ is a bilimit for the strict pseudofunctor $\mathrm{C} \to \mathcal{B}$ determined by this cospan. ◂

We translate this into a presentation closer to that for categorical pullbacks—namely, that given by Lemma 7.3.6 (page 237)—by showing that, for any $F : \mathrm{C} \to \mathcal{B}$, there exists an equivalence of categories $\mathrm{Hom}(\mathrm{C}, \mathcal{B})(\Delta B, F) \simeq B/F$, where each category B/F consists of iso-commuting squares and fill-ins.

Definition D.2. Let $\mathcal{B}$ be any bicategory, $B \in \mathcal{B}$ and $F : \mathrm{C} \to \mathcal{B}$ be a pseudofunctor. The category B/F has objects triples $(\gamma_1, \gamma_2, \overline{\gamma})$, where $\gamma_i : B \to Fi$ $(i = 1, 2)$ and $\overline{\gamma}$ is an invertible 2-cell as in the diagram

$$\begin{tikzcd}
& B \arrow[dl, "\gamma_1"'] \arrow[dr, "\gamma_2"] & \\
F1 \arrow[dr, "Fh_1"'] & \overset{\overline{\gamma}}{\Leftarrow} & F2 \arrow[dl, "Fh_2"] \\
& F0 &
\end{tikzcd}$$

Morphisms $(\gamma_1, \gamma_2, \overline{\gamma}) \to (\delta_1, \delta_2, \overline{\delta})$ are pairs of 2-cells $\Xi_i : \gamma_i \Rightarrow \delta_i$ $(i = 1, 2)$ such that

$$\begin{tikzcd}
F(h_2) \circ \gamma_2 \arrow[r, "F(h_2) \circ \Xi_2"] \arrow[d, "\overline{\gamma}"'] & F(h_2) \circ \delta_2 \arrow[d, "\overline{\delta}"] \\
F(h_1) \circ \gamma_1 \arrow[r, "F(h_1) \circ \Xi_1"'] & F(h_1) \circ \delta_1
\end{tikzcd}$$

The identity on $(\gamma_1, \gamma_2, \overline{\gamma})$ is $(\mathrm{id}_{\gamma_1}, \mathrm{id}_{\gamma_2})$ and composition is as in $\mathcal{B}$. ◂

The next lemma provides the components of the required equivalence.

Lemma D.3. Let $\mathcal{B}$ be a bicategory, C be the category $(1 \xrightarrow{h_1} 0 \xleftarrow{h_2} 2)$, and $F : \mathrm{C} \to \mathcal{B}$ a pseudofunctor. Then, for any $B \in \mathcal{B}$ there exists an equivalence of categories $\mathrm{Hom}(\mathrm{C}, \mathcal{B})(\Delta B, F) \simeq B/F$, where $\Delta : \mathcal{B} \to \mathrm{Hom}(\mathrm{C}, \mathcal{B})$ denotes the diagonal pseudofunctor.

Proof. We begin by defining functors $K : \mathrm{Hom}(\mathrm{C}, \mathcal{B})(\Delta B, F) \leftrightarrows B/F : L$. Take K first. For a pseudonatural transformation $(\mathsf{k}, \overline{\mathsf{k}}) : \Delta B \Rightarrow F$ with components as in the square

$$\begin{tikzcd}
B \arrow[r, "\mathrm{Id}_B"] \arrow[d, "\mathsf{k}_i"'] \arrow[dr, phantom, "\overset{\overline{\mathsf{k}}_i}{\Leftarrow}"] & B \arrow[d, "\mathsf{k}_0"] \\
Fi \arrow[r, "Fh_i"'] & F0
\end{tikzcd}$$

we define $K(\mathsf{k}, \overline{\mathsf{k}}) := (\mathsf{k}_1, \mathsf{k}_2, \overline{\gamma}_{(\mathsf{k}, \overline{\mathsf{k}})})$, where

$$\gamma_{(\mathsf{k}, \overline{\mathsf{k}})} := F(h_2) \circ \mathsf{k}_2 \xRightarrow{\overline{\mathsf{k}}_2^{-1}} \mathsf{k}_0 \circ \mathrm{Id}_B \xRightarrow{\overline{\mathsf{k}}_1} F(h_1) \circ \mathsf{k}_1 \tag{D.1}$$

For morphisms, suppose $\Xi : (\mathsf{k}, \overline{\mathsf{k}}) \to (\mathsf{j}, \bar{\mathsf{j}})$ is a modification. One thereby obtains 2-cells $\Xi_i : \mathsf{k}_i \Rightarrow \mathsf{j}_i$ $(i = 1, 2)$, and

$$\begin{tikzcd}
F(h_2) \circ \mathsf{k}_2 \arrow[r, "F(h_2) \circ \Xi_2"] \arrow[d, "\overline{\mathsf{k}}_2^{-1}"'] \arrow[dd, bend right=60, "\overline{\gamma}_{(\mathsf{k}, \overline{\mathsf{k}})}"'] \arrow[dr, phantom, "\underset{=}{\text{modif. law}}"] & F(h_2) \circ \mathsf{j}_2 \arrow[d, "\bar{\mathsf{j}}_2^{-1}"] \arrow[dd, bend left=60, "\overline{\gamma}_{(\mathsf{j}, \bar{\mathsf{j}})}"] \\
\mathsf{k}_0 \circ \mathrm{Id}_B \arrow[r, "\Xi_0 \circ \mathrm{Id}_B"'] \arrow[d, "\overline{\mathsf{k}}_1"'] \arrow[dr, phantom, "\underset{=}{\text{modif. law}}"] & \mathsf{j}_0 \circ \mathrm{Id}_B \arrow[d, "\bar{\mathsf{j}}_1"] \\
F(h_1) \circ \mathsf{k}_1 \arrow[r, "F(h_1) \circ \Xi_1"'] & F(h_1) \circ \mathsf{j}_1
\end{tikzcd}$$

So we may define $K(\Xi) := (\Xi_1, \Xi_2)$.

Going the other way, for a triple $(\gamma_1, \gamma_2, \overline{\gamma})$ we define $L(\gamma_1, \gamma_2, \overline{\gamma})$ to be the pseudonatural transformation with components

$$\mathsf{j}_i := B \xrightarrow{\gamma_i} Fi \qquad \text{for } i = 1, 2$$
$$\mathsf{j}_0 := B \xrightarrow{\gamma_2} F2 \xrightarrow{Fh_2} F0$$

and witnessing 2-cells

$$\begin{tikzcd}
B \arrow[r, "\mathrm{Id}_B"] \arrow[d, "\mathsf{j}_i"'] \arrow[dr, phantom, "\cong"] & B \arrow[d, "\mathsf{j}_i"] \\
Fi \arrow[r, bend left, "\mathrm{Id}_{Fi}"] \arrow[r, bend right, "F\mathrm{Id}_i"'] \arrow[r, phantom, "\overset{\psi^F}{\cong}"] & Fi
\end{tikzcd}
\qquad
\begin{tikzcd}
B \arrow[r, "\mathrm{Id}_B"] \arrow[dd, "\gamma_1"'] \arrow[dr, phantom, "\cong"] & B \arrow[d, "\gamma_2"] \arrow[ddl, bend right, "\gamma_1"'] \arrow[dd, bend left=60, "Fh_2\circ\gamma_2"] \\
 & F2 \arrow[d, "Fh_2"] \arrow[dl, phantom, "\overset{\overline{\gamma}}{\Leftarrow}"] \\
F1 \arrow[r, "Fh_1"'] & F0
\end{tikzcd}
\qquad
\begin{tikzcd}
B \arrow[r, "\mathrm{Id}_B"] \arrow[dd, "\gamma_2"'] \arrow[ddr, phantom, "\cong"] & B \arrow[d, "\gamma_2"] \arrow[dd, bend left=60, "Fh_2\circ\gamma_2"] \\
 & F2 \arrow[d, "Fh_2"] \\
F2 \arrow[r, "Fh_1"'] & F0
\end{tikzcd}$$

The naturality condition is trivial—there are no non-identity 2-cells in C—and the unit law holds by definition, so the only thing to check is the associativity law. For this one must verify the axiom for each of the possible composites in C, namely $\mathrm{Id}_i \circ \mathrm{Id}_i$, $\mathrm{Id}_0 \circ h_i$, and $h_i \circ \mathrm{Id}_i$. This is a long exercise.

On morphisms, for any (Ψ_1, Ψ_2) in B/F, we define $L(\Psi_1, \Psi_2)$ to be the modification with components

$$\Psi_i := \mathsf{k}_i \overset{\Psi_i}{\Longrightarrow} \mathsf{j}_i \qquad\qquad (i = 1, 2)$$
$$\Psi_0 := F(h_2) \circ \mathsf{k}_2 \xRightarrow{F(h_2)\circ\Psi_2} F(h_2) \circ \mathsf{j}_2$$

The only thing to check is the modification axiom, which we need to verify for the maps h_1, h_2 and $\mathrm{Id}_0, \mathrm{Id}_1, \mathrm{Id}_2$. Each of these is a simple calculation.

It remains to show that K and L form an equivalence. The composite $K \circ L$ is the identity. On the other hand, $LK(\mathsf{k}, \overline{\mathsf{k}})$ has components k_i for $i = 1, 2$ and $Fh_2 \circ \mathsf{k}_2$ for $i = 0$. One may then check that setting $\Xi_i^{(\mathsf{k},\overline{\mathsf{k}})} := \mathrm{id}_{\mathsf{k}_i}$ for $i = 1, 2$ and $\Xi_0^{(\mathsf{k},\overline{\mathsf{k}})} := \big(Fh_2 \circ \mathsf{k}_2 \overset{\overline{\mathsf{k}}_2^{-1}}{\Longrightarrow} \mathsf{k}_0 \circ \mathrm{Id}_B \overset{\cong}{\Rightarrow} \mathsf{k}_0\big)$ defines a modification $LK(\mathsf{k}, \overline{\mathsf{k}}) \to (\mathsf{k}, \overline{\mathsf{k}})$. It remains to show that the modifications $\Xi^{(\mathsf{k},\overline{\mathsf{k}})}$ are natural in $(\mathsf{k}, \overline{\mathsf{k}})$. The $i = 1$ and $i = 2$ cases are trivial, and for $i = 0$ one sees that, for any $\Psi : (\mathsf{k}, \overline{\mathsf{k}}) \to (\mathsf{j}, \overline{\mathsf{j}})$,

$$\begin{tikzcd}
KL(\mathsf{k}, \overline{\mathsf{k}})_0 \arrow[r, equal] \arrow[d, "(KL\Psi)_0"'] & Fh_2 \circ \mathsf{k}_2 \arrow[r, "\overline{\mathsf{k}}_2^{-1}"] \arrow[d, "Fh_2\circ\Psi_2"] \arrow[rr, bend left, "\Xi_0^{(\mathsf{k},\overline{\mathsf{k}})}"] & \mathsf{k}_0 \circ \mathrm{Id}_B \arrow[r, "\cong"] & \mathsf{k}_0 \arrow[d, "\Psi_0"] \\
KL(\mathsf{j}, \overline{\mathsf{j}})_0 \arrow[r, equal] & Fh_2 \circ \mathsf{j}_2 \arrow[r, "\overline{\mathsf{j}}_2^{-1}"'] \arrow[rr, bend right, "\Xi_0^{(\mathsf{j},\overline{\mathsf{j}})}"'] & \mathsf{j}_0 \circ \mathrm{Id}_B \arrow[r, "\cong"'] & \mathsf{j}_0
\end{tikzcd}$$

as required. It follows that $L \circ K \cong \mathrm{id}_{\mathrm{Hom}(\mathrm{C},\mathcal{B})(\Delta B, F)}$, which completes the proof. □

The mapping $B \mapsto B/F$ extends to a pseudofunctor as follows. For $f : B' \to B$, we define $f/F : B/F \to B'/F$ by setting $(f/F)(\gamma_1, \gamma_2, \overline{\gamma}) := (\gamma_1 \circ f, \gamma_2 \circ f, \overline{\gamma} \circ f)$. Then for $\alpha : f \Rightarrow f'$, the natural transformation α/F has components $\gamma_i \circ \alpha : \gamma_i \circ f \to \gamma_i \circ f'$. This defines a pseudofunctor with unit and associativity witnessed by structural isomorphisms. In fact this pseudofunctor is equivalent to $\mathrm{Hom}(\mathrm{C},\mathcal{B})(\Delta(-),F)$.

Lemma D.4. Let $\mathcal{B}$ be a bicategory, C be the category $(1 \xrightarrow{h_1} 0 \xleftarrow{h_2} 2)$, and $F : \mathrm{C} \to \mathcal{B}$ a pseudofunctor. Then, writing $K_B : \mathrm{Hom}(\mathrm{C},\mathcal{B})(\Delta B, F) \to B/F$ for the functor constructed in Lemma D.3, the diagram below commutes for any $f : B' \to B$ in $\mathcal{B}$:

$$\begin{tikzcd}
\mathrm{Hom}(\mathrm{C},\mathcal{B})(\Delta B, F) \arrow[r, "\mathrm{Hom}(\mathrm{C},\mathcal{B})(\Delta f, F)"] \arrow[d, "K_B"'] & \mathrm{Hom}(\mathrm{C},\mathcal{B})(\Delta B', F) \arrow[d, "K_{B'}"] \\
B/F \arrow[r, "f/F"'] & B'/F
\end{tikzcd}$$

Proof. For a pseudonatural transformation $(\mathsf{k},\overline{\mathsf{k}}) : \Delta B \Rightarrow F$, $(f/F \circ K_B)(\mathsf{k},\overline{\mathsf{k}})$ is the triple with 1-cells $\mathsf{k}_1 \circ f$ and $\mathsf{k}_2 \circ f$ and 2-cell

$$Fh_2 \circ (\mathsf{k}_2 \circ f) \overset{\cong}{\Rightarrow} (Fh_2 \circ \mathsf{k}_2) \circ f \xRightarrow{\gamma_{(\mathsf{k},\overline{\mathsf{k}})}} (Fh_1 \circ \mathsf{k}_2) \circ f \overset{\cong}{\Rightarrow} Fh_1 \circ (\mathsf{k}_2 \circ f)$$

Here $\gamma_{(\mathsf{k},\overline{\mathsf{k}})}$ is the composite defined in (D.1).

On the other hand, writing $f_* := \mathrm{Hom}(\mathrm{C},\mathcal{B})(\Delta f, F)$, one has that $f_*(\mathsf{k},\overline{\mathsf{k}})$ is the pseudonatural transformation with components $\mathsf{k}_i \circ f$ and witnessing 2-cells given by composing $\overline{\mathsf{k}}$ with the evident structural isomorphism:

$$\begin{tikzcd}
B' \arrow[r, "\mathrm{Id}_{B'}"] \arrow[d, "f"'] \arrow[dr, phantom, "\cong"] & B' \arrow[d, "f"] \\
B \arrow[r, "\mathrm{Id}_B"] \arrow[d, "\mathsf{k}_i"'] \arrow[dr, phantom, "\overset{\overline{\mathsf{k}}_i}{\Leftarrow}"] & B \arrow[d, "\mathsf{k}_i"] \\
Fi \arrow[r, "Fh_i"'] & F0
\end{tikzcd}$$

A short calculation shows that applying $K_{B'}$ to this pseudonatural transformation yields exactly $(f/F \circ K_B)(\mathsf{k},\overline{\mathsf{k}})$. □

It follows that the functors K_B are the components of a pseudonatural transformation. Since each K_B is an equivalence, one obtains the following.

Corollary D.5. Let $\mathcal{B}$ be a bicategory, C be the category $(1 \xrightarrow{h_1} 0 \xleftarrow{h_2} 2)$, and $F : \mathrm{C} \to \mathcal{B}$ a pseudofunctor. Then $\mathrm{Hom}(\mathrm{C},\mathcal{B})(\Delta(-),F) \simeq (-)/F$ in $\mathrm{Hom}(\mathcal{B}^{\mathrm{op}}, \mathbf{Cat})$. □

We can now use the fact that biequivalences preserve biuniversal arrows to rephrase the universal property of a bicategorical pullback. For any bicategory $\mathcal{B}$, let $(X_1 \xrightarrow{f_1} X_0 \xleftarrow{f_2} X_2)$ be any cospan and let F be the strict pseudofunctor $\mathrm{C} \to \mathcal{B}$ it determines. The pullback of this cospan, when it exists, is a biuniversal arrow $(P, \lambda : \Delta P \Rightarrow F)$ consisting of an object $P \in \mathcal{B}$ and a pseudonatural transformation $\lambda : \Delta P \Rightarrow F$. The universal property then requires that, for any other pseudonatural transformation $\gamma : \Delta Q \Rightarrow F$ there exists a 1-cell $u : Q \to P$ and a universal modification $\varepsilon : \lambda \circ \Delta u \Rightarrow \gamma$, such that both the unit and the counit ε are invertible.

We pass this data through the equivalence K. The pseudonatural transformations λ and γ become iso-commuting squares:

$$\begin{tikzcd}
 & P \arrow[dl, "\lambda_1"'] \arrow[dr, "\lambda_2"] & & & & Q \arrow[dl, "\gamma_1"'] \arrow[dr, "\gamma_2"] & \\
F1 \arrow[dr, "Fh_1"'] & \overset{\overline{\lambda}}{\Leftarrow} & F2 \arrow[dl, "Fh_2"] & & F1 \arrow[dr, "Fh_1"'] & \overset{\overline{\gamma}}{\Leftarrow} & F2 \arrow[dl, "Fh_2"] \\
 & F0 & & & & F0 &
\end{tikzcd}$$

The pseudonatural transformation $\lambda \circ \Delta u$ then becomes

$$\begin{tikzcd}
 & Q \arrow[d, dashed, "u"] \arrow[ddl, bend right, "\lambda_1 \circ u"'] \arrow[ddr, bend left, "\lambda_2 \circ u"] & \\
 & P \arrow[dl, "\lambda_1"'] \arrow[dr, "\lambda_2"] & \\
F1 \arrow[dr, "Fh_1"'] & \overset{\overline{\lambda}}{\Leftarrow} & F2 \arrow[dl, "Fh_2"] \\
 & F0 &
\end{tikzcd}$$

and the counit ε becomes a pair of 2-cells $\varepsilon_i : \lambda_i \circ u \Rightarrow \gamma_i$ which is universal among 2-cells satisfying the following:

$$\begin{tikzcd}
Fh_2 \circ (\lambda_2 \circ u) \arrow[r, "Fh_2 \circ \varepsilon_2"] \arrow[d, "\cong"'] & Fh_2 \circ \gamma_2 \arrow[ddd, "\overline{\gamma}"] \\
(Fh_2 \circ \lambda_2) \circ u \arrow[d, "\overline{\lambda} \circ u"'] & \\
(Fh_1 \circ \lambda_1) \circ u \arrow[d, "\cong"'] & \\
Fh_1 \circ (\lambda_1 \circ u) \arrow[r, "Fh_1 \circ \varepsilon_1"'] & Fh_1 \circ \gamma_1
\end{tikzcd}$$

Starting this diagram from $(Fh_2 \circ \lambda_2) \circ u$ and inverting the isomorphisms, one obtains the fill-in requirement from Lemma 7.3.6. One may now see that the remaining conditions of Lemma 7.3.6 are exactly those making ε universal.

Index of notation

With typing signature and page of first definition

$\mathsf{c}^{\Rightarrow}_{A,B}$ A 2-cell $\mathrm{q}^{\Rightarrow}_{A,B} \circ \mathrm{m}_{A,B} \Rightarrow \mathrm{Id}_{F(A \Rightarrow B)}$, part of the data of a cc-pseudofunctor $(F, \mathrm{q}^{\times}, \mathrm{q}^{\Rightarrow})$, page 144

$\mathsf{c}^{\times}_{A_\bullet}$ A 2-cell $\mathrm{q}^{\times}_{A_\bullet} \circ \langle F\pi_1, \ldots, F\pi_n \rangle \Rightarrow \mathrm{Id}_{(F \prod_i A_i)}$, part of the data of an fp-pseudofunctor $(F, \mathrm{q}^{\times})$, page 83

ε_t The counit for exponential structure, of type $\mathrm{eval}_{A,B} \circ (\lambda t \times A) \overset{\cong}{\Rrightarrow} t$, page 142

$\varpi^{(k)}_{t_1, \ldots, t_n}$ The kth component of the counit for product structure, of type $\pi_k \circ \langle t_\bullet \rangle \overset{\cong}{\Rrightarrow} t_k$, page 78

η_t The unit for exponential structure, of type $t \overset{\cong}{\Rrightarrow} \lambda\,(\mathrm{eval}_{A,B} \circ (t \times A))$, page 142

ς_t The unit for product structure, of type $t \overset{\cong}{\Rrightarrow} \langle \pi_1 \circ t, \ldots, \pi_n \circ t \rangle$, page 78

$\mathrm{m}_{A,B}$ The canonical map $F(A \Rightarrow B) \to (FA \Rightarrow FB)$ for an fp-pseudofunctor $(F, \mathrm{q}^{\times})$, defined as the transpose of $F(\mathrm{eval}_{A,B}) \circ \mathrm{q}^{\times}_{A \Rightarrow B, A}$, page 144

$\mathrm{q}^{\Rightarrow}_{A,B}$ An equivalence $(FA \Rightarrow FB) \to F\,(A \Rightarrow B)$ forming part of the data of a cc-pseudofunctor, page 144

$\mathsf{fuse}(h_\bullet; g_\bullet)$ The canonical 2-cell $(\prod_{i=1}^{n} h_i) \circ \langle g_1, \ldots, g_n \rangle \Rightarrow \langle h_1 \circ g_1, \ldots, h_n \circ g_n \rangle$, page 80

$\mathsf{f}_{h; f_\bullet; g_\bullet}$ The canonical 2-cell $\mathsf{f}_{h; f_\bullet; g_\bullet} : h[f_1 \times \cdots \times f_n]\,[g_1, \ldots, g_n] \Rightarrow h[f_1[g_1], \ldots, f_n[g_n]]$ in a biclone, page 49

$\mathsf{nat}_{f_\bullet}$ The 2-cells $\mathrm{q}^{\times}_{A_\bullet} \circ \prod_{i=1}^{n} Ff_i \Rightarrow F(\prod_{i=1}^{n} f_i) \circ \mathrm{q}^{\times}_{A_\bullet}$ witnessing that $\prod_{i=1}^{n} (F(-), \ldots, F(=)) \simeq (F \circ \prod_{i=1}^{n})\,(-, \ldots, =)$ for every fp-pseudofunctor $(F, \mathrm{q}^{\times})$, page 83

$\Phi_{h_\bullet, g_\bullet}$ The canonical 2-cell $\left(\prod_{i=1}^{n} h_i\right) \circ \left(\prod_{i=1}^{n} g_i\right) \Rightarrow \prod_{i=1}^{n} (h_i g_i)$ witnessing the pseudofunctorality of $\prod_n(-, \ldots, =)$, page 80

$\mathsf{post}(h_\bullet; g)$ The canonical 2-cell $\langle h_1, \ldots, h_n\rangle \circ g \Rightarrow \langle h_1 \circ g, \ldots, h_n \circ g\rangle$, page 80

$\mathrm{q}^{\times}_{A_\bullet}$ An equivalence $\prod_{i=1}^{n}(FA_i) \to F\left(\prod_{i=1}^{n} A_i\right)$ forming part of the data of an fp-pseudofunctor, page 82

$\mathsf{push}(f, g)$ The canonical 2-cell $\lambda(f) \circ g \Rightarrow \lambda\big(f \circ (g \times A)\big)$, page 143

$\mathsf{swap}_{h,f}$ The 2-cell of type $(f \times X) \circ \langle \mathrm{Id}_B, hf\rangle \Rightarrow \langle \mathrm{Id}_{B'}, h\rangle \circ f$, defined as the composite $(f \times X) \circ \langle \mathrm{Id}_B, hf\rangle \xRightarrow{\mathsf{fuse}} \langle f, hf\rangle \xRightarrow{\mathsf{post}^{-1}} \langle \mathrm{Id}_{B'}, h\rangle \circ f$, page 219

$\mathsf{e}^{\dagger}(\alpha)$ The unique mediating 2-cell $u \Rightarrow \lambda t$ corresponding to $\alpha : \mathrm{eval}_{A,B} \circ (u \times A) \Rightarrow t$, page 142

$\mathsf{p}^{\dagger}(\alpha_1, \ldots, \alpha_n)$ The unique mediating 2-cell $u \Rightarrow \langle t_1, \ldots, t_n\rangle$ corresponding to $\alpha_i : \pi_i \circ u \Rightarrow t_i$ $(i = 1, \ldots, n)$, page 78

$\mathsf{u}^{\mathbin{=\!\!\rhd}}_{A,B}$ A 2-cell $\mathrm{Id}_{(FA \mathbin{=\!\!\rhd} FB)} \Rightarrow \mathrm{m}_{A,B} \circ \mathrm{q}^{\mathbin{=\!\!\rhd}}_{A,B}$, part of the data of a cc-pseudofunctor $(F, \mathrm{q}^{\times}, \mathrm{q}^{\mathbin{=\!\!\rhd}})$, page 144

$\mathsf{unpack}_{f_\bullet}$ The 2-cell $\langle F\pi_1, \ldots, F\pi_n\rangle \circ F\langle f_1, \ldots, f_n\rangle \Rightarrow \langle Ff_1, \ldots, f_n\rangle$ ‘unpacking’ an n-ary tupling, page 84

$\mathsf{u}^{\times}_{A_\bullet}$ A 2-cell $\mathrm{Id}_{(\prod_i FA_i)} \Rightarrow \langle F\pi_1, \ldots, F\pi_n\rangle \circ \mathrm{q}^{\times}_{A_\bullet}$, part of the data of an fp-pseudofunctor $(F, \mathrm{q}^{\times})$, page 83

Bibliography

[Abb03] M. G. Abbott. *Categories of containers*. PhD thesis, University of Leicester, 2003.

[ACCL90] M. Abadi, L. Cardelli, P.-L. Curien, and J.-J. Levy. Explicit substitutions. In *Proceedings of the 17th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, POPL '90, pages 31–46, New York, NY, USA, 1990. ACM.

[ACD07] A. Abel, T. Coquand, and P. Dybjer. Normalization by evaluation for Martin-Löf type theory with typed equality judgements. In *Proceedings of the 22nd Annual IEEE Symposium on Logic in Computer Science*, LICS '07, pages 3–12, Washington, DC, USA, 2007. IEEE Computer Society.

[ADHS01] T. Altenkirch, P. Dybjer, M. Hofmannz, and P. Scott. Normalization by evaluation for typed lambda calculus with coproducts. In *Proceedings of the 16th Annual IEEE Symposium on Logic in Computer Science*, LICS '01, pages 303–, Washington, DC, USA, 2001. IEEE Computer Society.

[Agd] Agda contributors. The Agda proof assistant. `https://wiki.portal.chalmers.se/agda/pmwiki.php`.

[AHS95] T. Altenkirch, M. Hofmann, and T. Streicher. Categorical reconstruction of a reduction free normalization proof. In *Category Theory and Computer Science, 6th International Conference, CTCS '95, Cambridge, UK, August 7–11, 1995, Proceedings*, volume 953, pages 182–199, August 1995.

[AK16] T. Altenkirch and A. Kaposi. Normalisation by Evaluation for Dependent Types. In D. Kesner and B. Pientka, editors, *1st International Conference on Formal Structures for Computation and Deduction (FSCD 2016)*, volume 52 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 6:1–6:16, Dagstuhl, Germany, 2016. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[AK17] T. Altenkirch and A. Kaposi. Normalisation by Evaluation for Type Theory, in Type Theory. *Logical Methods in Computer Science*, Volume 13, Issue 4, October 2017.

[Ali95] M. Alimohamed. A characterization of lambda definability in categorical models of implicit polymorphism. *Theor. Comput. Sci.*, 146(1-2):5–23, July 1995.

[Awo10] S. Awodey. *Category Theory.* Number 52 in Oxford Logic Guides. Oxford University Press, 2nd edition, 2010.

[Bak] I. Baković. Bicategorical Yoneda lemma. Available at `https://www2.irb.hr/korisnici/ibakovic/yoneda.pdf`.

[Bar85] H. P. Barendregt. *The Lambda Calculus: Its Syntax and Semantics*, volume 103 of *Studies in Logic and the Foundations of Mathematics).* North-Holland, 1985. Revised edition.

[BBdPH93] N. Benton, G. Bierman, V. de Paiva, and M. Hyland. A term calculus for intuitionistic linear logic. In *Lecture Notes in Computer Science*, pages 75–90. Springer Berlin Heidelberg, 1993.

[BÉLM01] S. L. Bloom, Z. Ésik, A. Labella, and E. G. Manes. Iteration 2-theories. *Applied Categorical Structures*, 9(2):173–216, March 2001.

[Bén67] J. Bénabou. Introduction to bicategories. In *Reports of the Midwest Category Seminar*, pages 1–77, Berlin, Heidelberg, 1967. Springer Berlin Heidelberg.

[Bén85] J. Bénabou. Fibered categories and the foundations of naive category theory. *Journal of Symbolic Logic*, 50(1):10–37, 1985.

[Bén00] J. Bénabou. Distributors at work. Notes from a course given by the author in June 2000 at TU Darmstadt, 2000.

[BES98] U. E. Berger, M. Eberl, and H. Schwichtenberg. Normalization by evaluation. In B. Möller and J. V. Tucker, editors, *Prospects for Hardware Foundations: ESPRIT Working Group 8533 NADA — New Hardware Design Methods Survey Chapters*, pages 117–137. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998.

[BKP89] R. Blackwell, G. M. Kelly, and A. J. Power. Two-dimensional monad theory. *Journal of Pure and Applied Algebra*, 59(1):1–41, 1989.

[BKPS89] G. J. Bird, G. M. Kelly, A. J. Power, and R. Street. Flexible limits for 2-categories. *Journal of Pure and Applied Algebra*, 61(1):1–27, nov 1989.

[BKV18] K. Bar, A. Kissinger, and J. Vicary. Globular: an online proof assistant for higher-dimensional rewriting. *Logical Methods in Computer Science*, Volume 14, Issue 1, January 2018.

[BN98] F. Baader and T. Nipkow. *Term Rewriting and All That.* Cambridge University Press, 1998.

[Bor94] F. Borceux. *Bicategories and distributors*, volume 1 of *Encyclopedia of Mathematics and its Applications*, page 281–324. Cambridge University Press, 1994.

[BS91] U. Berger and H. Schwichtenberg. An inverse to the evaluation functional for typed λ-calculus. *Logic in Computer Science*, pages 203–211, 1991.

[CCRW17] S. Castellan, P. Clairambault, S. Rideau, and G. Winskel. Games and strategies as event structures. *Logical Methods in Computer Science*, 13, 2017.

[CD97] T. Coquand and P. Dybjer. Intuitionistic model constructions and normalization proofs. *Mathematical. Structures in Comp. Sci.*, 7(1):75–94, February 1997.

[CD98] D. Cubric and P. Dybjer. Normalization and the Yoneda embedding. *Mathematical Structures in Computer Science*, 1998.

[CD14] P. Clairambault and P. Dybjer. The biequivalence of locally cartesian closed categories and Martin-Löf type theories. *Mathematical Structures in Computer Science*, 24(6), 2014.

[CFW98] G. L. Cattani, M. Fiore, and G. Winskel. A theory of recursive domains with applications to concurrency. In *Proceedings of the Thirteenth Annual IEEE Symposium on Logic in Computer Science (LICS 1998)*, pages 214–225. IEEE Computer Society Press, June 1998.

[CHTM19] P.-L. Curien, C. Ho Thanh, and S. Mimram. A sequent calculus for opetopes. In *Proceedings of the Thirty-Fourth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, 2019.

[CKWW08] A. Carboni, G. M. Kelly, R. F. C. Walters, and R. J. Wood. Cartesian bicategories II. *Theory and Applications of Categories*, 19(6):93–124, 2008.

[Coh81] P. M. Cohn. *Universal Algebra*, volume 6 of *Mathematics and its applications.* Springer Netherlands, 1981.

[Cro94] R. L. Crole. *Categories for Types.* Cambridge University Press, 1994.

[Cur93] P.-L. Curien. Substitution up to isomorphism. *Fundam. Inf.*, 19(1-2):51–85, September 1993.

[CW87] A. Carboni and R. F. C. Walters. Cartesian bicategories I. *Journal of Pure and Applied Algebra*, 49(1):11–32, 1987.

[Day70] B. Day. On closed categories of functors. In S. Mac Lane, H. Applegate, M. Barr, B. Day, E. Dubuc, Phreilambud, A. Pultr, R. Street, M. Tierney, and S. Swierczkowski, editors, *Reports of the Midwest Category Seminar IV*, pages 1–38, Berlin, Heidelberg, 1970. Springer Berlin Heidelberg.

[DK97] R. Di Cosmo and D. Kesner. Strong normalization of explicit substitutions via cut elimination in proof nets. In *Proceedings of Twelfth Annual IEEE Symposium on Logic in Computer Science*, pages 35–46, June 1997.

[DL11] G. Dowek and J.-J. Lévy. *Introduction to the Theory of Programming Languages*, chapter 2, pages 15–31. Springer, London, 2011.

[DM13] P-E. Dagand and C. McBride. A categorical treatment of ornaments. In *Proceedings of the 2013 28th Annual ACM/IEEE Symposium on Logic in Computer Science*, LICS '13, pages 530–539, Washington, DC, USA, 2013. IEEE Computer Society.

[DS97] B. Day and R. Street. Monoidal bicategories and Hopf algebroids. *Advances in Mathematics*, 129(1):99–157, 1997.

[É99] Z. Ésik. Axiomatizing iteration categories. *Acta Cybern.*, 14(1):65–82, February 1999.

[FDCB02] M. Fiore, R. Di Cosmo, and V. Balat. Remarks on isomorphisms in typed lambda calculi with empty and sum types. In *Proceedings of the Seventeenth Annual IEEE Symposium on Logic in Computer Science (LICS 2002)*, pages 147–156. IEEE Computer Society Press, July 2002. DOI: 10.1109/LICS.2002.1029824.

[FGHW07] M. Fiore, N. Gambino, M. Hyland, and G. Winskel. The cartesian closed bicategory of generalised species of structures. *Journal of the London Mathematical Society*, 77(1):203–220, 2007.

[FGHW17] M. Fiore, N. Gambino, M. Hyland, and G. Winskel. Relative pseudomonads, Kleisli bicategories, and substitution monoidal structures. *Selecta Mathematica New Series*, 2017.

[Fio02] M. Fiore. Semantic analysis of normalisation by evaluation for typed lambda calculus. In *Proceedings of the 4th ACM SIGPLAN International Conference on Principles and Practice of Declarative Programming*, PPDP '02, pages 26–37, New York, NY, USA, 2002. ACM.

[Fio06] T. Fiore. *Pseudo Limits, Biadjoints, and Pseudo Algebras: Categorical Foundations of Conformal Field Theory.* Memoirs of the American Mathematical Society. AMS, 2006.

[Fio11] M. Fiore. Algebraic foundations for type theories. 18th Types for Proofs and Programs workshop, September 2011. Slides available at `https://www.cl.cam.ac.uk/~mpf23/talks/Types2011.pdf`.

[Fio16] M. Fiore. An algebraic combinatorial approach to opetopic structure. `https://www.mpim-bonn.mpg.de/node/6586`, 2016. Talk at the *Seminar on Higher Structures, Program on Higher Structures in Geometry and Physics*, Max Planck Institute for Mathematics, Bonn (Germany).

[Fio17] M. Fiore. On the concrete representation of discrete enriched abstract clones. *Tbilisi Mathematical Journal*, 10(3):297–328, 2017.

[FJ15] M. Fiore and A. Joyal. Theory of para-toposes. Talk at the Category Theory 2015 Conference. Departamento de Matematica, Universidade de Aveiro (Portugal), 2015.

[FM18] S. Forest and S. Mimram. Coherence of Gray categories via rewriting. In *3rd International Conference on Formal Structures for Computation and Deduction (FSCD 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

[FPT99] M. Fiore, G. Plotkin, and D. Turi. Abstract syntax and variable binding. In *Proceedings of the 14th Annual IEEE Symposium on Logic in Computer Science*, LICS '99, pages 193–, Washington, DC, USA, 1999. IEEE Computer Society.

[Fre91] P. Freyd. Algebraically complete categories. In *Lecture Notes in Mathematics*, pages 95–104. Springer Berlin Heidelberg, 1991.

[Fre19] J. Frey. A language for closed cartesian bicategories. Category Theory 2019, University of Edinburgh, Edinburgh, UK, July 2019.

[FS90] P. J. Freyd and A. Scedrov. *Categories, Allegories*. Elsevier North Holland, 1990.

[FS99] M. Fiore and A. Simpson. Lambda definability with sums via Grothendieck logical relations. In J.-Y. Girard, editor, *Typed Lambda Calculi and Applications*, pages 147–161, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.

[FS18] M. Fiore and P. Saville. Skew monoidal structures on categories of algebras. Category Theory 2018, University of Azores, Ponta Delgada, Portugal, July 2018.

[FS19] M. Fiore and P. Saville. A type theory for cartesian closed bicategories. In *Proceedings of the Thirty-Fourth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, 2019.

[Gan53] R. O. Gandy. *On axiomatic systems in mathematics and theories in physics.* PhD thesis, University of Cambridge, 1953.

[Gar09] R. Garner. Two-dimensional models of type theory. *Mathematical Structures in Computer Science*, 19(4):687–736, 2009.

[GdR99] N. Ghani, V. de Paiva, and E. Ritter. Categorical models of explicit substitutions. In *Proceedings of the Second International Conference on Foundations of Software Science and Computation Structure, Held As Part of the European Joint Conferences on the Theory and Practice of Software, ETAPS'99*, FoSSaCS '99, pages 197–211, Berlin, Heidelberg, 1999. Springer-Verlag.

[GFW98] G.L. Cattani, M. Fiore, and G. Winskel. A theory of recursive domains with applications to concurrency. In *Proceedings of the 13th Annual IEEE Symposium on Logic in Computer Science*, pages 214–225. IEEE Computer Society, 1998.

[Gha95] N. Ghani. *Adjoint rewriting.* PhD thesis, University of Edinburgh, 1995.

[Gib97] J. Gibbons. Conditionals in distributive categories. Technical report, University of Oxford, 1997.

[Gir72] J.-Y. Girard. *Interprétation fonctionnelle et élimination des coupures de l'arithmétique d'ordre supérieur.* PhD thesis, Université Paris Diderot - Paris 7, 1972.

[GJ17] N. Gambino and A. Joyal. On operads, bimodules and analytic functors. *Memoirs of the American Mathematical Society*, 249(1184):153–192, 2017.

[GK13] N. Gambino and J. Kock. Polynomial functors and polynomial monads. *Mathematical Proceedings of the Cambridge Philosophical Society*, 154(1):153–192, 2013.

[GPS95] R. Gordon, A. J. Power, and R. Street. *Coherence for tricategories.* Memoirs of the American Mathematical Society, 1995.

[Gra74] J. W. Gray. *Formal Category Theory: Adjointness for 2-Categories*, volume 391 of *Lecture Notes in Mathematics.* Springer, 1974.

[GTL89] J.-Y. Girard, P. Taylor, and Y. Lafont. *Proofs and Types.* Cambridge University Press, New York, NY, USA, 1989.

[Gur06] N. Gurski. *An Algebraic Theory of Tricategories.* University of Chicago, Department of Mathematics, 2006.

[Gur12] N. Gurski. Biequivalences in tricategories. *Theory and Applications of Categories*, 26(14):349–384, 2012.

[Gur13] N. Gurski. *Coherence in Three-Dimensional Category Theory.* Cambridge University Press, 2013.

[Har69] F. Harary. *Graph Theory.* Addison-Wesley Publishing Company, Boston, 1969.

[Her93] C. Hermida. *Fibrations, Logical Predicates and Indeterminates.* PhD thesis, University of Edinburgh, 1993.

[Her00] C. Hermida. Representable multicategories. *Advances in Mathematics*, 151(2):164–225, 2000.

[Hil96] B.P. Hilken. Towards a proof theory of rewriting: the simply typed 2λ-calculus. *Theoretical Computer Science*, 170(1):407–444, 1996.

[Hir13] T. Hirschowitz. Cartesian closed 2-categories and permutation equivalence in higher-order rewriting. *Logical Methods in Computer Science*, 9:1–22, 07 2013.

[Hou07] R. Houston. *Linear Logic without Units.* PhD thesis, University of Manchester, 2007.

[Hue76] G. Huet. *Résolution d'équations dans des langages d'ordre* $1, 2, \ldots, \omega$. PhD thesis, Université de Paris VII, 1976.

[Hue80] G. Huet. Confluent reductions: Abstract properties and applications to term rewriting systems. *Journal of the ACM*, 27(4):797–821, October 1980.

[Jac92] B. Jacobs. Simply typed and untyped lambda calculus revisited. In *Applications of Categories in Computer Science*, pages 119–142. Cambridge University Press, jun 1992.

[JG95] C. B. Jay and N. Ghani. The virtues of eta-expansion. *Journal of Functional Programming*, 5(2):135–154, 1995.

[Joh02] P. T. Johnstone. *Sketches of an Elephant: A Topos Theory Compendium Volume 2 (Oxford Logic Guides).* Clarendon Press, 2002.

[JS93] A. Joyal and R. Street. Braided tensor categories. *Advances in Mathematics*, 102(1):20–78, 11 1993.

[JT93] A. Jung and J. Tiuryn. A new characterization of lambda definability. In M. Bezem and J. F. Groote, editors, *Typed Lambda Calculi and Applications*, pages 245–257, Berlin, Heidelberg, 1993. Springer Berlin Heidelberg.

[Kel64] G.M. Kelly. On Mac Lane's conditions for coherence of natural associativities, commutativities, etc. *Journal of Algebra*, 1(4):397–402, 1964.

[Kel89] G. M. Kelly. Elementary observations on 2-categorical limits. *Bulletin of the Australian Mathematical Society*, 39(2):301–317, 1989.

[Lac07] S. Lack. Bicat is not triequivalent to Gray. *Theory and Applications of Categories*, 18(1):1–3, 2007.

[Lac10] S. Lack. *A 2-Categories Companion*, pages 105–191. Springer New York, New York, NY, 2010.

[Laf87] Y. Lafont. *Logiques, catégories et machines.* PhD thesis, Université Paris VII, 1987.

[Lam69] J. Lambek. Deductive systems and categories II: Standard constructions and closed categories. In *Category theory, homology theory and their applications I*, pages 76–122. Springer, 1969.

[Lam80] J. Lambek. From lambda calculus to cartesian closed categories. In *To H. B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism.* Academic Press, 1980.

[Lam86] J. Lambek. Cartesian closed categories and typed lambda calculi. In *Proceedings of the Thirteenth Spring School of the LITP on Combinators and Functional Programming Languages*, pages 136–175, London, UK, UK, 1986. Springer-Verlag.

[Lam89] J. Lambek. Multicategories revisited. In J. W. Gray and A. Scedrov, editors, *Categories in Computer Science and Logic: Proceedings of the AMS-IMS-SIAM Joint Summer Research Conference Held June 14-20, 1987 with Support from the National Science Foundation*, volume 92, pages 217–240. American Mathematical Society, 1989.

[Law17] F. W. Lawvere. Adjoints in and among bicategories. In *Logic and algebra*, pages 181–189. Routledge, 10 2017.

[Lei98] T. Leinster. Basic bicategories. Available at `https://arxiv.org/abs/math/9810017`, May 1998.

[Lei04] T. Leinster. *Higher operads, higher categories.* Number 298 in London Mathematical Society Lecture Note Series. Cambridge University Press, 2004.

[LH11] D. R. Licata and R. Harper. 2-dimensional directed type theory. *Electronic Notes in Theoretical Computer Science*, 276:263–289, 2011. Twenty-seventh Conference on the Mathematical Foundations of Programming Semantics (MFPS XXVII).

[LH12] D. R. Licata and R. Harper. Canonicity for 2-dimensional type theory. In *Proceedings of the 39th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, POPL '12, pages 337–348, New York, NY, USA, 2012. ACM.

[LM99] J.-J. Lévy and L. Maranget. Explicit substitutions and programming languages. In C. Pandu Rangan, V. Raman, and R. Ramanujam, editors, *Foundations of Software Technology and Theoretical Computer Science*, pages 181–200, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.

[LS86] J. Lambek and P. J. Scott. *Introduction to Higher Order Categorical Logic.* Cambridge University Press, New York, NY, USA, 1986.

[LS12] S. Lack and R. Street. Skew monoidales, skew warpings and quantum categories. *Theory and Applications of Categories*, 2012.

[LS14] S. Lack and R. Street. On monads and warpings. *Cahiers de Topologie et Géométrie Différentielle Catégoriques*, LV(4):244–266, 2014.

[LSR17] D. R. Licata, M. Shulman, and M. Riley. A fibrational framework for substructural and modal logics. In *FSCD*, 2017.

[Mac63] S. Mac Lane. Natural associativity and commutativity. *Rice University Studies*, 1963.

[Mac98] S. Mac Lane. *Categories for the Working Mathematician*, volume 5 of *Graduate Texts in Mathematics*. Springer-Verlag New York, second edition, 1998.

[Mak96] M. Makkai. Avoiding the axiom of choice in general category theory. *Journal of Pure and Applied Algebra*, 108(2):109 – 173, 1996.

[Mel95] P.-A. Melliès. Typed λ-calculi with explicit substitutions may not terminate. In M. Dezani-Ciancaglini and G. Plotkin, editors, *Typed Lambda Calculi and Applications*, pages 328–334, Berlin, Heidelberg, 1995. Springer Berlin Heidelberg.

[Mel09] P.-A. Melliès. Categorical semantics of linear logic. *Panoramas et synthèses*, 27:15–215, 2009.

[ML84] P. Martin-Löf. *Intuitionistic Type Theory.* Bibliopolis, 1984.

[Mog89] E. Moggi. Computational lambda-calculus and monads. In *Proceedings, Fourth Annual Symposium on Logic in Computer Science*. IEEE Comput. Soc. Press, 1989.

[Mog91] E. Moggi. Notions of computation and monads. *Information and Computation*, 93(1):55–92, jul 1991.

[MP85] S. Mac Lane and R. Paré. Coherence for bicategories and indexed categories. *Journal of Pure and Applied Algebra*, 37:59 – 80, 1985.

[MR77] M. Makkai and G. E. Reyes. *First Order Categorical Logic: Model-Theoretical Methods in the Theory of Topoi and Related Categories*. Springer, 1977.

[MR92] Q. M. Ma and J. C. Reynolds. Types, abstraction, and parametric polymorphism, part 2. In S. Brookes, M. Main, A. Melton, M. Mislove, and D. Schmidt, editors, *Mathematical Foundations of Programming Semantics*, pages 1–40, Berlin, Heidelberg, 1992. Springer Berlin Heidelberg.

[MS93] J. C. Mitchell and A. Scedrov. Notes on sconing and relators. In E. Börger, G. J., H. Kleine Büning, S. Martini, and M. M. Richter, editors, *Computer Science Logic*, pages 352–378, Berlin, Heidelberg, 1993. Springer Berlin Heidelberg.

[Oli20] F. Olimpieri. Intersection type distributors. *arXiv*, 2020. Available at `http://arxiv.org/abs/2002.01287v2`.

[Oua97] J. Ouaknine. A two-dimensional extension of Lambek's categorical proof theory. Master's thesis, McGill University, 1997.

[Paq20] H. Paquet. *Probabilistic concurrent game semantics.* PhD thesis, University of Cambridge, 2020.

[Pit87] A. M. Pitts. An elementary calculus of approximations (extended abstract). Unpublished manuscript, University of Sussex, December 1987, 1987.

[Pit00] A. M. Pitts. Categorical logic. In *Handbook of Logic in Computer Science*, chapter 2, pages 39–123. Oxford University Press, Oxford, UK, 2000.

[Plo73] G. D. Plotkin. Lambda-definability and logical relations. Technical report, University of Edinburgh School of Artificial Intelligence, 1973. Memorandum SAI-RM-4.

[Plo94] B. Plotkin. *Universal Algebra, Algebraic Logic, and Databases.* Springer, 1994.

[Pow89a] A. J. Power. An abstract formulation for rewrite systems. In D. H. Pitt, D. E. Rydeheard, P. Dybjer, A. M. Pitts, and A. Poigné, editors, *Category Theory and Computer Science*, pages 300–312, Berlin, Heidelberg, 1989. Springer Berlin Heidelberg.

[Pow89b] A. J. Power. Coherence for bicategories with finite bilimits I. In J. W. Gray and A. Scedrov, editors, *Categories in Computer Science and Logic: Proceedings of the AMS-IMS-SIAM Joint Summer Research Conference Held June 14–20, 1987 with Support from the National Science Foundation*, volume 92, pages 341–349. American Mathematical Society, 1989.

[Pow89c] A. J. Power. A general coherence result. *Journal of Pure and Applied Algebra*, 57(2):165–173, 1989.

[Pow98] A. J. Power. 2-categories. *BRICS Notes Series*, 1998.

[RBL11] K. H. Rose, R. Bloo, and F. Lang. On explicit substitution with names. *Journal of Automated Reasoning*, 49(2):275–300, mar 2011.

[RdP97] E. Ritter and V. de Paiva. On explicit substitutions and names (extended abstract). In P. Degano, R. Gorrieri, and A. Marchetti-Spaccamela, editors, *Automata, Languages and Programming*, pages 248–258, Berlin, Heidelberg, 1997. Springer Berlin Heidelberg.

[Rit99] E. Ritter. Characterising explicit substitutions which preserve termination. In *Proceedings of the 4th International Conference on Typed Lambda Calculi and Applications*, TLCA '99, pages 325–339, London, UK, UK, 1999. Springer-Verlag.

[RPW00] E. Ritter, D. Pym, and L. Wallen. Proof-terms for classical and intuitionistic resolution. *Journal of Logic and Computation*, 10(2):173–207, 04 2000.

[RS87] D. E. Rydeheard and J. G. Stell. Foundations of equational deduction: A categorical treatment of equational proofs and unification algorithms. In D. H. Pitt, A. Poigné, and D. E. Rydeheard, editors, *Category Theory and Computer Science*, pages 114–139, Berlin, Heidelberg, 1987. Springer Berlin Heidelberg.

[RS17] E. Riehl and M. Shulman. A type theory for synthetic ∞-categories. *Higher Structures*, 1(1):147–224, November 2017.

[Sea13] G. J. Seal. Tensors, monads and actions. *Theory and Applications of Categories*, 28(15):403–434, 2013.

[See84] R. A. G. Seely. Locally cartesian closed categories and type theory. *Mathematical Proceedings of the Cambridge Philosophical Society*, 95(1):33–48, jan 1984.

[See87] R. A. G. Seely. Modelling computations: A 2-categorical framework. In D. Gries, editor, *Proceedings of the Second Annual IEEE Symp. on Logic in Computer Science, LICS 1987*, pages 65–71. IEEE Computer Society Press, June 1987.

[Shu08] M. Shulman. Set theory for category theory. Preprint, `https://arxiv.org/abs/0810.1279`, 2008.

[Shu19] M. Shulman. A practical type theory for symmetric monoidal categories. Preprint, `http://arxiv.org/abs/1911.00818v1`, 2019.

[Sta85] R. Statman. Logical relations and the typed λ-calculus. *Information and Control*, 65:85–97, 1985.

[Sta13] S. Staton. An algebraic presentation of predicate logic. In F. Pfenning, editor, *Foundations of Software Science and Computation Structures*, pages 401–417, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[Str72] R. Street. The formal theory of monads. *Journal of Pure and Applied Algebra*, 2(2):149 –168, 1972.

[Str80] R. Street. Fibrations in bicategories. *Cahiers de Topologie et Géométrie Différentielle Catégoriques*, 21(2):111–160, 1980.

[Str95] R. Street. Categorical structures. In M. Hazewinkel, editor, *Handbook of Algebra*, volume 1, chapter 15, pages 529–577. Elsevier, 1995.

[Szl12] K. Szlachányi. Skew-monoidal categories and bialgebroids. *Advances in Mathematics*, 231(3):1694 – 1730, 2012.

[Tab11] N. Tabareau. Aspect oriented programming: A language for 2-categories. In *Proceedings of the 10th International Workshop on Foundations of Aspect-oriented Languages*, FOAL '11, pages 13–17, New York, NY, USA, 2011. ACM.

[Tai67] W. Tait. Intensional interpretations of functionals of finite type I. *The Journal of Symbolic Logic*, 32(2):198–212, 1967.

[Tay99] P. Taylor. *Practical Foundations of Mathematics*, volume 59 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, 1999.

[The13] The Univalent Foundations Program. *Homotopy Type Theory: Univalent Foundations of Mathematics*. `https://homotopytypetheory.org/book`, Institute for Advanced Study, 2013.

[TS00] A. S. Troelstra and H. Schwichtenberg. *Basic proof theory*. Number 43 in Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, second edition, 2000.

[Ver92] D. Verity. *Enriched categories, internal categories and change of base*. PhD thesis, University of Cambridge, 1992. TAC reprint available at `http://www.tac.mta.ca/tac/reprints/articles/20/tr20abs.html`.

[Vit10] E. M. Vitale. Bipullbacks and calculus of fractions. *Cahiers de Topologie et Géométrie Différentielle Catégoriques*, 51(2):83–113, 2010.

[Wei94] C. A. Weibel. *An Introduction to Homological Algebra*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 1994.

[YA18] N. Yamada and S. Abramsky. Dynamic game semantics. Preprint, `https://arxiv.org/abs/1601.04147`, October 2018.

[Yau16] D. Yau. *Colored Operads*. American Mathematical Society, 2016.