Adjunctions

Maarten Fokkinga* and Lambert Meertens†

Version of June 22, 1993

We present the category-theoretic notion of adjunction in a way that makes it easy to formally *calculate* with it; an acquaintance with its algebraic properties may greatly help in understanding the notion. It is illustrated by means of a lot of theorems and proofs. We also attempt to provide some intuitive understanding of adjunctions by various discussions. Our intended readership is familiar with the notion of category, functor, and naturality, and either about to learn about adjunctions or interested in a *calculational* approach to category theory.

Key Words and Phrases: category theory, adjunction, initiality, equational reasoning, algebraic calculation.

1 Introduction

1 Aim. The notion of 'adjunction' is one of the most important concepts in category theory. At the same time, it is one of the hardest to understand, due to the large numbers of entities involved. In this text we set out to present the notion of adjunction in a way that makes it easy to formally calculate with them (equational reasoning); an acquaintance with its algebraic properties may greatly help in understanding the notion. Thus we will prove a lot of theorems about adjunctions in an algebraic, calculational style. We also attempt to provide some intuitive understanding of adjunctions by various discussions.

Our intended readership is familiar with the notion of category, functor, and naturality, and either about to learn about adjunctions or interested in a *calculational* approach to category theory. All our theorems about adjunctions in general can be found in the standard book by Mac Lane [4], though in quite another wording and formulation, and proved in a quite different style.

* University of Twente, dept INF, PO Box 217, NL 7500 AE ENSCHEDE, the Netherlands;
e-mail: `fokkinga@cs.utwente.nl`

† CWI, dept Algorithmics and Architecture, PO Box 4079, NL 1009 AB AMSTERDAM, the Netherlands, and Utrecht University, dept Computing Science, Utrecht, the Netherlands;
e-mail: `lambert@cwi.nl`

2 Preliminaries. We assume the following concepts to be known. The default notation for the various entities is given in between parentheses. A *category* ($\mathcal{A},\mathcal{B},\dots$) is a collection of things ($A,B,\dots$) called *objects*, a collection of things ($f,g,\dots$) called *arrows*, a relation between an arrow and pair of objects ($f\colon\ A\to B$) called *typing*, a partial binary operation on arrows ($\_\ ;\ \_$) called *composition*, and for each object A a distinguished arrow (id_A) called the *identity* on A, satisfying the following axioms:

$$f\colon\ A\to B \text{ and } f\colon\ A'\to B' \quad\Rightarrow\quad A=A' \text{ and } B=B' \qquad \text{unique-TYPE}$$

$$f\colon\ A\to B \text{ and } g\colon\ B\to C \quad\Rightarrow\quad f\,;g\colon\ A\to C \qquad \text{composition-TYPE}$$

$$id_A\colon\ A\to A \qquad \text{identity-TYPE}$$

$$(f\,;g)\,;h = f\,;(g\,;h) \qquad \text{composition-ASSOC}$$

$$id\,;f = f = f\,;id \qquad \text{IDENTITY}$$

The collection of all arrows in $\mathcal{A}$ from A to B is denoted $(A\to B)$ (the standard notation is $\mathrm{Hom}(A,B)$), so that $f\colon\ A\to B$ is equivalent to $f\in(A\to B)$. An arrow term f is **well-typed** if: a typing $f\colon\ A\to B$ can be derived for some objects A,B according to these axioms (and the TYPE properties of defined notions that will be given in the sequel). Sometimes there are several categories under discussion. Then the name of the category may and must be added to the above notations, as a subscript or otherwise, in order to avoid ambiguity; for example, $f\colon\ A\to_{\mathcal{A}} B$ and $(A\to_{\mathcal{A}} B)$.

Convention. Whenever we write a term, we assume that the variables are typed (at their introduction — mostly an implicit universal quantification in front of the formula) in such a way that the term is well-typed. This convention allows us to simplify the formulations considerably; it has already been used in axioms composition-ASSOC and IDENTITY. The category axioms are so basic that we shall mostly use them tacitly. In particular, we shall use composition-ASSOC implicitly by omitting the parentheses in a composition.

Let $\mathcal{A}$ and $\mathcal{B}$ be categories. A *functor* ($F,G,\dots$) from $\mathcal{A}$ to $\mathcal{B}$ is a mapping F from $\mathcal{A}$ to $\mathcal{B}$ that preserves all categorical structure:

FA is an object in $\mathcal{B}$		whenever A is an object in $\mathcal{A}$	ftr-OBJECT
Ff is an arrow in $\mathcal{B}$		whenever f is an arrow in $\mathcal{A}$	ftr-ARROW
Ff	$:\ FA\to_{\mathcal{B}} FB$	whenever $f\colon\ A\to_{\mathcal{A}} B$	ftr-TYPE
$F\,id_A$	$=\ id_{FA}$	for each object A in $\mathcal{A}$	FUNCTOR
$F(f\,;g)$	$=\ Ff\,;Fg$	whenever $f\,;g$ is well-typed	FUNCTOR

Notation: $F\colon\ \mathcal{A}\to\mathcal{B}$. Composition of functors is denoted by juxtaposition: $(FG)x = F(G(x))$, and the identity functor is denoted I.

Let $\mathcal{A},\mathcal{B}$ be categories, and $F,G\colon\ \mathcal{A}\to\mathcal{B}$ be functors. A *transformation* ($\varepsilon,\eta,\dots$) from F to G is a family ε of arrows in $\mathcal{B}$ indexed by the objects of $\mathcal{A}$ such that:

$$\varepsilon_A \ :\ FA\to_{\mathcal{B}} GA \quad \text{for each } A \text{ in } \mathcal{A}. \qquad \text{trf-TYPE}$$

Notation: ε: $F \rightarrow_{\mathcal{B}} G$. A transformation ε from F to G is *natural* (ε: $F \dot{\rightarrow}_{\mathcal{B}} G$) if:

$$Ff \mathbin{;} \varepsilon_B \quad = \quad \varepsilon_A \mathbin{;} Gf \quad \text{for each } f\colon A \rightarrow_{\mathcal{A}} B \,, \qquad \text{NTRF}$$

that is, transformation ε *commutes with* an arbitrary arrow f: $A \rightarrow_{\mathcal{A}} B$ (suitably "lifted" to go from objects FA and GA, respectively). Actually, well-typedness of NTRF implies trf-TYPE.

3 Omitting subscripts. For readability and ease of formal calculation we shall often omit the subscripts to transformations and natural transformations when they can be retrieved from contextual information. Here is an example; you are not supposed to understand the meaning of the formulas.

Let the following be given: two functors F: $\mathcal{A} \rightarrow \mathcal{B}$ and G: $\mathcal{B} \rightarrow \mathcal{A}$, and two transformations ε: $FG \rightarrow I_{\mathcal{B}}$ and η: $I_{\mathcal{A}} \rightarrow GF$. Consider formula

$$\eta \mathbin{;} G\varepsilon \quad = \quad id \,.$$

The following procedure gives the most general subscripts that make the formula well typed. Use $a, b, c, \ldots$ as type variables (the "unknows"), use these as the subscripts, and write the source and target type within braces at the source and target side of the morphisms, thus:

$$\{a\} \quad \eta_b \ \{c\} \ ; \ \{d\} \quad G(\ \{e\} \quad \varepsilon_f \ \{g\}\ ) \ \{h\} \quad = \quad \{j\} \quad id_k \ \{l\} \,.$$

The typing axioms generate a collection of equations for the type variables:

typing η:	$a, c = b, GFb$	on account of trf-TYPE
typing $;$:	$c = d$	on account of composition-TYPE
typing $G\varepsilon$:	$d, h = Ge, Gg$	on account of ftr-TYPE
typing ε:	$e, g = FGf, f$	on account of trf-TYPE
typing id:	$j = k = l$	on account of identity-TYPE
typing $=$:	$a, h = j, l$.	

A most general (least constraining) solution for this collection of equations can be found by the unification algorithm, and yields

$$a = b = h = j = l = k = Gf$$
$$c = d = GFGf$$
$$e = FGf$$
$$g = f \,.$$

Hence, writing B for type variable f, and filling in the subscripts, the formula reads: for arbitrary object B in $\mathcal{B}$,

$$\eta_{GB} \mathbin{;} G\varepsilon_B \quad = \quad id_{GB} \quad : \quad GB \rightarrow_{\mathcal{A}} GB \,.$$

4 Universal property. The notion of "universal property" plays an important role in the categorical definition of various concepts, like finality, products, equalisers, pullbacks, limits, and adjointness and their dualisation. The notion is defined and discussed here, at a high level of abstraction. Our approach is the key to the calculational reasoning about adjunctions.

Let some entity a be given, and let $P(\_,\_,\_)$ be some predicate. We say that a is P**-universal** (for the free variables of P) if: for each x there is a unique solution y of the statement $P(a,x,y)$. Equivalently, a is P-universal if:

$$\forall x :: \quad \exists! y :: \quad P(a,x,y).$$

And yet another equivalent way is the following, using so-called skolemisation (after the logician Skolem). Entity a is P-universal if: there exists a mapping $\mathcal{F}$ such that

$$(*) \qquad \forall x,y :: \quad \mathcal{F}x = y \quad \equiv \quad P(a,x,y).$$

In the former formulation it is the existential quantification ($\exists y$) inside the scope of a universal one that hinders effective calculation. In the latter formulation the existence claim is brought to a more global level; a calculation (equational reasoning in category theory) need no longer be interrupted by the declaration and naming of the existence of a unique y that depends on x: it can be denoted just $\mathcal{F}x$.

In view of the frequent appearances of the various unique y's, these y's deserve a particular notation that triggers the reader of their particular properties. Below we employ the bracket notation $\llbracket x \rrbracket$ and $\lceil\!\lceil x \rceil\!\rceil$ for such $\mathcal{F}x$; Fokkinga [3] uses the notation $(\![x]\!)$ and $[\!(x)\!]$ for $\mathcal{F}x$ in the case of initiality and finality. As usual we omit in line $(*)$ the universal quantifications that are outermost, thus simplifying the formulation once more. So we say: a is P-universal if: there exists a mapping $\mathcal{F}$ such that

$$\mathcal{F}x = y \quad \equiv \quad P(a,x,y). \qquad \mathcal{F}\text{-}\textsc{Charn}$$

The name Charn derives from the fact that the equivalence is a characterisation (and definition) of $\mathcal{F}$. (Actually, 'P-' or 'P-$\mathcal{F}$-' might be a better prefix to Charn, especially in law Uniq below.) Two immediate corollaries are:

$$P(a,x,\mathcal{F}x) \qquad \mathcal{F}\text{-}\textsc{Self}$$

$$P(a,x,y) \wedge P(a,x,z) \quad \Rightarrow \quad y = z. \qquad \mathcal{F}\text{-}\textsc{Uniq}$$

Law Self asserts that there is at least one solution for y in $P(a,x,y)$, and more specifically, that $y = \mathcal{F}x$ is itself a solution; it follows from Charn by taking $y = \mathcal{F}x$ so that the left-hand side of Charn becomes true. Law Uniq asserts that there is at most one solution for y of $P(a,x,y)$, or in other words, that the solution is unique; it follows from Charn since both y and z are equal to $\mathcal{F}x$, hence equal to each other. Together they are equivalent to Charn:

$$\textsc{Self} \text{ and } \textsc{Uniq} \quad \equiv \quad \textsc{Charn}.$$

The $\Leftarrow$-part has been argued above; for the $\Rightarrow$-part we show equivalence CHARN by circular implication:

$$
\begin{array}{lll}
& P(a,x,y) & \text{(right-hand side of CHARN)} \\
\equiv & \quad \text{SELF} & \\
& P(a,x,\mathcal{F}x) \ \text{ and } \ P(a,x,y) & \\
\Rightarrow & \quad \text{UNIQ} & \\
& \mathcal{F}x = y & \text{(left-hand side of CHARN)} \\
\equiv & \quad \text{SELF} & \\
& P(a,x,\mathcal{F}x) \ \text{ and } \ \mathcal{F}x = y & \\
\Rightarrow & \quad \text{equality} & \\
& P(a,x,y) & \text{(right-hand side of CHARN)}
\end{array}
$$

In our experience, proving P-universality by separately establishing SELF (for some entity denoted $\mathcal{F}x$) and UNIQ is by no means simpler or more elegant than establishing CHARN directly by a series of equivalences or by circular implication as above.

As soon as there is some universal property, it is worthwhile to look for so-called fusion properties, that is, laws of the following shape:

$$\text{some condition on } x' \text{ and } x'' \quad \Rightarrow \quad (\mathcal{F}x) \,;\, x' = \mathcal{F}(x'') . \qquad \mathcal{F}\text{-FUSION}$$

And similarly with x' at the front instead of after $\mathcal{F}$. If the condition takes the form $x'' = E(x,x')$, then the fusion law simplifies to an unconditional one:

$$(\mathcal{F}x) \,;\, x' = \mathcal{F}(E(x,x')) \qquad \mathcal{F}\text{-FUSION}$$

Fusion laws are necessary in reasonings about $\mathcal{F}$.

5 Pre-orders. Many concepts of category theory can be considered as a generalisation of concepts about pre-orders, and this holds also for adjunctions. Therefore we recall that approach.

A pre-order $\mathcal{A} = (V, \sqsubseteq_A)$ can be seen as a category by taking the elements of the set V as the objects of $\mathcal{A}$, and defining

$$
\begin{array}{lcll}
(a \rightarrow_A b) & = & \{(a,b)\}, & \text{if } a \sqsubseteq_A b \\
& = & \{\,\}, & \text{otherwise} .
\end{array}
$$

In words, there is either exactly one arrow from a to b, namely if $a \sqsubseteq_A b$, or none. This leaves no choice but to define arrow composition and the identities thus:

$$
\begin{array}{lcl}
(a,b) \,;\, (b,c) & = & (a,c) \\
id_a & = & (a,a) .
\end{array}
$$

Thus the three axioms unique-TYPE, composition-ASSOC and IDENTITY are trivially satisfied; the two axioms composition-TYPE and identity-TYPE (and the well-definedness of these two definitions) precisely correspond to transitivity and reflexivity of $\sqsubseteq_{\mathcal{A}}$.

Let $\mathcal{B} = (W, \sqsubseteq_{\mathcal{B}})$ also be a pre-order, and F a function that maps the elements of V to elements of W. When can F be extended to a functor from $\mathcal{A}$ to $\mathcal{B}$, viewed as categories? We do not have much choice for the action of F on arrows. If (a, b) is an arrow of $\mathcal{A}$, the hopeful functor has to map it to an arrow in $(Fa \rightarrow_{\mathcal{A}} Fb)$. If this collection is empty for some arrow of $\mathcal{A}$, this is an impossibility and F cannot be extended to a functor. Otherwise the image under F is the unique arrow in this collection, and the requirement that a functor distributes over composition and preserves identity arrows is trivially satisfied. Translating these findings back into pre-order language, we have that F is (that is: can be viewed as) a functor from $\mathcal{A}$ to $\mathcal{B}$ precisely when

$$a \sqsubseteq_{\mathcal{A}} b \quad \Rightarrow \quad Fa \sqsubseteq_{\mathcal{B}} Fb$$

for all $a, b \in V$. In words, F is a functor if, and only if, F is monotonic.

This gives us a straightforward way to specialise results from category theory to pre-order theory. Conversely, given an order-theoretic concept, we can try to find a category-theoretical generalisation, that is, a corresponding concept expressed in category-speak whose meaningfulness does not depend on the specific properties of pre-orders. In text 6 we shall do so for Galois connections.

2 Adjunctions explained

In this section we shall provide three ways to "discover", as it were, the notion of adjunction. One is as a category-theoretical generalisation of Galois connections, another is a direct construction, and a last one is by abstracting from a specific example. Neither of these explanations is used in the formal calculations in the following sections.

6 Galois connections categorically. Let $\mathcal{A} = (V, \sqsubseteq_{\mathcal{A}})$ and $\mathcal{B} = (W, \sqsubseteq_{\mathcal{B}})$ be pre-orders, and $F\colon V \rightarrow W$ and $G\colon W \rightarrow V$ be mappings. These constitute a **Galois connection** if: for all $a \in V$ and $b \in W$

$$a \sqsubseteq_{\mathcal{A}} Gb \quad \equiv \quad Fa \sqsubseteq_{\mathcal{B}} b \,. \qquad \text{GALOIS}$$

Pre-orders turn up very frequently in mathematical reasoning, and Galois connections form an indispensable tool for effective, formal calculations in such situations. To give just one example, the floor function $\lfloor \_ \rfloor$ is characterised by the Galois connection:

$$n \leq_{\mathcal{N}} \lfloor r \rfloor \quad \equiv \quad n \leq_{\mathcal{R}} r$$

for all $\mathcal{N}$aturals n and all $\mathcal{R}$eals r. Here V is the set of naturals, W is the set of reals, F is the injection of the naturals into the reals, and G is the function $\lfloor \_ \rfloor$. Using the characterisation, and in addition the law of *indirect equality*:

$$k = m \quad \equiv \quad (\forall n :: \quad n \leq k \;\equiv\; n \leq m)\,,$$

one can now easily and elegantly prove such laws as: $\lfloor\sqrt{\lfloor r\rfloor}\rfloor = \lfloor\sqrt{r}\rfloor$. Backhouse et al. [1, Chapter 5] thoroughly discuss the use and theory of Galois connections, including this example.

Recalling the definition of the arrows for pre-order categories we see that the above equivalence Galois can be equivalently expressed as

$$(a \to_{\mathcal{A}} Gb) \quad \cong \quad (Fa \to_{\mathcal{B}} b)\,,$$

in which '$\cong$' means that there is a bijection between these two collections of arrows (since both are empty, or both are singleton collections).

Generalising this to arbitrary categories $\mathcal{A}$ and $\mathcal{B}$, we arrive as a first attempt at:

$$(A \to_{\mathcal{A}} GB) \quad \cong \quad (FA \to_{\mathcal{B}} B)$$

for each object A of $\mathcal{A}$ and B of $\mathcal{B}$. One can prove that equivalence Galois above implies that the mappings F and G are monotonic, so we expect, in the categorical generalisation, that F and G are functors, even if we do not require this a priori. This attempt sketches a situation (the existence of the bijections, and F, G being functors) that holds whenever there exists an adjunction. However, it is not yet quite strong enough. In order that there is an adjunction, the bijections between the arrow collections have, in some sense, to be *uniform* for varying A and B, or, in other words, to have the same structure independent of A and B. This is expressed categorically by requiring that the family bijections is *natural* in A and B. This, then, is the definition of 'adjunction'; it is worked out in detail in text 62.

Instead of pursuing this path here, we try to directly construct a uniform bijection in the next paragraph, and absorb what we need on the way as part of the definition of the concept of adjunction.

7 So what is an adjunction? To recapitulate the situation, we need two categories $\mathcal{A}$ and $\mathcal{B}$, two functors $F\colon \mathcal{A} \to \mathcal{B}$ and $G\colon \mathcal{B} \to \mathcal{A}$, and a "uniform" family of bijections between the arrow collections $(A \to_{\mathcal{A}} GB)$ and $(FA \to_{\mathcal{B}} B)$, for arbitrary A in $\mathcal{A}$ and B in $\mathcal{B}$. We denote these bijections in a bracketed form:

$$\begin{array}{lcl} \lceil\!\lceil \text{-} \rceil\!\rceil_{A,B} & : & (A \to_{\mathcal{A}} GB) \to (FA \to_{\mathcal{B}} B) \\ \lfloor\!\lfloor \text{-} \rfloor\!\rfloor_{A,B} & : & (FA \to_{\mathcal{B}} B) \to (A \to_{\mathcal{A}} GB) \end{array}$$

(but in the sequel the subscripts on $\lceil\!\lceil\ \rceil\!\rceil$ and $\lfloor\!\lfloor\ \rfloor\!\rfloor$ are omitted).

We shall use φ and ψ as names of arbitrary arrows in the two arrow collections involved:

$$\begin{array}{lcl} \text{in category } \mathcal{A}\text{:} & \varphi & : A \to_{\mathcal{A}} GB \\ & \lceil\!\lceil\ \rceil\!\rceil\!\downarrow\ \uparrow\!\lfloor\!\lfloor\ \rfloor\!\rfloor & \\ \text{in category } \mathcal{B}\text{:} & \psi & : FA \to_{\mathcal{B}} B. \end{array}$$

The requirement that $\lceil\!\lceil\ \rceil\!\rceil$ and $\lfloor\!\lfloor\ \rfloor\!\rfloor$ form a bijection can now be expressed by an appropriate universal quantification of the following equivalence between two arrow equations:

$$\varphi = \lfloor\!\lfloor \psi \rfloor\!\rfloor \quad \equiv \quad \lceil\!\lceil \varphi \rceil\!\rceil = \psi\,. \qquad \text{INVERSE}$$

It remains to express the "uniformity" of $\lceil\!\lceil\ \rceil\!\rceil$ and $\lfloor\!\lfloor\ \rfloor\!\rfloor$. Given an arrow φ from the $\mathcal{A}$ world, typed as above, how could we hope to translate it in a categorical fashion to an arrow ψ in the $\mathcal{B}$ world? The only ingredient we have that translates from $\mathcal{A}$ to $\mathcal{B}$ is the functor F. Applying it to φ gives us

$$F\varphi\colon\ FA \to_{\mathcal{B}} FGB\,.$$

We have made progress: at least the source object is right. If only we could bridge the remaining gap between the objects FGB and B, we would be done. So assume that there exists a family of arrows

$$\varepsilon_B\colon\ FGB \to_{\mathcal{B}} B\,,$$

one for each object B of $\mathcal{B}$, and define

$$\lceil\!\lceil \varphi \rceil\!\rceil \quad = \quad F\varphi \mathbin{;} \varepsilon\,. \qquad \text{rad-DEF}$$

To go the other direction we find, dually, from

$$G\psi\colon\ GFA \to_{\mathcal{A}} GB$$

that we need a family of arrows

$$\eta_A\colon\ A \to_{\mathcal{A}} GFA\,,$$

one for each object A of $\mathcal{A}$, so that we can define

$$\lfloor\!\lfloor \psi \rfloor\!\rfloor \quad = \quad \eta \mathbin{;} G\psi\,. \qquad \text{lad-DEF}$$

These ingredients together constitute an adjunction; to summarise, assuming all typings and definitions (of $\lfloor\!\lfloor\ \rfloor\!\rfloor$ and $\lceil\!\lceil\ \rceil\!\rceil$) as above:

an **adjunction** between the functors F and G arises when there exist families ε and η such that $\lfloor\!\lfloor\ \rfloor\!\rfloor$ and $\lceil\!\lceil\ \rceil\!\rceil$ are each other's inverse.	ADJUNCTION

Putting in the definitions of $\lfloor\!\lfloor\ \rfloor\!\rfloor$ and $\lceil\!\lceil\ \rceil\!\rceil$ and some typings, this reads:

$$\begin{array}{lccccl}
\text{in category } \mathcal{A}\text{:} & \varphi & = & \eta \mathbin{;} G\psi & : & A \to_{\mathcal{A}} GB \\
 & \lceil\!\lceil\ \rceil\!\rceil\!\downarrow & & \uparrow\!\lfloor\!\lfloor\ \rfloor\!\rfloor & & \\
\text{in category } \mathcal{B}\text{:} & F\varphi \mathbin{;} \varepsilon & = & \psi & : & FA \to_{\mathcal{B}} B.
\end{array}$$

Discussion. We have not stated what an adjunction *is*. The reason is that —considering $\mathcal{A}$ and $\mathcal{B}$ as fixed— there are altogether six ingredients, namely F, G, ε, η, $\lceil\!\lceil\ \rceil\!\rceil$ and $\lfloor\!\lfloor\ \rfloor\!\rfloor$, which are however not at all independent. In fact, each of those is expressible in terms of the others (as we shall prove later):

$$\begin{array}{rcl@{\qquad}rcl@{\qquad}rcl}
\lceil\!\lceil\varphi\rceil\!\rceil &=& F\varphi \mathbin{;} \varepsilon & \eta &=& \lfloor\!\lfloor id\rfloor\!\rfloor & Ff &=& \lceil\!\lceil f \mathbin{;} \eta\rceil\!\rceil \\
\lfloor\!\lfloor\psi\rfloor\!\rfloor &=& \eta \mathbin{;} G\psi & \varepsilon &=& \lceil\!\lceil id\rceil\!\rceil & Gg &=& \lfloor\!\lfloor\varepsilon \mathbin{;} g\rfloor\!\rfloor .
\end{array} \qquad \text{DEF}$$

In the last two equalities, f is an arbitrary arrow of $\mathcal{A}$, and g of $\mathcal{B}$; these equalities fully determine F and G, since for objects we have $FA =$ the target (or source) of $F id_A$, and similarly for G. Thus there are various alternative, equivalent formulations of ADJUNCTION, but all those are less symmetrical and so have something arbitrary. We shall give proofs in Section 3.

It is standard to call F a **left adjoint** of G, and G a **right adjoint** of F, which corresponds to their placement in the display above as long as the typing arrows are drawn from left to right. Adjoints are unique up to isomorphism, as shown in text 59, so one usually speaks of *the* left adjoint, and *the* right adjoint. The families ε and η are known, respectively, as the **co-unit** and the **unit** of the adjunction; it follows from ADJUNCTION that they are natural transformations. Bijections $\lfloor\!\lfloor\ \rfloor\!\rfloor$ and $\lceil\!\lceil\ \rceil\!\rceil$ are called **lad** and **rad**, respectively, from *left adjunct* and *right adjunct*. As a memory aid: the first of the lad symbols $\lfloor\!\lfloor\ \rfloor\!\rfloor$ has the shape of an 'L'. The bracket notation is not standard in category theory; the terminology 'rad' and 'right adjunct' is used by Mac Lane [4, page 79]. The adjunction property implies that $\lfloor\!\lfloor\ \rfloor\!\rfloor$ and $\lceil\!\lceil\ \rceil\!\rceil$ are natural transformations; see text 62.

8 An example. Here we give an example of an adjunction that should be clear to the functional programming community.

Consider the well-known category $\mathcal{S}et$ of sets and typed total functions, and the lesser known category $\mathcal{M}on$. An object in $\mathcal{M}on$ is, by definition, a *monoid operation* in $\mathcal{S}et$, that is, a binary operation that is associative and has a neutral element; an arrow in $\mathcal{M}on$ from $\oplus$ to $\otimes$ is, by definition, a function f satisfying, for all x, y:

$$\begin{array}{lcl}
f(x \oplus y) &=& (fx) \otimes (fy) \\
f(\text{neutral element of } \oplus) &=& \text{neutral element of } \otimes .
\end{array}$$

Such an f is called a *homomorphism* from $\oplus$ to $\otimes$, denoted $f\colon \oplus \to_{\mathcal{M}on} \otimes$. The composition and identities in $\mathcal{M}on$ are inherited from $\mathcal{S}et$. (Thus defined, $\mathcal{M}on$ satisfies all the category axioms except unique-TYPE; it is a precategory and not a category. In the present context the difference doesn't matter too much.) Notice that each object and arrow in $\mathcal{M}on$ is an arrow in $\mathcal{S}et$ too.

A *sequence* is a finite list of elements of a certain type, denoted $[a_0, \ldots, a_{n-1}]$. The set of sequences over A is denoted $Seq\,A$. Further operations are:

$$\begin{array}{lcl}
tip_A &=& a \mapsto [a] \\
 &:& A \to Seq\,A
\end{array}$$

$$
\begin{array}{lcl}
join_A & = & ([a_0, \ldots, a_{m-1}], [a_m, \ldots, a_{n-1}]) \mapsto [a_0, \ldots, a_{n-1}] \\
 & : & Seq\, A \times Seq\, A \to Seq\, A \\
Seq\, f & = & [a_0, \ldots, a_{n-1}] \mapsto [f\, a_0, \ldots, f\, a_{n-1}] \\
 & : & Seq\, A \to Seq\, B \quad \text{whenever} \quad f\colon\ A \to B \\
\oplus/ & = & [a_0, \ldots, a_{n-1}] \mapsto a_0 \oplus \ldots \oplus a_{n-1} \\
 & : & Seq\, A \to A \ \ \text{for monoid operation} \ \ \oplus\colon\ A \times A \to A
\end{array}
$$

Function $Seq\, f$ is often called $map\, f$. Function $\oplus/$ is called the *reduce-with-*$\oplus$ or the *fold-with-*$\oplus$; the neutral element of $\oplus$ is the outcome on the empty sequence $[\,]$. Associativity of $\oplus$ implies that the specification of $\oplus/$ is unambiguous, not depending on the parenthesisation within $a_0 \oplus \ldots \oplus a_{n-1}$. Notice that each $join_A$ is a monoid operation, its neutral element being the empty sequence $[\,]$.

Here is a law for sequences:

> "each homomorphism on sequences is uniquely determined (as a 'map' followed by a 'reduce') by its restriction to the singleton sequences, and vice versa."

To be precise, the law reads as follows.

> Let A be an arbitrary set, and $\otimes$ be an arbitrary monoid operation, say with target set B. Then, for all $f\colon\ A \to_{\mathcal{S}et} B$ and all $g\colon\ join_A \to_{\mathcal{M}on} \otimes$,
>
> $$f = tip_A \,;\, g \qquad \equiv \qquad Seq\, f \,;\, \otimes/ = g\,. \qquad\qquad \text{SEQADJ}$$

Thus we may call f the 'restriction of g to the tip elements' and write $f = \llbracket g \rrbracket_{A,\otimes} = tip_A \,;\, g$. Also, we may call g the 'extension of f to a homomorphism from $join_A$ to $\otimes$' and write $g = \lceil\!\lceil f \rceil\!\rceil_{A,\otimes} = Seq\, f \,;\, \otimes/$. With these definitions, and omitting the subscripts, the equivalence reads:

$$f = \llbracket g \rrbracket \qquad \equiv \qquad \lceil\!\lceil f \rceil\!\rceil = g\,.$$

This equivalence expresses that $\llbracket\ \rrbracket$ and $\lceil\!\lceil\ \rceil\!\rceil$ are each other's inverse, and constitute a bijection between functions (of a certain type) and homomorphisms (of a certain type). The significance of the equivalence may be evident from the consequences listed below.

Abstracting from the particulars, the situation above is described as follows.

- There are two categories $\mathcal{A}$ and $\mathcal{B}$.
  [In the above example $\mathcal{A} = \mathcal{S}et$ and $\mathcal{B} = \mathcal{M}on$.]

- There are two functors $F\colon\ \mathcal{A} \to \mathcal{B}$ and $G\colon\ \mathcal{B} \to \mathcal{A}$.
  [Above $Ff = Seq\, f$ and $Gg = g$ for arrows f and g. For objects the functors act as follows: $FA = join_A$, and $G(\oplus) =$ the target set of $\oplus$.]

- There are two transformations, $\varepsilon\colon\ FG \to I$ and $\eta\colon\ I \to GF$.
  [Above $\varepsilon_\otimes = \otimes/$ and $\eta_A = tip_A$.]

- There is the bijection $\lfloor\!\lfloor\ \rfloor\!\rfloor_{A,B}$ from arrow collection $(FA \to_{\mathcal{B}} B)$ to $(A \to_{\mathcal{A}} GB)$ (for arbitrary A in $\mathcal{A}$ and B in $\mathcal{B}$) defined by $\lfloor\!\lfloor \psi \rfloor\!\rfloor = \eta \mathbin{;} G\psi$, with inverse $\lceil\!\lceil\ \rceil\!\rceil$ defined by $\lceil\!\lceil \varphi \rceil\!\rceil = F\varphi \mathbin{;} \varepsilon$.
  [Above $\lfloor\!\lfloor g \rfloor\!\rfloor =$ 'the restriction of g to the tip elements', and $\lceil\!\lceil f \rceil\!\rceil =$ 'the extension of f to a homomorphism from *join* to $\otimes$'.]

By definition, such data F, G, ε, η, $\lfloor\!\lfloor\ \rfloor\!\rfloor$, and $\lceil\!\lceil\ \rceil\!\rceil$ constitute an adjunction.

It is rather easy to verify the given adjunction SeqAdj from the explicit definitions we have given for $\mathcal{M}on$, *tip*, *join*, *Seq*, and $\_/$. What makes it interesting is the large number of useful consequences it has. Here is a list of properties each of which can be proved from the adjunction property alone by pure categorical reasoning, without any reference to the explicit definitions of any of *tip*, *join*, *Seq*, and $\_/$:

$$
\begin{array}{lcll}
tip \mathbin{;} Seq\, f & = & f \mathbin{;} tip & \\
Seq\, g \mathbin{;} \otimes/ & = & \oplus/ \mathbin{;} g & \text{whenever } g\colon \oplus \to_{\mathcal{M}on} \otimes \\
tip \mathbin{;} Seq\, f \mathbin{;} \oplus/ & = & f & \\
Seq(tip \mathbin{;} g) \mathbin{;} \oplus/ & = & g & \text{whenever } g\colon join \to_{\mathcal{M}on} \oplus \\
tip \mathbin{;} \oplus/ & = & id & \\
Seq\, tip \mathbin{;} join/ & = & id\,. &
\end{array}
$$

Actually, since left adjoints are unique up to isomorphism, it follows that the "data type of sequences" is characterised by the adjunction property. To be precise, let F' be any left adjoint of the $G\colon \mathcal{M}on \to \mathcal{S}et$ above, say with unit tip' and co-unit $\_/'$ and $join'_A = F'A$. Define functor $Seq' = GF'\colon \mathcal{S}et \to \mathcal{S}et$, that is, $Seq'A =$ the target set of $join'_A$ and $Seq'\,f = F'f$. Then Seq' is isomorphic to Seq, implying amongst others that $join'_A$ is isomorphic to $join_A$ in $\mathcal{M}on$, and that $Seq'\,A$ is isomorphic to $Seq\,A$ in $\mathcal{S}et$, for each A.

3 Calculating with adjunctions

In this section we present the algebraic, calculational properties of an adjunction, and show them in action by proving various claims by calculation.

Theorem 11 asserts the equivalence of several statements. Each of them defines "F is left adjoint to G" and "F has right adjoint G".

So, in order to prove that F is left adjoint to G it suffices to establish just one of the statements, and when you know that F is left adjoint to G you may use *all* of the statements. Before we present the proof of the theorem, we also give some corollaries: additional properties of an adjunction.

9 Global constants. Let $\mathcal{A}$ and $\mathcal{B}$ be categories, and let $F\colon \mathcal{A} \to \mathcal{B}$ and $G\colon \mathcal{B} \to \mathcal{A}$ be functors, fixed throughout the section.

10 Default typing. Unless stated otherwise, variables A', A, f, φ (all in $\mathcal{A}$) and B, B', g, ψ (all in $\mathcal{B}$) are arbitrary, and have the typing indicated below.

$$\begin{array}{lcl@{\qquad\qquad}lcl}
A', A & \in & \text{Objects of } \mathcal{A} & B, B' & \in & \text{Objects of } \mathcal{B} \\
f & : & A' \to_{\mathcal{A}} A & g & : & B \to_{\mathcal{B}} B' \\
\varphi & : & A \to_{\mathcal{A}} GB & \psi & : & FA \to_{\mathcal{B}} B
\end{array}$$

In addition, families $\eta, \llbracket\ \rrbracket$ ("into $\mathcal{A}$") and $\varepsilon, \lceil\!\lceil\ \rrbracket$ ("into $\mathcal{B}$") will depend on F, G and have the following typing.

$$\eta_A \ :\ A \to_{\mathcal{A}} GFA \qquad\qquad \varepsilon_B \ :\ FGB \to_{\mathcal{B}} B \qquad\qquad \text{(co)unit-}\textsc{Type}$$

$$\frac{\psi\colon\ FA \to_{\mathcal{B}} B}{\llbracket\psi\rrbracket_{A,B}\colon\ A \to_{\mathcal{A}} GB} \qquad\qquad \frac{\varphi\colon\ A \to_{\mathcal{A}} GB}{\lceil\!\lceil\varphi\rrbracket_{A,B}\colon\ FA \to_{\mathcal{B}} B} \qquad\qquad \text{lad/rad-}\textsc{Type}$$

11 Theorem. The seven statements Adjunction ... Charns are equivalent. Moreover, the various $\llbracket\ \rrbracket$ that are asserted to exist, can all be chosen equal; the same holds for $\lceil\!\lceil\ \rrbracket$, ε, and η. (Each line is understood to be quantified with '$\forall A, B$' and '$\forall \varphi, \psi$' with A, B and φ, ψ typed as in text 10.)

Adjunction. There exist ε and η typed as in text 10 and satisfying

12 $\qquad \varphi = \eta \,;\, G\psi \quad \equiv \quad F\varphi \,;\, \varepsilon = \psi \qquad$ Adjunction

Units. There exist ε and η typed as in text 10 and satisfying

13 $\qquad \eta\colon\ I \dot{\to}_{\mathcal{A}} GF \qquad$ unit-Ntrf

14 $\qquad \varepsilon\colon\ FG \dot{\to}_{\mathcal{B}} I \qquad$ co-unit-Ntrf

15 $\qquad \eta \,;\, G\varepsilon = id \qquad$ unit-Inv

16 $\qquad F\eta \,;\, \varepsilon = id \qquad$ Inv-co-unit

LadAdj. There exist $\llbracket\ \rrbracket$ and ε typed as in text 10 and satisfying

17 $\qquad \varepsilon\colon\ FG \dot{\to} I \qquad$ co-unit-Ntrf

18 $\qquad Gg \ = \ \llbracket\varepsilon \,;\, g\rrbracket \qquad$ rightadjoint-Def

19 $\qquad \varphi = \llbracket\psi\rrbracket \quad \equiv \quad F\varphi \,;\, \varepsilon = \psi \qquad$ lad-Charn

Moreover, lad-Charn $\Rightarrow$ (co-unit-Ntrf $\equiv$ rightadjoint-Def).

RadAdj. There exist $\lceil\!\lceil\ \rrbracket$ and η typed as in text 10 and satisfying

20 $\qquad \eta\colon\ I \dot{\to} GF \qquad$ unit-Ntrf

21 $\quad Ff \;=\; \lceil\!\lceil f \,;\, \eta \rceil\!\rceil$ leftadjoint-Def

22 $\quad \lceil\!\lceil \varphi \rceil\!\rceil = \psi \quad\equiv\quad \varphi = \eta \,;\, G\psi$ rad-Charn

Moreover, rad-Charn $\Rightarrow$ (unit-Ntrf $\equiv$ leftadjoint-Def).

Fusions. There exist $\lfloor\!\lfloor\ \rfloor\!\rfloor$ and $\lceil\!\lceil\ \rceil\!\rceil$ typed as in text 10 and satisfying

23 $\quad \lfloor\!\lfloor Ff \,;\, \psi \,;\, g \rfloor\!\rfloor \;=\; f \,;\, \lfloor\!\lfloor \psi \rfloor\!\rfloor \,;\, Gg$ lad-Fusion

24 $\quad \lceil\!\lceil f \,;\, \varphi \,;\, Gg \rceil\!\rceil \;=\; Ff \,;\, \lceil\!\lceil \varphi \rceil\!\rceil \,;\, g$ rad-Fusion

25 $\quad \varphi = \lfloor\!\lfloor \psi \rfloor\!\rfloor \;\equiv\; \lceil\!\lceil \varphi \rceil\!\rceil = \psi$ Inverse

Moreover, Inverse $\Rightarrow$ (lad-Fusion $\equiv$ rad-Fusion).

Fusions′. There exist $\lfloor\!\lfloor\ \rfloor\!\rfloor$ and $\lceil\!\lceil\ \rceil\!\rceil$ typed as in text 10 and satisfying

26 $\quad \lfloor\!\lfloor \psi \rfloor\!\rfloor \;=\; \lfloor\!\lfloor id \rfloor\!\rfloor \,;\, G\psi$ lad-Fusion′

27 $\quad \lceil\!\lceil \varphi \rceil\!\rceil \;=\; F\varphi \,;\, \lceil\!\lceil id \rceil\!\rceil$ rad-Fusion′

28 $\quad \varphi = \lfloor\!\lfloor \psi \rfloor\!\rfloor \;\equiv\; \lceil\!\lceil \varphi \rceil\!\rceil = \psi$ Inverse

Charns. There exist $\lfloor\!\lfloor\ \rfloor\!\rfloor$ and $\lceil\!\lceil\ \rceil\!\rceil$, and ε, η typed as in 10 and satisfying

29 $\quad \varphi = \lfloor\!\lfloor \psi \rfloor\!\rfloor \;\equiv\; F\varphi \,;\, \varepsilon = \psi$ lad-Charn

30 $\quad \lceil\!\lceil \varphi \rceil\!\rceil = \psi \;\equiv\; \varphi = \eta \,;\, G\psi$ rad-Charn

31 $\quad \varphi = \lfloor\!\lfloor \psi \rfloor\!\rfloor \;\equiv\; \lceil\!\lceil \varphi \rceil\!\rceil = \psi$ Inverse

32 Corollary. Let F be left adjoint to G via $\varepsilon, \eta, \lfloor\!\lfloor\ \rfloor\!\rfloor, \lceil\!\lceil\ \rceil\!\rceil$. Then:

33 $\quad \eta = \lfloor\!\lfloor id \rfloor\!\rfloor$ unit-Def

34 $\quad \lfloor\!\lfloor \psi \rfloor\!\rfloor = \eta \,;\, G\psi$ lad-Def

35 $\quad F\lfloor\!\lfloor \psi \rfloor\!\rfloor \,;\, \varepsilon = \psi$ lad-Self

36 $\quad F\varphi \,;\, \varepsilon = F\varphi' \,;\, \varepsilon \;\Rightarrow\; \varphi = \varphi'$ lad-Uniq

37 $\quad \varepsilon = \lceil\!\lceil id \rceil\!\rceil$ co-unit-Def

38 $\quad \lceil\!\lceil \varphi \rceil\!\rceil = F\varphi \,;\, \varepsilon$ rad-Def

39 $\quad \eta \,;\, G\lceil\!\lceil \varphi \rceil\!\rceil = \varphi$ rad-Self

40 $\quad \eta \,;\, G\psi = \eta \,;\, G\psi' \;\Rightarrow\; \psi = \psi'$ rad-Uniq

41 Discussion.

Memorisation. A quick glance at the formulas of the theorem and the corollary reveals that the two composite expressions of ADJUNCTION turn up over and over again, namely

$\eta \,;\, G\psi$	possibly with a specific ψ, and
$F\varphi \,;\, \varepsilon$	possibly with a specific φ.

This observation may help to memorise the formulas. They also make clear that the *target* type of η begins with 'G', and the *source* type of ε begins with 'F'. A further consistency in the shape of the formulas is this: within each arrow expression and each arrow equation, φ and η, being entities in $\mathcal{A}$, occur to the left of ψ and ε, being entities in $\mathcal{B}$.

The definition for $\llbracket\ \rrbracket$ and $\lceil\!\lceil\ \rceil\!\rceil$, formula 34 and 38, can be read off directly from ADJUNCTION; it is then also immediate that $\llbracket\ \rrbracket$ and $\lceil\!\lceil\ \rceil\!\rceil$ are each other's inverse, as expressed by law INVERSE. Also, the left-hand side of unit-INV, formula 15, has the shape of '$\eta \,;\, G\psi$', namely with $\psi := \varepsilon$. The main equations of laws lad- and rad-CHARN are the same as the two equations of ADJUNCTION.

Universality. Another reading of ADJUNCTION is this: for each φ there is precisely one solution for ψ in the left-hand side equation ($\varphi = \eta \,;\, G\psi$), namely the ψ given by the right-hand side equation; and, also, for each ψ there is precisely one solution for φ in the right-hand side equation ($F\varphi \,;\, \varepsilon = \psi$), namely the one given by the left-hand side equation. The existence of a unique solution is also expressed by lad-CHARN and rad-CHARN separately, and the solutions themselves are given by $\llbracket\ \rrbracket$ and $\lceil\!\lceil\ \rceil\!\rceil$. So, in the terminology of text 4, η_A is P_A-universal (for G and A), where $P_A(\eta, \varphi, \psi) = (\varphi = \eta_A \,;\, G\psi)$. Dually for ε. The laws SELF and UNIQ in the corollary are thus instantiations of universality in general. The FUSION laws are fusion laws according to the explanation in text 4.

'Moreover'. The equivalence asserted in the 'Moreover' part of FUSIONS implies that, to check for adjunctionhood, it suffices to check for INVERSE and just one of lad- and rad-FUSION; the other is then true as well. A similar remark holds for LADADJ and RADADJ.

Inv, Self, Uniq. The two INV laws assert that ε and η have a pre-inverse and post-inverse, respectively. Law lad-UNIQ asserts a kind of monic-ness for ε, and rad-UNIQ asserts a kind of epic-ness for η. Law lad-SELF shows that the effect of $\llbracket\ \rrbracket$ can be undone; indeed, the definition of its inverse, $\lceil\!\lceil\ \rceil\!\rceil$, follows the pattern of the left-hand side of lad-SELF.

42 G **determines** F. Consider statement RADADJ. Suppose that functor G is given and rad-CHARN holds for some η and $\lceil\!\lceil\ \rceil\!\rceil$. This does not require F to be known on arrows: in the formulation F is nowhere applied to arrows (but F is applied on objects in the typing of η). The theorem then implies that F can be extended to a functor in such a way that it is a left adjoint of G. Indeed, by defining F on arrows as in leftadjoint-DEF the requirement leftadjoint-DEF is *by definition* satisfied, and it is also easily shown that the F thus defined is a functor. Notice that the naturality of η is not required a priori; it is not meaningful either if F is not yet defined on arrows.

In fact, F need not be known on objects either; it may be completely unknown. Satisfaction of rad-CHARN in RADADJ (with a suitable typing not involving F) is already sufficient to define a left adjoint of G. We shall make this precise. First recall that an equivalent formulation of rad-CHARN is: 'for each φ there is precisely one ψ with $\varphi = \eta \mathbin{;} G\psi$'. We'll use this formulation here for brevity's sake. Now consider the following statement.

RIGHTADJOINT. For each A there exists an object B_A and an arrow η_A: $A \rightarrow_{\mathcal{A}} GB_A$ satisfying

43 $\quad \forall B \;\; \forall \varphi\colon A \rightarrow_{\mathcal{A}} GB \;\; \exists!\psi\colon B_A \rightarrow_{\mathcal{B}} B :: \quad \varphi = \eta \mathbin{;} G\psi$ $\qquad$ rad-CHARN′

We claim that

44 $\quad G$ has a left adjoint if, and only if, RIGHTADJOINT holds.

The difference between RIGHTADJOINT and RADADJ is mainly a matter of skolemisation (that is, the replacement of a part $\forall x \exists y$ by $\exists F \forall x$ and substituting Fx for y):

$$
\begin{array}{lll}
 & & \text{RIGHTADJOINT} \\
 & \equiv & \quad \text{definition} \\
 & & \forall A \;\; \exists B_A \;\; \exists \eta_A{:}A{\rightarrow_{\mathcal{A}}}GB_A \;\; \forall B \;\; \forall \varphi{:}A{\rightarrow_{\mathcal{A}}}GB \;\; \exists!\psi{:}B_A{\rightarrow_{\mathcal{B}}}B :: \quad \varphi = \eta \mathbin{;} G\psi \\
 & \equiv & \quad \text{skolemisation (replacing } B_A \leftrightarrow FA\text{)} \\
(*) & & \exists F \;\; \forall A \;\; \exists \eta_A{:}A{\rightarrow_{\mathcal{A}}}GFA \;\; \forall B \;\; \forall \varphi{:}A{\rightarrow_{\mathcal{A}}}GB \;\; \exists!\psi{:}FA{\rightarrow_{\mathcal{B}}}B :: \quad \varphi = \eta \mathbin{;} G\psi \\
 & \equiv & \quad \text{skolemisation (replacing } \eta_A \leftrightarrow (\eta)_A\text{)} \\
 & & \exists F \;\; \exists \eta\text{: `see txt 10'} \;\; \forall A \;\; \forall B \;\; \forall \varphi{:}A{\rightarrow_{\mathcal{A}}}GB \;\; \exists!\psi{:}FA{\rightarrow_{\mathcal{B}}}B :: \quad \varphi = \eta \mathbin{;} G\psi \\
 & \equiv & \quad \text{definition RADADJ,} \\
 & & \quad \text{for `} \Rightarrow \text{' defining } F \text{ on arrows by leftadjoint-DEF} \\
(\star) & & \exists F \;\; \text{RADADJ.}
\end{array}
$$

The F in line $(*)$ is merely a mapping from objects to objects, while the F in line $(\star)$ is a functor (mapping also arrows to arrows).

Proof of the theorem. We shall now prove the theorem by circular implication:

$\cdots$ ADJUNCTION $\Rightarrow$ UNITS $\Rightarrow$ LADADJ $\Rightarrow$ FUSIONS $\Rightarrow$ RADADJ $\Rightarrow$ CHARNS $\Rightarrow$ $\cdots$

and separately FUSIONS $\equiv$ FUSIONS′. The 'Moreover' parts of LADADJ, RADADJ, and FUSIONS are proved along the way. We urge the readers to try and prove some of the implications themselves, before reading all of the proofs below. It is an excellent exercise to become familiar with the calculational properties of an adjunction.

45 Proof of ADJUNCTION $\Rightarrow$ UNITS. We establish co-unit-NTRF; and unit-INV along the way at line (∗).

$$\begin{array}{lll}
 & & \varepsilon\colon\ FG \dot{\to}_{\mathcal{B}} I \\
 & \equiv & \quad \text{definition of naturality:} \\
 & & \quad \text{For all } g\colon\ B \to_{\mathcal{B}} B' \\
 & & FGg \,;\, \varepsilon_{B'} = \varepsilon_B \,;\, g \\
 & \equiv & \quad \text{ADJUNCTION}[\varphi, \psi := Gg, (\varepsilon \,;\, g)] \text{ (from right to left)} \\
 & & Gg = \eta \,;\, G(\varepsilon \,;\, g) \\
 & \equiv & \quad \text{functor} \\
 & & Gg = \eta \,;\, G\varepsilon \,;\, Gg \\
 & \Leftarrow & \quad \text{Leibniz} \\
(*) & & id = \eta \,;\, G\varepsilon \qquad\qquad \text{(unit-INV)} \\
 & \equiv & \quad \text{ADJUNCTION}[\varphi, \psi := id, \varepsilon] \text{ (from left to right)} \\
 & & F\,id \,;\, \varepsilon = \varepsilon \\
 & \equiv & \quad \text{functor, identity} \\
 & & true.
\end{array}$$

Dually for unit-NTRF and INV-co-unit.

46 Proof of UNITS $\Rightarrow$ LADADJ. We establish the equivalence LADCHARN by circular 'follows from', defining the unknown $\llbracket\ \rrbracket$ along the way. We start with the side that doesn't contain the unknown.

$$\begin{array}{ll}
 & F\varphi \,;\, \varepsilon = \psi \\
\equiv & \quad \text{INV-co-unit} \\
 & F\varphi \,;\, \varepsilon = F\eta \,;\, \varepsilon \,;\, \psi \\
\equiv & \quad \text{co-unit-NTRF} \\
 & F\varphi \,;\, \varepsilon = F\eta \,;\, FG\psi \,;\, \varepsilon \\
\Leftarrow & \quad \text{functor, Leibniz} \\
 & \varphi = \eta \,;\, G\psi, \qquad = \llbracket\psi\rrbracket \ \text{ by } \textbf{defining } \llbracket\psi\rrbracket = \eta \,;\, G\psi \qquad \text{(left-hand side)} \\
\equiv & \quad \text{unit-INV} \\
 & \varphi \,;\, \eta \,;\, G\varepsilon = \eta \,;\, G\psi \\
\equiv & \quad \text{unit-NTRF} \\
 & \eta \,;\, GF\varphi \,;\, G\varepsilon = \eta \,;\, G\psi \\
\Leftarrow & \quad \text{functor, Leibniz} \\
 & F\varphi \,;\, \varepsilon = \psi.
\end{array}$$

Actually, the above calculation also shows UNITS $\Rightarrow$ ADJUNCTION.
Law co-unit-NTRF is already part of UNITS, and therefore trivially true. For the 'Moreover'

part lad-Charn $\Rightarrow$ (co-unit-Ntrf $\equiv$ rightadjoint-Def), we argue:

$$
\begin{array}{ll}
 & \varepsilon\colon\ FG \dot{\to} I \\
\equiv & \quad \text{definition of naturality} \\
 & \quad \text{For all } g: \\
 & FGg \mathbin{;} \varepsilon = \varepsilon \mathbin{;} g \\
\equiv & \quad \text{lad-Charn}[\varphi, \psi := Gg,\ \varepsilon \mathbin{;} g] \\
 & Gg = \llbracket \varepsilon \mathbin{;} g \rrbracket .
\end{array}
$$

47 Proof of LadAdj $\Rightarrow$ Fusions. We establish lad-Fusion as follows:

$$
\begin{array}{ll}
 & \llbracket Ff \mathbin{;} \psi \mathbin{;} g \rrbracket = f \mathbin{;} \llbracket \psi \rrbracket \mathbin{;} Gg \\
\equiv & \quad \text{lad-Charn}\,[\varphi, \psi := \text{rhs},\ \text{lhs}] \\
 & Ff \mathbin{;} \psi \mathbin{;} g = F(f \mathbin{;} \llbracket \psi \rrbracket \mathbin{;} Gg) \mathbin{;} \varepsilon \\
\equiv & \quad \text{functor, co-unit-Ntrf} \\
 & Ff \mathbin{;} \psi \mathbin{;} g = Ff \mathbin{;} F\llbracket \psi \rrbracket \mathbin{;} \varepsilon \mathbin{;} g \\
\Leftarrow & \quad \text{Leibniz} \\
 & \psi = F\llbracket \psi \rrbracket \mathbin{;} \varepsilon \\
\equiv & \quad \text{lad-Charn}[\varphi := \llbracket \psi \rrbracket\,] \\
 & \llbracket \psi \rrbracket = \llbracket \psi \rrbracket \\
\equiv & \quad \text{equality} \\
 & \mathit{true}.
\end{array}
$$

We establish Inverse, defining $\lceil\!\lceil\ \rceil\!\rceil$ along the way:

$$
\begin{array}{ll}
 & \varphi = \llbracket \psi \rrbracket \\
\equiv & \quad \text{lad-Charn} \\
 & F\psi \mathbin{;} \varepsilon = \psi \\
\equiv & \quad \textbf{define } \lceil\!\lceil \psi \rceil\!\rceil = F\psi \mathbin{;} \varepsilon \\
 & \lceil\!\lceil \varphi \rceil\!\rceil = \psi .
\end{array}
$$

For the 'Moreover' part Inverse $\Rightarrow$ (lad-Fusion $\equiv$ rad-Fusion), we argue:

$$
\begin{array}{ll}
 & \llbracket Ff \mathbin{;} \psi \mathbin{;} g \rrbracket = f \mathbin{;} \llbracket \psi \rrbracket \mathbin{;} Gg \\
\equiv & \quad \text{Inverse} \\
 & Ff \mathbin{;} \psi \mathbin{;} g = \lceil\!\lceil f \mathbin{;} \llbracket \psi \rrbracket \mathbin{;} Gg \rceil\!\rceil \\
\equiv & \quad \text{for `} \Rightarrow \text{' substitute } \psi := \lceil\!\lceil \varphi \rceil\!\rceil \text{ (hence by Inverse } \llbracket \psi \rrbracket = \varphi \text{), and} \\
 & \quad \text{for `} \Leftarrow \text{' substitute } \varphi := \llbracket \psi \rrbracket \text{ (hence by Inverse } \lceil\!\lceil \varphi \rceil\!\rceil = \psi \text{)} \\
 & Ff \mathbin{;} \lceil\!\lceil \varphi \rceil\!\rceil \mathbin{;} g = \lceil\!\lceil f \mathbin{;} \varphi \mathbin{;} Gg \rceil\!\rceil .
\end{array}
$$

Now rad-Fusion follows by this result.

48 Proof of FUSIONS $\Rightarrow$ RADADJ. First we establish rad-CHARN defining the unknown η along the way. We start with the side that doesn't contain the unknown:

$$\begin{array}{ll} & \llbracket \varphi \rrbracket = \psi \\ \equiv & \quad \text{INVERSE} \\ & \varphi = \llbracket \psi \rrbracket \\ \equiv & \quad \text{lad-FUSION} \\ & \varphi = \llbracket id \rrbracket \mathbin{;} G\psi \\ \equiv & \quad \textbf{define } \eta = \llbracket id \rrbracket \\ & \varphi = \eta \mathbin{;} G\psi . \end{array}$$

Now we establish leftadjoint-DEF:

$$\begin{array}{ll} & Ff = \llbracket f \mathbin{;} \eta \rrbracket \\ \equiv & \quad \text{rad-FUSION} \\ & Ff = Ff \mathbin{;} \llbracket \eta \rrbracket \\ \Leftarrow & \quad \text{Leibniz} \\ & id = \llbracket \eta \rrbracket \\ \equiv & \quad \text{INVERSE} \\ & \llbracket id \rrbracket = \eta \\ \equiv & \quad \text{defined above} \\ & true. \end{array}$$

The proof of the 'Moreover' part is dual to the one for LADADJ above. Hence unit-NTRF is true; it is also easy to prove unit-NTRF directly.

49 Proof of RADADJ $\Rightarrow$ CHARNS. First we establish INVERSE, defining $\llbracket\ \rrbracket$ along the way:

$$\begin{array}{ll} & \llbracket \varphi \rrbracket = \psi \\ \equiv & \quad \text{rad-CHARN} \\ & \varphi = \eta \mathbin{;} G\psi \\ \equiv & \quad \textbf{define } \llbracket \psi \rrbracket = \eta \mathbin{;} G\psi \\ & \varphi = \llbracket \psi \rrbracket . \end{array}$$

Next we establish lad-CHARN, defining the unknown ε along the way. We start with the side that doesn't contain the unknown:

$$\begin{array}{lll} & & \varphi = \llbracket \psi \rrbracket \\ & \equiv & \quad \text{INVERSE (just derived)} \\ & & \llbracket \varphi \rrbracket = \psi \\ (*) & \equiv & \quad \text{rad-FUSION (see below)} \end{array}$$

$$
\begin{array}{ll}
 & F\varphi \mathbin{;} \llbracket id \rrbracket = \psi \\
\equiv & \qquad \textbf{define } \varepsilon = \llbracket id \rrbracket \\
 & F\varphi \mathbin{;} \varepsilon = \psi .
\end{array}
$$

In step $(*)$ we have used rad-FUSION. This law follows from RADADJ in the same way as lad-FUSION follows from LADADJ, see text 47.

50 Proof of CHARNS $\Rightarrow$ ADJUNCTION.

$$
\begin{array}{ll}
 & \varphi = \eta \mathbin{;} G\psi \\
\equiv & \qquad \text{rad-CHARN} \\
 & \psi = \llbracket \varphi \rrbracket \\
\equiv & \qquad \text{INVERSE} \\
 & \llbracket \psi \rrbracket = \varphi \\
\equiv & \qquad \text{lad-CHARN} \\
 & F\varphi \mathbin{;} \varepsilon = \psi .
\end{array}
$$

51 Proof of FUSIONS $\equiv$ FUSIONS$'$. The implication $\Rightarrow$ is immediate by instantiating lad-FUSION with $f, \psi := id, id$, and similarly for rad. We prove the 'follows from' $\Leftarrow$ as follows. First an auxiliary result:

$$
\begin{array}{ll}
 & \llbracket \psi \mathbin{;} g \rrbracket = \llbracket \psi \rrbracket \mathbin{;} Gg \\
\equiv & \qquad \text{lad-FUSION}' \\
 & \llbracket id \rrbracket \mathbin{;} G(\psi \mathbin{;} g) = \llbracket id \rrbracket \mathbin{;} G\psi \mathbin{;} Gg \\
\equiv & \qquad \text{functor} \\
 & \mathit{true},
\end{array}
$$

and dually for rad. Now, for lad-FUSION we argue:

$$
\begin{array}{ll}
 & \llbracket Ff \mathbin{;} \psi \mathbin{;} g \rrbracket = f \mathbin{;} \llbracket \psi \rrbracket \mathbin{;} Gg \\
\equiv & \qquad \text{just derived for lad} \\
 & \llbracket Ff \mathbin{;} \psi \rrbracket \mathbin{;} Gg = f \mathbin{;} \llbracket \psi \rrbracket \mathbin{;} Gg \\
\Leftarrow & \qquad \text{Leibniz} \\
 & \llbracket Ff \mathbin{;} \psi \rrbracket = f \mathbin{;} \llbracket \psi \rrbracket \\
\equiv & \qquad \text{INVERSE} \\
 & Ff \mathbin{;} \psi = \llbracket f \mathbin{;} \llbracket \psi \rrbracket \rrbracket \\
\equiv & \qquad \text{just derived for rad} \\
 & Ff \mathbin{;} \psi = Ff \mathbin{;} \llbracket \llbracket \psi \rrbracket \rrbracket \\
\Leftarrow & \qquad \text{Leibniz}
\end{array}
$$

$$
\begin{array}{ll}
 & \psi = \lceil\!\lceil \llbracket \psi \rrbracket \rceil\!\rceil \\
\equiv & \qquad \text{INVERSE} \\
 & \llbracket \psi \rrbracket = \llbracket \psi \rrbracket \\
\equiv & \qquad \text{equality} \\
 & \mathit{true}.
\end{array}
$$

Dually for rad-FUSION.

This completes the proof of Theorem 11.

52 Proof of Corollary 32. Laws SELF and UNIQ are just instantiations of the SELF and UNIQ laws for universality in general as explained in text 4. So they need no proof here. For unit-DEF we argue:

$$
\begin{array}{ll}
 & \eta = \llbracket \mathit{id} \rrbracket \\
\equiv & \qquad \text{lad-CHARN} \\
 & F\eta \,;\, \varepsilon = \mathit{id} \\
\equiv & \qquad \text{INV-co-unit} \\
 & \mathit{true}.
\end{array}
$$

For lad-DEFwe argue:

$$
\begin{array}{ll}
 & \llbracket \psi \rrbracket = \eta \,;\, G\psi \\
\equiv & \qquad \text{ADJUNCTION}[\,\varphi := \llbracket \psi \rrbracket\,] \\
 & \psi = F \llbracket \psi \rrbracket \,;\, \varepsilon \\
\equiv & \qquad \text{lad-SELF} \\
 & \mathit{true}.
\end{array}
$$

The other parts are proved dually.

Exercises

53 Exc. For each $\mathcal{X}, \mathcal{Y} \in \{\,\text{ADJUNCTION} \ldots \text{CHARNS}\,\}$, see whether you can prove $\mathcal{X} \equiv \mathcal{Y}$ or $\mathcal{X} \Rightarrow \mathcal{Y}$ directly, without relying on Theorem 11. There are a lot of possibilities!! In particular do this with $\mathcal{X} =$ ADJUNCTION.

54 Exc. Give alternative proofs for each of the corollaries. Again there are a lot of possibilities.

55 Exc. Barr and Wells [2] present RADADJ as a definition of "F is adjoint to G", and they prove LADADJ as a proposition. Compare our calculational proof of LADADJ $\Rightarrow$ RADADJ with the two-and-a-half page proof of Barr and Wells (Proposition 12.2.2, containing eight diagrams).

56 Exc. Derive the typing (and the subscripts to $\llbracket\ \rrbracket$, $\lceil\!\lceil\ \rceil\!\rceil$, η, and ε) for each of the laws, following the procedure of text 3.

57 Exc. Let F be left-adjoint to G via ε, η and also via ε', η. Prove that $\varepsilon = \varepsilon'$.

58 Exc. Find F and G such that F is left-adjoint to G via ε, η as well as via ε', η' with $(\varepsilon, \eta) \neq (\varepsilon', \eta')$. (Hint: take $F = G = I$, and $\mathcal{A} = \mathcal{B} =$ a category with one object and two morphisms.) So an adjointness does not determine the unit and co-unit uniquely.

59 Exc. Suppose that F and F' are both left-adjoint to G. Prove that $F \cong F'$ (in category $\mathcal{F}tr(\mathcal{A}, \mathcal{B})$), that is, there exist natural transformations $\kappa\colon F \dot{\to} F'$ and $\kappa'\colon F' \dot{\to} F$ that are each other's inverse. (Hint: first calculate κ from the naturality requirement, and by symmetry also κ'; then show that $\kappa \,;\, \kappa' = id$ and, by symmetry, $\kappa' \,;\, \kappa = id$.) Conclude that κ and κ' are, in general, not uniquely determined by F, F', G. (Hint: see Exercise 58.)

60 Exc. Instantiate the properties of adjunctions in general to the case of sequences, discussed in text 8, and simplify the formulas so as to get the additional properties listed at the end of text 8. □

Yet another formulation

Here is the formulation of "F is adjoint to G" that we already hinted at when we generalised Galois connections to adjunctions in text 6. We first need some notation.

61 Notation. We introduce three pieces of notation. First, for arbitrary category $\mathcal{C}$ we define the two-place mapping $(\_ \to \_)$ by:

$$\begin{array}{lcll} (C \to_{\mathcal{C}} C') & = & \{h \text{ in } \mathcal{C} \mid \quad h\colon C \to_{\mathcal{C}} C'\}, & \quad \text{an object in } \mathcal{S}et \\ (h \to_{\mathcal{C}} h') & = & \lambda\chi. \quad h \,;\, \chi \,;\, h', & \quad \text{an arrow in } \mathcal{S}et \text{ typed} \\ & & (\text{tgt}\, h \to \text{src}\, h') \to_{\mathcal{S}et} (\text{src}\, h \to \text{tgt}\, h') & \end{array}$$

It follows that $(\_ \to_{\mathcal{C}} \_)$ is a (bi)functor, contravariant in its first parameter since $\text{src}\, h$ and $\text{tgt}\, h$ change place in the source and target type of $(h \to_{\mathcal{C}} h')$:

$$(\_ \to_{\mathcal{C}} \_) \quad : \quad \mathcal{C}^{op} \times \mathcal{C} \to \mathcal{S}et\,.$$

This functor is called the **hom-functor**, and is usually written $Hom_{\mathcal{C}}(\_,\_)$. Second, for bifunctor $\_ \oplus \_$, like $(\_ \to_{\mathcal{C}} \_)$, and functors F, G, we write $F \oplus G$ for the functor $x \mapsto Fx \oplus Gx$. Third, here and in the next theorem, let

$$\begin{array}{l} X\colon \mathcal{A}^{op} \times \mathcal{B} \to \mathcal{A}^{op} \\ Y\colon \mathcal{A}^{op} \times \mathcal{B} \to \mathcal{B} \end{array}$$

denote the obvious projection functors.

With all this notation, mappings $(X \to_{\mathcal{A}} GY)$ and $(FX \to_{\mathcal{B}} Y)$ are functors of type $\mathcal{A}^{op} \times \mathcal{B} \to \mathcal{S}et$ that satisfy the following equations:

$$(X \to_{\mathcal{A}} GY)(A, B) \quad = \quad (A \to_{\mathcal{A}} GB) \quad = \quad \{f \text{ in } \mathcal{A} \mid \quad f\colon A \to_{\mathcal{A}} GB\}$$

$$(X \to_{\scriptscriptstyle A} GY)(f,g) \quad = \quad (f \to_{\scriptscriptstyle A} Gg) \quad = \quad \lambda\varphi.\quad f \,;\, \varphi \,;\, Gg$$

and

$$\begin{array}{lllll}(FX \to_{\scriptscriptstyle B} Y)(A,B) & = & (FA \to_{\scriptscriptstyle B} B) & = & \{g \text{ in } \mathcal{B} \mid \; g\colon\; FA \to_{\scriptscriptstyle B} B\} \\ (FX \to_{\scriptscriptstyle B} Y)(f,g) & = & (Ff \to_{\scriptscriptstyle B} g) & = & \lambda\psi.\quad Ff \,;\, \psi \,;\, g\,.\end{array}$$

62 Theorem. The statement "F is left adjoint to G" is equivalent to IsoAdj.

IsoAdj.

63 $\qquad (X \to_{\scriptscriptstyle A} GY) \quad \cong \quad (FX \to_{\scriptscriptstyle B} Y)$ $\qquad$ Iso

Proof. The isomorphism in Iso is apparently in the category where functors are the objects and natural transformations are the morphisms. So Iso abbreviates the following:

there exist natural transformations

64 $\qquad \lceil\!\lceil\,\rceil\!\rceil_{\_,\_} \;:\; (X \to_{\scriptscriptstyle A} GY) \;\dot{\to}\; (FX \to_{\scriptscriptstyle B} Y)$ $\qquad$ rad-Ntrf

65 $\qquad \lfloor\!\lfloor\,\rfloor\!\rfloor_{\_,\_} \;:\; (FX \to_{\scriptscriptstyle B} Y) \;\dot{\to}\; (X \to_{\scriptscriptstyle A} GY)$ $\qquad$ lad-Ntrf

66 $\qquad$ that are each other's inverse. $\qquad$ Inverse

Now it is sufficient to show that these three statements are equivalent to Fusions. Law Inverse is the same as in Fusions, and lad-Ntrf is equivalent to lad-Fusion:

$$\begin{array}{ll} & \lfloor\!\lfloor\,\rfloor\!\rfloor\colon\; (FX \to Y) \;\dot{\to}\; (X \to GY) \\ \equiv & \quad \text{definition of naturality} \\ & \quad \text{For all } (f,g)\colon\, (A,B) \to (A',B') \text{ in } \mathcal{A}^{op} \times \mathcal{B}: \\ & (FX \to Y)(f,g) \,;\, \lfloor\!\lfloor\,\rfloor\!\rfloor_{A',B'} = \lfloor\!\lfloor\,\rfloor\!\rfloor_{A,B} \,;\, (X \to GY) \\ \equiv & \quad \text{property } (FX \to Y)(f,g) = (Ff \to g) \text{ and similarly for } G \\ & (Ff \to g) \,;\, \lfloor\!\lfloor\,\rfloor\!\rfloor_{A',B'} = \lfloor\!\lfloor\,\rfloor\!\rfloor_{A,B} \,;\, (f \to Gg) \\ \equiv & \quad \text{extensionality (in } \mathcal{S}et\text{)} \\ & \quad \text{For all } \psi \in (FA \to B): \\ & ((Ff \to g) \,;\, \lfloor\!\lfloor\,\rfloor\!\rfloor_{A',B'})\psi = (\lfloor\!\lfloor\,\rfloor\!\rfloor_{A,B} \,;\, (f \to Gg))\psi \\ \equiv & \quad \text{composition applied: } (\mathcal{F} \,;\, \mathcal{G})x = \mathcal{G}(\mathcal{F}x) \\ & \lfloor\!\lfloor\,\rfloor\!\rfloor_{A',B'}((Ff \to g)\psi) = (f \to Gg)\,(\lfloor\!\lfloor\,\rfloor\!\rfloor_{A,B}\psi) \\ \equiv & \quad \text{definition of hom-functor } (\_ \to \_)\text{, writing } \lfloor\!\lfloor\,\rfloor\!\rfloor_{\_,\_}xyz \text{ as } \lfloor\!\lfloor xyz \rfloor\!\rfloor_{\_,\_} \\ & \lfloor\!\lfloor Ff \,;\, \psi \,;\, g \rfloor\!\rfloor_{A',B'} = f \,;\, \lfloor\!\lfloor \psi \rfloor\!\rfloor_{A,B} \,;\, Gg. \end{array}$$

Dually for rad.

4 Initiality, colimits and adjointness

In this section we prove some results about initiality, colimits and adjunctions. We assume familiarity with the notion of initiality. Following the discussion about universality in general, text 4, we phrase the definition as follows. Let $\mathcal{A}$ be a category, and A an object in $\mathcal{A}$. Then A is **initial** in $\mathcal{A}$ if: there exists a mapping $(\![\;]\!)_{\mathcal{A},A}$ (from objects to arrows), satisfying

$$f = (\![B]\!)_{\mathcal{A},A} \quad \equiv \quad f\colon\ A \to_{\mathcal{A}} B \qquad\qquad \text{med-}\textsc{Charn}$$

Arrow $(\![B]\!)_{\mathcal{A},A}$ is called the mediating arrow, and $(\![\;]\!)_{\mathcal{A},A}$ is called the mediator. We omit the subscripts if no confusion can arise. In the terminology of text 4, A is initial iff it is P-universal, where $P(A,B,f) \equiv (f\colon A{\to}B)$. **Finality** is defined dually, this time using the notation $[\!(\;)\!]$ for the mediator. The notation $¡_B$ and $!_B$ is often used for $(\![B]\!)$ and $[\!(B)\!]$.

67 Left-adjoints preserve initiality. Let $\mathcal{A},\mathcal{B}$ be arbitrary categories, and suppose that $\mathcal{A}$ has an initial object 0 and that $\mathcal{A},\mathcal{B},F,G,\varepsilon,\eta,\lfloor\!\lfloor\;\rfloor\!\rfloor,\lceil\!\lceil\;\rceil\!\rceil$ is an adjunction. We claim that $F0$ is initial in $\mathcal{B}$. To prove this, we establish the equivalence med-$\textsc{Charn}\,[f,A,B,(\![\;]\!)_{\mathcal{A},A} := g,F0,B,(\![\;]\!)_{\mathcal{B},F0}]$, constructing the unknown $(\![\_]\!)_{\mathcal{B},F0}$ along the way. We start with the side that doesn't contain the unknown.

$$\begin{array}{ll}
& g\colon\ F0 \to_{\mathcal{B}} B \\
\equiv & \qquad \text{typing rules for } \lfloor\!\lfloor\;\rfloor\!\rfloor \\
& \lfloor\!\lfloor g\rfloor\!\rfloor\colon\ 0 \to_{\mathcal{A}} GB \\
\equiv & \qquad \text{med-}\textsc{Charn}\,[f,A,B := \lfloor\!\lfloor g\rfloor\!\rfloor, 0, GB] \text{ in } \mathcal{A} \\
& \lfloor\!\lfloor g\rfloor\!\rfloor = (\![GB]\!)_{\mathcal{A},A} \\
\equiv & \qquad \textsc{Inverse} \\
& g = \lceil\!\lceil (\![GB]\!)_{\mathcal{A},A} \rceil\!\rceil \\
\equiv & \qquad \textbf{defining } (\![B]\!)_{\mathcal{B},F0} = \lceil\!\lceil (\![GB]\!)_{\mathcal{A},A} \rceil\!\rceil \\
& g = (\![B]\!)_{\mathcal{B},F0}.
\end{array}$$

68 Initiality as a special adjointness. Let $\mathcal{B}$ be a category with an initial object 0. Then, for each category $\mathcal{A}$ with a final object 1, there is an adjunction between $\mathcal{A}$ and $\mathcal{B}$ (from which object 0 can be retrieved).
Proof. Let F and G be the constant functors $F = \underline{0}\colon\ \mathcal{A} \to \mathcal{B}$ and $G = \underline{1}\colon\ \mathcal{B} \to \mathcal{A}$. We claim that F is left-adjoint to G. To prove this, we establish $\textsc{Adjunction}$, constructing ε and η along the way. For arbitrary A,B and $f\colon\ A \to_{\mathcal{A}} GB$ and $g\colon\ FA \to_{\mathcal{B}} B$,

$$\begin{array}{ll}
& f = \eta_A \,;\, Gg \quad \equiv \quad Ff \,;\, \varepsilon_B = g \\
\equiv & \qquad \text{definition of } F \text{ and } G\text{, identity}
\end{array}$$

$$
\begin{array}{ll}
& f = \eta_A \quad \equiv \quad \varepsilon_B = g \\
\equiv & \quad \text{anticipating the next two steps, } \textbf{define } \eta_A = [\![A]\!]_{\mathcal{A}} \text{ and } \varepsilon_B = (\![B]\!)_{\mathcal{B}} \\
& f = [\![A]\!] \quad \equiv \quad (\![B]\!) = g \\
\equiv & \quad \text{dem-}\textsc{Charn} \text{ and med-}\textsc{Charn} \\
& f\colon\ A \to_{\mathcal{A}} 1 \quad \equiv \quad g\colon\ 0 \to_{\mathcal{B}} B \\
\equiv & \quad \text{typing of } f, g \text{, and definition of } F, G \\
& \mathit{true} \quad \equiv \quad \mathit{true}.
\end{array}
$$

Actually, we have shown both sides of the equivalence to be true, rather than to be the same truth value.
Exercise: show that $\mathcal{B}$ has an initial object if, and only if, there exists an adjunction between $\mathit{1}$ (the category with one object and one morphism) and $\mathcal{B}$.

69 Adjointness as a special initiality. Let $\mathcal{A}, \mathcal{B}$ be categories, and $G\colon \mathcal{B} \to \mathcal{A}$ be a functor. Then the statement that G has a left adjoint is equivalent to the statement that for each object A in $\mathcal{A}$ the category G/A (defined shortly) has an initial object. The proof outline is simple:

$$
\begin{array}{lll}
& & G \text{ has a left adjoint, say with unit } \eta \\
& \equiv & \quad \text{theorem 44, } \textsc{RightAdjoint} \text{ (see text 42)} \\
& & \forall A \quad \exists B_A \quad \exists \eta_A \quad \forall B \quad \forall \varphi \quad \exists! \psi :: \qquad \eta_A \,;\, G\psi = \varphi \\
(*) & \equiv & \quad \text{construction of precategory } G/'A \text{ below} \\
& & \forall A \quad \exists B_A \quad \exists \eta_A \quad \forall B \quad \forall \varphi \quad \exists! \psi :: \qquad \psi\colon\ \eta_A \to_{G/'A} \varphi \\
& \equiv & \quad \text{definition of initiality} \\
& & \forall A \quad \exists B_A \quad \exists \eta_A :: \qquad \eta_A \text{ is initial object in precategory } G/'A.
\end{array}
$$

(We leave it to the reader to add the typing; it is the same as in $\textsc{RightAdjoint}$.) Then, in a standard way (given below), one constructs a category G/A out of precategory $G/'A$; this construction preserves initiality of objects. So it remains to construct precategory $G/'A$ in such a way that step $(*)$ is valid, that is, the statement $\eta_A \,;\, G\psi = \varphi$ is equivalent to the statement $\psi\colon\ \eta_A \to_{G/'A} \varphi$. This requirement immediately suggests the definition of the objects and arrows of $G/'A$: the objects in $G/'A$ are $\mathcal{A}$-arrows φ with type $A \to_{\mathcal{A}} GB$ for some B; the arrows in $G/'A$, from φ to φ' say, are $\mathcal{B}$-arrows g satisfying $\varphi \,;\, Gg = \varphi'$; the identities and composition of $G/'A$ are inherited from $\mathcal{B}$. Thus defined, $G/'A$ satisfies all the axioms of a category, except unique-Type; hence it is a precategory and not necessarily a category. Now, by construction step $(*)$ is valid.

Here is the construction of category G/A out of precategory $G/'A$. The objects of G/A are the same as those of $G/'A$; the arrows in G/A from φ to φ' are triples (φ, g, φ') where $g\colon\ \varphi \to_{G/'A} \varphi'$ (this makes axiom unique-Type valid); composition is defined in the obvious way: $(\varphi, g, \varphi') \,;\, (\varphi', g', \varphi'') = (\varphi, (g \,;\, g'), \varphi'')$. It is straightforward to check that G/A is a category, and that φ is initial in $G/'A$ if, and only if, φ is initial in G/A.

70 Colimits. We recall here the definition of colimits. Throughout this text and the the following, A and f denote an arbitrary object and arrow in $\mathcal{A}$, and similarly for $B, g, \mathcal{B}$ and $C, h, \mathcal{C}$. Further, the constant functor mapping objects onto A, and arrows onto id_A, is denoted $\underline{A}$.

Let $H\colon \mathcal{C} \to \mathcal{A}$ be a functor (a 'diagram' in $\mathcal{A}$ of 'shape' $\mathcal{C}$). Then, a **cocone** in $\mathcal{A}$ for H is: a natural transformation $\alpha\colon H \dot{\to} \underline{A}$ for some object A in $\mathcal{A}$. Further, a cocone $\alpha\colon H \dot{\to} \underline{A}$ is called a **colimit** if: there exists a mapping $\alpha\backslash\_$ (from cocones to arrows) satisfying: for each cocone β for H and each $\mathcal{A}$-arrow f,

$$f = \alpha\backslash\beta \quad \equiv \quad \alpha \mathbin{;} f = \beta . \qquad\qquad \backslash\text{-}\textsc{Charn}$$

Here we define $(\alpha \mathbin{;} f)_C = \alpha_C \mathbin{;} f$, so that for $f\colon A \to_{\mathcal{A}} A'$ the expression $\alpha \mathbin{;} f$ denotes a cocone $\alpha \mathbin{;} f\colon H \dot{\to} \underline{A'}$. The arrow $\alpha\backslash\beta$ is called the mediating arrow. The notation $\_\backslash\_$ is motivated by the fact that its algebraic properties resemble those of division; see Fokkinga [3]. In the terminology of text 4, α is a colimit for H iff α is P_H-universal, where $P_H(\alpha, \beta, f) \equiv (\alpha \mathbin{;} f = \beta)$.

71 Left adjoints preserve colimits. Let $\mathcal{A}$ and $\mathcal{B}$ be arbitrary categories. Let $F\colon \mathcal{A} \to \mathcal{B}$ be left adjoint to $G\colon \mathcal{B} \to \mathcal{A}$, and let $H\colon \mathcal{C} \to \mathcal{A}$ be a functor (a diagram in $\mathcal{A}$ of shape $\mathcal{C}$). Suppose that α is a colimit in $\mathcal{A}$ for H. Then $F\alpha$ is a colimit in $\mathcal{B}$ for FH.

Proof. First observe that functors preserve cocones: if β is a cocone for $H\colon \mathcal{C} \to \mathcal{A}$, then $F\beta$ is a cocone for $FH\colon \mathcal{C} \to \mathcal{B}$. Now, we claim that $F\alpha$ is a colimit for FH. To prove this, we establish $\backslash$-Charn, constructing the unknown $F\alpha\backslash\_$ along the way. For arbitrary cocone β in $\mathcal{B}$ for FH, say $\beta\colon FH \dot{\to} \underline{B}$, and arbitrary $\mathcal{B}$-arrow g:

$$
\begin{array}{lll}
 & & F\alpha \mathbin{;} g = \beta \\
 & \equiv & \quad \text{extensionality, composition of cocone with an arrow} \\
 & & F\alpha_C \mathbin{;} g = \beta_C \qquad \text{for each } C \text{ in } \mathcal{C} \\
 & \equiv & \quad \textsc{Inverse}\text{, noting that both sides above have type } FHC \to_{\mathcal{B}} B \\
 & & \llfloor F\alpha_C \mathbin{;} g \rrfloor = \llfloor \beta_C \rrfloor \qquad \text{for each } C \text{ in } \mathcal{C} \\
 & \equiv & \quad \text{lad-}\textsc{Fusion} \\
 & & \alpha_C \mathbin{;} \llfloor g \rrfloor = \llfloor \beta_C \rrfloor \qquad \text{for each } C \text{ in } \mathcal{C} \\
(*) & \equiv & \quad \text{for } \Rightarrow\text{: } \textbf{define } \beta' \text{ by } \beta'_C = \llfloor \beta_C \rrfloor \text{ for each } C \text{ in } \mathcal{C} \\
 & & \quad \text{for } \Leftarrow\text{: note that by } (\star) \text{ we have } \beta'_C = \llfloor \beta_C \rrfloor \text{ for each } C \text{ in } \mathcal{C} \\
 & & \alpha \mathbin{;} \llfloor g \rrfloor = \beta' \\
 & \equiv & \quad \alpha \text{ is colimit for } H,\ \backslash\text{-}\textsc{Charn}[f, \beta := \llfloor g \rrfloor, \beta'] \\
 & & \llfloor g \rrfloor = \alpha\backslash\beta' \\
 & \equiv & \quad \textsc{Inverse} \\
 & & g = \llceil \alpha\backslash\beta' \rrceil \\
(\star) & \equiv & \quad \textbf{define } F\alpha\backslash\beta = \llceil \alpha\backslash\beta' \rrceil \text{ where } \beta'_C = \llfloor \beta_C \rrfloor\text{; observation below}
\end{array}
$$

$g = F\alpha \backslash \beta$.

The definition of $F\alpha\backslash\_$ in step $(\star)$ requires some care. First, even though in general α is not recoverable from $F\alpha$, here α is known from the data of the theorem. Second, the notation $...\alpha\backslash\beta'...$ requires that β' is a cocone in $\mathcal{A}$ for H, that is, β': $H \dot{\to} \underline{A'}$ for some object A' in $\mathcal{A}$. It is almost trivial that β' is a transformation from H to some $\underline{A'}$; indeed, for arbitrary C in $\mathcal{C}$:

$$
\begin{array}{ll}
 & \beta'_C \colon\ HC \to_{\mathcal{A}} A' \\
\Leftarrow & \qquad \text{definition of } \beta'_C = [\![\beta_C]\!] \text{, typing of } [\![\ ]\!] \\
 & \beta_C \colon\ FHC \to_{\mathcal{B}} B \ \text{ and } \ A' = GB \\
\Leftarrow & \qquad \text{assumption } \beta \colon\ FH \dot{\to} \underline{B} \text{, } \textbf{define } A' = GB \\
 & \mathit{true}.
\end{array}
$$

The verification of the naturality of β': $H \dot{\to} \underline{GB}$ is almost as simple. For arbitrary h: $C \to_{\mathcal{C}} C'$:

$$
\begin{array}{ll}
 & Hh \,;\, \beta'_{C'} = \beta'_C \,;\, id_{GB} \\
\equiv & \qquad \text{definition of } \beta' \text{, functor} \\
 & Hh \,;\, [\![\beta_{C'}]\!] = [\![\beta_C]\!] \,;\, Gid_B \\
\equiv & \qquad \text{lad-}\textsc{Fusion} \\
 & [\![FHh \,;\, \beta_{C'}]\!] = [\![\beta_C \,;\, id_B]\!] \\
\Leftarrow & \qquad \text{Leibniz} \\
 & FHh \,;\, \beta_{C'} = \beta_C \,;\, id_B \\
\equiv & \qquad \text{assumption } \beta \colon\ FH \dot{\to} \underline{B} \\
 & \mathit{true}.
\end{array}
$$

5 Examples

In this section we'll discuss some "standard" examples of adjunctions. We assume familiarity with the notion of categorical sum, but for completeness we'll recall the definition here.

Let $\mathcal{A}$ be a category, and A, B be objects in $\mathcal{A}$. Let $\mathit{inl}, \mathit{inr}$ be a pair of $\mathcal{A}$-arrows, typed:

$$
\begin{array}{lcl}
\mathit{inl} & : & A \to_{\mathcal{A}} S \\
\mathit{inr} & : & B \to_{\mathcal{A}} S \,,
\end{array}
$$

for some object S. Then $(\mathit{inl}, \mathit{inr})$ is called a **sum** of A and B if: there exists a mapping $\_ \triangledown \_$ (from pairs of arrows to arrows) satisfying:

$$
f = g \triangledown h \quad \equiv \quad \mathit{inl} \,;\, f = g \ \text{ and } \ \mathit{inr} \,;\, f = h \qquad \triangledown\text{-}\textsc{Charn}
$$

In order that this makes sense, f, g, h are typed thus:

$$f \quad : \quad S \to_{\mathcal{A}} C$$

and

$$\begin{array}{lll} g & : & A \to_{\mathcal{A}} C \\ h & : & B \to_{\mathcal{A}} C\,, \end{array}$$

for some object C. If (inl, inr) is a sum of A and B, then the arrows inl, inr are called the injections, the common target S of the injections is written $A+B$, and arrow $g \triangledown h$ is a 'case distinction', so that operator $\triangledown$ may be called 'dis'. Usually $g \triangledown h$ is written $[g, h]$. In the terminology of text 4, the pair (inl, inr) is $P_{A,B}$-universal, where

$$P_{A,B}((inl, inr), (g, h), f) \quad \equiv \quad inl \,;\, f = g \;\text{ and }\; inr \,;\, f = h\,.$$

When a sum exists for each pair of objects, we can define a bifunctor $\_+\_$:

$$\begin{array}{lll} A + B & = & \text{the common target of the sum } (inl, inr) \text{ for } A, B \\ f + g & = & (f \,;\, inl) \triangledown (g \,;\, inr)\,. \end{array}$$

One can prove that the following laws are valid:

$$\begin{array}{lll} inl \triangledown inr & = & id \qquad \triangledown\text{-ID} \\ f \triangledown g \,;\, h & = & (f \,;\, h) \triangledown (g \,;\, h) \qquad \triangledown\text{-FUSION} \\ f + g \,;\, h \triangledown j & = & (f \,;\, h) \triangledown (g \,;\, j) \qquad +\text{-}\triangledown\text{-DISTR} \end{array}$$

We also assume familiarity with the product category $\mathcal{A} \times \mathcal{B}$ of two categories $\mathcal{A}$ and $\mathcal{B}$: its objects are pairs (A, B) of objects, its arrows are pairs (f, g) of arrows, and the composition, identities, and typing is defined in the obvious way, componentwise. The doubling functor Δ: $\mathcal{A} \to \mathcal{A} \times \mathcal{A}$ is defined by $\Delta(x) = (x, x)$ for object and arrow x.

72 Adjoints of doubling. Suppose that category $\mathcal{A}$ has a sum for each pair of objects. We claim that the sum functor is left adjoint to the doubling functor. To show this, we need bijections $\llbracket\;\rrbracket$ and $\lceil\!\lceil\;\rceil\!\rceil$ between the arrows

$$\begin{array}{lll} \varphi & : & (A, B) \to_{\mathcal{A} \times \mathcal{A}} \Delta(C) \\ \lceil\!\lceil\;\rceil\!\rceil\!\downarrow\;\uparrow\!\llbracket\;\rrbracket & & \\ \psi & : & +(A, B) \to_{\mathcal{A}} \quad C, \end{array}$$

or, expanding the functors and using that φ is a pair of arrows from $\mathcal{A}$, between

$$\begin{array}{lll} (f, g) & : & (A, B) \to_{\mathcal{A} \times \mathcal{A}} (C, C) \\ \lceil\!\lceil\;\rceil\!\rceil\!\downarrow\;\uparrow\!\llbracket\;\rrbracket & & \\ \psi & : & A + B \to_{\mathcal{A}} \quad C. \end{array}$$

A construction downwards is given by $\triangledown$, since $f \triangledown g \colon A + B \to_{\mathcal{A}} C$ for f and g typed as above. In the other direction we can retrieve the components from ψ by using the injection functions to create $((\mathit{inl} \mathbin{;} \psi),\ (\mathit{inr} \mathbin{;} \psi)) \colon (A, B) \to_{\mathcal{A}\times\mathcal{A}} (C, C)$. Thus we define:

$$
\begin{array}{lcl}
\lceil\!\lceil (f, g) \rceil\!\rceil & = & f \triangledown g \\
\lfloor\!\lfloor \psi \rfloor\!\rfloor & = & ((\mathit{inl} \mathbin{;} \psi),\ (\mathit{inr} \mathbin{;} \psi)) .
\end{array}
$$

We have not *proved* though, yet, that we have an adjunction here. For this we will show that lad-Def, rad-Def, and Inverse are satisfied. Let us first determine the unit and co-unit. Assuming that we end up with an adjunction, we have

$$
\begin{array}{ll}
 & \eta \\
= & \qquad \text{unit-Def (formula 33)} \\
 & \lfloor\!\lfloor \mathit{id} \rfloor\!\rfloor \\
= & \qquad \text{definition of } \lfloor\!\lfloor\ \rfloor\!\rfloor \text{ above} \\
 & ((\mathit{inl} \mathbin{;} \mathit{id}),\ (\mathit{inr} \mathbin{;} \mathit{id})) \\
= & \qquad \text{identity} \\
 & (\mathit{inl}, \mathit{inr}).
\end{array}
$$

Further,

$$
\begin{array}{ll}
 & \varepsilon \\
= & \qquad \text{co-unit-Def (formula 37)} \\
 & \lceil\!\lceil \mathit{id} \rceil\!\rceil \\
= & \qquad \text{product category: } \mathit{id}_{\mathcal{A}\times\mathcal{A}} = (\mathit{id}_{\mathcal{A}}, \mathit{id}_{\mathcal{A}}) \\
 & \lceil\!\lceil (\mathit{id}, \mathit{id}) \rceil\!\rceil \\
= & \qquad \text{definition of } \lceil\!\lceil\ \rceil\!\rceil \text{ above} \\
 & \mathit{id} \triangledown \mathit{id}.
\end{array}
$$

So we define

$$
\begin{array}{lcl}
\eta & = & (\mathit{inl}, \mathit{inr}) \\
\varepsilon & = & \mathit{id} \triangledown \mathit{id} ,
\end{array}
$$

and it is now immediate that lad-Def is satisfied, and with $+$-$\triangledown$-Distr it easily follows that rad-Def is satisfied:

$$
\begin{array}{lcl}
\lfloor\!\lfloor \psi \rfloor\!\rfloor & = & \eta \mathbin{;} \Delta(\psi) \\
\lceil\!\lceil (f, g) \rceil\!\rceil & = & f + g \mathbin{;} \varepsilon .
\end{array}
$$

So, it only remains to give a proof of Inverse. Here it is.

$$
\begin{array}{ll}
 & (f, g) = \lfloor\!\lfloor \psi \rfloor\!\rfloor \\
\equiv & \qquad \text{definition of } \lfloor\!\lfloor\ \rfloor\!\rfloor \text{ above}
\end{array}
$$

$$
\begin{array}{lll}
 & & (f, g) = ((\mathit{inl} \mathbin{;} \psi),\ (\mathit{inr} \mathbin{;} \psi)) \\
(*) & \equiv & \qquad \triangledown\text{-}\textsc{Charn} \\
 & & f \triangledown g = \psi \\
 & \equiv & \qquad \text{definition of } \llbracket\ \rrbracket \text{ above} \\
 & & \llbracket (f, g) \rrbracket = \psi.
\end{array}
$$

So $+$ is left adjoint to Δ.

Dually, for a category with products, $\times$ is right adjoint to Δ.

Discussion. What we have shown here is that *if* a category has sums, *then* the sum functor is a left adjoint of doubling. In fact, these two statements are equivalent: if doubling has a left adjoint, then the category has sums and the left adjoint is a sum functor. On the one hand this follows from text 59, on the other hand we can prove it explicitly as follows. Suppose that the doubling functor has a left adjoint. Then we can find a definition for $\triangledown$ and $\mathit{inl}, \mathit{inr}$ satisfying $\triangledown$-Charn as follows. Cut the last calculation open at step $(*)$ with justification '$\triangledown$-Charn', paste the bottom line to the top line with justification 'Inverse' (step $(\star)$ below), and replace the definitions of $\llbracket\ \rrbracket$ and $\llbracket\ \rrbracket$ in terms of $\triangledown$ and $\mathit{inl}, \mathit{inr}$ by definitions of the latter in terms of the former:

$$
\begin{array}{lll}
 & & f \triangledown g = \psi \\
 & \equiv & \qquad \textbf{define } f \triangledown g = \llbracket (f, g) \rrbracket \\
 & & \llbracket (f, g) \rrbracket = \psi \\
(\star) & \equiv & \qquad \textsc{Inverse} \\
 & & (f, g) = \llbracket \psi \rrbracket \\
 & \equiv & \qquad \text{lad-}\textsc{Def} \\
 & & (f, g) = \eta \mathbin{;} \Delta(\psi) \\
 & \equiv & \qquad \textbf{define } (\mathit{inl}, \mathit{inr}) = \eta \\
 & & \qquad \text{product category} \\
 & & (f, g) = (\mathit{inl} \mathbin{;} \psi),\ (\mathit{inr} \mathbin{;} \psi)).
\end{array}
$$

Further, the left adjoint $+$ satisfies the definition of the categorical sum functor:

$$
\begin{array}{ll}
 & +(f, g) \\
= & \qquad \text{leftadjoint-}\textsc{Def} \\
 & \llbracket (f, g) \mathbin{;} \eta \rrbracket \\
= & \qquad \text{definition of } \mathit{inl} \text{ and } \mathit{inr} \text{ just made} \\
 & \llbracket (f, g) \mathbin{;} (\mathit{inl}, \mathit{inr}) \rrbracket \\
= & \qquad \text{product category, definition of } \triangledown \text{ just made} \\
 & (f \mathbin{;} \mathit{inl}) \triangledown (g \mathbin{;} \mathit{inr}).
\end{array}
$$

So, 'left adjointness to the doubling functor' gives a very snappy way of defining the categorical notion of 'sum'. It is also a somewhat cryptic way, but in a sense this is what category theory is about. While we —or at least most of us— are used to think in terms of specific constructions, implementations so to say, for something like a disjoint union (for example with tags), the categorical approach is to reason only from the *characterising properties*. One role of adjunctions is that they capture in a rather general way the process of defining concepts through characterisation.

Explicit constructions may still be needed to show that something *exists*, like the product of two categories, but the concrete definition of $\mathcal{A} \times \mathcal{B}$ suggested above is just one of several equally valid implementations.

73 Total and partial functions. Let $\mathcal{T}ot$ denote the category of typed *total* functions, and $\mathcal{P}ar$ that of typed *partial* functions. So $\mathcal{T}ot$ is the same as $\mathcal{S}et$. These two categories have the same objects (namely some collection of sets), but $\mathcal{P}ar$ has more arrows than $\mathcal{T}ot$. In fact, $\mathcal{T}ot$ is a subcategory of $\mathcal{P}ar$, since a total function is (or may be considered as) a partial function that happens to be defined everywhere. This gives us an embedding functor

$$E \quad : \quad \mathcal{T}ot \to \mathcal{P}ar\,,$$

which semantically is a null action both on objects and on arrows. Can we go the other way? In this text we shall *construct* a functor X from $\mathcal{P}ar$ to $\mathcal{T}ot$. Throughout the sequel variable f denotes a $\mathcal{T}ot$-arrow, g denotes a $\mathcal{P}ar$-arrow, and $\omega_{A,B}\colon A \to_{\mathcal{P}ar} B$ denotes the everywhere undefined function.

There is a standard construction for turning a partial function $g\colon A \to_{\mathcal{P}ar} B$ into a total function $\overline{g}$. Namely, extend the target type with some new element $\perp$, and define $\overline{g}$ like g on those arguments for which g is defined, but have it yield $\perp$ wherever g is not defined. The requirement that the element with which B is extended be 'new' amounts to taking a disjoint union of B with $\{\perp\}$, which is categorically best modelled by taking $B+1$ as the target type of $\overline{g}$. So we have

$$\overline{g}\colon A \to_{\mathcal{T}ot} B+1 \qquad\qquad \text{whenever } g\colon A \to_{\mathcal{P}ar} B\,.$$

Instead of formally defining this 'totalisator' operation, we list a few crucial properties, for which we give only informal justifications.

The first one is that the totalisator is injective:

$$\overline{g_0} = \overline{g_1} \quad \equiv \quad g_0 = g_1\,.$$

Next, when a totalised function $\overline{g}$ is pre-composed with a total function f, then this function can be absorbed into the totalisation. Formally expressed, this amounts to the following law:

$$f \,;\, \overline{g} \quad = \quad \overline{Ef \,;\, g}\,.$$

which we will refer to as 'totalisator fusion'.

Further, there are two ‘extreme cases’ in which it is possible to eliminate the totalisator operation. One extreme is if the argument of the totalisator happens to be an already total function. The effect of totalising a total function is simply that the results get injected into the left component of the sum $B+1$. This property can be expressed as:

$$\overline{Ef} \quad = \quad f \,;\, \mathit{inl}\,.$$

The other extreme is for the everywhere undefined function ω. In that case everything “ends up” in the 1 component, which is expressed by:

$$\overline{\omega} \quad = \quad ! \,;\, \mathit{inr}\,,$$

where $!$ denotes the unique arrow (family) $!_A$: $A \rightarrow_{\mathcal{T}ot} 1$.

Finally, when g has the form $g_0 \triangledown g_1$, we can distribute the totalisator over the case distinction:

$$\overline{g_0 \triangledown g_1} \quad = \quad \overline{g_0} \triangledown \overline{g_1}\,.$$

So much for some properties of totalisation.

What we have in our hands now already looks much like an adjunction. To make this even more explicit we write the extension $B+1$ as XB:

$$XB \quad = \quad B+1\,.$$

Our wish is to extend this object mapping to a full-fledged functor

$$X \quad : \quad \mathcal{P}ar \rightarrow \mathcal{T}ot\,.$$

For the moment we can only guess what the action of X on functions is, but if we have indeed an adjunction, with X as the right adjoint, then we have the tools to find this out, namely law rightadjoint-Def, formula 18. (Note that $Xg = g + \mathit{id}$ will not do, since the result is not an arrow in $\mathcal{T}ot$.) A display of the present ingredients is given by

$$\begin{array}{lll} \text{in category } \mathcal{T}ot\text{:} & \varphi & : A \rightarrow_{\mathcal{T}ot} XB \\ & \Big\uparrow\overline{} & \\ \text{in category } \mathcal{P}ar\text{:} & \psi & : EA \rightarrow_{\mathcal{P}ar} B. \end{array}$$

What is missing to complete the picture into an adjunction is a way to go from a φ-arrow to a ψ-arrow. From the dual statement of text 44, we know that all that is needed is a family of arrows

$$\varepsilon \quad : \quad EXB \rightarrow_{\mathcal{P}ar} B$$

satisfying the characterisation lad-Charn′:

$$\varphi = \overline{\varphi} \quad \equiv \quad E\varphi \,;\, \varepsilon = \psi$$

for all appropriately typed φ and ψ. Let us not worry for a moment about the characterisation, but try to find some arrow family of the right type. Expanding the functors, we have that $\varepsilon_B\colon B+1 \to_{\mathcal{P}ar} B$, so we can express ε_B as a case distinction: $\varepsilon_B = \varepsilon'_B \triangledown \varepsilon''_B$ in which the two components are typed thus:

$$\begin{array}{lcl} \varepsilon'_B & : & B \to_{\mathcal{P}ar} B \\ \varepsilon''_B & : & 1 \to_{\mathcal{P}ar} B\,. \end{array}$$

To find a candidate for ε' is easy enough: just take id. For ε'' we really need that we are in the category $\mathcal{P}ar$ here, and we take the everywhere undefined function ω:

$$\varepsilon \quad = \quad \mathit{id} \triangledown \omega\,.$$

We come now to the crucial question: is the characterisation lad-CHARN$'$ satisfied? Let us calculate. We start with the most complicated side.

$$\begin{array}{llll}
 & & E\varphi \,;\, \varepsilon = \psi & \\
 & \equiv & \qquad \text{definition of } \varepsilon & \\
 & & E\varphi \,;\, \mathit{id} \triangledown \omega = \psi & \\
 & \equiv & \qquad \text{totalisator is injective} & \\
 & & \overline{E\varphi \,;\, \mathit{id} \triangledown \omega} = \overline{\psi} & \\
 & \equiv & \qquad \text{totalisator fusion} & \\
 & & \varphi \,;\, \overline{\mathit{id} \triangledown \omega} = \overline{\psi} & \\
 & \equiv & \qquad \text{totalisator distributes over } \triangledown & \\
 & & \varphi \,;\, \overline{\mathit{id}} \triangledown \overline{\omega} = \overline{\psi} & \\
(*) & \equiv & \qquad \text{extreme cases} & \\
 & & \varphi \,;\, \mathit{inl} \triangledown (!_1 \,;\, \mathit{inr}) = \overline{\psi} & \\
 & \equiv & \qquad \text{by uniqueness, } !_1 = \mathit{id}_1 & \\
 & & \varphi \,;\, \mathit{inl} \triangledown \mathit{inr} = \overline{\psi} & \\
 & \equiv & \qquad \triangledown\text{-ID: } \mathit{inl} \triangledown \mathit{inr} = \mathit{id}\text{, identity} & \\
 & & \varphi = \overline{\psi}. &
\end{array}$$

This completes the proof that the object mapping X is right adjoint to E.

We can now complete the definition of the functor X with its action on arrows by using rightadjoint-DEF, formula 18, and obtain

$$Xg \quad = \quad \overline{\varepsilon \,;\, g}\,.$$

The right-hand side simplifies as follows:

$$\begin{array}{ll}
 & \overline{\varepsilon \,;\, g} \\
= & \qquad \text{definition of } \varepsilon
\end{array}$$

$$\overline{id \triangledown \omega \,;\, g}$$

$=$ $\triangledown$-FUSION

$$\overline{(id \,;\, g) \triangledown (\omega \,;\, g)}$$

$=$ id is identity and ω is zero of composition

$$\overline{g \triangledown \omega}$$

$=$ totalisator distributes over $\triangledown$

$$\overline{g} \triangledown \overline{\omega}$$

$=$ as shown above at the steps from $(*)$ onwards, $\overline{\omega} = inr$

$$\overline{g} \triangledown inr.$$

This completes the extension of object mapping X to a functor X: $\mathcal{P}ar \to \mathcal{T}ot$, using the conjecture (and subsequent proof) that the object mapping X is right adjoint to the embedding functor E: $\mathcal{T}ot \to \mathcal{P}ar$.

References

[1] R.C. Backhouse et al. A relational theory of datatypes. Presented at the STOP 1992 Summerschool on Constructive Algorithmics, Ameland, Netherlands, 1992., 1992.

[2] M. Barr and C. Wells. *Category Theory for Computing Science*. Prentice Hall, 1990.

[3] M.M. Fokkinga. Calculate categorically! *Formal Aspects of Computing*, 1993. To appear. An abbreviated version has appeared in *Computing Science in The Netherlands*, published by Stichting Mathematisch Centrum, Amsterdam, pages 211–230, 1991.

[4] S. Mac Lane. *Categories for the Working Mathematician*, volume 5 of *Graduate Texts in Mathematics*. Springer Verlag, 1971.