Beginning Category Theory

A gentle, still unfinished, introduction

Peter Smith

LOGIC MATTERS

This is very much work in progress, but I hope you find it helpful and interesting. Two friendly requests:

1. Please do send any comments, suggestions and corrections – however small! – to this address. They are always welcome. It can help a lot if you give the date of the pdf you are commenting on: this version is dated April 27, 2022.
2. Please do spread the word about this gentle intro. But please don't directly share this pdf or place it on a website as it could be superseded in a few days! Instead, share the url where the latest version can always be found, namely logicmatters.net/categories.

Contents

Preface

The project In around 2015, I put together a lengthy set of notes, *Category Theory: A Gentle Introduction.* I didn't really intend to write that near-book. But background notes for a rather less elementary project grew and grew, and they started taking on a life of their own as I tried to organize them more logically. The emerging result was an elementary introduction to some entry-level category theory, a beginner's guide of the kind that I myself would have rather welcomed when starting out in this area, and which I hoped that others might find both helpful and intriguing. In a rough and ready way, I tried to cover most of the really basic notions of category theory – explaining the very idea of a category, then treating e.g. limits, functors, natural transformations, representables, and adjunctions, with some pointers forward to further ideas. The aim was to get far enough to give a reader a reasonable grounding from which to tackle other texts of various kinds with some confidence.

Those notes were long given their limited coverage, because – as advertised – they went at a pretty gentle pace. I don't apologize at all for this: there are plenty of fast-track alternative introductions available. But experience strongly suggests that getting a secure understanding of categorial[1] ways of thinking by initially taking things slowly does make later adventures exploring beyond the basics *very* much more manageable.

A corrected and slightly revised version of the notes as published online in 2018. That was still very much a work in progress; so there were chapters at different levels of development and with different degrees of integration with what's around them (and no doubt different levels of reliability). Despite their half-baked character, these notes have been downloaded a lot, often over a thousand times a month in a month. Which is pleasing but also embarrassing, as I know how flawed they are.

I needed to set the *Gentle Introduction* aside for a few years, longer than I planned, to get on with some other projects. But those are at last finished, or at least can be set aside for a while in their turn: so I now have time to return to thinking a bit about categories, and can start working on a new and (let's hope!) improved version of the notes. To avoid confusion, I have given this a slightly different main title, hence *Beginning Category Theory.*

[1]Logicians already have a quite different use for 'categorical'. So when talking about categories, I much prefer the adjectival form 'categorial', even though it is the minority usage.

Here then are some initial revised/expanded/improved chapters (currently fourteen of them). Note: these are *not* final versions. Revised chapters get added when I think that they are definitely better than what they replace, not when I think they are as good as they could be! *So all comments and corrections will be very gratefully received.*

Then – to save juggling between two documents – I have appended the remaining old chapters to make a single PDF. There's a headline warning on old pages saying that they are indeed unrevised! (No doubt some cross-linkages between the various archeological layers have been broken.)

Who are these notes for? What do you need to bring to the party? I imagine one reader to be a mathematics student who wants a first introduction to some categorial ideas, perhaps as a preliminary warm-up before taking on an industrial-strength graduate-level course. Another reader might be a philosopher interested in the foundations of mathematics who wants a relatively accessible introduction to give them an initial sense of what the categorial fuss is about, so that they can tell if they want to find out more.

Now, you obviously can't be well placed to appreciate how category theory gives us a story about the ways in which different parts of modern abstract mathematics hang together if you really know *nothing* beforehand about modern mathematics! But I try to presuppose relatively little. Suppose you know a few basic facts about groups (there's some revision in Chapter 2!), know a little about different kinds of orderings, are acquainted with some elementary topological ideas, and know a few more bits and pieces; then you should in fact be able to cope fairly easily. And if some later illustrative examples pass you by, don't panic. I usually try to give multiple illustrations of important concepts and constructs; so feel free simply to skip those examples that happen not to work so well for you.

Theorems as exercises Almost all the proofs of the theorems you meet as you begin category theory are *very* straightforward. Surprisingly often, you just have to 'do the obvious thing': there's little ingenious trickery needed at the outset. So you can usually think of the statement of a theorem as in fact presenting you with an exercise which you could, and even should, attempt to work through for yourself in order to fix ideas. The ensuing proof which I spell out is then the answer (or at least, *an* answer) to the exercise. For a few tougher theorems, I give preliminary hints about how the argument ought to go.

I haven't yet added further batches of exercises.

Notation and terminology I try to keep to settled notation and terminology, and where there are standard alternatives often mention them too: what I say here should therefore be easy to relate to other discussions of the same material.

'Iff', as usual, abbreviates 'if and only if'. In addition to using the familiar '$\square$' as an end-of-proof marker (or to conclude the statement of a theorem that needs no proof), I also use '$\triangle$' as an end-of-definition marker.

And from now on, I mostly follow the usual mathematicians' practice of omitting quotation marks when mentioning symbolic expressions, if no confusion is likely to result. Logicians can get irritatingly pernickety about this sort of thing, and I try to avoid that.

Thanks! Andrew Bacon, Malcolm F. Lowe and Mariusz Stopa very kindly sent long lists of corrections to an early ancestor of these notes. A lot of the mistakes were obvious typos, but there were also enough mislabelled arrows or fumbling of notation mid-proof and the like that I should certainly apologize to readers who found themselves scratching their heads in puzzlement. I had then further corrections from Malcolm F. Lowe, David Ozonoff, Jan Thiemann, Zoltán Tóth, and Adrian Yee. And I haven't yet dealt with very substantial further comments about the unrevised chapters from Matthias Falk.

And, as I start work on revising all this, I've now had more corrections and suggestions on early chapters, in particular from Sam Butchart and Rowsety Moid. Very warm thanks to everyone!

1 Introduction

1.1 The categorial imperative

Modern pure mathematics explores abstract structures. And these mathematical structures cluster in families.

Take a family of structures together with a good helping of the structure-preserving maps between them. Then we can think of this family as forming a further structure – a structure-of-structures, if you like – something else to explore mathematically.

(1) Here's a basic example. A particular *group* is a structure which comprises some objects equipped with a binary operation defined on them, where the operation obeys the well-known axioms. But we can also think of a whole family of groups, together with appropriate maps between them – i.e. homomorphisms which preserve group structure – as forming a further structure-of-structures.

(2) Another example: any particular *topological space* is a structure, this time comprising some objects, 'points', which are equipped with a topology. But again, a whole family of these spaces, together with appropriate maps between them – this time, the continuous functions which preserve topological structure – forms another structure-of-structures.

(3) And so it goes. Perhaps what interests you are some *well-ordered objects*: these constitute another mathematical structure. In fact, there is a whole family of such well-ordered structures together with order-preserving maps between them. We are interested in the structure of this family (perhaps in the guise of the theory of ordinals, the theory of order-types of well-orderings). We want to know too about other kinds of families of ordered objects and the relations between them.

In each of these various cases, then, we not only investigate *individual* structures (the particular groups, particular topological spaces, particular collections of ordered objects), but we can also explore *families* of such structures (families of groups, families of topological spaces, families of ordered objects), with the family itself structured by the maps or morphisms between its members.

As a further step, we can next go on to consider the interrelations between these structures-of-structures. This will involve looking at an additional level of structure-preserving maps, the so-called *functors*, this time linking structures-of-structures (as when we map a family of topological spaces with base points to their corresponding fundamental groups). And even this is not the end of it. Going up yet another level of abstraction, we will find ourselves wanting to consider operations which map one functor to another while preserving their functorial character (in ways we will later explain).

So here is *one* central mathematical imperative: to explore these upper levels of increasing abstract structure.

Let's agree straight away that this project certainly doesn't appeal to all – or even most – mathematicians. A vast amount of pure mathematics is of course carried on at much less exalted levels. Still, the hyper-abstracting project can resonate with a certain systematizing cast of mind. And evidently, if we *are* going to set out on such an enquiry, we will want a framework for dealing with these upper layers of abstraction in a disciplined and illuminating way.

This is where category theory comes into play for us: it provides exactly what we need, at least as we first set out to explore the territory, because suitably structured families of structures are prime examples of categories. Category theory's basic ideas and constructions will provide a general toolkit for systematically probing structures-of-structures and even structures-of-structures-of-structures. And it is the theory in *this* role that will be our main concern in this beginners' guide.

1.2 From a bird's eye view

But what do we really gain by ascending through those levels of abstraction and by developing tools for imposing some order on what we find?

For a start, we should get a richer conceptual understanding of how various parts of mathematics relate to each other. And we might reasonably say that, in *one* sense of that contested label, this will be a 'philosophical' gain. After all, many philosophers, pressed for a crisp characterization of their discipline, like to quote a famous remark by Wilfrid Sellars,

> The aim of philosophy, abstractly formulated, is to understand how things in the broadest possible sense of the term hang together in the broadest possible sense of the term. (Sellars 1963, p. 1)

Category theory indeed provides us with a suitable unifying framework for exploring in depth some of the ways in which a lot of mathematics hangs together. That's why it should be of considerable interest to philosophers of mathematics as well as to mathematicians interested in the conceptual shape of their own discipline.

But note, category theory does much more than give us a helpful way of relating aspects of structures that we already know about. As Tom Leinster so very nicely puts it, the theory

> ... takes a bird's eye view of mathematics. From high in the sky, details become invisible, but we can spot patterns that were impossible to detect from ground level. (Leinster 2014, p. 1)

From its highly abstract vantage point, category theory crucially reveals *new* connections we hadn't made before. What are called 'adjunctions' are a prime example, as we will eventually see.

Seeing recurrent patterns in different families of structures and making new connections between them in turn enables new mathematical discoveries. And it was because of the depth and richness of the resulting discoveries in e.g. algebraic topology that category theory first came to prominence. But it would be distracting to investigate those roots in this book. I will stick to very much more elementary concerns, with an emphasis on unification and conceptual clarification. This will still give us more than enough to explore. And this way, I hope to keep everything relatively accessible.

1.3 A slow ascent

The gadgets of basic category theory do fit together rather beautifully in multiple ways. These intricate interconnections mean that there certainly isn't a single best route into the theory. Different treatments can take topics in significantly divergent orders, all illuminating in their various ways.

I will follow the simplest plan, however, and make a slow ascent to the categorial heights. We begin then at that first new level of abstraction, one step up from talking about particular structures. In other words, we start by talking about *categories*. For, as we said, many paradigm cases of categories are in fact structured-families-of-structures. And we go on to develop ways of describing what happens inside a category. In this new setting, we revisit many familiar ideas about maps between structures, and about ways of forming new structures by e.g. taking products or taking quotients. Which gives us our topics for what can be thought of as Part I of this book, i.e. Chapters 4 to 19.

Only after extended exploration of categories taken singly do we move up another level to consider *functors*, maps between categories (typically, maps between families of structures). And only after we have spent a number of chapters thinking about how particular functors work (and how they interact with products, quotients and the like) do we next move up a further level to define operations sending one functor to another – these are the so-called *natural transformations* and *natural isomorphisms*. We then explore these notions, and the related idea of one functor being a *representation* of another, at some length before we at last start exploring the key notion of *adjunctions*. All this will be covered in Part II of the book, i.e. Chapters 20 to 35.

Finally, having climbed to these heights, there are some pointers forward to other topics that now come into view, including topics to be taken up in a planned sequel on logic and sets treated categorially.

In summary, my chosen route here into the basics of category theory steadily ascends through the increasing levels of abstraction in a particularly natural way (which has some logical appeal). True, this does mean that it takes us quite a while to reach some of the *really* novel and exciting categorial ideas. However, this disadvantage is (I hope) considerably outweighed by the gain in secure understanding which comes from taking our gently sloping path. I will just have to do my best to make sure some of the views we glimpse along the way still seem interesting enough.

2 One structured family of structures

Category theory gives us a framework in which we can think systematically about structured families of mathematical structures: or at least, it is this aspect of the theory which is going to be our focus.

I said that one paradigm case of such a structured family comprises some groups organized by homomorphisms between them. And by the end of this chapter, I'll have officially defined the idea of a category of groups. I'll begin, however, by reviewing some *very* elementary facts about groups. These facts will quite likely be entirely familiar, so I hope we can take things quite briskly without needing to give very much by way of motivation and examples. Still, it will prove useful to highlight some themes which are already there in pre-categorial mathematics, themes which we will soon encounter again in a categorial guise.

You'll spot straight away that, in one respect, the definitions I give in this chapter are not quite the usual ones. But I'll explain the reason for their (only mildly) deviant character in Chapter 3, so do indulge me!

2.1 Groups revisited

(a) Here then is my preferred way of characterizing groups:

Definition 1. The objects G (including the distinguished object e), equipped with a binary operation $*$ (where for any x, y among G, $x * y$ is also among G), form a *group* iff

(i) $*$ is associative, i.e. for any x, y, z among G, $(x * y) * z = x * (y * z)$;
(ii) e acts as a group identity, i.e. for any x among G, $x * e = x = e * x$;
(iii) every object has a group inverse, i.e. for any x among G, there is at least one object y also among G such that $x * y = e = y * x$. △

Don't read too much into 'equipped'. It's a standard turn of phrase here; but it means no more than that we are dealing with some objects G *and* an operation defined over them.

If e and e' are both identities for the group of objects G equipped with $*$, then $e = e * e' = e'$; so group identities are unique. Likewise, inverses are unique.

(b) Note the variety of objects and operations that can form a group. In fact, *any* object e, whatever you like, equipped with the only possible binary operation $*$ such that $e * e = e$, forms a trivial one-object group. Similarly, any two objects

e, j, whatever you like, form a group when equipped with the binary operation $*$ for which e is the identity and $j * j = e$.

Less trivially, there are additive groups of numbers (e.g. the integers equipped with addition, or with addition mod n, with 0 as the identity), and there are multiplicative groups of numbers (e.g. non-zero rationals equipped with multiplication, with 1 as the identity). These examples are *abelian*, i.e. the binary operation is commutative.

Likewise, there are groups of functions. For a simple case, take the group of permutations of the first n naturals, with functional composition as the group operation and the do-nothing permutation as the group identity. If $n > 2$, then this permutation group is non-abelian. Or consider groups of geometrical transformations – for instance the non-abelian group of symmetries of a regular polygon (i.e. the rotation and reflection operations which map the polygon to itself).

Then there are various groups of real invertible matrices, groups of closed directed paths through a base point in a topological space (with concatenation of paths as the group operation), and so on and on it goes. Groups are indeed very many and various! But you knew that.

(c) We need to agree some notation. So let's use '$(G, *, e)$' simply to abbreviate 'the objects G equipped with the operation $*$ and with distinguished object e'. Similarly, of course, for e.g. '$(H, \star, d)$' etc. And when convenient we can abbreviate such expressions further by '$\mathcal{G}$', '$\mathcal{H}$', etc.

If $(G, *, e)$ satisfy the conditions for forming a group, then let's briskly write 'the group $(G, *, e)$' (or simply 'the group $\mathcal{G}$') rather than 'the group consisting in $(G, *, e)$'.

As we have seen, the group operation can be significantly different from case to case (all that is required is that it satisfies Defn. 1). But it is customary to default in general to using multiplication-like notation and to talk generically of group 'products';[1] we will correspondingly default to denoting the inverse of a group object x by x^{-1}.

2.2 A very quick word about 'objects'

There is a view, introduced into modern logic by Frege, according to which there are *absolute* type-theoretic distinctions to be made between objects (individual things) and first-level functions sending objects to objects, and then between those items and second-level functions sending first-level functions to first-level functions, etc.

Whatever the virtues of that view, I should emphasize again that when we talk about the objects of a group, the notion of object in play here is a *relative* one. A group involves a group operation (a binary function of some type, whose inputs and outputs must be at the same type-level); and then this group's 'objects' are the items (of whatever type) which are the inputs and outputs for that function.

[1] Additive notation, however, is commonly used when dealing with abelian groups in particular.

These items can be objects-as-individuals (like numbers); but as already noted, the items can equally well be first-level functions (like permutations of some numbers, i.e. bijections between those numbers); or they can be of other types too.

Looking ahead, we will meet the same relative use of the notion of object when we get round to defining the general notion of a category.

2.3 New groups from old

(a) Given one or more groups, we can form further groups from them in a number of natural ways. For a start, there are subgroups, in the obvious sense:

Definition 2. $(G', *, e)$ is a *subgroup* of $(G, *, e)$ iff (i) G' are some of the objects G, and (ii) these objects G' are closed with respect to the group operation and taking inverses (meaning that all $*$-products and $*$-inverses of objects among G' are also among G'). △

Example: the even integers, under addition, form a subgroup of the additive group of integers. For another example: the complex numbers on the unit circle, under multiplication, form a subgroup of the multiplicative group of non-zero complex numbers.

(b) Next, products. And, as a preliminary, we first need the general idea of a *pairing scheme*:

Definition 3. A scheme for pairing objects G with objects G' provides

(i) some pair-objects O (which can be any suitable objects, and may or may not be already among G or G');

(ii) a binary pairing function which we can notate '$\langle\ ,\ \rangle$' which sends x from among G and x' from among G' to a pair-object $\langle x, x'\rangle$ among O (with every pair-object being some such $\langle x, x'\rangle$);

(iii) two unpairing functions which send a pair-object $\langle x, x'\rangle$ to x and x' respectively. △

Note, it is immediate from this definition that the pairing function sends distinct pairs x, x' and y, y' to distinct pair-objects $\langle x, x'\rangle$ and $\langle y, y'\rangle$.

Don't jump to over-interpreting the notation here. The angle-brackets might remind you of some standard set-theoretic construction of ordered pairs. But all we need for a pairing scheme are *some* objects to 'code' for pairs together with interlocking pairing and unpairing functions. For example, if the group objects G and G' are in both cases natural numbers, then we could perfectly well take the pair-object $\langle m, n\rangle$ to be the number $2^m 3^n$, with the obvious pairing and unpairing functions.

With Defn. 3 to hand, we can now define the notion of a product group:

Definition 4. Suppose we have the groups $\mathcal{G}$ and $\mathcal{G}'$, i.e. $(G, *, e)$ and $(G', *', e')$, together with some pairing scheme which maps an object x from G with an object x' from G' to a pair-object $\langle x, x'\rangle$. Let H be all the resulting pair-coding objects. Define $d = \langle e, e'\rangle$, and define multiplication of pairs componentwise, i.e. put $\langle x, x'\rangle \star \langle y, y'\rangle = \langle x * y, x' *' y'\rangle$. Then $(H, \star, d)$ is a *product* of the groups $\mathcal{G}$ and $\mathcal{G}'$ (which we can notate $\mathcal{G} \times \mathcal{G}'$). △

It is routine to check that $(H, \star, d)$ really is a group.

For a very simple example, suppose $\mathcal{J}$ is a group comprising just the two objects e, j. If $\mathcal{K}_1$ is to be a product of $\mathcal{J}$ with itself, it will need to comprise four distinct objects $\langle e, e\rangle$, $\langle e, j\rangle$, $\langle j, e\rangle$, $\langle j, j\rangle$, with the first of these being the group identity. For brevity's sake, call these four pair-objects $1, a, b, c$ respectively. $\mathcal{K}_1$'s group operation $\star$ is then defined by the following table (the entry at row r, column s, gives $r \star s$):

$\star$	1	a	b	c
1	1	a	b	c
a	a	1	c	b
b	b	c	1	a
c	c	b	a	1

The symmetry of the table reflects the fact that $\mathcal{K}_1$ is abelian.

Note, then, that we speak here of 'a' product of $\mathcal{J}$ with itself, not 'the' product. Why? Because there are unlimitedly many alternative schemes for coding pairs of objects, and different schemes will give rise to different product groups. In the present example, *any* four distinct objects we like can play the role of the required pair-objects, as long as we have pairing and unpairing functions to match. However, the resulting different groups *will* be equivalent-as-groups: each way of forming a product group from a two-object group and itself always give us a group describable by reinterpreting the same table. And the point of course generalizes. Products $\mathcal{G} \times \mathcal{G}'$ produced by using different pairing schemes will always be equivalent, in a familiar sense we'll clarify shortly.

(c) Now for a third, rather more interesting, way of forming new groups. We start with another general idea, and define a *quotient scheme*:

Definition 5. If G are some objects, and $\sim$ is an equivalence relation defined over G, then a corresponding quotient scheme provides

(i) some quotient-objects O (which can be any suitable objects, which may or may not be already among G),

(ii) a unary function which we can notate '[ ]' which sends x from among G to a quotient-object $[x]$ among O (with every quotient-object being some such $[x]$), where

(iii) for all x, y among G, $[x] = [y]$ iff $x \sim y$. △

So $[x]$ behaves in the crucial respect like an $\sim$-equivalence class containing x.

But note, just as pair-objects in pairing schemes do not have to be sets, we similarly do *not* require $[x]$ to be a set. For example, take the integers Z and consider the equivalence relation $\equiv_8$, i.e. congruence mod 8. Then we can simply put $[x]$ to be the remainder when x is divided by 8, and trivially $[x] = [y]$ iff $x \equiv_8 y$.

With our new definition to hand, we can now define the notion of a quotient group in two steps:

Definition 6. (i) If $\mathcal{G}$, i.e. $(G, *, e)$, is a group, then $\sim$ is a *congruence* relation for the group iff it is an equivalence relation which respects the group structure of $\mathcal{G}$. In other words, for any objects x, y, z from G, given $x \sim y$, then $x * z \sim y * z$ and $z * x \sim z * y$ (that is to say, 'multiplying' equivalent objects by the same object yields equivalent results).

(ii) Suppose $\mathcal{G}$, i.e. $(G, *, e)$ again, is a group, and $\sim$ is a congruence relation for the group; and suppose we also have a quotient scheme for $\sim$, which sends x among G to $[x]$. Let $[G]$ be all the objects $[x]$ for x among G, and put $[x] \star [y] = [x * y]$. Then $([G], \star, [e])$ is a *quotient* of the original group $\mathcal{G}$ with respect to $\sim$, which we symbolize $\mathcal{G}/\sim$. △

For this definition to be in order, we need to show that we have successfully defined a genuine function $\star$. In particular, then, we need to show that the result of $\star$-multiplication does not depend on how we pick out the multiplicands. In other words – *without* yet assuming $\star$ is a function so we can trivially substitute identicals! – we need to show that if $[x] = [x']$ then (i) $[x] \star [y] = [x'] \star [y]$, and (ii) $[y] \star [x] = [y] \star [x']$. But for (i), just note that if $[x] = [x']$, then by definition $x \sim x'$, hence (since $\sim$ respects group structure) $x * y \sim x' * y$, hence $[x * y] = [x' * y]$, hence by definition $[x] \star [y] = [x'] \star [y]$. We derive (ii) similarly.

It remains to check that $\mathcal{G}/\sim$ is then a group with $\star$ the group operation. But that's straightforward.

Note, we again talk of 'a' quotient group rather than 'the' quotient group. There will be many ways of finding quotient schemes for $\sim$, hence many alternative objects $[G]$ from which to build a quotient group $\mathcal{G}/\sim$ (though, as with product groups, quotient groups constructed using different quotient schemes will all 'look the same').

Let's take a quick example, to reinforce the point that the objects forming a quotient group need not be sets. Suppose $(Z, +, 0)$ is the additive group of the integers, $\mathcal{Z}$ for short, and consider again the equivalence relation of congruence mod 8 defined over Z. This equivalence relation respects the additive structure of the integers; for if $x \equiv_8 y$ then $x + z \equiv_8 y + z$ and $z + x \equiv_8 z + y$. As suggested before, we can take our quotient scheme for this equivalence relation simply to send x to the remainder on dividing x by 8; this gives us as quotient-objects the eight numbers from 0 to 7, call them together $\overline{8}$. Then $(\overline{8}, +_8, 0)$ is evidently a group (where $+_8$ is addition mod 8), and it is a quotient $\mathcal{Z}/{\equiv_8}$.

(d) Given groups $\mathcal{G}$ and $\mathcal{G}'$, do they always have a product? Given a group $\mathcal{G}$ and a congruence $\sim$, is there always a quotient group $\mathcal{G}/\sim$?

On the one hand, nothing we have said so far assumes positive answers. On the other hand, it is very natural to work on the very modest assumption that such constructions are freely available, i.e. we take it that product schemes and quotient schemes are always available when we want them. Elementary discussions of group theory typically proceed as if that modest assumption is permitted. We'll return to this point.

2.4 Group homomorphisms

(a) Next, let's equally briskly recall some basic facts about structure-preserving maps between the groups. My preferred – though again slightly deviant – style of definition is:

Definition 7. A *group homomorphism* from the group $(G, *, e)$ as source to the group $(H, \star, d)$ as target is a function f defined over the objects G with values among H such that:

(i) for every x, y among G, $f(x * y) = f(x) \star f(y)$,
(ii) $f(e) = d$. △

So a homomorphism sends products in the source group to corresponding products in the target group. It similarly sends the identity object in the source group to the identity in the target group.

Since $f(x) \star f(x^{-1}) = f(x * x^{-1}) = f(e) = d$, and similarly $f(x^{-1}) \star f(x) = d$, $f(x^{-1})$ is the inverse of $f(x)$. In other words, a homomorphism sends inverses to inverses.

Thought of simply in its role of mapping objects to objects, the function $f\colon G \to H$ is said to be the underlying function of the homomorphism. When thought of in its role as a structure-preserving homomorphism we can use the notation $f\colon (G, *, e) \to (H, \star, d)$, or $f\colon \mathcal{G} \to \mathcal{H}$.

(b) Some initial trivial examples:

(1) Let $(G, *, e)$ form a group $\mathcal{G}$, and let $\mathsf{1}$ be any one-object group. Then there is a homomorphism $f\colon \mathcal{G} \to \mathsf{1}$, which sends every object among G to the sole object of the target group. And this is the unique homomorphism from $\mathcal{G}$ to $\mathsf{1}$.

(2) Likewise, there is a homomorphism $g\colon \mathsf{1} \to \mathcal{G}$ which sends the single object of $\mathsf{1}$ to the group identity of $\mathcal{G}$. And this is the unique homomorphism from $\mathsf{1}$ to $\mathcal{G}$.

(3) Relatedly, there is always a 'collapse' homomorphism $h\colon \mathcal{G} \to \mathcal{G}$ which sends every $\mathcal{G}$-object to its group identity e.

These cases remind us that, although homomorphisms are often described as *preserving* group structure, this doesn't mean replicating *all* structure. A homomorphism from $\mathcal{G}$ to $\mathcal{H}$ can compress many or most aspects of $\mathcal{G}$'s structure simply by mapping distinct $\mathcal{G}$-objects to one and the same $\mathcal{H}$-object. Perhaps it would be rather better to talk of homomophisms as *respecting* group structure.

Three more elementary examples:

(4) There is a homomorphism from $\mathcal{Z}$, the additive group of integers $(Z, +, 0)$, to any two object group $\mathcal{J}$ which sends even numbers to $\mathcal{J}$'s identity, and

sends odd numbers to $\mathcal{J}$'s other object. The underlying function here is surjective but not injective.

(5) There is a homomorphism from $\mathcal{Z}$ to $\mathcal{Q}$, the additive group of rationals $(Q, +, 0)$, which sends an integer n to the corresponding rational $n/1$. As a function from Z to Q, this is injective but not surjective.

(6) Let R be the real numbers, and C^* the non-zero complex numbers. The reals form a group under addition, and the non-zero complex numbers form a group under multiplication. Define $j\colon (R, +, 0) \to (C^*, \times, 1)$ by putting $j(x) = \sin x + i \cos x$. Then we have a homomorphism whose underlying function is neither injective nor surjective.

(c) Let's pause to see what can be said about group homomorphisms in general, various though they have already proved to be.

Theorem 1. *(1) Any two homomorphisms $f\colon \mathcal{G} \to \mathcal{H}$, $g\colon \mathcal{H} \to \mathcal{J}$, with the target of the first being the source of the second, will compose to give a homomorphism $g \circ f\colon \mathcal{G} \to \mathcal{J}$.*

(2) Composition of homomorphisms is associative. In other words, if f, g, h are group homomorphisms which can compose so that one of $h \circ (g \circ f)$ and $(h \circ g) \circ f$ is defined, then the other composite is defined, and the two composites are equal.

(3) For any group $\mathcal{G}$, there is an identity homomorphism $1_{\mathcal{G}}\colon \mathcal{G} \to \mathcal{G}$ which sends each object to itself. Then for any $f\colon \mathcal{G} \to \mathcal{H}$ we have $f \circ 1_{\mathcal{G}} = f = 1_{\mathcal{H}} \circ f$.

Proof. For (1) we, of course, simply take $g \circ f$ ('g following f') applied to an object x among the objects of $\mathcal{G}$ to be $g(f(x))$ and then check that $g \circ f$ so defined does satisfy the condition for being a homomorphism given that g and f do.

For (2), associativity of homomorphisms is inherited from the associativity of ordinary functional composition for the underlying functions.

(3) is also immediate. □

(d) A quick but important remark. Note that this, our very first theorem, is *not* a mere logical consequence of our definitions of groups and group homomorphisms. Our proof plainly depends on invoking background assumptions about functions, such as the assumption that functional composition is associative. And so it goes. Almost *nothing* in group theory just follows from the definitions alone.

2.5 Group isomorphisms and automorphisms

(a) Now we highlight the special case where the underlying function of a homomorphism is both injective and surjective, so it gives rise to a nice one-to-one correspondence between two groups (or a group and itself).

Definition 8. A *group isomorphism* $f\colon \mathcal{G} \xrightarrow{\sim} \mathcal{H}$ is a homomorphism where the underlying function is a bijection between the objects of $\mathcal{G}$ and the objects of $\mathcal{H}$.

We say that the groups $\mathcal{G}$ and $\mathcal{H}$ are *isomorphic* as groups iff there is a group isomorphism $f\colon \mathcal{G} \xrightarrow{\sim} \mathcal{H}$, and then write $\mathcal{G} \simeq \mathcal{H}$

A *group automorphism* is a group isomorphism $f\colon \mathcal{G} \xrightarrow{\sim} \mathcal{G}$ whose source and target are the same. △

Again, let's have some elementary examples:

(1) Any two two-object groups are isomorphic. Take the group comprising e, j, equipped with the only possible group operation $*$, and the group comprising e', j', equipped with $*'$. Then the map which sends the group identity e to the group identity e' and j to j' is obviously a group isomorphism.

(2) There are two automorphisms from the additive group $\mathcal{Z}$ to itself. One is the trivial identity homomorphism; the other is the function which sends an integer j to $-j$.

(3) There are infinitely many automorphisms from the group $\mathcal{Q}$ to itself. Take any non-zero rational q: then the map $x \mapsto qx$ 'stretches/compresses' the rationals, perhaps reversing their order, while still preserving additive structure.

(4) Let $\mathcal{K}_2$ be the group consisting in the numbers $1, 3, 5, 7$ equipped with multiplication mod 8. And let $\mathcal{K}_3$ be the group of symmetries of a non-equilateral rectangle whose four 'objects' are the operations of leaving the rectangle in place, vertical reflection, horizontal reflection and rotation through 180°, with the group operation being simply composition of geometric operations. Then $\mathcal{K}_2 \simeq \mathcal{K}_3$.

The easiest way to see this is by constructing the abstract 'multiplication table'. First, take $1, a, b, c$ to be respectively the numbers $1, 3, 5, 7$, and take $\star$ to be multiplication mod 8. Second, take $1, a, b, c$ to be the geometric operations on a rectangle in the order just listed and take $\star$ to be composition. Both times we get the same table as for $\mathcal{K}_1$ that we met in §2.3. Matching up the two new interpretations of $1, a, b, c$ and the two corresponding interpretations of $\star$ gives us the claimed isomorphism $f\colon \mathcal{K}_2 \xrightarrow{\sim} \mathcal{K}_3$. By the same reasoning, both groups are isomorphic to $\mathcal{K}_1$.

This illustrates an obvious general point. Groups that can interpret the same 'multiplication table' are isomorphic; conversely, isomorphic groups can be described by the same (possibly infinite) table, when suitably reinterpreted.

(5) In defining a product of two groups, we were allowed to invoke any scheme for coding pairs of objects from the two groups. But whichever scheme we choose, the resulting product (we said) will 'look the same', and have the same multiplication table. We can now put it like this: suppose $\mathcal{H}_1$ and $\mathcal{H}_2$ are both products of $\mathcal{G} \times \mathcal{G}'$; then $\mathcal{H}_1 \simeq \mathcal{H}_2$.

Why? Just take the bijection which sends the pair-object $\langle x, x' \rangle_1$ which pairs x from $\mathcal{G}$ and x' from $\mathcal{G}'$ according to the pairing scheme used in constructing $\mathcal{H}_1$ to the corresponding pair-object $\langle x, x'\rangle_2$ formed according to the pairing scheme used in constructing $\mathcal{H}_2$. This is trivially seen to be a group isomorphism from $\mathcal{H}_1$ to $\mathcal{H}_2$.

Likewise, suppose $\mathcal{H}_1$ and $\mathcal{H}_2$ are different quotients of a group $\mathcal{G}$ with respect to a congruence relation $\sim$, different because they rely on different quotient schemes for, in effect, representing $\sim$-equivalent classes of objects from $\mathcal{G}$. Take the bijection that sends the quotient-object $[x]_1$ according to the first quotient scheme to the corresponding object $[x]_2$ according to the second scheme. Then by a similar argument we again have $\mathcal{H}_1 \simeq \mathcal{H}_2$.

(b) Another very easy result, for future reference:

Theorem 2. *A group homomorphism $f\colon \mathcal{G} \to \mathcal{H}$ is an isomorphism iff it has a two-sided inverse, i.e. there is a homomorphism $g\colon \mathcal{H} \to \mathcal{G}$ such that $g \circ f = 1_{\mathcal{G}}$ and $f \circ g = 1_{\mathcal{H}}$.*

Proof. Suppose $f\colon (G, *, e) \to (H, \star, d)$ is an isomorphism. Then by definition the underlying function $f\colon G \to H$ is a bijection and so has a two-sided inverse $g\colon H \to G$. We now need to show that this inverse function g gives rise to a homomorphism $g\colon (H, \star, d) \to (G, *, e)$. But since f is a homomorphism, and g is its two-sided inverse, we have $g(x \star y) = g(fgx \star fgy) = gf(gx * gy) = gx * gy$. In addition, as required, $gd = gfe = e$.

Conversely, suppose f is a homomorphism with a two-sided inverse. Then its underlying function must have a two-sided inverse; but it is a familiar elementary result that a function with a two-sided inverse is a bijection. □

Evidently, a group is isomorphic to itself (by the identity homomorphism) and the composition of two group isomorphisms is also an isomorphism. And given that isomorphisms are homomorphisms with two-sided inverses which are homomorphisms, it is immediate that the inverse of an isomorphism is also an isomorphism.

Therefore, just as we would want,

Theorem 3. *Being isomorphic is an equivalence relation between groups.* □

2.6 Another way of forming new groups from old

Take any group $\mathcal{G}$ and consider its automorphisms $Aut_{\mathcal{G}}$. There is of course at least one such automorphism, namely the identity map $1_{\mathcal{G}}$. Note too that any two of $\mathcal{G}$'s automorphisms f, g compose to give us a new automorphism $g \circ f$. Composition here is associative. And we've just noted that isomorphisms in general, and hence automorphisms in particular, have inverses with respect to composition. Hence:

Theorem 4. *For any group $\mathcal{G}$, $(Aut_{\mathcal{G}}, \circ, 1_{\mathcal{G}})$ form a group, the automorphism group of $\mathcal{G}$, $AUT(\mathcal{G})$.* □

For example, we've already remarked that there are exactly two automorphisms from $\mathcal{Z}$ to itself; so $AUT(\mathcal{Z})$ is a two-object group. And what is the automorphism group of that two-object group? A trivial one-object group.

By contrast, since 'stretching by a non-zero rational' is an automorphism for the additive group $\mathcal{Q}$, and stretchings can be composed by multiplying the stretching factor, the corresponding automorphism group $AUT(\mathcal{Q})$ will be isomorphic to the multiplicative group of non-zero rationals.

For one more example, look at the 'multiplication table' for $\mathcal{K}_1$ again. We see that if we swap the three entries a, b, c around, we keep the same structure. So $AUT(\mathcal{K}_1)$ will be a group of permutations of three objects. And what does the automorphism group of *that* look like? It turns out to be the same again, a group of permutations of three elements. What fun!

2.7 Homomorphisms and constructions

In §2.3 we considered some basic ways of forming new groups from old, yielding subgroups, product groups and quotient groups. In §2.4 we introduced structure-preserving maps between groups. We now bring the two themes together, foreshadowing a central motif of category theory.

(a) For the simplest case, start by noting how homomorphisms give rise to subgroups and vice versa.

Theorem 5. *For any homomorphism $f\colon \mathcal{G} \to \mathcal{H}$, the f-image of $\mathcal{G}$ is a subgroup of $\mathcal{H}$. Conversely, for every subgroup of $\mathcal{H}$, there is a homomorphism $f\colon \mathcal{G} \to \mathcal{H}$ such that that subgroup is the f-image of $\mathcal{G}$.*

Proof. Given a group homomorphism $f\colon (G, *, e) \to (H, \star, d)$, let $f[G]$ be all the objects which are f-images of objects from among G, so they include d, i.e. $f(e)$. Define $f(\mathcal{G})$, the f-image of the group $\mathcal{G}$, in the obvious way as $(f[G], \star, d)$. We now need to check this too is a group, and hence a subgroup of $\mathcal{H}$.

(i) Suppose y_1 and y_2 are among $f[G]$. By assumption, they are f-images of some objects x_1, x_2 among G. So we have $y_1 \star y_2 = fx_1 \star fx_2 = f(x_1 * x_2)$, and hence $y_1 \star y_2$ will also be among $f[G]$ as required.

(ii) Since $\star$ is associative and d an identity for that operation, it only remains to show that if y is among $f[G]$ its inverse is too. But y is by assumption $f(x)$ for some object x among G, and homomorphisms send inverses to inverses. So the inverse of y, i.e. $(fx)^{-1}$, is $f(x^{-1})$ and hence is among $f[G]$.

That establishes the first half of our theorem. For the converse half, just note that any subgroup $\mathcal{G}$ of $\mathcal{H}$ gives rise to a trivial injection map $i\colon \mathcal{G} \to \mathcal{H}$ which sends an object from $\mathcal{G}$ to the same object now considered as an object of $\mathcal{H}$. $\square$

Hence we can characterize the subgroups of a given group $\mathcal{H}$ in terms of group-homomorphisms with the target $\mathcal{H}$. Putting it roughly, then, we can trade in claims about what goes on *inside* various groups when forming subgroups for claims about corresponding homomorphisms *between* groups.

(b) I'll quickly mention another essential link between homomorphisms and subgroups. We start by introducing what turns out to be another important idea of group theory:

Definition 9. $(K, *, e)$ is a *normal subgroup* of $(G, *, e)$ iff it is a subgroup and, for any k among K and any g among G, $g * k * g^{-1}$ is also among K.

Then we have, in particular, the following result:

Theorem 6. *Suppose $f\colon (G, *, e) \to (H, \star, d)$ is a group homomorphism. Let K be the objects among G which f maps to the identity d. Then $(K, *, e)$ is a normal subgroup of $(G, *, e)$.*

Proof. We need to show first that the objects K are closed under the operation $*$. But suppose k_1 and k_2 are among K. Then $f(k_1 * k_2) = fk_1 \star fk_2 = d \star d = d$. Hence $(k_1 * k_2)$ is also among K.

By the definition of a homomorphism, $f(e) = d$, the identity e is among K. Then recall that homomorphisms send inverses to inverses. So if $f(k) = d$, then $f(k^{-1}) = d^{-1} = d$; so the inverse of an object among K is still among K. Hence $(K, *, e)$ is indeed a subgroup of $(G, *, e)$.

For normality, we simply note that for any k among K and g among G, $f(g * k * g^{-1}) = f(g) \star f(k) \star f(g^{-1}) = f(g) \star e \star f(g)^{-1} = e$, so $g * k * g^{-1}$ is also among K as required. ☐

There is a converse theorem too, that every normal subgroup for a group $\mathcal{G}$ is the kernel of some homomorphism with the source $\mathcal{G}$. So again, we can trade in claims about what goes on *inside* various groups, making them normal subgroups, for claims about corresponding homomorphisms *between* groups.

(c) I'll skip past product groups for now, and next consider quotient groups arising from suitable equivalence relations. We then have the following result:

Theorem 7. *Given a group homomorphism $f\colon \mathcal{G} \to \mathcal{H}$, and x, y among $\mathcal{G}$'s objects, put $x \sim y$ iff $fx = fy$. Then $f(\mathcal{G})$, the f-image of the group $\mathcal{G}$, is a quotient group $\mathcal{G}/{\sim}$. Conversely, given a quotient group of $\mathcal{G}$ with respect to a congruence relation $\sim$, we can find a group $\mathcal{H}$ and homomorphism $f\colon \mathcal{G} \to \mathcal{H}$, such that $\mathcal{G}/{\sim}$ is $f(\mathcal{G})$.*

Proof. The relation $\sim$ of being equalized-by-f is trivially an equivalence relation. But we need to check that $\sim$ respects $\mathcal{G}$'s group operation $*$ so that $\mathcal{G}/{\sim}$ exists. In other words, we need to show that for any group objects x, y, z, given $x \sim y$, then (i) $x * z \sim y * z$ and (ii) $z * x \sim z * x$.

But for (i), if $x \sim y$, then $fx = fy$, hence $f(x*z) = fx \star fz = fy \star fz = f(y*z)$, hence $x * z \sim y * z$ (here, $\star$ is of course $\mathcal{H}$'s group operation). Case (ii) is exactly similar.

By the definition of $\sim$, the f-images of objects among G act like quotient-objects with respect to $\sim$; so it is immediate that $f(\mathcal{G})$ is a quotient group $\mathcal{G}/{\sim}$.

For the converse result, suppose $\mathcal{G}/\sim$ is a quotient of $\mathcal{G}$ with respect to some equivalence relation $\sim$, with $f_\sim : x \mapsto [x]$ giving us the relevant quotient scheme. Then $f_\sim : \mathcal{G} \to \mathcal{G}/\sim$ is easily checked to be a homomorphism, and $f_\sim(\mathcal{G})$ is the whole of $\mathcal{G}/\sim$. □

So again we can trade in certain claims about the structure of certain groups, this time about their quotient structure, for corresponding claims about homomorphisms between groups.

And note further that this trade reveals something that was not obvious before, namely that there is a kind of duality between the relation of being a quotient group and the relation of being a subgroup: given a homomorphism $f : \mathcal{G} \to \mathcal{H}$, $f(\mathcal{G})$ is a quotient of $\mathcal{G}$ and a subgroup of $\mathcal{H}$.

(d) Similarly, it turns out that claims about the structure of product groups can also be traded in for claims about corresponding homomorphisms between groups (we use the fact that pairing schemes essentially involve pairing and unpairing functions that behave in the right way). But I'll leave the proof of this for later. For now, I'll just flag up the general point that these sorts of trades – i.e. trades between claims about the 'internals' of structures and claims about 'external' maps between structures – will turn out to be an absolutely central motif of category theory.

2.8 'Identical up to isomorphism'

(a) We have met the groups $\mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3$ which are isomorphic to each other. They are also isomorphic to any other group whose four objects can be labelled $1, a, b, c$ in such a way that the same 'multiplication table' in §2.3 applies again. Call such groups *Klein four-groups*. And note, the way in which the various Klein four-groups differ from each other, namely in the internal constitution of their various *objects*, is not relevant to their core behaviour as groups, for that depends just on the *functional relations between the objects*. In other words, despite the differences between their objects, the groups are the same at least as far as their structural properties – i.e. the properties as determined by their shared 'multiplication table' – are concerned.

A bit of care is needed in describing the situation, however. Consider, for example, the following from a rightly well-regarded algebra text:

> The groups $\mathcal{G}$ and $\mathcal{H}$ are isomorphic if there is a bijection between them which preserves the group operations. Intuitively, $\mathcal{G}$ and $\mathcal{H}$ are the same group except that the elements and the operations may be written differently in $\mathcal{G}$ and $\mathcal{H}$. (Dummit and Foote 2004, p. 37)

But that surely isn't a very happy way to putting things. We have just reminded ourselves that $\mathcal{K}_2$ and $\mathcal{K}_3$ are isomorphic groups. But $\mathcal{K}_2$ comprises four *numbers* as its objects, and $\mathcal{K}_3$ comprises four *operations* on a non-equilateral rectangle; and there is no sense in which numbers and geometric operations can be thought of as the same things 'written differently'.

If anything, then, it is exactly the other way around: we have here distinct groups comprising different elements and different group operations which, however, can be 'written the same', in the sense of being summed up by the same table differently interpreted.

A rather happier, and widely used, way of putting things is this: $\mathcal{K}_2$ and $\mathcal{K}_3$ are identical *up to isomorphism*. And (now reading that quotation more charitably) for many purposes, group theory can indeed ignore the differences between groups which are identical up to isomorphism.

(b) It is common, then, to talk of *the* Klein group $\mathcal{K}$, and similarly to talk of *the* permutation group of three elements $\mathcal{S}_3$, *the* free group over the generators G, and so on. And in most contexts, we can let this pass quite happily. Though if we are being pernickety, such talk can be cashed out in one of two or perhaps three ways:

1. Most simply, talk of *the* Klein group can typically be treated as just generalizing talk about Klein groups. So 'the Klein group is abelian' is to be understood as simply saying that any Klein group is abelian. Similarly, 'There is a unique homomorphism from the Klein group to the one object group' says that for any Klein group and any one object group, there is a unique homomorphism from the first to the second. And so on.

2. Sometimes, though, a paradigm case, a canonical exemplar, is introduced. For example, *the* free group with certain generators is often defined by a specific construction using 'words' formed from the generators.

3. Occasionally, and more mysteriously, some flirt with the idea that, as well as 'concrete' Klein groups (to return to that example), i.e. groups whose elements have an independent nature (which could be numbers, pairs of numbers, rotations and reflections, whatever), there is also a more purely 'abstract' Klein group. This has the right multiplication table, but is supposedly built up from objects with no properties at all over and above being sent to each other by the group operation according to the given table. It is this purely abstract group comprising de-natured elements which is then said to be, properly speaking, *the* Klein group.

For now, we'll hang fire on the question whether the third option makes much sense. We will in due course need to revisit this sort of question.

2.9 Categories of groups

That should hopefully be enough for present purposes by way of some brisk revision notes on groups and their homomorphisms!

(a) Now, we said at the beginning of the first chapter that 'suitably structured families of structures' are prime examples of categories. Then at the start of this chapter, we picked out a paradigm case, namely a family of groups organized by homomorphisms between them.

Given some groups, how many homomorphisms between them do we need to make them into a category of groups? We will impose just two very natural conditions. First, the homomorphisms in the category should be closed under composition. And second, the identity homomorphisms for each relevant group needs to be included. It's that simple!

(b) But now let's say exactly the same thing again, but this time in laborious detail, for clarity's sake. So:

Definition 10. A *category of groups* comprises

(1) some groups *Grp*, and

(2) some group homomorphisms *Hom*,

where these groups and homomorphisms are governed by the following conditions:

Sources and targets For each homomorphism $f\colon \mathcal{G} \to \mathcal{H}$ among *Hom*, both its source group $\mathcal{G}$ and its target group $\mathcal{H}$ are among *Grp*.

Composition For any two homomorphisms $f\colon \mathcal{G} \to \mathcal{H}$, $g\colon \mathcal{H} \to \mathcal{I}$ among *Hom*, where the target of f is the source of g, the homomorphism $g \circ f\colon \mathcal{G} \to \mathcal{I}$ is also among *Hom*.

Identity homomorphisms For every group $\mathcal{G}$ among *Grp*, the identity homomorphism $1_\mathcal{G} : \mathcal{G} \to \mathcal{G}$ is among *Hom*.

The homomorphisms also satisfy the following conditions:

Associativity of composition. For any $f\colon \mathcal{G} \to \mathcal{H}$, $g\colon \mathcal{H} \to \mathcal{I}$, $h\colon \mathcal{I} \to \mathcal{J}$, we have $h \circ (g \circ f) = (h \circ g) \circ f$.

Identity homomorphisms do behave as identities. For any $f\colon \mathcal{G} \to \mathcal{H}$ we have $f \circ 1_\mathcal{G} = f = 1_\mathcal{H} \circ f$. △

Of course, we know the last two conditions will automatically be satisfied in the case of group homomorphisms because of Theorem 1. But I'm (redundantly) mentioning those conditions here so that our account of categories of groups matches up nicely with our later general definition of categories.

(c) Just as groups are many and various, so too are categories of groups. For example, a single group $\mathcal{G}$ together with its identity homomorphism $1_\mathcal{G} : \mathcal{G} \to \mathcal{G}$ counts as a trivial category of groups. So too does any uncommunicative bunch of groups equipped only with their identity homomorphisms.

But those are *very* unexciting cases! Things can get more interesting when the groups in a category start to communicate (so to speak).

Consider next, then, the category which comprises all the finite groups whose objects are natural numbers together with all the isomorphisms between them. Now there is a *bit* of structure to the category, with the isomorphic groups at least connected together by the maps between them. But this is still of relatively little interest: we have different islands of isomorphic groups, and a group inhabiting one island knows nothing about groups inhabiting other islands.

So let's move on to consider the category comprising those same finite groups but this time combined with *all* the homomorphisms between them (whether isomorphisms or not). And *now* non-isomorphic groups can 'see' each other. We have enough homomorphisms in play to be able e.g. distinguish the one-object groups by saying that these are the groups which have one and only one homomorphism to and from every other group, as indicated in §2.4. We can also use these homomorphisms to tell a story about e.g. subgroups and quotient groups living in the category, as indicated in our preliminary sketch in §2.7. Developing this sort of story will be a primary item of business in the coming chapters.

But first, we do really need to pause to say something about the universe where we can by default take categories of groups to live (along with all the other categories we will want). That's business for the next chapter.

3 Groups and sets

The last chapter reviewed some elementary facts about groups and their homomorphisms. And everything we said, at least before we mentioned categories in the final section, was probably very familiar. Except in one respect. I almost entirely avoided talking about *sets*.

This was the mildly deviant feature of the presentation. There was a reason – a good enough reason, as I hope to make clear – for introducing groups in my way, and avoiding for the moment the conventional set-theoretic idiom; but there is also a good reason for now bringing sets back to centre stage. This chapter explains.

3.1 Sets, virtual classes, plurals

(a) We need to begin by getting an important distinction into focus.

Following Cantor, I'll understand a set – properly so called – to be a unity, a thing in itself over and above its members (so the 'set of' operator takes zero, one, or many things, and outputs a single new thing).

But if this is the guiding idea, then the first point to note is that *a great deal of elementary informal set talk is really no more than a façon de parler*. Yes, it is a useful and familiar idiom for talking about many things at once; but in many elementary contexts informal talk of a set doesn't really carry any serious commitment to there being any *additional* object over and above those many things. On the contrary, apparent singular talk about *the set of Xs* can often be paraphrased away into talk directly about those Xs, without loss of content. Talk about the set of prime numbers, for example, can typically just be taken as a way of talking about the prime numbers themselves.

When it can be paraphrased away, talk of sets is said to be talk of *virtual classes*.[1] And the idea that there is a distinction to be made between sets and virtual classes is an old one. Here is Paul Finsler, writing a century ago:

> It would surely be inconvenient if one always had to speak of many things in the plural; it is much more convenient to use the singular

[1] See W.V.O. Quine's famous discussion in the opening chapter of his *Set Theory and its Logic*: "Much ... of what is commonly said of classes with the help of '∈' can be accounted for as a mere manner of speaking, involving no real reference to classes nor any irreducible use of '∈'. ... [T]his part of class theory ... I call the virtual theory of classes."(Quine 1963, p. 16)

> and speak of them as a class. ... A class of things is understood as being the things themselves, while the set which contains them as its elements is a single thing, in general distinct from the things comprising it. ... Thus a set is a genuine, individual entity. By contrast, a class is singular only by virtue of linguistic usage; in actuality, it almost always signifies a plurality.[2]

Finsler writes 'almost always', I take it, because a class term may in fact denote just one thing, or even – perhaps by misadventure – none.

Nothing at all hangs, of course, on the choice of particular word here, 'class' vs 'set'. What matters is the distinction between non-committal, eliminable, talk – talk of merely virtual sets/classes/pluralities (whatever we call them) – and uneliminable talk of sets as entities in their own right.

(b) I'm not so sure that Finsler is right, though, about the inconvenience of plural talk. In fact, for clarity's sake, I do think the best policy is simply to eschew class talk when we can, and to stick to plural locutions when there's no particular need to invoke sets in the Cantorian sense. So that will be the general policy in this book.

Do note that there is nothing suspect or unnatural about the use of plural terms. Consider, for example, terms such as 'the real numbers between 0 and 1', 'the points where line L intersects curve C', 'the finite groups of order 8', 'Hilbert's axioms for geometry', 'the symmetries of a rectangle', 'the ordinals', etc. Mathematicians of course are always using such terms which (taken at face value) refer plurally, to many things – and they use them without the slightest sense of strain or impropriety.

And don't be tempted by the thought that, all the same, we should really construe informal plural talk about Xs as disguised singular talk referring to *the set of* X's (where the set is a single item and something distinct, over and above its members). For you already know that *that* can't always be done. We can't, for example, trade in universally generalizing plural talk about the ordinals for singular talk about the set of ordinals because, on standard assumptions, there *is* no set of ordinals (there are as many ordinals as sets – set-many, for short – and that is too many to form a set).

The same goes, as we will see, when it comes to defining categories (and this matters). We can't in general treat a category as comprising a *set* of items together with the maps between them. For there may be too many relevant items to form a set. We will meet a first (and typical) example at the end of this chapter.

(c) In sum, plural talk (for example, of the kind I used in talking about groups in the previous chapter) is in perfectly good logical order as it is, taken at face value, without needing to be re-interpreted as referring to sets.[3]

[2]Finsler 1926, p. 106, quoted in Incurvati 2020, p. 3.

[3]That is still true, even if your measure of being in good logical order is formalizability: for an extended formal treatment of how to argue with plural terms and plural quantifiers, taking them at face value, see e.g. Oliver and Smiley's *Plural Logic* (2016).

And, as just hinted, we will need plural talk – and preferably not disguised as talk of (virtual) classes – in framing our theory of categories.

3.2 Group theory again

(a) Let's return to thinking in general terms about what it takes to develop group theory, initially as informal mathematics in the standard sort of way.

As noted before, even as soon as we reach our very elementary Theorem 1 we are going beyond the mere logical consequences of our definitions of groups and group homomorphisms. So what do we need to bring to the table to get group theory going? Roughly: the usual mathematical stock-in-trade of a body of assumptions about *functions* together with a generous repertoire of available *constructions*.

For example, we assume that any association of inputs to single outputs (nicely specifiable or not) constitutes a function. And functions always do compose when they can (i.e. when the target of the first is the source of the second), and composition is associative. We assume that there can exist functions of any possible type we want (i.e. functions of objects, functions of functions, functions of functions-of-functions, and functions of mixed types too). At a finer grain, we assume e.g. facts about injections, surjections, bijections. We assume that it makes sense, e.g. to talk about all the bijections between some objects and those same objects. And so on, and so forth.

Again, we typically assume that we can construct (what will serve as) pairs ad libitum; and we assume that when an equivalence relation partitions some objects we can somehow get representatives for the partitions. That is to say, in our earlier terms, we assume pairing schemes and quotient schemes are available whenever we want them. And not only we can construct pairs and finite tuples but infinite sequences too. We also assume that we can freely construct 'copies' of whatever structures we already have. And so on, and so forth.

That's vague, but intentionally so. I am simply gesturing at the way that textbook developments of group theory help themselves from the outset to whatever unproblematic background assumptions are needed as they go along.

(b) Now, as acknowledged at the beginning of this chapter, it is of course conventional to adopt a set-theoretic idiom in introducing groups. So instead of saying, as I did, that a group $(G, *, e)$ is some objects G equipped with a suitable binary operation $*$ and a distinguished object e, we more usually define a group as a set whose members are a non-empty *set* of objects, etc. Similarly, we usually take group homomorphisms to be functions which have sets as source and target, etc., and conventionally treat these functions as themselves sets of ordered pairs.

But how much heavy lifting is really being done here by the conventional invocation of sets? At the beginning of his fine book *Algebra, Chapter 0*, Paulo Aluffi frankly remarks that the informal set idiom which he adopts is in fact "little more than a system of notation and terminology" (Aluffi 2009, p. 1). And we can indeed develop group theory quite extensively without talking of sets or making essentially set-theoretic assumptions; part of the point of the previous chapter was to show how to make a start on this.

Still, framing the presentation of group theory as dealing with sets does give us a bit more than Aluffi says. For a start, it provides us with standard ways of constructing pairing schemes and quotient schemes whenever we want them. More generally, a background universe of sets can provide implementations for all the normal functions and constructions we can want. But, for all that, it is worth emphasizing that those functions and constructions are not intrinsically set-theoretic: and it is rather important for our purposes to highlight the distinction between the gadgetry we need in developing group theory, for example, and the set-theoretic underpinnings we might offer for that gadgetry. For we want to leave room for the thought that other underpinnings might be available.[4]

To take one example, a suitable *topos* (that's a distinctively category-theoretic notion) can arguably provide an alternative, even improved, framework in which we can regiment our ordinary mathematical gadgetry (including constructions like pairing and quotienting). And that claim would be very puzzling if we have already jumped too quickly to assuming that ordinary mathematics is already fixedly set-theoretic through and through.

3.3 Implementing structures in universes of sets

I've just talked of a universe of sets as providing *implementations*[5] for the functions and constructions of informal mathematics, and thereby left room for the idea that other implementations might be possible. But why talk of 'implementations'? – aren't functions standardly just *defined* to be certain sets? Aren't e.g. ordered pairs standardly defined as certain sets too?

Indeed they are: but we mustn't over-interpret 'defined' here. The point should, I hope, be a familiar one. This section is for those who need the point spelt out.

(a) Consider, for a first simple example, the case of one-place functions. Agree for the moment on some way of implementing ordered pairs as sets, e.g. as Kuratowski pairs $\langle x, y\rangle_K$ ($=_{\text{def}}$ $\{\{x\}, \{x, y\}\}$). The following definition is entirely standard:

Definition 11. Given a function $f\colon X \to Y$, the *graph* of f is the corresponding set $\hat{f}$ of ordered pairs $\langle x, y\rangle_K$ where x is among the objects X and y is among Y, and $fx = y$. △

[4]To complicate the story a bit: different underpinnings – set theoretic or otherwise – can give, at the margins, different stories about what functions are available e.g. to be group homomorphisms, and hence lead to some different answers to some of the more arcane group-theoretic questions. For a neat illustration, involving automorphism groups, see Hamkins (2002). But we needn't pursue this thought here.

[5]I'll to stick to that term. None is perhaps ideal, but alternatives like 'representation', 'proxy', 'surrogate' have their potentially misleading connotations, and 'implementation' is both common and relatively colourless.

Then it is a conventional textbook policy to 'define' a function $f\colon X \to Y$ as being the graph $\hat{f}$.[6] But we should certainly resist over-interpreting this as an outright *identification* of a function with its graph. Why so?

For a start, let's consider the function which maps an object to its singleton. Then – by the set-theorists' own lights – it doesn't have a graph: the totality of pairs $\langle x, \{x\}\rangle_K$, pairing-up every set x with its singleton, is the size of the universe of sets and so is 'too big' to be a set. Likewise, the function which maps every ordinal to its successor is also 'too big' to have a graph. Therefore not all functions can be identified with their graphs.

Just one counterexample is enough to defeat a universal claim. It might be suggested, though, that the cases where a function applies to too many things to be a set are in some sense rogue cases. So, in a concessive spirit, let's put such cases aside for a moment and see where that gets us.

Well, next note that the definition of a function as a set of ordered pairs involves arbitrary choices:

(i) For a start, it is arbitrary to fix on Kuratowski's particular implementation of pairs as sets.

(ii) And even relative to a choice of set-theoretic pairing scheme, we could equally well implement a function using the set of pairs $\langle y, x\rangle$ where $f(x) = y$, rather than by the set of pairs $\langle x, y\rangle$ – some textbooks do just this. Other choices are also possible.

However, if various permutations of choices at stages (i) and (ii) are pretty much as workable as each other, then we surely can't suppose that – when we choose to define a function as its graph – we have made the uniquely *right* choice, i.e. the choice that correctly identifies which set that function really is. But if there is no determinate fact of the matter about which sets functions are, then functions aren't sets. What remains true is that, for many purposes, appropriate sets can be used as proxies for functions or, as I put it, functions can be *implemented* as sets.

(b) It is worth digging a bit deeper. The key underlying point is that *functions just aren't the right logical type of thing to be sets.* As Alonzo Church puts it:

> it lies in the nature of any given [one-place] function to be applicable to certain things and, when applied to one of them as argument, to yield a certain value. (Church 1956, p. 15)

And to show that it isn't just philosophically-minded logicians who care about this, here is Terence Tao on the same theme:

[6] 'Define' is the word most often used. Fine print: some would prefer, for rather good reasons which chime with category theory, to define a function as a set-theoretic *triple* whose members are the function's graph, its domain (treated as a set, rather than plurally, of course), and its co-domain. But our comments about the simpler version of the standard account will carry over, mutatis mutandis, to the fancier version, so we need not delay over this just now. Compare §4.6(b).

> functions are not sets, and sets are not functions; it does not make sense to ask whether an object x is an element of a function f, and it does not make sense to apply a set A to an input x to create an output $A(x)$. (Tao 2016, p. 51)

For example, a function such as the factorial defined over the natural numbers is, of its nature, the type of thing which yields a numerical value when given a number as argument. By contrast a set doesn't, of its nature, take an argument or yield a value. And what applies to sets in general applies to e.g. sets of ordered pairs of numbers in particular.[7]

Which isn't for a moment to deny that we can make use of the graph of a function (a glorified input-output look-up table) in mapping an input object to an output value. But to do this, we need to deploy *another* function, namely a two-place evaluation function which takes an object x and the graph, and outputs y if and only if the pair $\langle x, y\rangle_K$ is in the graph. And unless we are planning to set off on an infinite regress, we had better not seek to again trade in this evaluation function for another set.

(c) We can briskly make parallel remarks about another standard 'definition'. Recall another bit of terminology:

Definition 12. Given a binary relation R which holds between objects X and Y, the *extension* of R is the set $\hat{R}$ of ordered pairs $\langle x, y\rangle_K$ where x is among X, y is among Y, and xRy. △

Then the usual textbook policy is to simply 'define' the relation R to be the set $\hat{R}$.

But again we should resist any outright identification here. For a start, some relations are 'too big' to have extensions according to standard set theories (consider e.g. the relation that holds between a singleton and its sole member). And in any case the arbitrariness built into the conventional rendition of ordered pairs prevents us from justifiably saying that a relation really *is* its extension as just defined. But if there is no determinate fact of the matter about which sets relations are, then relations aren't sets. What remains true is that, for many purposes, appropriate sets are workable proxies for relations or, as I put it, relations can be implemented as sets.

And again, we can dig deeper and press the point further: *relations just aren't the right logical type of thing to be sets.* Start with the following observation:

> it lies in the nature of a relation that it holds or does not hold of things. (Oliver and Smiley 2016, p. 156)

[7] A well-known Fregean metaphor might help. Functions of their nature are 'unsaturated', having a number of empty slots waiting to be filled appropriately when the function is applied to the right number of arguments. By contrast, a set is already 'saturated', it is self-standing, with no empty slots waiting to be filled. (A philosopher of Fregean bent might say that here we have the makings of a very general distinction between objects-as-individuals and functions. Be that as it may. For our purposes we only need the point that at least functions and sets are of different logical types.)

But a set doesn't, of its nature, hold of anything; and that applies to any set, including sets of ordered pairs. Hence an extension isn't the type of thing that a binary relation is.[8]

(d) In summary, functions aren't sets, strictly speaking. So we need to read that familiar 'definition' of functions as sets as really introducing a way of *implementing* functions in the world of sets. It tells us how we can render a claim of the form $f(x) = y$ by an equivalent set-theoretic claim of the form $\langle x, y \rangle_K \in \hat{f}$. Similarly, relations aren't sets, and the only true relation in the universe of sets is the membership relation. What we can do in a set-theoretic environment, though, is implement other relations which aren't 'too big' by using their extensions, and so render a claim of the binary form xRy by an equivalent set-theoretic claim of the form $\langle x, y \rangle_K \in \hat{R}$. Which is all absolutely fine, of course, so long as we are clear that that's what we are doing (and, as I said at the beginning of this section, this should really all be familiar).

Back then to structures like groups or well-ordered objects. These comprise some objects and functions, or some objects and relations on those objects, and so on. And while speaking strictly functions and relations aren't sets, we can *implement* groups or well-orderings in a universe of sets. And these set-theoretic implementations for structures will *of course* serve us perfectly well in familiar ways. So yes, three cheers for set theory, in its place!

Still, it *is* important for our future purposes to be clear about what this place is, about the role that the universe of sets can play here. It provides *one* generous arena where we can find implementations for all the structures we want for ordinary mathematical purposes.[9] However – and here is the important point again – this way of looking at things does leave the door open to the possibility that other kinds of universe might do the same job. They *might* even do the job better in some respects, e.g. by more faithfully respecting type-differences between objects as contrasted with functions, which are both to be contrasted with relations.

3.4 'The' category of groups?

(a) Let's return now to categories of groups.

The finite groups whose objects are natural numbers are countable, and so are the homomorphisms between these groups. Hence the category we defined as comprising them is equally a tamely countable structure-of-structures. But there are much larger, more inclusive, categories of groups. Indeed, we might now wonder: is there perhaps an all-inclusive category of *all* groups and *all* the homomorphisms between them?

[8]The same Fregean metaphor might help. Relations of their nature are 'unsaturated' and have a certain number of empty slots waiting to be filled appropriately when the relation is applied to the right number of things. By contrast, as we said, a set is already 'saturated', it is self-standing, with no empty slots waiting to be filled.

[9]The phrase 'generous arena' is borrowed from Penelope Maddy's very helpful discussion of the idea of set-theoretic foundations. See Maddy (2017).

"But can this really make sense? For a start, can we stably pin down *all* the groups? To take a silly example, if I cut out a new cardboard non-equilateral rectangle, then – lo and behold! – won't there spring into being a new Klein group, the group of its own rotation/reflection symmetries?" Fair questions, given that we were previously entirely permissive about where we can find groups: on our definition, we just need some new objects (in the broad sense) and a suitable operation on them and we get another group. But on the other hand, a new physically realized Klein group is surely neither here nor there as far as the mathematics of groups is concerned. As we said before, group theory will for most purposes ignore the differences between groups which are identical up to isomorphism; so, in particular, it can concentrate on more abstract exemplars.

OK: suppose then that we can assume that we are working in a capacious enough mathematical universe which can implement all the groups we will ever want (so we won't care about any additional isomorphic copies of these groups which are roaming outside in the wild). Then perhaps *that* universe can be the arena in which we can hope to locate a determinate category of 'all' groups and their homomorphisms. And where can we find a capacious enough mathematical universe? We have already trailed the now entirely predictable answer: Take a large enough universe of sets. Then we can hope to implement all the groups we want; and – now working with these set-theoretic surrogates – there will consequently be a category Grp living there which comprises all the groups we want and the homomorphisms between them.

(b) But this gives us something of a presentational quandary.

In developing category theory – our abstract theory for handling structures of structures – there is (as we have been stressing in previous sections) a rather good reason for *not* baldly assuming from the start that mathematical structures must all have their home in some universe of sets. We want to leave room for the possibility of other 'foundational' frameworks, perhaps ones that in fact turn out to be more category-theoretic in character. So, on the one hand, we would rather like to avoid taking on any specifically set-theoretic commitments at the outset.

On the other hand, we'll soon want to talk about large categories like Grp which supposedly comprise (implementations for) 'all' groups and the homomorphisms between. But as we've just said, if that's to make sense, it seems that we will need to think of the relevant groups as living in some definite but sufficiently capacious universe. And a universe of sets is quite likely to be the only familiar candidate we have available at this point. So the obvious default is to understand categories like Grp as living in a universe of sets.

Now, we could lean in the first direction and try to proceed at some remove from any direct engagement with sets. But this would in fact involve too big a departure from the standard mode of presentation of elementary category theory, which I judge would be quite inappropriate for an introductory account. There's nothing for it, then, but to start off by going in the second direction. So we now *will* default to assuming (and for quite a long time) that we are working in a universe of sets which is capacious enough to implement all the structures that we want. And then we can coherently talk about categories like Grp living in such a universe.

An aside: note that on the usual choice of background set theory to work in, Grp involves too many groups to form a set of them, and two many homomorphisms to form a set of *them* either. So we have already got an example to illustrate that we don't want in general to define a category as comprising a *set* of items and the maps between them.

(c) But do note that what Grp comprises will be relative to our favoured universe of sets, and that's arguably not a unique choice.

We are very familiar with the fact that our canonical set theory first-order ZFC has multiple models (some where the continuum hypothesis holds, some where it doesn't, and so on and so forth, in a proliferating multiverse of models). How do we fix on a model to work in? Even if we go second-order, that still doesn't determine a unique universe – there are different models of second-order ZFC with different heights. But then, on second thoughts, do we need such rich universes as models of full ZFC for ordinary mathematical purposes? It has been argued e.g. that the much weaker Mac Lane set theory is strong enough to model standard mathematics which is not directly connected with set theory or logic. And arguably more radically deviant set theories like NFU provide equally competent generous arenas for modelling the gadgetry of ordinary mathematics.

Now, we most certainly don't want to get bogged down into further investigations of such contentious issues here at the very beginning. What to do? In fact, it is quite common for introductions to category theory to talk loosely about large inclusive categories like Grp as if they are determinate families of mathematical structures, without fussing too much about the relativity to a choice of a background universe to implement those structures. We will do the same. Because, as we will see, this relativity will turn out to be harmless for many category-theoretic purposes. It will be a while, however, before we can show exactly why. And it will be a while longer before we can perhaps begin to untie our developing category theory from its initial anchorage in a particular world of sets. There is a lot of ground to cover first.

4 Categories defined

We have met only one sort of category so far, namely categories comprising some groups and enough homomorphisms between them. Here, 'enough' stands in for pretty minimal requirements – just that (i) compositions of homomorphisms in the category are also in the category, and (ii) the identity homomorphism for each group in the category is also present.

We now make our real start on category theory by generalizing to . . . .

4.1 The very idea of a category

We said that many paradigm examples of categories are – as in the case of categories of groups – families of structures with structure-preserving maps between them. But what can we say about such families at an abstract level?

One sufficiently general thought is this: if, within a family of structures including A, B, and C we have a structure-preserving map f from A to B, and another structure-preserving map g from B to C, then we should to be able to compose these maps. That is to say, the first map f followed by the second g should also count as a structure-preserving map $g \circ f$ from A to C.

What principles will govern such composition of maps? Associativity, surely. Using a natural diagrammatic notation, if we are given maps

$$A \xrightarrow{\quad f \quad} B \xrightarrow{\quad g \quad} C \xrightarrow{\quad h \quad} D$$

it really ought not matter how we carve up the journey from A to D. It ought not matter whether we apply the map f followed by the composite g-followed-by-h, or alternatively apply the composite map f-followed-by-g and then afterwards apply h.

What else can we say at the same level of stratospheric generality about families of structures and structure-preserving maps? Very little. Except that there presumably will always in principle be the limiting case of a 'do nothing' identity map, which applied to any structure A leaves it untouched.

That apparently doesn't give us a great deal to work with. But in fact it is already enough to shape our following definition of categories. However, it is useful to abstract even further from the idea of structures with structure-preserving maps between them, and – using more neutral terminology – we'll speak very generally of *objects* and of *arrows* between them. Then we say:

Definition 13. A category $\mathscr{C}$ comprises two kinds of things:

(1) $\mathscr{C}$-*objects* (which we will typically notate by $A, B, C, \ldots$)

(2) $\mathscr{C}$-*arrows* (which we typically notate by $f, g, h, \ldots$).

These $\mathscr{C}$-objects and $\mathscr{C}$-arrows are governed by the following axioms:

Sources and targets For each arrow f, there are unique associated objects $src(f)$ and $tar(f)$, respectively the *source* and *target* of f, not necessarily distinct. We write $f\colon A \to B$ or $A \xrightarrow{f} B$ to notate that f is an arrow with $src(f) = A$ and $tar(f) = B$.

Composition For any two arrows $f\colon A \to B$, $g\colon B \to C$, where $src(g) = tar(f)$, there exists an arrow $g \circ f\colon A \to C$, 'g following f', which we call the *composite* of f with g.

Identity arrows Given any object A, there is an arrow $1_A\colon A \to A$ called the *identity arrow* on A.

We also require the arrows to satisfy the following further axioms:

Associativity of composition. For any $f\colon A \to B$, $g\colon B \to C$, $h\colon C \to D$, we have $h \circ (g \circ f) = (h \circ g) \circ f$.

Identity arrows behave as identities. For any $f\colon A \to B$ we have $f \circ 1_A = f = 1_B \circ f$. △

Evidently, a category of groups as originally defined will be a category in this sense. And given what we have already said, the objects which are mathematical structures of a particular kind taken together with enough arrows which are structure-preserving maps between them should also satisfy those axioms, and hence should count as forming a category.

Here are six quick remarks on terminology and notation:

(i) The objects and arrows of a category are very often called the category's *data*. That's a helpfully neutral term if you don't read too much into it, and I will occasionally adopt this common way of speaking.

(ii) The label 'objects' for the first kind of data is quite standard. But note that, just as with the 'objects' of groups (see §2.2), the 'objects' in categories needn't be objects-as-individuals in a type-theoretic sense which contrasts objects with entities like relations or functions. There are perfectly good categories whose objects are actually relations, and other categories where they are functions. And in a category of groups, an object is of course a group (some group-objects, whatever they are, equipped with an operation defined over them).

(iii) Borrowing familiar functional notation $f\colon A \to B$ for arrows in categories is entirely natural given that arrows in many categories *are* (structure-preserving) functions: in fact, that is the motivating case. But again, as we'll soon see, not all arrows in categories are functions. Which means that not all arrows are morphisms either, in the usual sense of that term. Which is why I rather prefer the colourless 'arrow' to the equally common term 'morphism' for the second sort of data in a category. (Not that that will stop me talking of morphisms or maps when context makes it natural!)

(iv) In keeping with the functional paradigm, the source and target of an arrow are frequently called, respectively, the 'domain' and 'codomain' of the arrow (for usually, when arrows are functions, that's what the source and target are). But that usage has the potential to mislead when arrows aren't functions (or aren't functions 'in the right direction', cf. §6.1), which is why I prefer our common alternative terminology.

(v) Note again the order in which we write the components of a composite arrow, because some from computer science writing about categories do things the other way about. Our notational convention is again suggested by the functional paradigm. When $f\colon A \to B$, $g\colon B \to C$ are both functions in the ordinary sense, then $(g \circ f)(x) = g(f(x))$. Occasionally, to reduce clutter, we may allow ourselves to write simply 'gf' rather than '$g \circ f$'.

(vi) Initially, we will explicitly indicate which object an identity arrow has as both source and target, as in '1_A'. Again to reduce clutter, we will later allow ourselves simply write '1' when context makes it clear which identity arrow is in question.

4.2 Identity arrows

The definition of a category implies our first mini-result:

Theorem 8. *Identity arrows on a given object are unique; and the identity arrows on distinct objects are distinct.*[1]

Proof. For the first part, suppose A has identity arrows 1_A and $1'_A$. Then applying the identity axioms for each, we immediately have $1_A = 1_A \circ 1'_A = 1'_A$.

For the second part, we simply note that $A \neq B$ entails $src(1_A) \neq src(1_B)$ which entails $1_A \neq 1_B$. □

So there's a one-one correlation between objects in a category and identity arrows; and we can in fact pick out such identity arrows by the special way they interact with all the other arrows. Hence we could in principle give a variant definition of categories framed entirely in terms of arrows.[2] But I am not unusual in finding this bit of trickery rather unhelpful. As we will see, a central theme of category theory is indeed the idea that we should probe the objects in a category by considering the arrows between them; but that's no reason to write the objects out of the story altogether.

[1]As this illustrates, the most trivial of lemmas, as well as run-of-the-mill propositions, interesting corollaries, and the weightiest results, will continue to be labelled 'theorems' without distinction. I did initially try to mark a distinction between, as-it-were, capital-'T' theorems and the rest, but that didn't really work out well!

[2]For an account of how to do this, see Adámek et al. (2009, pp. 41–43).

4.3 Monoids and pre-orderings

Let's continue by looking at two simple but instructive types of categories, one algebraic, one order-theoretic.

(a) We have already met the algebraic case of various categories of groups, and the particular inclusive category Grp. But it is worth thinking now about the case where the algebraic structure is cut to the bone.

Recall, then, that a *monoid* is, so to speak, a group minus the requirement for inverses. And a monoid homomorphism is a function which respects monoid structure. So, more carefully, we have:

Definition 14. The objects M (including the distinguished object e), equipped with a binary operation $*$ (where for any x, y among M, $x * y$ is also among M), form a *monoid* $\mathcal{M} = (M, *, e)$ iff

(i) $*$ is associative, i.e. for any x, y, z among M, $(x * y) * z = x * (y * z)$;
(ii) e acts as a monoid unit or identity, i.e. for any x among M, $x*e = x = e*x$.

Further, a *monoid homomorphism* from the monoid $(M, *, e)$ as source to the monoid $(N, \star, d)$ as target is a function $f \colon M \to N$ such that:

(i) for every x, y among M, $f(x * y) = fx \star fy$,
(ii) $f(e) = d$. △

Just as in the case of groups, thought of simply in its role of mapping objects to objects, the function $f \colon M \to N$ is said to be the underlying function of the homomorphism. When thought of in its role as a structure-preserving homomorphism we can use the notation $f \colon (M, *, e) \to (N, \star, d)$, or $f \colon \mathcal{M} \to \mathcal{N}$.

It is evident that, again just as in the group case, monoid homomorphisms $f \colon \mathcal{M} \to \mathcal{N}$ and $g \colon \mathcal{N} \to \mathcal{O}$ compose to give a homomorphism $g \circ f \colon \mathcal{M} \to \mathcal{O}$. Composition of homomorphisms is associative (because composition of the underlying functions is). And the identity function on objects M is a homomorphism $f \colon \mathcal{M} \to \mathcal{M}$ which acts as an identity with respect to composition.

Evidently, then, some monoids together with enough homomorphisms will form a category – where by 'enough' we mean as before that (i) compositions of homomorphisms in the category are also in the category, and (ii) the identity homomorphism for each monoid in the category is also present (I won't keep repeating that gloss on 'enough'!).

But now assuming that we are working in some suitably inclusive universe, we can also sensibly say:

(C1) Mon is the category whose objects are all monoids and whose arrows are all the monoid homomorphisms.

We mean here of course that the objects are all the monoids in the relevant universe, and the arrows are all the homomorphisms between *them*.

Fine print: as indicated in the last chapter, we are by default working in a universe of sets – so if you are being really pernickety you could say that what we have in Mon will strictly speaking be implementations of monoids and their homomorphisms. But we won't worry about that. For note that these implementations of monoids and their homomorphisms can count perfectly well as objects and arrows (in particular, note that arrows don't have to be kosher functions). So, Mon is a genuine category of (proxies for) monoids, and not a proxy category!

(b) Next, an example involving ordered objects; and again we'll cut structure to the bone by considering the simplest case, pre-orderings.

Definition 15. Some objects M equipped with a relation $\preccurlyeq$, for short $(M, \preccurlyeq)$, are pre-ordered iff for all a, b, c from among M,

(i) if $a \preccurlyeq b$ and $b \preccurlyeq c$, then $a \preccurlyeq c$,
(ii) $a \preccurlyeq a$.

A monotone map $f\colon (M, \preccurlyeq) \to (N, \sqsubseteq)$ is then defined to be a function $f\colon M \to N$ between the underlying objects which respects order, i.e. such that for any a, b among M, if $a \preccurlyeq b$, then $fa \sqsubseteq fb$. △

Call the likes of $(M, \preccurlyeq)$ a pre-ordering. It is obvious that monotone maps between pre-orderings compose to give monotone maps, and the identity map on some objects gives rise to an identity monotone map on them. So some pre-orderings equipped with enough monotone maps will form a category.

And assuming, again that we are working in some suitable universe, we can also sensibly say:

(C2) Ord is the category whose objects are all pre-orderings and whose arrows are the monotone maps between such collections.

We mean of course that the objects here are all the pre-orderings in the relevant universe, and the arrows are all the monotone maps between *them*.

Fine print: In fact, since by default we are working in a universe of sets, actually what we have in Ord are suitable set-implementations for pre-orderings and their homomorphisms. But we won't worry about that, for the same reason as before. If we want to be precise again, Ord is genuine category of (proxies for) pre-orderings, not a proxy category.

But such pernicketiness gets very tiresome. So, for now on, let's mostly take this sort of fine print as read.

4.4 Some rather sparse categories

(a) So far, so very unsurprising.

However, note that monoids can get into the story in a second way. As we've seen, monoids as objects taken together with enough monoid homomorphisms as arrows can form a category. However, any single monoid taken just by itself can also be thought of giving rise to a category. Here's how:

(C3) Take any monoid $(M, *, e)$. Then define a corresponding category $\mathscr{M}$ whose data is as follows:

(i) $\mathscr{M}$'s sole object is some arbitrary entity – choose whatever you like, it *doesn't* have to be one of the objects M, and dub it '$\bullet$';

(ii) An $\mathscr{M}$-arrow $a\colon \bullet \to \bullet$ is then just any of the monoid's objects a (we put $src(a) = tar(a) = \bullet$). Composition of arrows $a \circ b$ is defined to be the monoid product $a * b$, and the identity arrow $1_\bullet$ is defined to be the monoid identity e.

It is trivial that the category axioms are satisfied.

Conversely, any one-object $\mathscr{M}$ category gives rises to an associated monoid built from its arrows, with multiplication in the monoid being composition of arrows. So we can think of many-object categories as, in a sense, generalizing from the case of the one-object categories which are tantamount to monoids.

Note in this case, since the 'object' in the category $\mathscr{M}$ can be anything you like, it needn't be an object in any ordinary sense (let alone be a structure). And unless the objects of the original monoid M happen to be functions, the arrows of the associated category $\mathscr{M}$ will also not be functions or morphisms or maps in any ordinary sense. So this sort of single-monoid-as-a-category won't usually be anything like a 'structure of structures'.

(b) Similarly, while we can put pre-orderings and their monotone maps together to form a category, we can also think of any one pre-ordering – some objects equipped with a pre-order – as forming a category just by itself. Here's how:

(C4) Take any pre-ordered objects $(P, \preccurlyeq)$. Then define a corresponding category $\mathscr{P}$ whose data is as follows:

(i) $\mathscr{P}$'s objects are the objects P again;

(ii) there is a (single) $\mathscr{P}$-arrow from A to B just in case $A \preccurlyeq B$ – this arrow might as well be identified as an ordered pair $\langle A, B\rangle$, and then we can define composition by putting $\langle B, C\rangle \circ \langle A, B\rangle = \langle A, C\rangle$. Take the identity arrow 1_A to be $\langle A, A\rangle$.

It is trivial that, so defined, the arrows for $\mathscr{P}$ satisfy the identity and associatively axioms, so we do have another category here – and again, this isn't one comprising structures and structure-preserving maps.

Conversely, if you think about it, any category with objects O and where there is at most one arrow between objects can be regarded as a pre-ordering $(O, \preccurlyeq)$, where for A, B among O, $A \preccurlyeq B$ just in case there is an arrow from A to B in the category. It is therefore natural to call a category with at most one arrow between objects a *pre-order category*. And so we can think of the unrestricted notion of a category as a generalization of the case of pre-order categories.

(c) Monoids-as-categories and pre-ordered-objects-as-categories can give us very small categories with few objects and/or arrows. And here are some more sparse categories.

(C5) For any collection of objects M, there is a *discrete category* on those objects. This is the category whose objects are just the members of M, and which has as few arrows as possible, i.e. just the identity arrow for each object in M.

(C6) For convenience, we can allow the empty category, with zero objects and zero arrows. Otherwise, the smallest discrete category is $\mathbb{1}$ which has exactly one object and one arrow (the identity arrow on that object). Let's picture it in all its glory!

$$\begin{tikzcd} \bullet \arrow[loop above] \end{tikzcd}$$

(C7) And having mentioned the one-object category $\mathbb{1}$, here's another very small category, this time with two objects, the necessary identity arrows, and one further arrow between them. We can picture it like this:

$$\begin{tikzcd} \bullet \arrow[loop left] \arrow[r] & \star \arrow[loop right] \end{tikzcd}$$

Call this category $\mathbb{2}$. We could think of as arising from the von Neumann ordinal 2, i.e. the set $\{\varnothing, \{\varnothing\}\}$; take the ordinal's members as objects of the category, and let there be an arrow between objects when the source is a subset of the target. Other von Neumann ordinals, finite and infinite, similarly give rise to other categories.

But hold on! Should we in fact talk about *the* category $\mathbb{1}$ (or *the* category $\mathbb{2}$, etc.)? Won't different choices of object make for different one-object categories, etc.? Well, yes and no! We can have, in our mathematical universe, different cases of single objects equipped with an identity arrow – *but they will be indiscernible from within category theory.* So as far as category theory is concerned, they are all 'essentially the same' – in the same spirit as e.g. different Klein four-groups are 'essentially the same' in group theory. We will return to this point.

4.5 More categories

Let's continue our list of sorts of categories, first generalizing from our basic algebraic and order-theoretic examples in the last section, and then adding some geometric and other categories. And for brevity's sake, in most cases, we'll jump straight to the maximal version living in our default universe (i.e. the version which stands to other instances of the same general sort as e.g. Grp does to each category of groups).

Categories of monoids and categories of groups are just the first of a family of similar cases, where the objects are algebraic structures – comprising objects equipped with some functions and/or with certain distinguished objects picked out – and the arrows are the homomorphisms preserving the relevant amount of structure. Adding more structure to our objects, then, we can get :

(C8) Ab is the category whose objects are abelian groups, and whose objects are group homomorphisms again.

(C9) Rng is, the category of rings, whose objects are predictably enough all rings and whose objects are ring homomorphisms.

(C10) And Bool is the category of Boolean algebras and structure-preserving maps between them.

And so on it goes!

We similarly have further categories of ordered objects. Enrich the notion of a pre-order, take as structures objects-equipped-with-the-richer-order, take as arrows enough order-preserving functions, and we get another category. For example (taking maximal cases as before),

(C11) Pos is the category whose objects are objects-equipped-with-a-partial-order (where that's a pre-order which is anti-symmetric), and the arrows are order-preserving maps again.

(C12) Tot is the category whose objects are objects-equipped-with-a-total-order (where that's a partial order where any two objects stand in the order relation, one way round or the other). The arrows are as you would now expect.

And so on it goes!

Now for another paradigm type of case, namely geometric categories (even more central to the original development of category theory than the cases of algebraic categories or order categories).

(C13) Top is the category with

(i) objects: all the topological spaces;

(ii) arrows: the continuous maps between spaces.

(C14) Met is also a category: this has

(i) objects: metric spaces, which we can take to be some points S equipped with a real metric d;

(ii) arrows: the non-expansive maps, where – in an obvious notation – $f\colon (S,d) \to (T,e)$ is non-expansive iff $d(x,y) \geqslant e(f(x), f(y))$.

(C15) Vect_k is a category with

(i) objects: vector spaces over the field k (each such space comprising vectors equipped with vector addition and multiplication by scalars in the field k);

(ii) arrows: linear maps between the spaces.

And finally in this section, let's have a logical example.

(C16) Suppose $\mathcal{L}$ is a first-order formal language (the details don't matter). Then there is a category of propositions $\mathsf{Prop}_\mathcal{L}$ with

(i) objects: propositions, closed sentences $X, Y, \ldots$ of the formal language;

(ii) arrows: there is a (unique) arrow from X to Y iff $X \vDash Y$, i.e. X semantically entails Y.

The reflexivity and transitivity of semantic entailment means we get the identity and composition laws which ensure that this is a category.

4.6 The category of sets

(a) Categories like Mon and Ord whose objects are sets-equipped-with-some-structure and whose arrows are structure-preserving-functions are conventionally called *concrete* categories. As we have seen, lots of categories are not concrete in this sense – for example, neither a monoid-as-category nor a pre-ordering-as-category will count. We'll revisit the distinction between 'concrete' and 'abstract' categories in due course, and give a sharper technical account once we have the idea of a functor in play. But I thought I should mention the standard distinction straight away.

(b) Now, the monoids in Mon are sets equipped with not-very-much structure. Likewise for pre-orderings in Ord. Going in one direction, we get concrete categories whose objects are sets equipped with a richer structure and whose arrows are functions constrained to preserve this richer structure. Going in the other direction, we get categories whose objects are simply sets (equipped with no additional structure at all) and whose arrows are functions between these sets (any old functions so long as they are closed under composition, and we include the relevant identity functions: there is no requirement that functions preserve structure because there is no structure to preserve). The maximal case is:

(C17) Set is the category with

(i) objects: all sets.

(ii) arrows: for any sets X, Y, every (total) set-function $f\colon X \to Y$ is an arrow.

There's an identity function on any set. And set-functions $f\colon A \to B$, $g\colon B \to C$ (where the source of g is the target of f) always compose. And so the axioms for being a category are evidently satisfied.

Some initial remarks:

(i) Note that the arrows in Set, like any arrows, must come with determinate targets/codomains. But the standard way of treating functions set-theoretically is simply to define a function f as its *graph* $\hat{f}$, i.e. with the set of pairs $\langle x, y\rangle$ such that $f(x) = y$. This definition is lop-sided in that it fixes the function's source/domain, the set of first elements in the pairs, but it doesn't determine the function's target. (For a quite trivial example, consider the Set-arrows $z\colon \mathbb{N} \to \mathbb{N}$ and $z'\colon \mathbb{N} \to \{0\}$ where both functions send every number to zero. These have the same graphs, but the functions have different targets and correspondingly different properties – e.g. the second is surjective, the first isn't.)

Perhaps set theorists themselves ought really to define a set-function $f: A \to B$ as a triple $\langle A, \hat{f}, B \rangle$. But be that as it may, that's how category theorists ought officially to regard arrows $f: A \to B$ in Set, and in other concrete categories too.

(ii) We should perhaps remind ourselves why there *is* an identity arrow for $\varnothing$ in Set. Vacuously, for *any* target set Y, there is exactly one set-function $f: \varnothing \to Y$, i.e. the one whose graph is the empty set. Hence in particular there is a function $1_\varnothing: \varnothing \to \varnothing$.

(iii) Note that in Set, the empty set is in fact the *only* set such that there is exactly one arrow from it to any other set. This gives us a simple example of how we can characterize a significant object in a category not by its internal constitution, so to speak, but by what arrows it has to and from other objects.

For another example, note that we can define singletons in Set by relying on the observation that there is exactly one arrow from any set *to* a singleton (why?).

(iv) So now choose a singleton $\{\bullet\}$, it won't matter which one (as elsewhere, treat the bullet as a wildcard). Call your chosen singleton '1'. And consider the possible arrows (i.e. set-functions) from 1 to A.[3]

We can represent the arrow from 1 to A which sends the element of the singleton 1 to $x \in A$ as $\vec{x}: 1 \to A$ (the over-arrow here is simply a helpful reminder that we are notating an arrow). Then there is evidently a one-one correspondence between these arrows $\vec{x}$ and the elements $x \in A$. So talk of such arrows $\vec{x}$ is available as a category-speak surrogate for talking about elements x of A.

More on this sort of thing in due course: but it gives us another glimpse ahead of how we might trade in talk of sets-and-their-elements for categorial talk of sets-and-arrows-between-them.

(c) So far, so straightforward. But let's just pause to note again that the make-up of the category Set of course is relative to our background universe. We haven't determinately fixed that. But we'll live with that for the moment. You can just interpret our talk of sets and the category Set in your preferred way assuming that this isn't too idiosyncratic!

And note again that familiar size considerations kick in. The category of sets has all sets (in your favoured universe) as its objects. Unless you are an NF-iste,[4] however, there is no set of all sets – such a collection is, in a familiar way, 'too

[3] We are overloading notation – here '1' is a special object, while in other contexts '1' is a special arrow, an identity arrow. You'll need to get used to this sort of thing, where we rely on context to disambiguate shared notations for objects and arrows.

[4] That is to say, a devotee of Quine's deviant set theory NF which does have a universal set, and avoids paradox by constraining our comprehension principle.

big' to be a set. Hence on the standard view, the category of sets is itself too big to be a set or to be modelled as a set. Not wanting to rule out the standard view of sets, that's another reason why our initial definition of a category did not say e.g. that a category always comprises a *set* of objects, but used a plural characterization.

4.7 Yet more examples

(a) Let's finish our initial list of examples of categories. And now we can go more briskly:

(C19) There is a category FinSet whose objects are the hereditarily finite sets (i.e. sets with at most finitely many members, these members in turn having at most finitely many members, which in turn ...), and whose arrows are the set-functions between such objects.

(C20) Pfn is the category of sets and *partial* functions. Here, the objects are all the sets again, but an arrow $f\colon A \to B$ is a function not necessarily everywhere defined on A (one way to think of such an arrow is as a total function $f\colon A' \to B$ where $A' \subseteq A$). Given arrows-qua-partial-functions $f\colon A \to B$, $g\colon B \to C$, their composition $g \circ f\colon A \to C$ is defined in the obvious way, though you need to check that this succeeds in making composition associative.

(C21) $\mathsf{Set}_\star$ is the category (of 'pointed sets') with

(i) objects: all the non-empty sets, with each set A having a distinguished member $\star_A$.

(ii) arrows: all the total functions $f\colon A \to B$ which map $\star_A$ to $\star_B$, for any objects A, B.

As we'll show later, Pfn and $\mathsf{Set}_\star$ are in a good sense equivalent categories (it is worth pausing to think why we should expect that).

(C22) The category Rel again has naked sets as objects, but this time an arrow $A \to B$ in Rel is (not a function but) any relation R between A and B. We can take this officially to be a triple $(A, \hat{R}, B)$, where $\hat{R} \subseteq A \times B$ is R's extension, the set of pairs $\langle a, b\rangle$ such that aRb.

The identity arrow on A is then the diagonal relation with the graph $\{\langle a, a\rangle \mid a \in A\}$. And $S \circ R\colon A \to C$, the composition of arrows $R\colon A \to B$ and $S\colon B \to C$, is defined by requiring $a\, S \circ R\, c$ if and only if $\exists b(aRb \wedge bSc)$. It is easily checked that composition is associative.

So here we have yet another example where the arrows in a category are *not* functions.

(b) And that will do for the moment as an introductory list. There certainly is no shortage of categories, then!

Indeed, by this stage, you might very reasonably be wondering whether it isn't just *too* easy to be a category. If such very different sorts of structures as e.g. a particular small monoid on the one hand and the whole universe of topological spaces on the other hand equally count as categories, how much mileage can there be theorizing in general about categories and their interrelations?

Well, that's exactly what we hope to find out over the coming chapters.

5 Diagrams, informally

We can diagrammatically represent objects related by arrows in a very natural way – we've already seen some trivial mini-examples. And in particular, we can represent facts about the equality of arrows using so-called commutative diagrams.

We'll soon be using such diagrams a great deal: so we'd better make some headline points about them straight away. And these are important enough to deserve a brief chapter to themselves.

5.1 Diagrams, in two senses

Talk of diagrams is in fact used by category theorists in three related ways. Later, in §22.1, we will give a sharp characterization of a more technical notion of a diagram. But for the moment, we can be informal and work with two looser but more immediately intuitive notions:

Definition 16. A *representational diagram* is a directed graph with nodes representing objects from a given category $\mathscr{C}$, and edges (drawn as arrows) between nodes representing arrows of $\mathscr{C}$. Nodes and edges are usually appropriately labelled.

Two different nodes in a diagram can be joined by zero, one, or more drawn arrows. A drawn arrow labelled 'f' from the node labeled 'A' to the node labeled 'B' of course represents the arrow $f\colon A \to B$ of $\mathscr{C}$.

There can also be arrows looping round from a node to itself, representing the identity arrow on an object or representing some other arrow whose source and target is the same. △

Definition 17. A *diagram in a category* $\mathscr{C}$ is what is represented by a representational diagram – in other words, it will be some $\mathscr{C}$-objects and some $\mathscr{C}$-arrows between them. △

I'm being mildly fussy in distinguishing the two ideas here, the diagram-as-picture, and the diagram-as-what-is-pictured. But having made the distinction, we will rarely need to bother about it, and can let context determine a sensible reading of informal claims about diagrams.

A very important point, though, is that diagrams (in either sense) needn't be *full*. That is to say, a diagram-as-a-picture need only show *some* of the objects and arrows in a category; and a diagram-as-what-is-pictured need only be a portion of the whole category in question.

5.2 Commutative diagrams

(a) Within a representational diagram, we may be able to follow a directed path through more than two nodes, walking along the connecting drawn arrows (from an original source to an ultimate target). So a path in a representational diagram from node A to node E (for example) might look like this

$$A \xrightarrow{f} B \xrightarrow{g} C \xrightarrow{h} D \xrightarrow{j} E$$

The axiom about composition tells us that there is also an arrow $j \circ (h \circ (g \circ f))$ from A to E which you get by composing along the path. And because of the associativity of composition we needn't actually worry about bracketing here, and can simply write $j \circ h \circ g \circ f$. (From now on, then, we freely insert or omit brackets in writing composites, doing whatever promotes local clarity.)

(b) Then we say:

Definition 18. A diagram *commutes* iff, for any two directed paths along edges in the diagram from a node X to a node Y (where at least one path includes more than one edge), the arrow you get by composing along the first path is equal to the arrow you get composing along the second path. △

Hence, for example, the associativity axiom $h \circ (g \circ f) = (h \circ g) \circ f$ can be presented by saying that the following diagram always commutes:

$$\begin{tikzcd}
A \arrow[r, "f"] \arrow[dr, "g \circ f"'] & B \arrow[d, "g"] \arrow[dr, "h \circ g"] & \\
& C \arrow[r, "h"] & D
\end{tikzcd}$$

Each of the two triangles in this diagram commutes just by the definition of composition. And then by associativity we can paste the triangles together to get a larger commutative diagram.

Here's another example. If the two squares on the left commute, then by associativity we can paste them together along the common arrow to get the larger commutative diagram:

$$\begin{tikzcd}
A \arrow[r, "f"] \arrow[d, "j"] & B \arrow[d, "c"] \\
D \arrow[r, "k"'] & E
\end{tikzcd}
\qquad
\begin{tikzcd}
B \arrow[r, "g"] \arrow[d, "c"] & C \arrow[d, "h"] \\
E \arrow[r, "l"'] & F
\end{tikzcd}
\qquad
\begin{tikzcd}
A \arrow[r, "f"] \arrow[d, "j"] & B \arrow[r, "g"] \arrow[d, "c"] & C \arrow[d, "h"] \\
D \arrow[r, "k"'] & E \arrow[r, "l"'] & F
\end{tikzcd}$$

To check this, note that

$$h \circ (g \circ f) = (h \circ g) \circ f = (l \circ c) \circ f = l \circ (c \circ f) = l \circ (k \circ j)$$

with the equations holding alternately by associativity and by the assumed commutativity of the squares.

We will meet many more examples of commutative diagrams in the coming chapters, so I won't give more illustrations just now. For the moment, let's just emphasize three preliminary points, the first two of them very obvious:

1. To say a given diagram commutes is just a helpfully vivid way of saying that certain identities hold between composites – it is the identities that matter!

2. So merely drawing a diagram with different routes from e.g. A to D in the relevant category doesn't always mean that we have a *commutative* diagram: the identity of the composites along the paths in each case has to be established!

3. We said that, for a commutative diagram, arrows along any two paths from X to Y must be equal, so long as at least one path includes more than one edge. What was the point of the added condition? Well, for convenience, we will want a commutative diagram to be able to include two 'parallel' arrows $f\colon X \to Y$ and $g\colon X \to Y$ which have the same source and target, without f and g having to be equal. For example, we will later be encountering diagrams like this:
$$\begin{tikzcd} E \arrow[r, "e"] & X \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & Y \end{tikzcd}$$
And it is convenient to count such a diagram as commuting so long as $f \circ e = g \circ e$, without requiring the parallel arrows f and g to be equal.

5.3 A reality check

We've met some very small categories, such as the one pictured by the following commutative diagram:

$$\begin{tikzcd} \bullet \arrow[loop left] \arrow[r] & \star \arrow[loop right] \end{tikzcd}$$

Now consider: is there a small category that can be pictured by the the following commutative diagram?

$$\begin{tikzcd} \bullet \arrow[loop left, "1_\bullet"] \arrow[r, "g"] & \star \arrow[loop right, "1_\star"] \arrow[l, bend right, "f"'] \arrow[l, bend left, "h"] \end{tikzcd}$$

Reading the diagram, this tells us that $g \circ f = 1_\star$, because the composites along the two paths from $\star$ to itself must be equal if the diagram is to count as commuting. Similarly $h \circ g = 1_\bullet$. Hence

$$(h \circ g) \circ f = 1_\bullet \circ f = f \neq h = h \circ 1_\star = h \circ (g \circ f).$$

Hence composition isn't associative here, and therefore we aren't dealing with a category!

6 Categories beget categories

We already know that categories are very plentiful! And in this chapter we are going to introduce yet more, by describing a number of general constructions which give us new categories from old. We'll meet further construction methods later, but these first ones will be more than enough to be going on with.

6.1 Duality

(a) An easy but particularly important way of getting one category from another is to simply *reverse all the arrows.*

More carefully, let's say:

Definition 19. Given a category $\mathscr{C}$, then its *opposite* or *dual* $\mathscr{C}^{op}$ is the category such that

(1) The objects of $\mathscr{C}^{op}$ are just the objects of $\mathscr{C}$ again.
(2) If f is an arrow of $\mathscr{C}$ with source A and target B, then f is also an arrow of $\mathscr{C}^{op}$ but now it is assigned source B and target A.
(3) Identity arrows remain the same, i.e. $1_A^{op} = 1_A$.
(4) Composition-in-$\mathscr{C}^{op}$ is defined in terms of composition-in-$\mathscr{C}$ by putting $f \circ^{op} g = g \circ f$ △

Here $\circ^{op}$ is, of course, composition in the new opposite category, and condition (4) should be made transparent by the following pair of diagrams:

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[dr, "g \circ f"'] & B \arrow[d, "g"] \\ & C \end{tikzcd} \quad \Rightarrow \quad \begin{tikzcd} A & B \arrow[l, "f"'] \\ & C \arrow[u, "g"'] \arrow[ul, "f \circ^{op} g"] \end{tikzcd}$$

$\mathscr{C}$ $\qquad\qquad$ $\mathscr{C}^{op}$

It is then trivial to check that our definition is in good order and that $\mathscr{C}^{op}$ really is a category.

It is also trivial to check that $(\mathscr{C}^{op})^{op}$ is $\mathscr{C}$: this means *every* category is also the opposite of some other category.

(b) A bit of care is required. Take for example Set^{op}. By definition, an arrow $f\colon A \to B$ in Set^{op} is the same thing as an arrow $f\colon B \to A$ in Set, which is of course a set-function from B to A.

But this means that $f\colon A \to B$ in Set^{op} typically *won't* be a function from *its* source to its target – it's an arrow in that direction but usually only a function in the opposite one![1]

Set^{op} is, as we will see, a very different sort of category to Set. And in general, taking the opposite category gives us something essentially new. But not always. Consider the category Rel^{op}, for example, and just remember that every relation comes as one of a pair with its converse or opposite.

(c) Let's get a bit formal for a moment. Take $\mathcal{L}$ to be the elementary pure language of categories – meaning a two-sorted first-order language with identity, with one sort of variable for objects, $A, B, C \ldots$, and another sort for arrows $f, g, h, \ldots$. It has built-in function-expressions '*src*' and '*tar*' (denoting two operations taking arrows to objects), a built-in relation '... is the identity arrow for ...', and a two place function-expression '$\ldots \circ \ldots$' which expresses the function which takes two composable arrows to another arrow.

Definition 20. Suppose φ is a wff of $\mathcal{L}$. Then its *dual* φ^{op} is the wff you get by (i) swapping '*src*' and '*tar*' and (ii) reversing the order of composition, so '$f \circ g$' becomes '$g \circ f$', etc. △

Now, the duals of the axioms for a category are also instances of axioms, as is quickly checked – which is why $\mathscr{C}^{op}$ is a category. And *that* observation gives us the following *duality principle*:

Theorem 9. *Suppose φ is an $\mathcal{L}$-sentence (a wff with no free variables) – so φ is a general claim about objects/arrows in an arbitrary category. Then if the axioms of category theory entail φ, they also entail the dual claim φ^{op}.*

Since we are dealing with a first-order theory, syntactic and semantic entailment come to the same, and we can prove the theorem either way:

Syntactic proof. If there's a first-order proof of φ from the axioms of category theory, then by taking the duals of every wff in the proof we'll get a proof of φ^{op} from the duals of the axioms. But those duals of axioms are themselves axioms, so we have a proof of φ^{op} from the axioms of category theory. □

Semantic proof. If φ always holds, i.e. holds in every category $\mathscr{C}$, then φ^{op} will hold in every $\mathscr{C}^{op}$ – but the $\mathscr{C}^{op}$s comprise every category again, since every category is the opposite of some category, so φ^{op} also holds in every category. □

The duality principle is very simple but also a hugely labour-saving result; we'll see this time and time again, starting in the next chapter.

[1] So this is one of those cases where talking of 'domains' and 'codomains' instead of 'sources' and 'targets' could initially encourage confusion, since the 'domain' of an arrow in Set^{op} is its codomain as a function. Hence my preference for the source/target terminology.

6.2 Subcategories, product and quotient categories

(a) Three familiar ways of getting new widgets from old are by taking sub-widgets, forming products of widgets, and quotienting by an equivalence relation. We met these sorts of constructions on groups in §2.3. And we can do the same constructions with categories, as we will now see.

The simplest way of getting a new category is by slimming down an old one:

Definition 21. Given a category $\mathscr{C}$, if $\mathscr{S}$ consists of the data

(i) objects: some or all of the $\mathscr{C}$-objects,
(ii) arrows: some or all of the $\mathscr{C}$-arrows,

subject to the conditions

(iii) for each $\mathscr{S}$-object A, the $\mathscr{C}$-arrow 1_A is also an $\mathscr{S}$-arrow,
(iv) for any $\mathscr{S}$-arrows $f\colon A \to B$, $g\colon B \to C$, the $\mathscr{C}$-arrow $g \circ f\colon A \to C$ is also an $\mathscr{S}$-arrow,

then, with composition of arrows in $\mathscr{S}$ defined as in the original category $\mathscr{C}$, $\mathscr{S}$ is a *subcategory* of $\mathscr{C}$. △

Plainly, the conditions in the definition – containing identity arrows for the remaining objects and being closed under composition – are there to ensure that the slimmed-down $\mathscr{S}$ is still a category.

Some cases where we prune an existing category will leave us with constructions of no particular interest. Other cases can be more significant, and we have already met some examples:

(1) Lots of categories of groups will be subcategories of Grp
(2) Set is a subcategory of Pfn,
(3) FinSet is a subcategory of Set,
(4) Ab is a subcategory of Grp,
(5) The discrete category on the objects of $\mathscr{C}$ is a subcategory of $\mathscr{C}$ for any category.

So, we can shed objects and/or arrows in moving from a category to a subcategory. In examples (2) and (5) we keep all the objects but shed some or all of the non-identity arrows. But cases (3) and (4) are ones where we drop some objects while keeping all the existing arrows between those objects retained in the subcategory, and there is a standard label for such cases:

Definition 22. If $\mathscr{S}$ is a subcategory of $\mathscr{C}$ where, for all $\mathscr{S}$-objects A and B, the $\mathscr{S}$-arrows from A to B are *all* the $\mathscr{C}$-arrows from A to B, then $\mathscr{S}$ is said to be a *full subcategory* of $\mathscr{C}$. △

We'll meet more cases of full subcategories later.

(b) It is also easy to form products of categories (so we'll give the definition now, though we won't really make use of products for a good while):

Definition 23. If $\mathscr{C}$ and $\mathscr{D}$ are categories, then a product category $\mathscr{C} \times \mathscr{D}$ is such that

(1) Its objects are pairs $\langle C, D\rangle$ where C is a $\mathscr{C}$-object and $\mathscr{D}$ is a $\mathscr{D}$-object;
(2) Its arrow from $\langle C, D\rangle$ to $\langle C', D'\rangle$ are the pairs $\langle f, g\rangle$ where $f\colon C \to C'$ is a $\mathscr{C}$-arrow and $g\colon D \to D'$ is a $\mathscr{D}$-arrow.
(3) For object $\langle C, D\rangle$ we define the identity arrow on this object by putting $1_{\langle C,D\rangle} = \langle 1_C, 1_D\rangle$;
(4) Composition is defined componentwise in the obvious way: $\langle f, g\rangle \circ \langle f', g'\rangle = \langle f \circ_{\mathscr{C}} f', g \circ_{\mathscr{D}} g'\rangle$. △

Obviously, this definition requires us to have suitable pairing schemes in play for the relevant objects and arrows: but assuming those are available, it is trivial to check that this well-defines a sort of category.

(c) Next, quotients. Remember, we say arrows f, g are parallel when they have the same source and target. Then, following closely what we said about quotients for groups in Defn. 6, we can say:

Definition 24. (i) If $\mathscr{C}$ is a category, then the relation $\sim$ is a *congruence* on its arrows iff it is an equivalence relation which respects composition.

That is to say, $f \sim g$ is an equivalence such that (i) if $f \sim g$, then they are parallel arrows (ensuring that equivalent arrows can compose in the same way), and (ii) if $f \sim g$, then $f \circ h \sim g \circ h$ and $k \circ f \sim k \circ g$ whenever the composites are defined.

(ii) Suppose $\mathscr{C}$ is a category, and suppose $\sim$ is a congruence on its arrows. And suppose we have a quotient scheme for $\sim$. Then $\mathscr{C}/{\sim}$ is the category whose objects are the same as those of $\mathscr{C}$ and whose arrows are the quotient objects $[f]$ for f in $\mathscr{C}$, with $[f]$ assigned the same source and target as an arrow in $\mathscr{C}/{\sim}$ as f has in $\mathscr{C}$. △

We've defined the notion of congruence so that it becomes trivial to check that $\mathscr{C}/{\sim}$ actually is a category.

For a natural example, take the category Top; and consider the congruence $\sim$ which holds between two of its arrows, i.e. two continuous maps between spaces, when one map can be continuously deformed into the other, i.e. there is a so-called homotopy between the maps (why is that a congruence?). Then $\mathsf{Top}/{\sim}$ is the important homotopy category hTop.

6.3 Slice categories

It is quite instructive to think through the further definitions in this and the next section. But equally, they can readily be skipped at a first pass.

(a) Suppose that $\mathscr{C}$ is a category, and X a particular $\mathscr{C}$-object. We next define a new category from $\mathscr{C}$, the so-called 'slice' category $\mathscr{C}/X$, where each of the new category's objects is a $\mathscr{C}$-arrow $f\colon A \to X$.

So $\mathscr{C}/X$'s *objects* f and g are the same as $\mathscr{C}$'s *arrows*, as it might be $f\colon A \to X$ and $g\colon B \to X$. Here they are:

$$\begin{tikzcd}
A \arrow[dr, "f"] & & B \arrow[dl, "g"'] \\
& X &
\end{tikzcd}$$

Then what can be a $\mathscr{C}/X$-arrow from f to g? Well, if we are constructing $\mathscr{C}/X$ from $\mathscr{C}$, then we'll surely need to use a $\mathscr{C}$-arrow j which sends A to B. However, not any old arrow $j\colon A \to B$ will do: we'll want j to interact appropriately with the arrows f and g. The obvious suggestion is to require j to be such that adding it gives us a commuting diagram (in $\mathscr{C}$, of course).

This motivates the following definition (and to keep things clear but brief, let's continue to use '$\mathscr{C}$-object' and '$\mathscr{C}$-arrow' to refer to the old objects and arrows in $\mathscr{C}$, and reserve plain 'object' and 'arrow' for the data to be found in $\mathscr{C}/X$):

Definition 25. Let $\mathscr{C}$ be a category, and X be a $\mathscr{C}$-object. Then the category $\mathscr{C}/X$, the *slice category over* X, has the following data:

(1) The objects are the $\mathscr{C}$-arrows $f\colon A \to X$ for any $\mathscr{C}$-object A

(2) An arrow from $f\colon A \to X$ to $g\colon B \to X$ is a $\mathscr{C}$-arrow $j\colon A \to B$ such that $g \circ j = f$ in $\mathscr{C}$.

(3) The identity arrow on $f\colon A \to X$ is the $\mathscr{C}$-arrow $1_A\colon A \to A$.

(4) Given arrows $j\colon f \to g$ and $k\colon g \to h$, their composition $k \circ j\colon f \to h$ is the $\mathscr{C}$-arrow $k \circ j$. $\triangle$

To be clear: the source and target of j as an arrow in $\mathscr{C}$ are respectively A, B. But the source and target of j as an arrow in the slice category $\mathscr{C}/X$ are respectively f and g.

Of course, we need to check that these data do together satisfy the axioms for constituting a category. So let's do that. In particular, we need to confirm that our definition of $k \circ j$ for composing $\mathscr{C}/X$-arrows works.

$$\begin{tikzcd}
A \arrow[d, "j"] \arrow[dr, "f"] & \\
B \arrow[r, "g"] \arrow[d, "k"] & X \\
C \arrow[ur, "h"'] &
\end{tikzcd}$$

We are given that $j\colon A \to B$ in $\mathscr{C}$ is such that $g \circ j = f$; and suppose likewise that $k\colon B \to C$ in $\mathscr{C}$ is such that $h \circ k = g$. So putting things together we get our commutative diagram. Or in equations, we have $(h \circ k) \circ j = f$ in $\mathscr{C}$, and therefore $h \circ (k \circ j) = f$. Hence $(k \circ j)$ really does count as an arrow in $\mathscr{C}/X$ from f to h, as we require.

The remaining checks to confirm $\mathscr{C}/X$ satisfies the axioms for being a category are then trivial.

(b) There's a dual notion we can define here, namely the idea of a *co-slice category* $X/\mathscr{C}$ (or the slice category *under* X). This category has as objects $\mathscr{C}$-arrows going opposite direction, i.e. they are arrows of the form $f\colon X \to A$. Then the rest of the definition is as you would predict given our explanation of duality: just go through the definition a slice category reversing arrows and the order of composition. (Check that this works!)

(c) Here are two quick examples of slice and co-slice categories, one of each kind:

(1) Pick a singleton set '1'. We have mentioned before the idea that we can think of any element x of X as an arrow $\vec{x}\colon 1 \to X$.

So now think about the co-slice category $1/\mathsf{Set}$. Its objects are the arrows $\vec{x}\colon 1 \to X$ for each X. We can think of such an arrow as providing us with a set X (its target) and then a selected distinguished element $x \in X$; in other words, it in effect gives us a pointed set. And then the arrows $1/\mathsf{Set}$ from some $\vec{x}$ to $\vec{y}$ are all the maps $f\colon X \to Y$ in Set such that $f \circ \vec{x} = \vec{y}$: so we can think of such maps as the maps which preserve basepoints.

Hence $1/\mathsf{Set}$ is (or, in some strong sense to be later explained, comes to the same as) the category Set_* of pointed sets.

(2) Second, take an n-membered index set $I_n = \{c_1, c_2, c_3, \ldots, c_n\}$. Think of the members of I_n as 'colours'. Then an arrow $S \to I_n$, can therefore be thought of as giving us a set (its source) whose members are coloured from that palette of n colours.

Hence we can think of FinSet/I_n as the category of n-coloured finite sets, exactly the sort of thing that combinatorialists are interested in.

More generally, we can think of a slice category Set/I as a category of 'indexed' sets, with I providing the indices.

6.4 Arrow categories

Our slice categories took some arrows from an old category and made them the objects of a new category. Let's finish with a more general way of doing that.

First let's think informally. We are going to take a category $\mathscr{C}$ and build from its materials a new category $\mathscr{C}^{\to}$ whose objects are *all* the arrows of the old category. OK: so here are two objects of the new category,

$$\begin{tikzcd} X_1 \arrow[d, "f_1"] & X_2 \arrow[d, "f_2"] \\ Y_1 & Y_2 \end{tikzcd}$$

where f_1 and f_1 are $\mathscr{C}$-arrows. Now, what sort of thing could be an arrow relating these two objects? The only obvious candidate for linking these together, if we are to use materials provided by $\mathscr{C}$, is to use more $\mathscr{C}$-arrows to make a commutative square. What else? So let's put

Definition 26. Given a category $\mathscr{C}$, the derived *arrow category* $\mathscr{C}^{\rightarrow}$ has the following data:

(1) $\mathscr{C}^{\rightarrow}$'s objects, its first sort of data, are simply the *arrows* of $\mathscr{C}$,

(2) Given $\mathscr{C}^{\rightarrow}$-objects f_1, f_2 (i.e. $\mathscr{C}$-arrows $f_1\colon X_1 \to Y_1$, $f_2\colon X_2 \to Y_2$), a $\mathscr{C}^{\rightarrow}$-arrow $f_1 \to f_2$ is a pair $\langle j, k\rangle$ of $\mathscr{C}$-arrows such that the following diagram commutes in $\mathscr{C}$:

$$\begin{tikzcd}
X_1 \arrow[r, "j"] \arrow[d, "f_1"] & X_2 \arrow[d, "f_2"] \\
Y_1 \arrow[r, "k"] & Y_2
\end{tikzcd}$$

(3) The identity arrow on $f\colon X \to Y$ is defined to be the pair $\langle 1_X, 1_Y\rangle$.

(4) And composition of arrows $\langle j, k\rangle\colon f_1 \to f_2$ and $\langle j', k'\rangle\colon f_2 \to f_3$ is then defined in the obvious way to be $\langle j' \circ j, k' \circ k\rangle\colon f_1 \to f_3$. △

It is straightforward to check that this definition does characterize a category, but it is worth doing so to fix ideas. In particular, in order to check that the definition of composition in $\mathscr{C}^{\rightarrow}$ works, just think of pasting together two of those commuting squares along a shared arrow f_2.

There are moderately fancy examples of arrow categories which do arise tolerably naturally e.g. in topology, but we won't delay over them now. I mention such categories here mainly as an exercise, and to reinforce once more the point that what makes data count as objects and arrows in a given category is not a matter of their intrinsic nature but of the respective roles they play.

7 Kinds of arrows

So where have we got to? We have given the general definition of a category – and note again that we have cast our net widely, going beyond the initial motivating idea of a family of structures equipped with enough structure-preserving maps between them. We have met a lot of examples, and then we've explained how to construct yet more categories from old ones in various ways.

It's now natural to want to impose some order on this proliferating universe of categories. That's why we are going to be centrally interested in *functors*, maps between categories which preserve categorial structure. But not yet. We are first going to spend a number of chapters looking *inside* categories before looking at relations *between* categories. In this chapter and the next, we make a start by characterizing a number of different kinds of arrows by the way they interact with other arrows. This will give us some elementary examples of categorial, arrow-theoretic, (re)definitions of familiar notions.

We have to introduce a number of standard but not-exactly-memorable technical terms as we go along. Sorry about this! But since you'll certainly meet the jargon elsewhere, there's nothing for it but to explain it here too.

7.1 Monomorphisms, epimorphisms

(a) Let's begin with a simple (and natural enough) definition:

Definition 27. An arrow $f\colon C \to D$ in the category $\mathscr{C}$ is *left-cancellable* iff for every parallel pair of arrows $g\colon B \to C$ and $h\colon B \to C$, if $f \circ g = f \circ h$ then $g = h$. △

But why is this notion interesting? First note that we have the following general result for concrete categories:

Theorem 10. *In a category where the arrows are indeed functions, such as* Set *or* Grp*, if f is injective as a function, then f is left-cancellable as an arrow.*

Proof. Suppose $f\colon C \to D$ is injective. Then in particular for any x, and any functions $g\colon A \to C$ and $h\colon A \to C$, we have $f(g(x)) = f(h(x))$ implies $g(x) = h(x)$. So in arrow-speak, if $f \circ g = f \circ h$ then $g = h$, so f is left-cancellable. □

And in many categories where the arrows are functions, the reverse is true. For example,

Theorem 11. *In* Set *and* Grp*, if f is left-cancellable as an arrow, it is injective as a function.*

Proof. For Set, suppose $f\colon C \to D$ is not injective. So, for some x, y we have $f(x) = f(y)$ but not $x = y$. But x and y will be respectively picked out as the values (for the only inputs) of functions $\vec{x}\colon 1 \to C$ and $\vec{y}\colon 1 \to C$, where 1 is your favourite singleton. Hence we have $f \circ \vec{x} = f \circ \vec{y}$ but not $\vec{x} = \vec{y}$. So the non-injective f in Set isn't left-cancellable. Contraposing gives us our wanted result.

For Grp, suppose that $f\colon C \to D$ is a group homomorphism between the groups $(C, *, e_C)$ and $(D, \star, e_D)$ but is not injective. So for some x, y we have $f(x) = f(y)$ but not $x = y$. Now, note that

$$f(x^{-1} * y) = f(x^{-1}) \star f(y) = f(x^{-1}) \star f(x) = f(x^{-1} \cdot x) = f(e_C) = e_D.$$

Let K (for 'kernel'!) be the objects that f sends e_D. So $x^{-1} * y$ belongs to K. And e_C is another *distinct* object that f sends to e_D (for if $x^{-1} * y = e_C$, then $x = x * e_C = x * x^{-1} * y = y$ contrary to hypothesis). Hence K includes more than one object.

Now define $g\colon K \to C$ to be the obvious inclusion map (which sends an object from K to the same element of C), while $h\colon K \to C$ sends everything to e_C. Since K has more than one element, $g \neq h$. But obviously, $f \circ g = f \circ h$ (both send everything in K to e_D). So the non-injective f in Grp isn't left-cancellable. Contraposing gives us our wanted result. □

So, putting things together, we have now proved that

Theorem 12. *In* Set *and* Grp *the left-cancellable arrows are exactly the injective functions.* □

And the same applies in most other categories where arrows are functions. But not all, because we can, with a bit of effort, find categories where arrows are functions but a left-cancellable function needn't be injective.[1]

(b) Now let's introduce the obvious twin notion:

Definition 28. An arrow $f\colon C \to D$ in the category $\mathscr{C}$ is *right-cancellable*, iff for every parallel pair of arrows $g\colon B \to C$ and $h\colon B \to C$, if $g \circ f = h \circ f$ then $g = h$. △

Left and right cancellability are evidently dual properties – i.e. f is right-cancellable in $\mathscr{C}$ if and only if it is left-cancellable in $\mathscr{C}^{op}$. And we easily get a companion result to Theorem 10:

Theorem 13. *In a category where the arrows are functions, such as* Set *or* Grp*, if f is surjective as a function, then f is right-cancellable as an arrow.*

[1]For those who know about such things, an example is provided by the category of divisible groups.

Proof. Suppose $f\colon C \to D$ is surjective. And consider any two further functions onwards from the target of f, namely $g, h\colon D \to E$.

Suppose $g \neq h$. Then for some d, $g(d) \neq h(d)$. But by the surjectivity of f, we know that $d = f(c)$ for some c in f's source domain, and hence $g(f(c)) \neq h(f(c))$. So in arrow-speak, $g \circ f \neq h \circ f$.

Contraposing, if $g \circ f = h \circ f$, then $g = h$. Hence, in sum, the surjectivity of f entails that it is right-cancellable. □

And there is an easy converse result in the special case of Set:

Theorem 14. *In* Set*, if f is right-cancellable as an arrow, then it is surjective as a function.*

Proof. Suppose $f\colon C \to D$ is not surjective, so $f[C] \neq D$. Consider two functions $g\colon D \to E$ and $h\colon D \to E$ which agree on $f[C]$ but disagree on the rest of D. Then $g \neq h$, even though by hypothesis $g \circ f$ and $h \circ f$ will agree everywhere on C, so f is not right-cancellable. Contraposing, if f is right-cancellable in Set, it is surjective. □

So putting the last two results together we have

Theorem 15. *In* Set *the right-cancellable arrows are exactly the surjective functions.* □

We can also show e.g. that in Grp, the surjective functions are right-cancellable; but this is certainly not trivial.[2] And in §8.1 we'll meet an easy case where we have a right-cancellable arrow which *is* a function but which is *not* surjective.

(c) There is a notational convention that we use some special drawn arrows in representing cancellable arrows, and we will follow this convention occasionally but not religiously:

$$f\colon C \rightarrowtail D \text{ or } C \xrightarrow{f} D \text{ represents a left-cancellable } f,$$

$$f\colon C \twoheadrightarrow D \text{ or } C \xrightarrow{f} D \text{ represents a right-cancellable } f.$$

That convention is easy enough to remember: just note that a left cancellable arrow gets notated by an extra decoration on the left of the arrow, and a right cancellable arrow gets an extra decoration on the right.

But now we need to introduce some distinctly less memorable but absolutely standard terminology that you need to know:

Definition 29. An arrow is a *monomophism* (or is *monic*) iff it is left-cancellable. And an arrow is an *epimorphism* (or is *epic*) iff it is right-cancellable.

The best I can offer by way of a mnemonic here is to go by the alphabetic proximity of *ML* and of *PR*: a *M*onomorphism is *L*eft cancellable, while an e*P*imorphism is *R*ight cancellable. Well, it works for me!

[2] Why can't we recycle the proof of Theorem 14? Because while there may be such *functions* as the g and h there, that's not enough – we need functions-as-arrows, which in this case means functions which are *group homomorphisms*.

(d) As the very gentlest of exercises, let's add for the record a mini-theorem:

Theorem 16. *(1) Identity arrows are always monic. Dually, they are always epic too.*

(2) If f, g are monic, so is $f \circ g$. If f, g are epic, so is $f \circ g$.

(3) If $f \circ g$ is monic, so is g. If $f \circ g$ is epic, so is f.

Proof. (1) is trivial.

For (2), we need to show that if $(f \circ g) \circ j = (f \circ g) \circ k$, then $j = k$. So suppose the antecedent. By associativity, $f \circ (g \circ j) = f \circ (g \circ k)$. Whence, assuming f is monic, $g \circ j = g \circ k$. Whence, assuming g is monic, $j = k$.

Interchanging f and g, if f and g are monic, so is $(g \circ f)$. Being epic is dual to being monic. So applying the duality principle from §6.1, it follows that f and g are epic, so is $(f \circ g)$.

For (3) assume $f \circ g$ is monic. Suppose $g \circ j = g \circ k$. We need to show $j = k$. But $f \circ (g \circ j) = f \circ (g \circ k)$, hence $(f \circ g) \circ j = (f \circ g) \circ k$, hence since $f \circ g$ is monic we have $j = k$. Dually again for epics. □

7.2 Inverses

(a) We define some more types of arrow (and now we are back for a while to self-explaining terminology!):

Definition 30. Given an arrow $f : C \to D$ in the category $\mathscr{C}$,

(1) $g : D \to C$ is a *right inverse* of f iff $f \circ g = 1_D$.
(2) $g : D \to C$ is a *left inverse* of f iff $g \circ f = 1_C$.
(3) $g : D \to C$ is an *inverse* of f iff it is both a right inverse and a left inverse of f. △

Three remarks. First, on the use of 'left' and 'right'. Note that if we represent the situation in (1) like this

$$\begin{tikzcd} D \arrow[r, "g"] \arrow[rr, bend right, "1_D"'] & C \arrow[r, "f"] & D \end{tikzcd}$$

then f's right inverse g appears on the left! It is just a matter of convention that we standardly describe handedness by reference to the representation '$f \circ g = 1_D$' rather than by reference to our representing diagram. (Similarly, of course, earlier in defining left-cancellability, etc.)

Second, note that $g \circ f = 1_C$ in $\mathscr{C}$ iff $f \circ^{op} g = 1_C$ in $\mathscr{C}^{op}$. So a left inverse in $\mathscr{C}$ is a right inverse in $\mathscr{C}^{op}$. And vice versa. The notions of a right inverse and left inverse are therefore, exactly as you would expect, dual to each other; and the notions of an inverse is its own dual.

Third, if f *has* a right inverse g, then it *is* a left inverse (of g, of course!). Dually, if f *has* a left inverse, then it *is* a right inverse.

(b) Let's start by considering what happens in concrete categories. Here's a *very* easy result:

Theorem 17. *In a category where arrows are functions, if f has a left-inverse as an arrow, it is injective as a function. And if f has a right-inverse, it is surjective as a function.*

Proof. For the first part, we just note that if $f(x) = f(y)$, then applying f's left inverse to both sides we can infer $x = y$.

For the second part, suppose $f\colon C \to D$ has a right inverse $g\colon D \to C$. Take any d in D. Then $f \circ g$ applied to d gives back d. In other words, there is an object c in C, where $c = g(d)$, such that $f(c) = d$. So f is surjective. □

So, putting together this last theorem with Theorems 10 and 13, the following hold for concrete categories:

$$\begin{array}{ccccc} f \text{ has left inverse} & \Rightarrow & f \text{ is injective} & \Rightarrow & f \text{ is left-cancellable.} \\ f \text{ has right inverse} & \Rightarrow & f \text{ is surjective} & \Rightarrow & f \text{ is right-cancellable.} \end{array}$$

What about categories where the arrows aren't functions (so the question of being injective or surjective doesn't arise)? Well, the first item on each line still implies the last. Or to ring the changes on the terminology, since you need to get used to this, we have the first part of the following theorem. But we also need to note that the converse implications do not in general hold.

Theorem 18. *(1) Every right inverse is monic, and every left inverse is epic.*

(2) But in general, not every monomorphism is a right inverse; and dually, not every epimorphism is a left inverse.

Proof. For (1), suppose f is a right inverse for e, which means that $e \circ f = 1$ (the identity arrow on the relevant object). Now suppose $f \circ g = f \circ h$. Then $e \circ f \circ g = e \circ f \circ h$, and hence $1 \circ g = 1 \circ h$, i.e. $g = h$, so f is monic. Similarly for the dual.

(2) can be shown by a toy example. Take the two-object category 2 which we met back in §4.7:

$$\begin{tikzcd} \bullet \arrow[loop left] \arrow[r, "f"] & \star \arrow[loop right] \end{tikzcd}$$

The non-identity arrow f can only compose with an identity arrow. So, for example, when we have $f \circ g = f \circ h$ it can only be because $g = h = 1_\bullet$. Hence f is monic. Similarly f is epic. But it lacks both a left and a right inverse. □

(c) So monics need not in general be right inverses nor epics left inverses. But how do things pan out in the particular case of the category Set?

Theorem 19. *In Set, every monomorphism is a right inverse apart from arrows of the form $\varnothing \to D$. Also in Set, the proposition that every epimorphism is a left inverse is (a version of) the Axiom of Choice.*

Proof. Suppose $f\colon C \to D$ in Set is monic, hence one-to-one between C and $f[C]$. Consider a function $g\colon D \to C$ that reverses f on $f[C]$ and maps everything in $D - f[C]$ to some particular object in C. Such a g is always possible to find in Set unless C is the empty set. So $g \circ f = 1_C$, and f is a right inverse.

Now suppose $f\colon C \to D$ in Set is epic, and hence a surjection. Assuming the Axiom of Choice, there will be a function $g\colon D \to C$ which maps each $d \in D$ to some chosen one of the elements c such that $f(c) = d$ (but note that this time, in the general case, we do have to make an infinite number of choices, picking out one element among the pre-images of d for every $d \in D$: that's why Choice is involved). Given such a function g, $f \circ g = 1_D$, so f is a left inverse.

Conversely, suppose we have a partition of C into disjoint subsets indexed by (exactly) the elements of D. Let $f\colon C \to D$ be the function which sends an object in C to the index of the partition it belongs to; then f is surjective, hence epic. Suppose f is also a left inverse, so for some $g\colon D \to C$, $f \circ g = 1_D$. Then g is evidently a choice function, picking out one member of each partition. So the claim that every epic is a left inverse gives us the Axiom of Choice. $\square$

(d) An arrow can zero or one left inverse. It can also have more than one. For a mini-example in Set, consider $f\colon \{0,1\} \to \{0,1,2\}$ where $f(0) = 0$, $f(1) = 1$. Then $g\colon \{0,1,2\} \to \{0,1\}$ is a left inverse so long as $g(0) = 0$, $g(1) = 1$; we have two choices for $g(2)$, and hence two left inverses. By the duality principle, an arrow can also have zero, one, or many right inverses. However,

Theorem 20. *If an arrow has both a right inverse and a left inverse, then these are the same and are the arrow's unique inverse.*

Proof. Suppose $f\colon C \to D$ has right inverse $r\colon D \to C$ and left inverse $s\colon D \to C$. Then

$$r = 1_C \circ r = (s \circ f) \circ r = s \circ (f \circ r) = s \circ 1_D = s.$$

Or, to put it diagrammatically, the following commutes:

$$\begin{tikzcd}
D \arrow[r, "r"] \arrow[rrr, bend left=40, "r"] \arrow[rr, bend right, "1_D"'] \arrow[rrr, bend right=40, "s"'] & C \arrow[r, "f"] \arrow[rr, bend left, "1_C"] & D \arrow[r, "s"] & C
\end{tikzcd}$$

Hence $r = s$ and r is an inverse.

Suppose now that f has inverses r and r'. Then r will be a right inverse and r' a left inverse for f, so as before $r = r'$. Therefore inverses are unique. $\square$

(e) There really is an annoying oversupply of other jargon hereabouts, also in pretty common use elsewhere. I suppose I should note some of the alternatives for the record (but we won't be making any further use of the terminology here, so I won't highlight these in official definitions).

Assume we have a pair of arrows $f\colon C \to D$, and $g\colon D \to C$ such that $g \circ f = 1_C$. Then f is a right inverse of g, but f is also called a *section* of g; and g, which is a left inverse of f, is also said to be a *retraction* of f. (In this usage, f *is* a section iff it *has* a retraction, etc.)

Further, if f has a left inverse/is a right inverse, then f is said to be a *split monomorphism*; if g has a right inverse/is a left inverse, then g is a *split epimorphism*. (In this usage, we can say e.g. that the claim that every epimorphism splits in Set is the categorial version of the Axiom of Choice.)

Note that Theorem 18 tells us that right inverses are monic, so a split monomorphism is properly called a monomorphism. Dually, a split epimorphism is an epimorphism.

7.3 Aside: groups as categories

Recall that we can consider a particular monoid as itself giving rise to a category – see §4.4 (C3). Let's just pause to remark that in the same way, a particular group gives rise to a category, one with a bit more structure.

So take a group $(G, *, e)$ and define $\mathscr{G}$ to be the corresponding category whose sole object $\bullet$ is whatever you like, and whose arrows are the simply the group objects G, with e the identity arrow. Composition of arrows in $\mathscr{G}$ is defined as group-multiplication of elements in G.

Now, since every element in the group has an inverse, it follows immediately that every arrow in the corresponding category $\mathscr{G}$ has an inverse. This is the key difference from a monoid-as category.

In sum then, a group-as-a-category is a category with one object and whose every arrow has an inverse.[3]

[3] There's a more general notion around, of a category with perhaps more than one object but whose arrows all still have inverses: this is called a *groupoid*. But we won't be needing this idea.

8 Isomorphisms

Before we ever encounter category theory, we are familiar with the notion of an isomorphism between groups, between metric spaces, between topological spaces, between orderings, etc. – it's a bijection between the underlying objects which preserves all the relevant structure.

How can we redefine this idea in arrow-theoretic, categorial, terms?

8.1 What doesn't work

In the extremal case, in the category Set of sets with no additional structure, the bijections are the arrows which are both monic and epic. Can we generalize from this case and define the isomorphisms of any category to be arrows which are monic and epic there?

No. Isomorphisms properly so called need to have inverses. But being monic and epic doesn't always imply having an inverse. We can use again the toy case of the two-object category which has just one non-identity arrow. That non-identity arrow, we saw in proving Theorem 18, is trivially both monic and epic, but lacks an inverse. Or here's a generalized version of the same idea:

(1) Take the category $\mathscr{N}$ corresponding to the pre-ordered objects $(N, \preccurlyeq)$, as in §4.4 (C4). Then there is at most one arrow between any given objects of $\mathscr{N}$. But if $f \circ g = f \circ h$, then g and h must share the same object as source and same object as target, hence $g = h$, so f is monic. Similarly f must be epic. But no arrows other than identities have inverses.

The arrows in that example aren't functions, however. So here's a revealing case where the arrows *are* functions but where being monic and epic *still* doesn't imply having an inverse:

(2) Consider the category Mon of monoids. Among its objects are $\mathcal{N} = (\mathbb{N}, +, 0)$ and $\mathcal{Z} = (\mathbb{Z}, +, 0)$ – i.e. the monoid of natural numbers equipped with addition and the monoid of positive and negative integers equipped with addition. Let $i \colon \mathcal{N} \to \mathcal{Z}$ be the map which sends a natural number to the corresponding positive integer. This map obviously does not have an inverse in Mon. But it is both monic and epic.

That last claim is interesting enough to be worth pausing to prove it:

(i) First, for the easy half, suppose $\mathcal{M} = (M, \cdot, 1_M)$ is some monoid and we have two arrows $g, h\colon \mathcal{M} \to \mathcal{N}$, where $g \neq h$. There is then some $\mathcal{M}$-object m such that the natural numbers $g(m)$ and $h(m)$ are different, which means that the corresponding integers $i(g(m))$ and $i(h(m))$ are different, so $i \circ g \neq i \circ h$. Contraposing, this means i is monic in the category.

(ii) Second, again take a monoid $\mathcal{M}$ and this time consider any two monoid homomorphisms $g, h\colon \mathcal{Z} \to \mathcal{M}$ such that $g \circ i = h \circ i$. Then g and h must agree on all integers from zero up. We'll now show that g and h agree on negative integers too, starting from -1. So note we have

$$\begin{aligned} g(-1) &= g(-1) \cdot 1_M = g(-1) \cdot h(0) = g(-1) \cdot h(1 + -1) \\ &= g(-1) \cdot h(1) \cdot h(-1) = g(-1) \cdot g(1) \cdot h(-1) \\ &= g(-1 + 1) \cdot h(-1) = g(0) \cdot h(-1) = 1_M \cdot h(-1) = h(-1). \end{aligned}$$

But if $g(-1) = h(-1)$, then

$$\begin{aligned} g(-2) &= g(-1 + -1) = g(-1) \cdot g(-1) = h(-1) \cdot h(-1) \\ &= h(-1 + -1) = h(-2), \end{aligned}$$

and the argument iterates, so we have $g(z) = h(z)$ for all $z \in \mathbb{Z}$, positive and negative. Hence $g = h$ and i is right-cancellable, i.e. epic.

And note too, picking up a point from the end of §7.1(b), i is also an example of an epic arrow which is a function but isn't surjective.

8.2 Isomorphism defined

(a) The moral of our last examples? If we want isomorphisms to be invertible, then we'll just have to build in that feature by definition! So:

Definition 31. An *isomorphism* in category $\mathscr{C}$ is an arrow which has an inverse. We conventionally represent isomorphisms by decorated arrows, thus: $\xrightarrow{\sim}$. △

From what we have already seen, we know or can immediately check that

Theorem 21. *(1) Identity arrows are isomorphisms.*

(2) An isomorphism $f\colon C \xrightarrow{\sim} D$ has a unique inverse which we can call $f^{-1}\colon D \xrightarrow{\sim} C$, such that $f^{-1} \circ f = 1_C$, $f \circ f^{-1} = 1_D$, $(f^{-1})^{-1} = f$, and f^{-1} is also an isomorphism.

(3) If f and g are isomorphisms, then $g \circ f$ is an isomorphism if it exists, whose inverse will be $f^{-1} \circ g^{-1}$.

Let's give some simple examples of isomorphisms in different categories:

(1) In Set, the isomorphisms are the bijective set-functions.

(2) In Grp, the isomorphisms are the bijective group homomorphisms.

(3) In Vect_k, the isomorphisms are invertible linear maps.

(4) In a group treated as a category, every arrow is an isomorphism.

(5) But as we noted, in a pre-order category, the only isomorphisms are the identity arrows.

(b) Isomorphisms are monic and epic by Theorem 18. And we now know that arrows which are monic and epic need not be isomorphisms.

However, we do have this:

Theorem 22. *If f is both monic and has a right inverse (or both epic and has a left inverse), then f is an isomorphism.*

Proof. If f has a right inverse, there is a g such that $f \circ g = 1$. Then $(f \circ g) \circ f = f$, whence $f \circ (g \circ f) = f \circ 1$. Hence, given that f is also mono, $g \circ f = 1$. So g is both a left and right inverse for f, i.e. f has an inverse. Dually for the other half of the theorem. □

We will also mention another easy result in the vicinity:

Theorem 23. *If f and g are both monic arrows with the same target, and each factors through the other, i.e. there are i, j such that $f = g \circ i$ and $g = f \circ j$, then the factors i and j are isomorphisms and inverse to each other.*

In other words, if each of the triangles in the following diagram commutes, then so does the whole diagram:

$$\begin{tikzcd}
X \arrow[rr, shift left, "i"] \arrow[dr, tail, "f"'] & & Y \arrow[ll, shift left, "j"] \arrow[dl, tail, "g"] \\
& Z &
\end{tikzcd}$$

Proof. We have $f \circ 1_X = f = g \circ i = f \circ j \circ i$. Since f is monic, $j \circ i = 1_X$. Similarly, $i \circ j = 1_Y$. So i and j are each other's two-sided inverse, and both are isomorphisms. □

(c) Finally, we should mention a bit of standard terminology:

Definition 32. A category $\mathscr{C}$ is *balanced* iff every arrow which is both monic and epic is in fact an isomorphism.

Then we have seen that some categories like Set are balanced, while others like Mon are not. Top is another example of an unbalanced category.

8.3 Isomorphic objects

(a) We can now introduce another key notion:

Definition 33. If there is an isomorphism $f : C \xrightarrow{\sim} D$ in $\mathscr{C}$ then the objects C, D are said to be *isomorphic* in $\mathscr{C}$, and we write $C \cong D$. △

From the ingredients of Theorem 21, we immediately get the desirable result that

Theorem 24. *Isomorphism between objects in a category $\mathscr{C}$ is an equivalence relation.*

An isomorphism between objects in a category also induces a bijection between the arrows to (or from) those objects:

Theorem 25. *If $C \cong D$ in $\mathscr{C}$, then there is a one-one correspondence between arrows $X \to C$ and $X \to D$ for all objects X in $\mathscr{C}$, and likewise a one-one correspondence between arrows $C \to X$ and $D \to X$.*

Proof. If $C \cong D$ then there is an isomorphism $j\colon C \xrightarrow{\sim} D$. Consider the map which sends an arrow $f\colon X \to C$ to $\overline{f} = j \circ f\colon X \to D$. This map $f \mapsto \overline{f}$ is injective (for $\overline{f} = \overline{g}$ entails $j^{-1} \circ \overline{f} = j^{-1} \circ \overline{g}$ and hence $f = g$). It is also surjective (for any $g\colon X \to D$, put $f = j^{-1} \circ g$ then $\overline{f} = g$). Similarly for the other part. $\square$

(b) We might wonder, however, how far this notion of isomorphism between objects in a category actually captures the idea of two objects amounting to the same as far as their ambient category is concerned.

We mentioned before the example where we have, living in Grp, lots of instances of a Klein four-group which are group-theoretically indiscernible by virtue of being isomorphic (indeed, between any two instances, there is a unique isomorphism). And yes, we then cheerfully talk about *the* Klein four-group.

There is a real question, however, about just what this way of talking amounts to, when we seemingly identify isomorphic objects. Some claim that category theory itself throws a lot of light on this very issue (see e.g. Mazur 2008). And certainly, category theory typically doesn't care about distinguishing isomorphic objects in a category.

Note though that there is a unique bijection between any two singleton sets – so, in the category Set any two singletons count as isomorphic. And it would strike us as odd to say that we can always happily talk about *the* singleton. To be sure, there are contexts where any singleton will do, as for example when we associate elements x of a set X with arrows $\vec{x}\colon 1 \to X$. But in other contexts, the pairwise distinctness of singletons could be important, e.g. when we treat $\{\varnothing\}, \{\{\varnothing\}\}, \{\{\{\varnothing\}\}\}, \{\{\{\{\varnothing\}\}\}\}, \ldots$ as a sequence of *distinct* singletons in one possible construction (Zermelo's) for the natural numbers.

But we can't delay to explore this issue any further just at the moment: we are just flagging up that there are questions we'll at some point want to discuss around and about the idea of isomorphism-as-sameness.

9 Initial and terminal objects

When we defined an isomorphism in the previous chapter, we characterized a type of arrow not by (so to speak) its internal workings – not by how it operated on its source and target domains – but by reference to its interaction with another arrow, its inverse. This is entirely typical of a category-theoretic (re)definition of a familiar notion: we look for similarly external, relational, characterizations of arrows and/or structured objects.

Here is Awodey, offering some similarly arm-waving

> remarks about category-theoretical definitions. By this I mean characterizations of properties of objects and arrows in a category in terms of other objects and arrows only, that is, in the language of category theory. Such definitions may be said to be abstract, structural, operational, relational, or external (as opposed to internal). The idea is that objects and arrows are determined by the role they play in the category via their relations to other objects and arrows, that is, by their position in a structure and not by what they 'are' or 'are made of' in some absolute sense. (Awodey 2006, p. 25)

We proceed, then, in this spirit to give some further examples of external category-theoretic definitions of a range of familiar notions. A prime exhibit will be the illuminating treatment of products, starting in the next chapter. In this chapter, however, we warm up by considering a particularly simple pair of cases.

9.1 Initial and terminal defined

(a) As we noted in §4.6, in a nice category of sets,

(i) For any set X, there is one and only one set-function from the empty set $\varnothing$ to X – namely the empty function. Moreover, if the set S is such that for every X there is one and only one set-function from S to X, then S is the empty set.

(ii) For any set X, there is one and only one set-function from X to a singleton set $\{\star\}$ – namely the empty function if X is the empty set, or otherwise the function which maps every member of X to $\star$. Moreover, if the set S

is such that for every X there is one and only one set-function from X to S, then S is a singleton.

In category-speak: in Set the empty set is distinguished by being such that there is one and only one arrow *from* it to any object. And a singleton is distinguished by being such that there is one and only one arrow *to* it from any object.

Let's now introduce a pair of quite natural concepts:

Definition 34. The object I is an *initial* object of the category $\mathscr{C}$ iff, for every $\mathscr{C}$-object X, there is a unique arrow $!\colon I \to X$.

Dually, the object T is a *terminal* object of $\mathscr{C}$ iff, for every $\mathscr{C}$-object X, there is a unique arrow $!\colon X \to T$.[1] △

Then, in summary, we've just noted that

(1) The empty set is initial in Set, while any singleton is terminal.

Let's immediately have some more examples;

(2) In the pre-ordering $(\mathbb{N}, \leqslant)$ thought of as a category, zero is trivially the unique initial object and there is no terminal object. By contrast $(\mathbb{Z}, \leqslant)$ has neither initial nor terminal objects.

    More generally, $(S, \leqslant)$-treated-as-a-category has an initial object iff the pre-order has a minimum, an object which $\leqslant$-precedes all the others. Dually for terminal objects/maxima.

(3) $\mathsf{Set}_\star$, recall, is the category whose objects are non-empty sets equipped with a distinguished member and whose arrows are functions preserving distinguished members. Such a function from a singleton in $\mathsf{Set}_\star$ must map its (automatically distinguished) member to the distinguished member of its target X. And any such function from X to a singleton will be unique. Hence in $\mathsf{Set}_\star$ each singleton is both initial and terminal.

(4) In Rel, the category of sets and relations, the empty set is both the sole initial and sole terminal object.

(5) In Top, the empty set (considered as a trivial topological space) is the initial object. Any one-point singleton space is a terminal object.

(6) As in effect noted in §2.4, in Grp the trivial one-element group is an initial object. The same one-element group is also terminal.

(7) In the category Bool, the trivial one-object algebra is terminal. While the two-object algebra on $\{0, 1\}$ familiar from propositional logic is initial – for a homomorphism of Boolean algebras from $\{0, 1\}$ to B must send 0 to the bottom object of B and 1 to the top object, and there's a unique map that does that.

[1] The use of '!' to signal the unique arrows from an initial object (or to a terminal object) is common. If we want explicitly to indicate the source (or target) of such a unique arrow, we can write $!_X$. By the way, some call terminal objects *final*; and then that frees up 'terminal' to mean *initial or final*.

(8) Recall: in the slice category $\mathscr{C}/X$ an object is a $\mathscr{C}$-arrow like $f\colon A \to X$, and a $\mathscr{C}/X$ arrow from $f\colon A \to X$ to $g\colon B \to X$ is a $\mathscr{C}$-arrow $j\colon A \to B$ such that $g \circ j = f$ in $\mathscr{C}$.

Consider the $\mathscr{C}/X$ object which is the $\mathscr{C}$-arrow 1_X. A $\mathscr{C}/X$ arrow from $f\colon A \to X$ to $1_X\colon X \to X$ is a $\mathscr{C}$-arrow $j\colon A \to X$ such that $1_X \circ j = f$, i.e. such that $j = f$ – which always exists and is unique! So 1_X is terminal in $\mathscr{C}/X$.

Such various cases show that a category may have zero, one or many initial objects, and (independently of that) may have zero, one or many terminal objects. Further, an object can be both initial and terminal.

There is, incidentally, a standard bit of jargon for the last case:

Definition 35. An object O in the category $\mathscr{C}$ is a *null object* of the category $\mathscr{C}$ iff it is both initial and terminal.[2] △

9.2 Uniqueness up to unique isomorphism

Evidently, the ideas of being initial and being terminal are dual, as they can be interrelated by reversing arrows. So for every general result about initial objects, there is a dual result about terminal objects.

Now, a category $\mathscr{C}$, to repeat, may have no initial objects, or only one, or have many. However, we do have the following key result:

Theorem 26. *Initial objects, when they exist, are 'unique up to unique isomorphism': i.e. if the $\mathscr{C}$-objects I and J are both initial in the category $\mathscr{C}$, then there is a unique isomorphism $f\colon I \xrightarrow{\sim} J$ in $\mathscr{C}$. Dually for terminal objects.*

Further, if I is initial and $I \cong J$, then J is also initial. Dually for terminal objects.

Proof. Suppose I and J are both initial objects in $\mathscr{C}$. By definition there must be unique $\mathscr{C}$-arrows $f\colon I \to J$, and $g\colon J \to I$. Then $g \circ f$ is an arrow from I to itself. Another arrow from I to itself is the identity arrow 1_I. But since I is initial, there can only be one arrow from I to itself, so $g \circ f = 1_I$. Likewise $f \circ g = 1_J$. Hence the unique arrow f has a two-sided inverse and is an isomorphism. (Note this pattern of argument: we'll be using it a lot!)

Next suppose I is initial and $I \cong J$, so that there is an isomorphism $i\colon I \xrightarrow{\sim} J$. Then for any X, there is a unique arrow $f\colon I \to X$.

Now take any arrow $g\colon J \to X$. Then $g \circ i\colon I \to X$, and so by uniqueness, $g \circ i = f$. Hence g is required to equal $f \circ i^{-1}$. In other words, for any X there is a unique arrow g from J to X, thus J is also initial.

Duals of these two arguments deliver, of course, the dual results. □

[2] Null objects are often alternatively called 'zero' objects. But that perhaps doesn't sit happily with the standard practice we will adopt of using '0' for an initial object. For 0 (in the sense of an initial object) typically isn't a zero (in the sense of null) object.

It is standard to introduce notation for arbitrary initial and terminal objects (since categorially, we often won't care about distinctions among instances):

Definition 36. We use '0' to denote an initial object of $\mathscr{C}$ (assuming one exists), and likewise '1' to denote a terminal object. △

9.3 Elements

(a) Consider the category Set again. As we have remarked before, arrows $\vec{x}\colon 1 \to X$ from a terminal object (a singleton!) correlate one-to-one with elements $x \in X$: so, when working in Set, we can think of talk of such arrows $\vec{x}\colon 1 \to X$ as the categorial version of talking of members of X.

We had better check, though, that it doesn't matter which terminal object 1 we take here. So suppose 1 and $1'$ are two terminal objects in Set. There is a unique isomorphism $j\colon 1 \xrightarrow{\sim} 1'$. Then if $\vec{x}'\colon 1' \to X$ picks out a certain member of X, then $\vec{x} = \vec{x}' \circ j\colon 1 \to X$ picks out the same object, and there will be a one-to-one correspondence between arrows $\vec{x}'$ and $\vec{x}$.

(b) We now generalize and carry this idea over to other categories:

Definition 37. In a category $\mathscr{C}$ with a terminal object 1, an *element* or *point* of the $\mathscr{C}$-object X is an arrow $f\colon 1 \to X$.[3] △

And here's a little theorem to help fix ideas:

Theorem 27. *Point elements $\vec{x}\colon 1 \to X$ in a category are monic.*

Proof. Suppose $\vec{x} \circ f = \vec{x} \circ g$; then, for the compositions to be defined and equal, both f and g must be morphisms $Y \to 1$, for the same Y. Hence $f = g$ since 1 is terminal. □

(c) We can immediately see, however, that in categories $\mathscr{C}$ other than Set, these so-called 'elements' $1 \to X$ won't always line up nicely with the elements of X in the intuitive sense. In Grp, for example, a homomorphism from 1 (remember, that's a one-element group) to a group X has to send the only group element of 1 to the identity element e of X: so there is only one possible homomorphism $\vec{e}\colon 1 \to X$, independently of how many elements there are in the group X.

We can put this last observation in more categorial terms. Let's say:

Definition 38. Suppose the category $\mathscr{C}$ has a terminal object 1. And suppose that for any objects X, Y in $\mathscr{C}$, and parallel arrows $f, g\colon X \to Y$, $f = g$ if for all $\vec{x}\colon 1 \to X$, $f \circ \vec{x} = g \circ \vec{x}$. Then $\mathscr{C}$ is said to be *well-pointed.* △

Then Set is, in this sense, well-pointed. There are enough elements-as-arrows to ensure that parallel arrows with domain X which act identically on all relevant elements of X are in fact identical.

[3] Other standard terminology for such an element is '*global* element', picking up from a paradigm example in topology – but we won't fuss about that.

By contrast, we have just noted that Grp is not well-pointed. Take any two group homomorphisms $f, g\colon X \to Y$ where $f \neq g$. Still, for all possible $\vec{x}\colon 1 \to X$, both $f \circ \vec{x}$ and $g \circ \vec{x}$ must send the sole member of 1 to the identity element of the group Y, so are equal.

(d) Our definition of well-pointedness invokes a choice of the terminal object 1 in terms of which we define elements $\vec{x}\colon 1 \to X$. If the notion of well-pointedness is to be useful, though, the choice of terminal object should not matter. And it doesn't:

Theorem 28. *Take two terminal objects* 1 *and* $1'$ *and define two different types of elements of* X *in* $\mathscr{C}$ *as arrows* $1 \to X$ *and* $1' \to X$. $\mathscr{C}$ *is well-pointed with respect to elements of the first kind iff it is well-pointed with respect to elements of the second kind.*

Proof. We need only prove one direction. By the same argument as we used in the case of Set, if 1 and $1'$ are terminal, there is a unique isomorphism $j\colon 1 \to 1'$, and we can set up a one-one correspondence between elements $\vec{x}\colon 1 \to X$ and $\vec{x}'\colon 1' \to X$ by putting $\vec{x} = \vec{x}' \circ j$.

Assume $\mathscr{C}$ is well-pointed with respect to elements of the first kind. Then, for all $f, g\colon X \to Y$, if $f \circ \vec{x}' = g \circ \vec{x}'$, then $f \circ \vec{x} = f \circ \vec{x}' \circ j = g \circ \vec{x}' \circ j = g \circ \vec{x}$, and therefore by well-pointedness with respect to elements of the first kind $f = g$. Which proves well-pointedness with respect to the second sort of element. $\square$

9.4 Generalized elements

(a) We have just seen in the case of Grp that, even when arrows in a category are functions, acting the same way on elements need not imply being the same arrow. An obvious question arises: can we generalize the notion of an element so that acting the same way on generalized elements *does* imply being the same arrow? Well, suppose we say:

Definition 39. A *generalized element* (of shape S) of the object X in $\mathscr{C}$ is an arrow $s\colon S \to X$. $\triangle$

Generalized elements give us more ways of interacting with the data of a category than the original point elements. And now we indeed have

Theorem 29. *Parallel arrows in a category* $\mathscr{C}$ *are identical if and only if they act identically on all* generalized *elements.*

Proof. If $f, g\colon X \to Y$ act identically on *all* generalized elements, they act identically on $1_X\colon X \to X$: so $f \circ 1_X = g \circ 1_X$, and $f = g$. The converse is trivial. $\square$

We might wonder though whether it isn't a bit of a jump to go from point elements of X (arrows from a terminal object to X) to generalized elements of X (arrows from *any* object to X). We'll have to see later whether such a sweepingly generalized notion is useful.

10 Pairs and products, pre-categorially

The discussion in the last chapter illustrates a key categorial theme. We defined initial objects and terminal objects (with numerous examples) in terms of the arrows for which they are source or target, and we showed that the objects defined this way are 'unique up to unique isomorphism'. This is a pattern which will keep recurring in rather more exciting contexts, starting in the next chapter where we meet categorial products.

We are familiar in pre-categorial maths with constructing products for all kinds of widgets. The paradigm case, of course, is where we take sets X and Y and form their Cartesian product, the set of ordered pairs of their elements. But let's pause to ask: what *are* ordered pairs? That's the question for this chapter, and our answer will point forward to a categorial treatment of products.

10.1 Two ways of pairing numbers

(a) Suppose for a moment that we are working in a theory of arithmetic and we need to start considering ordered pairs of natural numbers. Perhaps we want to go on to use such pairs in constructing integers or rationals.

Then we can easily handle such ordered pairs of natural numbers as single objects, and without taking on any new commitments, by the trick of using *code-numbers*. For example, if we want a bijective coding between pairs of naturals and all the numbers, we could adopt the scheme of coding the ordered pair (m, n) by the single number $\langle m, n\rangle_B = \{(m+n)^2 + 3m + n\}/2$. Or, if we don't insist on every number coding a pair, we could adopt the simpler policy of using powers of primes, setting $\langle m, n\rangle_P =_{\text{def}} 2^m 3^n$, which allows rather simpler decoding functions for extracting m and n from $\langle m, n\rangle_P$. Relative to this coding scheme, we can call such code-numbers $\langle m, n\rangle_P$ *pair-numbers*, and by a slight abuse of terminology we might refer to m as the first element of the pair, and n as the second element.

(b) Now, you might be very tempted to protest that this coding trick is unnatural compared with the set-theoretic way of dealing with ordered pairs of numbers. After all,

(i) a single pair-number $\langle m, n\rangle_P$ as just defined is really neither ordered nor a twosome;

(ii) the number m is a member of (or is one of) the pair of m with n, but a number can't be a genuine member of a pair-number $\langle m,n\rangle_P$; and
(iii) such a coding scheme is quite arbitrary (e.g. we could equally well have used 3^m5^n as a code for the pair m,n).

Which is all true. But note that we can lay *exactly* analogous complaints against e.g. the familiar Kuratowski implementation of ordered pairs that we all know and love. This treats the ordered pair of m with n as the set $\langle m,n\rangle_K = \{\{m\},\{m,n\}\}$. But then:

(i′) $\langle m,n\rangle_K$ is not intrinsically ordered (after all, it is just a *set*!), nor is it always two-membered (consider the case where $m = n$);
(ii′) even when it is a twosome, its members are not the members of the pair: in standard set theories, m cannot be a member of $\{\{m\},\{m,n\}\}$; and
(iii′) the construction again involves quite arbitrary choices: thus $\{\{n\},\{m,n\}\}$ or $\{\{\{m\}\},\{\{m,n\}\}\}$ etc., etc., would have done just as well as alternative implementations.

On these counts, then, coding pairs of numbers by using pair-numbers in fact involves no worse a trick than coding them using Kuratowski's standard gadget.

There is indeed a rather neat symmetry between the adoption of pair numbers as representing ordered pairs of numbers and another very familiar procedure adopted by the enthusiast for working in ZFC. For remember that standard ZFC knows only about pure sets. So to get natural numbers into the story at all – and hence to get Kuratowski pair-sets of natural numbers – the enthusiast for sets has to choose some convenient sequence of sets to implement the numbers (or to 'stand proxy' for numbers, 'simulate' them, 'play the role' of numbers, or even 'define' them – whatever your favourite way of describing the situation is). But someone who, for her purposes, has opted to play the game this way, treating pure sets as basic and dealing with natural numbers by selecting some convenient sets to implement them, is hardly in a position to complain about someone else who, for his purposes, goes in the opposite direction and treats numbers as basic and deals with ordered pairs of numbers by choosing some convenient code-numbers to implement *them*. Both theorists are in the implementation game.

(c) It might be retorted that the Kuratowski trick has the virtue of being an all-purpose device, available not just when you want to talk about pairs of *numbers*, while e.g. the powers-of-primes coding is of much more limited use. Again true. Similarly you can use sledgehammers to crack all sorts of things, while nutcrackers are only useful for dealing with nuts. But that's not particularly to the point if it happens to be nuts you currently want to crack, efficiently and with light-weight resources. Similarly, if we want to implement ordered pairs of numbers without ontological inflation – say in pursuing the project of 'reverse mathematics' (with its eventual aim of exposing the minimum commitments required for e.g. doing classical analysis, as in Simpson 2010) – then pair-numbers are *exactly* the kind of thing we need!

10.2 Pairing schemes

(a) So: pair-numbers $\langle m, n\rangle_P$ and Kuratowski pairs $\langle m, n\rangle_K$ belong to two different schemes for pairing up numbers, each of which works well enough (though a particular surrounding context might lead us to prefer one to the other). Let's now ask: what does it take more generally to have such a workable scheme for pairing numbers with numbers? Or to have a scheme for pairing up other Xs with Ys?

We've been here before, with Defn. 3. To repeat, we need some objects O to code up pairs; we need a binary function that sends a given x from the Xs and a given y from the Ys to a particular pair-coding object o; and (of course!) we need a couple of functions which allow us to recover x and y from o. And the point suggested by the case of rival pairing schemes for numbers is that maybe we shouldn't care too much about the 'internal' nature of the objects O, so long as we can associate them with suitable pairing and unpairing functions which fit together in the right way (roughly, pairing and then unpairing must get us back to where we started, and likewise unpairing followed by pairing).

Which motivates the following general definition, sharpening Defn. 3 – though just for local typographical neatness, we'll now use 'pr' generically for a pairing function (rather than '$\langle\ ,\ \rangle$'):

Definition 40. Suppose X are some objects, Y are some objects, and O are also some objects (not necessarily all distinct). Let $pr\colon X, Y \to O$ be a two-place function, while $\pi_1\colon O \to X$, and $\pi_2\colon O \to Y$, are one-place functions. Then (O, pr, π_1, π_2) *form a pairing scheme for* X *with* Y iff for all x, y and o,

(a) $\pi_1(pr(x, y)) = x$ and $\pi_2(pr(x, y)) = y$,
(b) $pr(\pi_1(o), \pi_2(o)) = o$,

where 'x' is a typed variable running over the objects X, etc. The objects O will be said to be the *pair-objects* of the pairing scheme, with pr the associated *pairing function*, while π_1 and π_2 are unpairing or *projection* functions. △

It hardly needs to be said that, if O are all the natural numbers of the form 2^m3^n and $pr(m, n) = \langle m, n\rangle_P = 2^m3^n$, with $\pi_1(o)$ (or $\pi_2(o)$) returning the exponent of 2 (or 3) in the factorization of o, then (O, pr, π_1, π_2) officially form a scheme for pairing naturals with naturals. And if O' are all the Kuratowski pairs $\langle m, n\rangle_K$, with $pr'(m, n) = \langle m, n\rangle_K$, and with π'_1 (π'_2) taking a pair $\langle m, n\rangle_K$ and returning its first (second) element, then $(O', pr', \pi'_1, \pi'_2)$ form another scheme for pairing naturals with naturals.[1]

Two simple facts about pairing schemes:

Theorem 30. *If (O, pr, π_1, π_2) is a pairing scheme, then (i) different pairs of objects are sent by pr to different pair-objects, i.e. $pr(x, y) = pr(x', y')$ iff $x = x'$ and $y = y'$; and (ii) pr, π_1 and π_2 are all surjective.*

[1] There is no need to over-interpret the brackets in (O, pr, π_1, π_2): they are no more than punctuation, so you can read 'the objects O, taken together with the functions pr, π_1 and π_2'.

Proof. For (i) suppose $pr(x, y) = pr(x', y')$. Then by condition (a) on pairing schemes, $x = \pi_1(pr(x, y)) = \pi_1(pr(x', y')) = x'$, and likewise $y = y'$.

We want (ii) to be true so that O are no more than just the relevant pair-objects, and so that every x among X is the first projection of a pair, etc. And it is indeed immediate that pr is surjective by (b). The projection function π_1 is surjective because, given x among X, we can take any y among Y and put $o = pr(x, y)$, and then by (a), $x = \pi_1 o$. Similarly for π_2. □

As we'd also expect, for given candidate pair-objects O, a pairing function fixes the two corresponding projection functions, and vice versa, in the following sense:

Theorem 31. *(1) If (O, pr, π_1, π_2) and (O, pr, π_1', π_2') are both pairing schemes for X with Y, then $\pi_1 = \pi_1'$ and $\pi_2 = \pi_2'$.*

(2) If (O, pr, π_1, π_2) and (O, pr', π_1, π_2) are both pairing schemes for X with Y, then $pr = pr'$.

Proof. For (1), take any o among O. Suppose $o = pr(x, y)$ (there must be such x and y since pr is surjective). Hence, applying (a) to both schemes, $\pi_1 o = x = \pi_1' o$. Hence $\pi_1 = \pi_1'$. Similarly $\pi_2 = \pi_2'$.

For (2), take any x among X, and y among Y, and let $pr(x, y) = o$, so $\pi_1 o = x$ and $\pi_2 o = y$. Then by (b) applied to the second scheme, $pr'(\pi_1 o, \pi_2 o) = o$. Whence $pr'(x, y) = pr(x, y)$. □

Further, there is a sense in which all schemes for pairing X with Y are equivalent up to a unique isomorphism. More carefully,

Theorem 32. *If (O, pr, π_1, π_2) and $(O', pr', \pi_1', \pi_2')$ are both schemes for pairing X with Y, then there is a unique bijection $f\colon O \to O'$ which respects product structure, i.e. which is such that for all x, y, $pr'(x, y) = f(pr(x, y))$.*

Putting it another way, there is a unique bijection f such that, if we pair x with y using pr (in the first scheme), use f to send the resulting pair-object o to o', and then retrieve elements using π_1' and π_2' (from the second scheme), we get back to the original x and y.

Proof. Define $f\colon O \to O'$ by putting $f(o) = pr'(\pi_1 o, \pi_2 o)$. Then it is immediate that $f(pr(x, y)) = pr'(x, y)$.

To show that f is injective, suppose $f(o) = f(o^*)$. Then $pr'(\pi_1 o, \pi_2 o) = pr'(\pi_1 o^*, \pi_2 o^*)$. Apply π_1' to each side and then use principle (a), and it follows that $\pi_1 o = \pi_1 o^*$. And likewise $\pi_2 o = \pi_2 o^*$. Therefore $pr(\pi_1 o, \pi_2 o) = pr(\pi_1 o^*, \pi_2 o^*)$. Whence by condition (b), $o = o^*$.

To show that f is surjective, take any o' among O'. Then put $o = pr(\pi_1' o', \pi_2' o')$. By the definition of f, $f(o) = pr'(\pi_1 o, \pi_2 o)$; plugging the definition of o twice into the right hand side and simplifying using rules (a) and (b) confirms that $f(o) = o'$.

So f is a bijection with the right properties. And since any object among O is $pr(x, y)$ for some x, y, the requirement that $f(pr(x, y)) = pr'(x, y)$ fixes f uniquely. □

10.3 Defining products, almost categorially

(a) Here's another simple theorem about pairing schemes, with X, Y, O and associated typed variables as before:

Theorem 33. *Suppose the functions* $\pi_1 : O \to X$, $\pi_2 : O \to Y$ *are such that there is a* unique *two-place function* $pr : X, Y \to O$ *satisfying the condition*

(a) $\forall x \forall y(\pi_1(pr(x, y)) = x \wedge \pi_2(pr(x, y)) = y)$.

Then

(b) $\forall o\, pr(\pi_1(o), \pi_2(o)) = o$,

and hence (O, pr, π_1, π_2) *forms a scheme for pairing* X *and* Y.

Proof. We argue that the uniqueness of *pr* ensures that the function *pr* is surjective, and then that its surjectivity implies (b).

Suppose *pr* is not surjective. Then for some o, there is no x and y such that $pr(x, y) = o$. So $pr(\pi_1 o, \pi_2 o) = o' \neq o$.

Consider then the function pr' which agrees with *pr* on all inputs except that $pr'(\pi_1 o, \pi_2 o) = o$. For all cases other than $x = \pi_1 o, y = \pi_2 o$ we still have $\pi_1(pr'(x, y)) = x \wedge \pi_2(pr'(x, y)) = y$, and by construction for the remaining case $\pi_1(pr'(\pi_1 o, \pi_2 o)) = \pi_1 o \wedge \pi_2(pr'(\pi_1 o, \pi_2 o)) = \pi_2 o$. So condition (a) holds for pr', where $pr' \neq pr$. Contraposing, if *pr* *uniquely* satisfies the condition (a), it is surjective.

Because *pr* is surjective, every o among O is $pr(x, y)$ for some x, y. But by (a) $\pi_1 o = x \wedge \pi_2 o = y$, and hence $pr(\pi_1 o, \pi_2 o) = pr(x, y) = o$. Which proves (b). □

(b) Pairing up X with Y through a pairing scheme, then, gives us the pair-objects O. On the model of forming Cartesian products, we can therefore think of O as a serving as a *product* of X with Y (relative to that scheme).

But we don't want to identify the resulting product *simply* with the objects O, because it depends crucially on the rest of the pairing scheme whether O can play the right role. Our last theorem, however, makes the following an appropriate definition:

Definition 41. Given objects X and objects Y, then (O, π_1, π_2) form *a product of* X *with* Y, where O are some objects, and $\pi_1 : O \to X, \pi_2 : O \to Y$ are functions, so long as there is a *unique* two-place function $pr : X, Y \to O$ such that $\forall x \forall y(\pi_1(pr(x, y)) = x \wedge \pi_2(pr(x, y)) = y)$. △

Which is certainly a categorically flavoured definition, defining a product in terms of there being a unique map doing a certain job.

But there is a major obstacle to turning this into a kosher arrow-theoretic definition. Arrows have single sources (so in categories where arrows are functions, they are monadic functions); but the function *pr* here is essentially binary. What to do? The next chapter explains!

10.4 Logical pairing?

But first, a final pre-categorial remark about ordered pairs.

Ordinarily, we wouldn't think it makes any sense to say that the ordered pair of 4 and 2 is divisible by 9 or to say that the pair's square root is even: ordered pairs are surely not the right sort of thing to have divisors or to have square roots. But on our powers-of-primes pairing scheme, it is actually *true* that the pair $\langle 4, 2\rangle_P$ is divisible by 9, and *true* that its square root is even. These are unwanted side-effects of this pairing scheme, you might well suppose.

We can generalize. Typically, the pair-objects O in a scheme for pairing objects X with Y will have properties over and above being the target of a pairing function and the source of the matching unpairing functions; so the pair-objects will have properties that are extraneous to their role as ordered pairs.

Can there however also be – so to speak – a minimal pairing scheme, involving ordered pairs which are objects with no built-in properties other than those they logically *need* to have to be pairs?

Those coming from a certain logical background might perhaps be attracted by the following line of thought. The clauses in Defn. 40 are tantamount to interlocking natural-deduction-style introduction and elimination rules for the pairing function (given the projection functions):

$$\frac{x = \pi_1(o) \qquad y = \pi_2(o)}{pr(x,y) = o}\ (pr\text{-I}) \qquad \frac{pr(x,y) = o}{x = \pi_1(o)} \qquad \frac{pr(x,y) = o}{y = \pi_2(o)}\ (pr\text{-E})$$

Equally – a mini-exercise! – we can see that the same clauses in Defn. 40 are tantamount to interlocking introduction and elimination rules for the projection functions (given the pairing function). Thus, for π_1 we have:

$$\frac{pr(x,y) = o}{x = \pi_1(o)}\ (\pi_1\text{-I}) \qquad\qquad \frac{x = \pi_1(o) \qquad \begin{matrix}\overline{pr(x,t) = o}^{\,(1)} \\ \vdots \\ C\end{matrix}}{C}\,{}^{(1)}\ (\pi_1\text{-E})$$

where, Gentzen-style, t is parametric in the subproof and doesn't appear in its conclusion C. There is of course a similar pair of rules for introducing and eliminating π_2. Now, suppose we are inclined to take introduction and elimination rules to fix the content of logical operators like connectives and quantifiers. Then can we similarly take these new introduction and elimination rules for the ingredients of a pairing scheme as sufficing both to introduce purely logical pair-objects and to fix the content of their associated pairing and unpairing functions?

Perhaps so, but we can't explore the issues here.[2] And it seems in keeping with the general tenor of a categorial approach not to worry about such issues either, and to insist that – as we said at the outset – all that really matters about a pairing scheme is that its pair-objects are suitably related by interlocking pairing

[2]For some relevant discussion, see Tennant (2009).

and unpairing functions to the objects being paired up. Rightly or wrongly, from a categorial point of view, the pairing scheme just needs to have the required structure to *work*, while the intrinsic character of its objects (richer or more minimally logical) doesn't matter.

11 Categorial products introduced

In the previous chapter, we arrived at Defn. 41, which characterized a product of some objects X and some objects Y as consisting in some objects O equipped with a pair of projection functions $\pi_1 \colon O \to X$ and $\pi_2 \colon O \to Y$ satisfying a certain condition (C): there must be a *unique* binary function $pr \colon X, Y \to O$ such that – for any x among X and any y among Y –

$$\pi_1(pr(x, y)) = x \ \wedge \ \pi_2(pr(x, y)) = y.$$

Which, as we pointed out, is a definition *almost* in the style of category theory, except for the fact that pr is binary while an arrow in a category is always unary, with just one of the category's objects as its source.

So how can we turn our very natural story about products into a properly categorial account?

11.1 Products defined categorially

(a) Let's take things in stages. So:

(i) First note that we can reformulate condition (C) above like this: (C′) for any x among X and y among Y, there is a *unique* corresponding o such that $\pi_1(o) = x \ \wedge \ \pi_2(o) = y$.

Why so? Because if (C′) holds, then the map from each x and y to the corresponding o is (C)'s required pairing function pr, which will be unique because otherwise (for at least some x and y) there would after all be alternative candidates for o.

Now suppose for a moment that we are working in a well-pointed category like Set, where 'elements' in the sense of Defn. 37 do behave sufficiently like how elements intuitively should behave. Then

(ii) In this setting, instead of talking of an object x (one of X) and object y (one of Y), we can talk instead of two arrows $\vec{x} \colon 1 \to X$ and $\vec{y} \colon 1 \to Y$. Again, instead of talking of some object o which is one of the objects O, we can talk of an arrow $\vec{o} \colon 1 \to O$.

(iii) So, instead of saying as in condition (C′) that $\pi_1(o) = x \ \wedge \ \pi_2(o) = y$, we could then equivalently say $\pi_1 \circ \vec{o} = \vec{x} \wedge \ \pi_2 \circ \vec{o} = \vec{y}$.

(iv) And *that* is trivially equivalent to saying that the following diagram commutes:

$$\begin{tikzcd}
 & 1 \arrow[dl, "\vec{x}"'] \arrow[d, "\vec{o}"] \arrow[dr, "\vec{y}"] & \\
X & O \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & Y
\end{tikzcd}$$

(v) Hence we can recast our Defn. 41 by saying that the objects O equipped with projection arrows $\pi_1 \colon O \to X$ and $\pi_2 \colon O \to Y$ form a product for X and Y in our category just if for each $\vec{x} \colon 1 \to X$ and $\vec{y} \colon 1 \to Y$ there is a *unique* arrow $\vec{o} \colon 1 \to O$ which makes our diagram commute.

(b) So far, so good. But this will only give us what we want in well-pointed categories with 'enough' elements-as-arrows; consider what would happen if we were working e.g. in the category Grp. However, we do now know a potential way of generalizing claims to non-well-pointed categories: just replace talk about point elements with talk of generalized elements (see §9.4). Which motivates the following crucial definition:

Definition 42. In any category $\mathscr{C}$, *a (binary) product* (O, π_1, π_2) *for the objects* X *with* Y is an object O together with projection arrows $\pi_1 \colon O \to X, \pi_2 \colon O \to Y$, such that for any object S and arrows $f_1 \colon S \to X$ and $f_2 \colon S \to Y$ there is always a unique 'mediating' arrow $u \colon S \to O$ such that the following diagram commutes:

$$\begin{tikzcd}
 & S \arrow[dl, "f_1"'] \arrow[d, dashed, "u"] \arrow[dr, "f_2"] & \\
X & O \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & Y
\end{tikzcd}$$

△

Note, by the way, that we now adopt the following common convention: in a commutative diagram, we use a dashed arrow $\dashrightarrow$ to indicate an arrow which is to be uniquely fixed by the requirement that the diagram commutes.

(c) Now it is true that you can just stare hard at Defn. 42 if it is served up neat, without preceding ceremony, and 'see' that it is the sort of thing we need in a categorial context if O with its projection arrows is to do the work of a product of X with Y. Arm-waving more than a bit: the fact that our diagram always commutes for some u tells us that O encodes *enough* for us always to be able extract again data relating to X and Y; and the fact that the mediating arrow u is unique tells that O does the encoding without redundancy, it is (so to speak) *no more* than we need.

But it has been well worth taking the longer route our destination, and showing that our definition does arise quite naturally by generalizing elementary facts about what we want from a pairing scheme.

11.2 Examples

Let's now have some examples of products in categories.

(1) In Set, as you would certainly hope, the usual Cartesian product treated as the set $X \times Y$ of Kuratowksi pairs $\langle x, y\rangle$ of objects from X and Y, together with the obvious projection functions $\langle x, y\rangle \xmapsto{\pi_1} x$ and $\langle x, y\rangle \xmapsto{\pi_2} y$, form a binary product.

Let's just confirm this. Suppose we are given any set S and functions $f_1 \colon S \to X$ and $f_2 \colon S \to Y$. If, for $s \in S$, we put $u(s) = \langle f_1(s), f_2(s)\rangle$, the diagram evidently commutes. Now trivially, for any pair $p \in X \times Y$, $p = \langle \pi_1 p, \pi_2 p\rangle$. Hence if $u' \colon S \to X \times Y$ is another candidate for completing the diagram, $u'(s)$ is a pair, so $u'(s) = \langle \pi_1 u'(s), \pi_2 u'(s)\rangle = \langle f_1(s), f_2(s)\rangle = u(s)$. Therefore u is unique.

Motivated by this paradigm case, we will often use the notation $X \times Y$ for the object in a binary product of X with Y, thus $(X \times Y, \pi_1, \pi_2)$.

Continuing our examples:

(2) Recall Defn. 3 from Chapter 2. For groups in the category Grp – which we are taking as living in a universe of sets – we can use the same Kuratowksi construction for pairs. And then, using that pairing scheme, the direct product of the groups $\mathcal{G} = (G, *, e)$ and $\mathcal{G}' = (G', *', e')$ will be the group $\mathcal{G} \times \mathcal{G}' = (G \times G', \star, d)$, where $\star$ is as before defined component-wise (so $\langle x, x'\rangle \star \langle y, y'\rangle = \langle x * y, x' *' y'\rangle$) and $d = \langle e, e'\rangle$.

The projection function which sends each $\langle x, x'\rangle$ to x is easily checked to be a group homomorphism $\pi_1 \colon \mathcal{G} \times \mathcal{G}' \to \mathcal{G}$. We define π_2 similarly, of course. So we now need to check that $\mathcal{G} \times \mathcal{G}'$ equipped with π_1 and π_2 is indeed a categorial product. That's easy, following the same line of argument as in example (1).

(3) Similarly a product of topological spaces defined in the usual way, equipped with the trivial projection functions recovering the original spaces, is a categorial product of topological spaces in Top.

(4) Now revisit the category $\mathsf{Prop}_{\mathscr{L}}$ introduced in §4.5, (C16). Its objects are propositions, closed wffs of a given first-order language $\mathscr{L}$, and there is a unique arrow from X to Y iff $X \vDash Y$, i.e. iff X semantically entails Y.

In this case, consider the *logical* product of X with Y, i.e. their conjunction $X \wedge Y$. Take this together with the obvious projections $X \wedge Y \to X$, $X \wedge Y \to Y$ (these are arrows because they encode entailments!). Then this gives us a *categorial* product of X with Y in $\mathsf{Prop}_{\mathscr{L}}$.

Why? Take any arrows in $\mathsf{Prop}_{\mathscr{L}}$ from A to X and A to Y – i.e. assume $A \vDash X$ and $A \vDash Y$. Then of course we have $A \vDash X \wedge Y$, and we get the required and necessarily unique mediating arrow from A to $X \wedge Y$!

So far, then, so good: categorial products are lining up nicely with products intuitively understood.

Let's have one more example to be going on with:

(5) Take pre-ordered objects $(P, \preccurlyeq)$ considered as a category $\mathscr{P}$ as in §4.4, (C4). So, recall, there is an arrow $p \to q$ in the category iff $p \preccurlyeq q$.

Then what is a product of p and q in $\mathscr{P}$? It will be an object $p \times q$ with projection arrows to p and q such that for any pair of arrows from s to p and s to q there is a unique arrow from s to $p \times q$ making the diagram commute:

$$\begin{tikzcd} & s \arrow[dl] \arrow[d, dashed] \arrow[dr] & \\ p & p \times q \arrow[l] \arrow[r] & q \end{tikzcd}$$

Which means that $p \times q \preccurlyeq p$ and $p \times q \preccurlyeq q$, and whenever $s \preccurlyeq p$ and $s \preccurlyeq q$, we have $s \preccurlyeq p \times q$. Which means in turn that the object in a categorial product of p and q in $\mathscr{P}$ must be their 'meet' or greatest lower bound.

A simple moral from this last example: since pairs of objects in a pre-ordering need not in general have greatest lower bounds, this shows that a category in general need not have products (other than some trivial ones, as we shall see).

11.3 Products as terminal objects

(a) Defn. 42 defines the notion of *a* product of objects X and Y in a category. But we can in fact loosely talk of *the* categorial product of two objects – because products are unique up to unique isomorphism. We will prove that in the next section. But it is helpful and illuminating first to introduce a slightly different, though equivalent, way of defining products.

We need an auxiliary notion. Let's say

Definition 43. A *wedge to X and Y* (in category $\mathscr{C}$) is an object S and a pair of arrows $f_1 \colon S \to X$, $f_2 \colon S \to Y$. △

Then a wedge $\begin{tikzcd}[row sep=small] & X \\ O \arrow[ur, "\pi_1"] \arrow[dr, "\pi_2"'] & \\ & Y \end{tikzcd}$ is a product of X with Y iff, for any wedge $\begin{tikzcd}[row sep=small] & X \\ S \arrow[ur, "f_1"] \arrow[dr, "f_2"'] & \\ & Y \end{tikzcd}$ to X and Y, there exists a unique arrow u such that the following diagram commutes:

$$\begin{tikzcd} & & X \\ S \arrow[urr, bend left, "f_1"] \arrow[drr, bend right, "f_2"'] \arrow[r, dashed, "u"] & O \arrow[ur, "\pi_1"] \arrow[dr, "\pi_2"'] & \\ & & Y \end{tikzcd}$$

That's just our previous definition put in different terms, with the diagram rotated! No mystery here.

We will say in such a case that f_1 'factors' as $\pi_1 \circ u$ and f_2 as $\pi_2 \circ u$, and hence the whole wedge from S into X and Y *(uniquely) factors through* the product wedge via the mediating arrow u.

(b) Now for another definition involving wedges. Recall, the category $\mathscr{C}/X$, the slice category of $\mathscr{C}$ over X, has as it objects the $\mathscr{C}$-arrows of the form $f\colon O \to X$ (see §6.3). We are now going to introduce a new category $\mathscr{C}/XY$, the wedge category of $\mathscr{C}$ over X and Y. Its objects are going to be pairs of $\mathscr{C}$-arrows of the form $f\colon O \to X$, $g\colon O \to Y$, i.e. wedges to X and Y. Looking at the definition of slice categories, the corresponding definition for wedge categories should be predictable (pause to work it out!).

Thus, we will say:

Definition 44. Given a category $\mathscr{C}$ and $\mathscr{C}$-objects X, Y, then the *wedge category* $\mathscr{C}/XY$ has the following data.

(i) Its objects are all the wedges (O, f_1, f_2) from O to X, Y.

(ii) And an arrow from (O, f_1, f_2) to (O', f_1', f_2') is a $\mathscr{C}$-arrow $g\colon O \to O'$ such that the two resulting triangles commute: i.e. $f_1 = f_1' \circ g$, $f_2 = f_2' \circ g$.

The identity arrow on (O, f_1, f_2) is 1_O, and the composition of arrows in $\mathscr{C}/XY$ is the same as their composition as arrows of $\mathscr{C}$. △

The definition of the arrows in $\mathscr{C}/XY$ should make sense if you think again about slice categories and/or if you meditate on the following diagram!

$$\begin{tikzcd}
 & O \arrow[r, "g"] \arrow[dl, "f_1"'] \arrow[drr, "f_2"] & O' \arrow[dll, "f_1'"'] \arrow[dr, "f_2'"] & \\
X & & & Y
\end{tikzcd}$$

And with the given definition for arrows, it is easily confirmed that $\mathscr{C}/XY$ really is a category.

(c) Finally, our new notion of the derived category $\mathscr{C}/XY$ to hand, we can revisit our previous definition of a product. A moment's reflection shows that it is straightforwardly equivalent to

Definition 45. A product of X with Y in $\mathscr{C}$ is a terminal object of the wedge category $\mathscr{C}/XY$. △

Which is rather cute!

11.4 Uniqueness up to unique isomorphism

As noted, products need not exist for arbitrary objects X and Y in a given category $\mathscr{C}$; and when they exist, they need not be strictly unique. However, when they do exist, then – as announced – they *are* 'unique up to unique isomorphism' (compare Theorem 32). That is to say,

Theorem 34. *If both (O, π_1, π_2) and (O', π_1', π_2') are products for X with Y in the category $\mathscr{C}$, then there is a unique isomorphism $f\colon O \xrightarrow{\sim} O'$ commuting with the projection arrows (i.e. such that $\pi_1' \circ f = \pi_1$ and $\pi_2' \circ f = \pi_2$).*

Note the statement of the theorem carefully. It is *not* being baldly claimed that there is a unique isomorphism between any objects O and O' which are components of products for some given X, Y. That's false. For a very simple example, in Set, take the standard product object $X \times X$ comprising Kuratowski pairs: there are evidently two isomorphisms between it and itself, given by the maps $\langle x, x' \rangle \mapsto \langle x, x' \rangle$, and $\langle x, x' \rangle \mapsto \langle x', x \rangle$. The claim is, to repeat, that there is a unique isomorphism between the objects of any two products for X with Y *which commutes with the products' respective projection arrows.*

We are now going to prove our theorem twice over: ploddingly from first principles, and then zippily using our redefinition of products as terminal objects.

Plodding proof. Since (O, π_1, π_2) is a product for X and Y in $\mathscr{C}$, every wedge to X and Y factors uniquely through it, including itself. In other words, there is a unique u such that this diagram commutes:

$$\begin{tikzcd}
 & O \arrow[dl, "\pi_1"'] \arrow[d, dashed, "u"] \arrow[dr, "\pi_2"] & \\
X & O \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & Y
\end{tikzcd}$$

But evidently putting 1_O for the central arrow trivially makes the diagram commute. So by the uniqueness requirement we know that

(i) Given a product (O, π_1, π_2) and an arrow $u\colon O \to O$, if $\pi_1 \circ u = \pi_1$ and $\pi_2 \circ u = \pi_2$, then $u = 1_O$.

Now, assuming (O', π_1', π_2') is also a product, (O, π_1, π_2) has to uniquely factor through it:

$$\begin{tikzcd}
 & O \arrow[dl, "\pi_1"'] \arrow[d, dashed, "f"] \arrow[dr, "\pi_2"] & \\
X & O' \arrow[l, "\pi_1'"'] \arrow[r, "\pi_2'"] & Y
\end{tikzcd}$$

In other words, there is a unique $f\colon O \to O'$ commuting with the projection arrows, i.e. such that

(ii) $\pi_1' \circ f = \pi_1$ and $\pi_2' \circ f = \pi_2$.

And since (O, π_1, π_2) is also a product, (O', π_1', π_2') has to uniquely factor through *it*. That is to say, there is a unique $g\colon O' \to O$ such that

(iii) $\pi_1 \circ g = \pi'_1$ and $\pi_2 \circ g = \pi'_2$.

Whence,

(iv) $\pi_1 \circ g \circ f = \pi'_1 \circ f = \pi_1$ and $\pi_2 \circ g \circ f = \pi_2$.

From which it follows – given our initial observation (i) – that

(v) $g \circ f = 1_O$

The situation with the products is symmetric so we also have

(vi) $f \circ g = 1_{O'}$

Hence f has a two-sided inverse, i.e. is an isomorphism. □

However, you'll recognize the key proof idea here is akin to the one we used in proving that initial/terminal objects are unique up to unique isomorphism. And we indeed can just appeal to that earlier result:

Proof using the alternative definition of products. (O, π_1, π_2) and (O', π'_1, π'_2) are both terminal objects in the wedge category $\mathscr{C}/XY$. So by Theorem 26 there is a unique $\mathscr{C}/XY$-isomorphism f between them. But, by definition, this has to be a $\mathscr{C}$-arrow $f\colon O \to O'$ commuting with the projection arrows. And it is immediate that an isomorphism in $\mathscr{C}/XY$ is also an isomorphism in $\mathscr{C}$. □

11.5 Some more properties of products

(a) Let's next check that binary products have the following trio of nice properties:

Theorem 35. *In a category which has a terminal object 1,*

(1) Products $1 \times X$ and $X \times 1$ exist, and $1 \times X \cong X \cong X \times 1$.

In a category where the relevant products exist,

(2) $X \times Y \cong Y \times X$,
(3) $X \times (Y \times Z) \cong (X \times Y) \times Z$.

Here, when we talk of a product object in a category, take this to be equipped with suitable projection arrows. Results (2) and (3) are intuitively desirable (why?); and (1) shows that our notation for terminal objects is rather happily chosen.

Proof for (1). Following the notational convention of Defn. 34, we will use $!_X$ for the unique arrow from X to the terminal object 1. Consider then the wedge (V) $1 \xleftarrow{!_X} X \xrightarrow{1_X} X$, and take any other wedge to 1 and X, namely $1 \xleftarrow{!_Y} Y \xrightarrow{f} X$. The following diagram trivially commutes:

$$\begin{tikzcd} & Y \arrow[dl, "!_Y"'] \arrow[d, "f"] \arrow[dr, "f"] & \\ 1 & X \arrow[l, "!_X"] \arrow[r, "1_X"'] & X \end{tikzcd}$$

(the triangle on the left commutes because there can only be one arrow from Y to 1 which forces $!_X \circ f = !_Y$). And obviously f is the only vertical arrow which makes this commute. Hence $(X, !_X, 1_X)$ satisfies the conditions for being a product of 1 with X. So, by Theorem 34, given any product $(1 \times X, \pi_1, \pi_2)$, we have $1 \times X \cong X$. Exactly similarly, $X \times 1 \cong X$. □

Proof for (2). Suppose $(X \times Y, \pi_1, \pi_2)$ is a product of X with Y; then $(X \times Y, \pi_2, \pi_1)$ will obviously serve as a product of Y with X. Hence, by Theorem 34 again, there is an isomorphism between the object in that product and the object $Y \times X$ of any other product of Y with X. □

Proof for (3) postponed. It is a just-about-useful reality check to prove this by appeal to our initial definition of a product, using brute force. Masochists are invited to try! But we'll meet a slicker proof in §13.1. □

(b) Question: In parallel to the result (1), do we similarly have $0 \times X \cong 0$ in categories with an initial object and the relevant product? Answer: Not always.

Theorem 36. *There are categories where the product $0 \times X$ or $X \times 0$ always exists but is not generally isomorphic to* 0.

Proof. Take a category which has a null object, i.e. where $0 = 1$. Then since every product $1 \times X$ exists, so does $0 \times X$. Now suppose $0 \times X \cong 0$. Then we would have $X \cong 1 \times X = 0 \times X \cong 0$.

Take then a category like Grp which has a null object (and so all products $0 \times X$ exist), but which also has other non-isomorphic objects, so we don't always have $X \cong 0$. It follows that in Grp it can't always be the case that $0 \times X \cong 0$. □

11.6 A notation for mediating arrows

Definition 46. Suppose (O, π_1, π_2) is a binary product for the objects X with Y, and the wedge $X \xleftarrow{f_1} S \xrightarrow{f_2} Y$ factors through it. We will now notate the unique mediating arrow $\langle f_1, f_2 \rangle \colon S \to O$. △

In other words, we can write the commuting diagram like this:

$$\begin{tikzcd} & S \arrow[dl, "f_1"'] \arrow[d, dashed, "{\langle f_1, f_2 \rangle}"] \arrow[dr, "f_2"] & \\ X & O \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & Y \end{tikzcd}$$

We should check that this product-style notation $\langle f_1, f_2 \rangle$ for mediating arrows doesn't mislead. But indeed we have:

Theorem 37. *If* $\langle f_1, f_2 \rangle = \langle g_1, g_2 \rangle$, *then* $f_1 = g_1$ *and* $f_2 = g_2$.

Proof. Evidently, $f_1 = \pi_1 \circ \langle f_1, f_2 \rangle = \pi_1 \circ \langle g_1, g_2 \rangle = g_1$, and similarly $f_2 = g_2$. □

So we will adopt this new notation when helpful.

11.7 'Universal mapping properties'

Let's pause for a moment to make two quick points.

(a) We have defined a binary product for X with Y categorially as a special sort of wedge to X and Y. And what makes some wedge a product for X with Y is that it has a certain universal property – i.e. *any* other wedge to X and Y factors uniquely through a product wedge via a unique arrow.

Since arrows are typically functions or maps, we can therefore say that products are defined by a *universal mapping property.* We've already met other examples of universal mapping properties: terminal and initial objects are defined by how any other object has a unique map/arrow to or from them. We will meet lots more examples.

It is perhaps too soon to attempt a formal definition of what it is to be defined by a universal mapping property. For the moment, then, take the notion as an informal gesture towards a common pattern of categorial definition which we start to recognize when we come across it.

(b) We noted at the beginning of this chapter that arrows in categories are unary, with single sources. We don't have true binary maps of the type $f : X, Y \to Z$. We now know how to get round this issue in a category with appropriate products – we can use arrows $f : X \times Y \to Z$.

But we won't say more about this device now; let's wait until we start putting it to real work later, beginning in Chapter 18.

11.8 Coproducts

(a) We are going now to discuss the duals of products. But first, we should note a common terminological device:

Definition 47. Duals of categorially defined widgets are very often called *co-widgets.* Thus a *co-widget* of the category $\mathcal{C}$ is a widget of $\mathcal{C}^{op}$. △

For just one example, we have met co-slice categories, the duals of slice categories. True, there is a limit to this sort of thing – no one, as far as I know, talks e.g. of 'co-monomorphisms' (instead of 'epimorphisms'). But still, the general convention is used widely. In particular, it is absolutely standard to talk of the duals of products as 'co-products' – though in this case, as in some others, the hyphen is usually dropped.

(b) The definition of a coproduct is immediately obtained, then, by reversing all the arrows in our definition of products. Thus:

Definition 48. In any category $\mathscr{C}$, *a (binary) coproduct* (O, ι_1, ι_2) *for the objects* X *with* Y is an object O together with two 'injection' arrows $\iota_1 \colon X \to O, \iota_2 \colon Y \to O$, such that for any object S and arrows $f_1 \colon X \to S$ and $f_2 \colon Y \to S$ there is always a unique 'mediating' arrow $v \colon O \to S$ such that the following diagram commutes:

$$\begin{tikzcd} & S & \\ X \arrow[ur, "f_1"] \arrow[r, "\iota_1"'] & O \arrow[u, dashed, "v"] & Y \arrow[ul, "f_2"'] \arrow[l, "\iota_2"] \end{tikzcd}$$

The object O in a coproduct for X with Y can be notated '$X \oplus Y$' or '$X \amalg Y$'. △

Note, however, that the 'injections' in this sense need not be injective or even monic.

Let's say that objects and arrows arranged as $X \xrightarrow[\iota_1]{} O \xleftarrow[\iota_2]{} Y$ form a *corner* (or we could say 'co-wedge'!) from X and Y with vertex O. Then a coproduct of X with Y can be thought of as a corner from X and Y which factors through any other corner from X and Y via a unique map between the vertices of the corners.

We could now go on to define a category of corners from X and Y on the model of a category of wedges to X and Y, and then redefine a coproduct of X with Y as an initial object of this category. It is a useful reality check to work through the details.

(c) Let's have some examples of coproducts. Start with easy cases:

(1) In Set, disjoint unions are instances of coproducts.

Given sets X and Y, let $X \oplus Y$ be the set with members $\langle x, 0\rangle$ for $x \in X$ and $\langle y, 1\rangle$ for $y \in Y$. And let the injection arrow $\iota_1 \colon X \to X \oplus Y$ be the function $x \mapsto \langle x, 0\rangle$, and similarly let $\iota_2 \colon Y \to X \oplus Y$ be the function $y \mapsto \langle y, 1\rangle$. Then $(X \oplus Y, \iota_1, \iota_2)$ is a coproduct for X with Y.

To show this, take any object S and arrows $f_1 \colon X \to S$ and $f_2 \colon Y \to S$, and then define the function $v \colon X \oplus Y \to S$ as sending an element $\langle x, 0\rangle$ to $f_1(x)$ and an element $\langle y, 1\rangle$ to $f_2(y)$.

By construction, this will make both triangles commute in the diagram in the definition above.

Moreover, if v' is another candidate for completing the diagram, then $v'(\langle x, 0\rangle) = v' \circ \iota_1(x) = f_1(x) = v(\langle x, 0\rangle)$, and likewise $v'(\langle y, 1\rangle) = v(\langle y, 1\rangle)$, whence $v' = v$, which gives us the necessary uniqueness.

(2) In $\mathsf{Prop}_{\mathcal{L}}$ (which we met in §11.2) the disjunction $X \vee Y$ (with the obvious injections $X \to X \vee Y$, $Y \to X \vee Y$) is a coproduct of X with Y.

(3) In the case of pre-ordered objects $(P, \preccurlyeq)$ considered as a category then a coproduct of p and q would be an object c such that $p \preccurlyeq c, q \preccurlyeq c$ and such that for any object d such that $p \preccurlyeq d, q \preccurlyeq d$ there is a unique arrow from c to d, i.e. $c \preccurlyeq d$. Which means that the coproduct of p and q, if it exists, must be their least upper bound (equipped with the obvious two arrows).

(d) In some cases, however, the story about coproducts gets markedly more complicated. Just for enthusiasts I'll mention one more example here: but the details really aren't going to matter later, so do by all means skip:

(4) In the category Grp, coproducts are (isomorphic to) the so-called 'free products' of groups.

Take the groups $\mathcal{G} = (G, \cdot, e)$, $\mathcal{H} = (H, \odot, d)$. Assume that we have doctored the groups if necessary so that now $e = d$ while ensuring the objects G and H are otherwise disjoint. Form all the finite 'reduced words' $G \star H$ you get by concatenating objects from G and/or H, and then multiplying out neighbouring G-objects by $\cdot$ and neighbouring H-objects by $\odot$ as far as you can. Equip these objects $G \star H$ with the operation $\diamond$ of concatenation-of-words-followed-by-reduction. Then $\mathcal{G} \star \mathcal{H} = (G \star H, \diamond, e)$ is a group – the so-called free product of the two groups $\mathcal{G}$ and $\mathcal{H}$ – and there are obvious 'injection' group homomorphisms $\iota_1 \colon \mathcal{G} \to \mathcal{G} \star \mathcal{H}$, $\iota_2 \colon \mathcal{H} \to \mathcal{G} \star \mathcal{H}$.

Claim: $(\mathcal{G} \star \mathcal{H}, \iota_1, \iota_2)$ is a coproduct for the groups $\mathcal{G}$ and $\mathcal{H}$. That is to say, for any group $\mathcal{K} = (K, *, k)$ and group homomorphisms $f_1 \colon \mathcal{G} \to \mathcal{K}$, $f_2 \colon \mathcal{H} \to \mathcal{K}$, there is a unique v such that this commutes:

$$\begin{tikzcd}
 & \mathcal{K} & \\
\mathcal{G} \arrow[ur, "f_1"] \arrow[r, "\iota_1"'] & \mathcal{G} \star \mathcal{H} \arrow[u, dashed, "v"] & \mathcal{H} \arrow[ul, "f_2"'] \arrow[l, "\iota_2"]
\end{tikzcd}$$

Proof. Put $v \colon \mathcal{G} \star \mathcal{H} \to \mathcal{K}$ to be the group homomorphism that sends a word such as $g_1 h_1 g_2 h_2 \cdots g_r$ (for g_i among G, and h_i among H) to $f_1(g_1) * f_2(h_1) * f_1(g_2) * f_2(h_2) * \cdots * f_1(g_r)$. By construction, $v \circ \iota_1 = f_1$, $v \circ \iota_2 = f_2$. So that makes the diagram commute.

Let v' be any other candidate group homomorphism to make the diagram commute. Then, to take a simple example, consider gh (one of the objects $G \star H$). Then $v'(gh) = v'(g) * v'(h) = v'(i_1(g)) * v'(i_2(h)) = f_1(g) * f_2(g) = v(i_1(g)) * v(i_1(g)) = v(i_1(g) * i_1(g)) = v(gh)$. Similarly $v'(hg) = v(hg)$. So by induction over the length of words w we can go on to show quite generally $v'(w) = v(w)$. Hence, as required, v is unique. $\square$

12 Binary products explored

This chapter continues to explore binary products. Really, we are just reading into the record a handful of theorems which will be useful later, illustrating some characteristic proof strategies as we go. So for now you might very well want to skip on to the following chapter which discusses products of more than two objects.

Of course, everything in these two chapters will dualize: but let's leave it as an exercise to supply all the corresponding theorems about coproducts.

12.1 Two more simple results

(a) To reduce clutter, we drop subscripts from unique arrows to terminal objects (so write simply '!' rather than '$1_{1\times X}$'), and drop subscripts from identity arrows (so write simply '1' rather than '$1_{1\times X}$'). It is a nice reality check to mentally replace subscripts in the following.

Then our first result is this:

Theorem 38. *If* $1 \xleftarrow{!} 1 \times X \xrightarrow{i} X$ *is a product, then* i *is an isomorphism. Similarly for the mirror image result.*

We know from Theorem 35 that there is an isomorphism between $1 \times X$ and X; but that doesn't rule out other arrows between them. So it takes another argument to show that in any product wedge (W) $1 \xleftarrow{!} 1 \times X \xrightarrow{i} X$, i has to be an isomorphism.

Proof. Consider, then, the following diagram:

$$\begin{tikzcd}
 & 1 \times X \arrow[ddl, "!"'] \arrow[d, "i"] \arrow[ddr, "i"] & \\
 & X \arrow[dl, "!"'] \arrow[d, "u"] \arrow[dr, "1"] & \\
1 & 1 \times X \arrow[l, "!"] \arrow[r, "i"'] & X
\end{tikzcd}$$

This commutes. The wedge (V) $1 \xleftarrow{!} X \xrightarrow{1} X$ must factor through the product (W) via a unique mediating arrow u, and then $i \circ u = 1$.

Similarly (W) factors through (V) as shown.[1] But putting the triangles together means that (W) factors through (W) via the (unique) mediating arrow $u \circ i$. But since (W) also factors through itself via 1, it follows that $u \circ i = 1$.

Having inverses on both sides, i is therefore an isomorphism. □

(b) And now second, again for future use, we should remark on a non-theorem.

Suppose we have a pair of parallel composite arrows built up using the same projection arrow like this: $X \times Y \xrightarrow{\pi_1} X \overset{f}{\underset{g}{\rightrightarrows}} X'$. In Set, the projection arrow here just 'throws away' the second component of pairs living in $X \times Y$, and all the real action then happens on X: so if $f \circ \pi_1 = g \circ \pi_1$, we should also have $f = g$. Generalizing, we might then suppose that, in any category, projection arrows in products are always right-cancellable, i.e. are epic.

This is wrong. Here's a brute-force counterexample. Consider the mini category with just four objects together with the following diagrammed arrows (labelled suggestively but noncommittally), plus all identity arrows, and the necessary two composites:

$$
\begin{tikzcd}
X' & X \arrow[l, shift right, "f"'] \arrow[l, shift left, "g"] & V \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & Y
\end{tikzcd}
$$

If that is all the data we have to go on, we can consistently stipulate that in this mini-category $f \neq g$ but $f \circ \pi_1 = g \circ \pi_1$.

Now, there is only one wedge of the form $X \longleftarrow ? \longrightarrow Y$, so trivially all wedges of that shape uniquely factor through it. In other words, the wedge $X \xleftarrow{\pi_1} V \xrightarrow{\pi_2} Y$ is trivially a product and π_1 is indeed a projection arrow. But by construction it isn't epic.

12.2 Diagonal arrows

Definition 49. Suppose we are working in a category with the relevant products. Then the wedge $X \xleftarrow{1} X \xrightarrow{1} X$ must factor uniquely through the product $X \times X$ via an arrow $\delta_X : X \to X \times X$. That unique arrow δ_X, i.e. $\langle 1_X, 1_X \rangle$, is *the diagonal arrow* from X to $X \times X$. △

In Set, thinking of $X \times X$ in the usual way, δ_X sends an element $x \in X$ to $\langle x, x \rangle$ (imagine elements $\langle x, x \rangle$ lying down the diagonal of a two-dimensional array of pairs $\langle x, y \rangle$: hence the label 'diagonal' and the notation δ).

Theorem 39. *Given an arrow $q \colon S \to X$, $\delta_X \circ q = \langle q, q \rangle$.*

Proof. Consider the following diagram:

[1] Reality check: The top left triangle commutes, i.e. $!_X \circ i = !_{1 \times X}$ because arrows to the same terminal object are unique.

$$\begin{tikzcd}
 & S \arrow[dl, "q"'] \arrow[d, "q"] \arrow[dr, "q"] \arrow[dd, dashed, bend left, "\langle q,q\rangle"] & \\
 & X \arrow[dl, "1"] \arrow[d, "\delta_X"] \arrow[dr, "1"'] & \\
X & X \times X \arrow[l, "\pi_1"] \arrow[r, "\pi_2"'] & X
\end{tikzcd}$$

The inner triangles commute, hence $\delta_X \circ q$ is a mediating arrow factoring the wedge $X \xleftarrow{q} S \xrightarrow{q} X$ through the product $X \times X$. But by definition, the unique mediating arrow which does that is $\langle q, q \rangle$. □

Theorem 40. *Assuming $\langle f, g \rangle$ and e compose, $\langle f, g \rangle \circ e = \langle f \circ e, g \circ e \rangle$.*

Proof. Another, rather similar, diagram gives the proof:

$$\begin{tikzcd}
 & R \arrow[dl, "f\circ e"'] \arrow[d, "e"] \arrow[dr, "g\circ e"] \arrow[dd, dashed, bend left, "\langle f\circ e, g\circ e\rangle"] & \\
 & S \arrow[dl, "f"] \arrow[d, "\langle f,g\rangle"'] \arrow[dr, "g"'] & \\
X & X \times Y \arrow[l, "\pi_1"] \arrow[r, "\pi_2"'] & Y
\end{tikzcd}$$

Again the inner triangles commute, hence $\langle f, g \rangle \circ e$ is a mediating arrow factoring the wedge with apex R through the product $X \times Y$. But by definition, the unique mediating arrow is $\langle f \circ e, g \circ e \rangle$. □

12.3 Maps between two products

(a) Suppose we have two arrows $f \colon X \to X'$, $g \colon Y \to Y'$. Then we might want to characterize an arrow between products, $f \times g \colon X \times Y \to X' \times Y'$, which works component-wise – i.e., putting it informally, the idea is that $f \times g$ sends the product of elements x and y to the product of $f(x)$ and $g(y)$.

In more categorial terms, we require $f \times g$ to be such that the following diagram commutes:

$$\begin{tikzcd}
X \arrow[d, "f"'] & X \times Y \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] \arrow[d, "f\times g"] & Y \arrow[d, "g"] \\
X' & X' \times Y' \arrow[l, "\pi_1'"'] \arrow[r, "\pi_2'"] & Y'
\end{tikzcd}$$

Note, however, that the vertical arrow is then a mediating arrow from the wedge $X' \xleftarrow{f\circ\pi_1} X \times Y \xrightarrow{g\circ\pi_2} Y'$ through the product $X' \times Y'$. Therefore $f \times g$ is indeed fixed uniquely by the requirement that that diagram commutes, and must equal $\langle f \circ \pi_1, g \circ \pi_2\rangle$. This shows that the following definition is in good order:

Definition 50. Given the arrows $f\colon X \to X'$, $g\colon Y \to Y'$, and the products $(X \times Y, \pi_1, \pi_2)$ and $(X' \times Y', \pi'_1, \pi'_2)$, then $f \times g\colon X \times Y \to X' \times Y'$ is the unique arrow such that $\pi'_1 \circ f \times g = f \circ \pi_1$ and $\pi'_2 \circ f \times g = g \circ \pi_2$. △

(b) And just to check everything works as it ought to, let's prove a pair of theorems which should look obvious if you have been following the various definitions.

Theorem 41. *Suppose we have arrows $f\colon X \to X$ and $g\colon Y \to Y$, and an order-swapping isomorphism $o\colon X \times Y \to Y \times X$. Then $o \circ (f \times g) = (g \times f) \circ o$.*

Proof. Suppose we have products $(X \times Y, \pi_1, \pi_2)$ and $(Y \times X, \pi'_1, \pi'_2)$, and an isomorphism $o\colon X \times Y \to Y \times X$, as in the proof of Theorem 35 (2). And now consider the following pair of diagrams:

$$\begin{tikzcd}
X \arrow[d, "f"'] & X \times Y \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] \arrow[d, "f \times g"] & Y \arrow[d, "g"] \\
X \arrow[d, "1_X"'] & X \times Y \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] \arrow[d, "o"] & Y \arrow[d, "1_Y"] \\
X & Y \times X \arrow[l, "\pi'_2"'] \arrow[r, "\pi'_1"] & Y
\end{tikzcd}
\qquad
\begin{tikzcd}
X \arrow[d, "1_X"'] & X \times Y \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] \arrow[d, "o"] & Y \arrow[d, "1_Y"] \\
X \arrow[d, "f"'] & Y \times X \arrow[l, "\pi'_2"'] \arrow[r, "\pi'_1"] \arrow[d, "g \times f"] & Y \arrow[d, "g"] \\
X & Y \times X \arrow[l, "\pi'_2"'] \arrow[r, "\pi'_1"] & Y
\end{tikzcd}$$

(Careful with the directions of the projection arrows!). Both diagrams commute, revealing that the same wedge factors through the bottom product via both $o \circ (f \times g)$ and $(g \times f) \circ o$. Those arrows must therefore be equal by the uniqueness of mediating arrows. □

Theorem 42. *Suppose we have parallel arrows $f, g\colon X \to Y$ in a category with binary products. Then the arrow $\langle f, g\rangle$ is equal to the composite $(f \times g) \circ \delta_X$.*

Proof. The idea is that it should not matter whether we apply f and g separately to an element of X and take the product, or take the product of that element with itself and apply f and g componentwise. So take the diagram

$$\begin{tikzcd}
 & X \arrow[dl, "1"'] \arrow[d, "\delta_X"] \arrow[dr, "1"] & \\
X \arrow[d, "f"'] & X \times X \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] \arrow[d, "f \times g"] & X \arrow[d, "g"] \\
Y & Y \times Y \arrow[l, "\pi'_1"'] \arrow[r, "\pi'_2"] & Y
\end{tikzcd}$$

This commutes by the definitions of δ_X and $f \times g$. Hence the following also commutes:

$$\begin{tikzcd} & X \arrow[dl, "f"'] \arrow[d, "(f\times g)\circ\delta_X"] \arrow[dr, "g"] & \\ Y & Y \times Y \arrow[l, "\pi_1'"] \arrow[r, "\pi_2'"'] & Y \end{tikzcd}$$

Which makes $(f \times g) \circ \delta_X$ the mediating arrow in a product diagram, so by uniqueness and the definition of $\langle f, g\rangle$, we have $(f \times g) \circ \delta_X = \langle f, g\rangle$. ☐

(c) Here's a special case: sometimes we have an arrow $f\colon X \to X'$ and we want to define an arrow from $X \times Y$ to $X' \times Y$ which applies f to the first component of a product and leaves the second alone. Then $f \times 1_Y$ will do the trick.

Now, it is tempting to suppose that if we have parallel maps $f, g\colon X \to X'$ and $f \times 1_Y = g \times 1_Y$, then $f = g$. But this actually fails in some categories – for example, in the toy category we met in §(b), whose only arrows are as diagrammed

$$\begin{tikzcd} X' & X \arrow[l, shift right, "f"'] \arrow[l, shift left, "g"] & V \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & Y \end{tikzcd}$$

together with the necessary identities and composites, and where by stipulation $f \neq g$ but $f \circ \pi_1 = g \circ \pi_1$ (and hence $f \times 1_Y = g \times 1_Y$).

(d) Lastly, an easy lemma and a final general result. The little lemma is this:

Theorem 43. *Given a product $(X \times Y, \pi_1, \pi_2)$ and arrows $S \underset{v}{\overset{u}{\rightrightarrows}} X \times Y$, then, if $\pi_1 \circ u = \pi_1 \circ v$ and $\pi_2 \circ u = \pi_2 \circ v$, it follows that $u = v$.*

Proof. The assumptions tell us that the same wedge $X \leftarrow S \rightarrow Y$ factors through the product both via u and via v: :

$$\begin{tikzcd} & S \arrow[dl, "\pi_1\circ u/\pi_1\circ v"'] \arrow[d, shift right, "u"'] \arrow[d, shift left, "v"] \arrow[dr, "\pi_2\circ u/\pi_2\circ v"] & \\ X & X \times Y \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & Y \end{tikzcd}$$

Hence $u = v$ by uniqueness of mediating arrows. ☐

Then here's the result we will need later:

Theorem 44. *Assume that there are arrows*

$$\begin{tikzcd} X \arrow[r, "f"] & X' \arrow[r, "j"] & X'' \\ Y \arrow[r, "g"] & Y' \arrow[r, "k"] & Y'' \end{tikzcd}$$

Assume there are products $(X \times Y, \pi_1, \pi_2)$, $(X' \times Y', \pi_1', \pi_2')$ and $(X'' \times Y'', \pi_1'', \pi_2'')$. Then $(j \times k) \circ (f \times g) = (j \circ f) \times (k \circ g)$.

Proof. Pause to think why we should want this to be true!

By the defining property of arrow products applied to the three different products we get,

$$\pi_1'' \circ (j \times k) \circ (f \times g) = j \circ \pi_1' \circ (f \times g) = j \circ f \circ \pi_1 = \pi_1'' \circ (j \circ f) \times (k \circ g).$$

Similarly

$$\pi_2'' \circ (j \times k) \circ (f \times g) = \pi_2'' \circ (j \circ f) \times (k \circ g)$$

The theorem then immediately follows by our preceding lemma. □

13 Products more generally

So far we have talked of binary products. But we can generalize in obvious ways. For example, there are ...

13.1 Ternary products

(a) The definition should be obvious:

Definition 51. In any category $\mathscr{C}$, *a ternary product* (O, π_1, π_2, π_3) *for the objects* X_1, X_2, X_3 is an object O together with projection arrows $\pi_i \colon O \to X_i$ (for $i = 1, 2, 3$) such that for any object S and arrows $f_i \colon S \to X_i$ there is always a unique arrow arrow $u \colon S \to O$ such that $f_i = \pi_i \circ u$. △

And then, exactly as we would expect, using just the same proof ideas as in the binary case, we can prove

Theorem 45. *If the ternary products* (O, π_1, π_2, π_3) *and* $(O', \pi'_1, \pi'_2, \pi'_3)$ *for* X_1, X_2, X_3 *both exist in* $\mathscr{C}$, *then there is a unique isomorphism* $f \colon O \xrightarrow{\sim} O'$ *commuting with the projection arrows.*

We can safely leave filling in the details as an exercise.

(b) We now note that if $\mathscr{C}$ has binary products for all pairs of objects, then it automatically has ternary products too, for

Theorem 46. $(X_1 \times X_2) \times X_3$ *together with the obvious projection arrows forms a ternary product of* X_1, X_2, X_3.

Proof. Assume $(X_1 \times X_2, \pi_1, \pi_2)$ is a product of X_1 with X_2, and also that $((X_1 \times X_2) \times X_3, \rho_1, \rho_2)$ is a product of $X_1 \times X_2$ with X_3.

Take any object S and arrows $f_i \colon S \to X_i$. By our first assumption, (a) there is a unique $u \colon S \to X_1 \times X_2$ such that $f_1 = \pi_1 \circ u$, $f_2 = \pi_2 \circ u$. And by our second assumption, (b) there is then a unique $v \colon S \to (X_1 \times X_2) \times X_3$ such that $u = \rho_1 \circ v$, $f_3 = \rho_2 \circ v$.

Therefore $f_1 = \pi_1 \circ \rho_1 \circ v$, $f_2 = \pi_2 \circ \rho_1 \circ v$, $f_3 = \rho_2 \circ v$

Now consider the triple wedge $((X_1 \times X_2) \times X_3, \pi_1 \circ \rho_1, \pi_2 \circ \rho_1, \rho_2)$. This, we claim, is indeed a ternary product of X_1, X_2, X_3.

$$\begin{tikzcd}
& & S \arrow[ddll, "f_1"'] \arrow[ddl, "u"'] \arrow[dd, "f_2"] \arrow[ddr, bend left, "f_3"] \arrow[dr, "v"] & \\
& & & (X_1 \times X_2) \times X_3 \arrow[dll, "\rho_1"'] \arrow[d, "\rho_2"] \\
X_1 & X_1 \times X_2 \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & X_2 & X_3
\end{tikzcd}$$

Everything in the diagram commutes, so – just as we have noted – the triple wedge with vertex S and arrows $f_i\colon S \to X_i$ factors through $(X_1 \times X_2) \times X_3$ via the arrow v. It remains to confirm v's uniqueness in this new role.

Suppose we have $w\colon S \to (X_1 \times X_2) \times X_3$ where $f_1 = \pi_1 \circ \rho_1 \circ w$, $f_2 = \pi_2 \circ \rho_1 \circ w$, $f_3 = \rho_2 \circ w$. Then $\rho_1 \circ w\colon S \to X_1 \times X_2$ is such that $f_1 = \pi_1 \circ (\rho_1 \circ w)$, $f_2 = \pi_2 \circ (\rho_1 \circ w)$. Hence by (a), $u = \rho_1 \circ w$. But now invoking (b), that together with $f_3 = \rho_2 \circ w$ entails $w = v$. $\square$

Of course, an exactly similar argument will show that the product $X_1 \times (X_2 \times X_3)$ together with the obvious projection arrows will serve as another ternary product of X_1, X_2, X_3.

Hence we are now at last in a position to neatly prove

Theorem 35. (3) $X \times (Y \times Z) \cong (X \times Y) \times Z$.

Proof. Both $(X_1 \times X_2) \times X_3$ and $X_1 \times (X_2 \times X_3)$ (with their projection arrows) are ternary products of X_1, X_2, X_3. So Theorem 45 entails that $X_1 \times (X_2 \times X_3) \cong (X_1 \times X_2) \times X_3$. $\square$

13.2 More finite products

What goes for ternary products goes for n-ary products defined in a way exactly analogous to Defn. 51. If $\mathscr{C}$ has binary products for all pairs of objects it will have quaternary products such as $((X_1 \times X_2) \times X_3) \times X_4$, quinary products, and n-ary products more generally, for any finite $n \geqslant 2$.

So, to round things out, how do things go for the nullary and unary cases?

Following the same pattern of definition, a *nullary* product in $\mathscr{C}$ would be an object O together with *no* projection arrows, such that for any object S there is a unique arrow $u\colon S \to O$. Which is just to say that a nullary product is a terminal object of the category.

And a *unary* product of X would be an object O and a single projection arrow $\pi_1\colon O \to X$ such that for any object S and arrow $f\colon S \to X$ there is a unique arrow $u\colon S \to O$ such that $\pi \circ u = f$. Putting $O = X$ and $\pi = 1_X$ evidently fits the bill. So the basic case of a unary product of X is not quite X itself, but rather X equipped with its identity arrow (and like any product, this is unique up to unique isomorphism). Trivially, unary products for all objects exist in all categories.

In sum, suppose we say

Definition 52. A category $\mathscr{C}$ *has all binary products* iff for all $\mathscr{C}$-objects X and Y, there exists a binary product of X with Y in $\mathscr{C}$. $\mathscr{C}$ *has all finite products* iff $\mathscr{C}$ has n-ary products for any n objects, for all $n \geqslant 0$. △

Then our preceding remarks establish

Theorem 47. *A category $\mathscr{C}$ has all finite products iff $\mathscr{C}$ has a terminal object and has all binary products.* □

13.3 Infinite products

We can now generalize still further in an obvious way, going beyond finite products to infinite cases.

Definition 53. Suppose that we are dealing with $\mathscr{C}$-objects X_j indexed by items j in some suite of indices J (not now assumed finite). Then the product of the X_j, if it exists in $\mathscr{C}$, is an object O together with a projection arrow $\pi_j \colon O \to X_j$ for each index j. It is required that for any object S and family of arrows $f_j \colon S \to X_j$ (one for each index), there is always a unique arrow arrow $u \colon S \to O$ such that $f_j = \pi_j \circ u$. △

For the same reasons as before, such a generalized product will be unique up to unique isomorphism.

Now, we are in fact only going to be really interested in cases where the suite of indices J can be treated as a set in standard set theory. In other words, we are really only going to be interested in cases where we take products of set-many objects. Ignoring the over-sized cases, we then say:

Definition 54. A category $\mathscr{C}$ *has all small products* iff for any $\mathscr{C}$-objects X_j, for $j \in J$ where J is some index set, these objects have a product. We notate the object in the product of such X_j for $j \in J$ by '$\prod_{j \in J} X_j$'. △

Here, 'small' is a joke. It doesn't mean small by any normal standards – it just indicates that we are taking products over collections of objects that are not too many to form a set. We'll be returning to such issues of size in Chapter 21.

14 Equalizers

Terminal and initial objects, products and coproducts, are defined by universal mapping properties. In this chapter, we look at another dual pair of constructs defined by such mapping properties, so-called equalizers and co-equalizers.

14.1 Equalizers

It was useful, when defining products, to introduce the idea of a 'wedge' (Defn. 43) for a certain configuration of objects and arrows in a category. Here's a similar definition that is going to be useful in defining the equalizers:

Definition 55. A *fork* (from S through X to Y) consists of arrows $k\colon S \to X$ with $f\colon X \to Y$ and $g\colon X \to Y$, such that $f \circ k = g \circ k$. △

So a fork is a commuting diagram $S \xrightarrow{k} X \underset{g}{\overset{f}{\rightrightarrows}} Y$, with the composite arrows from S to Y being equal.[1]

Now, as we saw, a product wedge from O to X and Y is a limiting case: it's a wedge such that any other wedge from S to X and Y uniquely factors through it. Likewise, an equalizing fork from E through parallel arrows $f, g\colon X \to Y$ is another limiting case: it's a fork such that any other fork from an object S through f, g uniquely factors through it. In other, clearer, words:

Definition 56. Let $\mathscr{C}$ be a category and $f, g\colon X \to Y$ be a pair of parallel arrows in $\mathscr{C}$. Then the object E and arrow $e\colon E \to X$ form an *equalizer* in $\mathscr{C}$ for those arrows iff $f \circ e = g \circ e$ (so $E \xrightarrow{e} X \underset{g}{\overset{f}{\rightrightarrows}} Y$ is indeed a fork), and for any fork $S \xrightarrow{k} X \underset{g}{\overset{f}{\rightrightarrows}} Y$ there is a unique mediating arrow $u\colon S \to E$ such that the following diagram commutes:

$$\begin{tikzcd}
S \arrow[dr, "k"] \arrow[dd, dashed, "u"'] & & \\
& X \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & Y \\
E \arrow[ur, "e"'] & &
\end{tikzcd}$$

[1] Check again our Defn. 18 for why this diagram counts as commuting even though the parallel arrows from X to Y need not be equal.

△

We now note that, just as with products (see Defn. 44), we can give an alternative definition which defines equalizers in terms of a terminal object in a suitable category. First we say

Definition 57. Given a category $\mathscr{C}$ and parallel arrows $f, g\colon X \to Y$, then the derived category of forks $\mathscr{C}_{F(XY)}$ has as objects all forks $S \xrightarrow{k} X \overset{f}{\underset{g}{\rightrightarrows}} Y$.

And an arrow from $S \xrightarrow{k} \cdots$ to $S' \xrightarrow{k'} \cdots$ in $\mathscr{C}_{F(XY)}$ is a $\mathscr{C}$-arrow $g\colon S \to S'$ such that the resulting triangle commutes: i.e. such that $k = k' \circ g$.

The identity arrow in $\mathscr{C}_{F(XY)}$ on the fork $S \xrightarrow{k} \cdots$ is the identity arrow 1_S in $\mathscr{C}$; and the composition of arrows in $\mathscr{C}_{F(XY)}$ is defined as the composition of the arrows as they feature in $\mathscr{C}$. △

It is again easily checked that this indeed defines a category. Our definition of an equalizer then comes to this:

Definition 58. An equalizer of $f, g\colon X \to Y$ in $\mathscr{C}$ is some (E, e) (where E is a $\mathscr{C}$-object, and e is a $\mathscr{C}$-arrow $E \to X$) such that the fork $E \xrightarrow{e} X \overset{f}{\underset{g}{\rightrightarrows}} Y$ is terminal in $\mathscr{C}_{F(XY)}$. △

Here, then, are a few examples of equalizers:

(1) Suppose in Set we have parallel arrows $X \overset{f}{\underset{g}{\rightrightarrows}} Y$. Then let $E \subseteq X$ be the set such that $x \in E$ iff $fx = gx$, and let $e\colon E \to X$ be the simple inclusion map. By construction, $f \circ e = g \circ e$. So $E \xrightarrow{e} X \overset{f}{\underset{g}{\rightrightarrows}} Y$ is a fork. We show that (E, e) is in fact an equalizer for f and g.

Suppose $S \xrightarrow{k} X \overset{f}{\underset{g}{\rightrightarrows}} Y$ is any other fork through f, g, which requires $f(k(s)) = g(k(s))$ for each $s \in S$ and hence $k[S] \subseteq E \subseteq X$. Defining the mediating arrow $u\colon S \to E$ to agree with $k\colon S \to X$ on all inputs will make the diagram for equalizers commute. And this is the unique possibility: for the diagram to commute we need $k = e \circ u$, and since the inclusion e doesn't affect the values of the function, k and u must indeed agree on all inputs.

(2) Equalizers in categories whose objects are sets-with-structure behave similarly. Take as an example the category Mon. Given a pair of monoid homomorphisms $(X, \cdot, 1_X) \overset{f}{\underset{g}{\rightrightarrows}} (Y, *, 1_Y)$, take the subset E of X on which the functions agree. Evidently E must contain the identity element of X (since f and g agree on this element: being homomorphisms, both must send 1_X to the 1_Y). And suppose $e, e' \in E$: then $f(e \cdot e') = f(e) * f(e') = g(e) * g(e') = g(e \cdot e')$, which means that E is closed under products of members.

So take E together with the monoid operation from $(X, \cdot, 1_X)$ restricted to members of E. Then $(E, \cdot, 1_X)$ is a monoid – for the shared identity element still behaves as an identity, E is closed under the operation, and the operation is still associative. And if we take $(E, \cdot, 1_X)$ and equip it with the injection homomorphism into $(X, \cdot, 1_X)$, this will evidently give us an equalizer for f and g.

(3) Similarly, take Top. What is the equalizer for a pair of continuous maps $X \overset{f}{\underset{g}{\rightrightarrows}} Y$? Well, take the subset of (the underlying set of) X on which the functions agree, and give it the subspace topology. This topological space equipped with the injection into X is then the desired equalizer. (This works because of the way that the subspace topology is defined – we won't go into details).

(4) A special case. Suppose we are in Grp and have a group homomorphism, $f\colon X \to Y$. There is also another trivial homomorphism $o\colon X \to Y$ which sends any element of the group X to the identity element in Y, i.e. is the composite $X \to 1 \to Y$ of the only possible homomorphisms. Now consider what would constitute an equalizer for f and o.

Suppose K is the kernel of f, i.e. the subgroup of X whose objects are the elements which f sends to the identity element of Y, and let $i\colon K \to X$ be the inclusion map. Then $K \xrightarrow{i} X \overset{f}{\underset{o}{\rightrightarrows}} Y$ is a fork since $f \circ i = o \circ i$.

Let $S \xrightarrow{k} X \overset{f}{\underset{o}{\rightrightarrows}} Y$ be another fork. Now, $o \circ k$ sends every element of S to the unit of Y. Since $f \circ k = o \circ k$, k must send any element of S to some element in the kernel K. So let $k'\colon S \to K$ agree with $k\colon S \to X$ on all arguments.

Then the following commutes:

$$\begin{tikzcd}
S \arrow[d, "k'"'] \arrow[dr, "k"] & & \\
K \arrow[r, "i"'] & X \arrow[r, shift left, "f"] \arrow[r, shift right, "o"'] & Y
\end{tikzcd}$$

And evidently k' is the only possible homomorphism to make the diagram commute.

So the equalizer of f and o is f's kernel K equipped with the inclusion map into the domain of f. Or putting it the other way about, we can define kernels of group homorphisms categorially in terms of equalizers.

(5) Finally we remark that the equalizer of a pair of maps $X \overset{f}{\underset{g}{\rightrightarrows}} Y$ where in fact $f = g$ is simply $[X, 1_X]$.

Consider then a poset $(P, \preccurlyeq)$ considered as a category whose objects are the members of P and where there is a unique arrow $X \to Y$ (for $X, Y \in P$) iff $X \preccurlyeq Y$. So the only cases of parallel arrows from X to Y are cases of equal arrows which then, as remarked, have equalizers. So in sum, a poset category has all possible equalizers.

14.2 Uniqueness again

Just as products are unique up to unique isomorphism, equalizers are too. That is to say,

Theorem 48. *If both the equalizers* $[E, e]$ *and* $[E', e']$ *exist for* $X \underset{g}{\overset{f}{\rightrightarrows}} Y$, *then there is a unique isomorphism* $j \colon E \xrightarrow{\sim} E'$ *commuting with the equalizing arrows, i.e. such that* $e = e' \circ j$.

Plodding proof from first principles. We can use an argument that goes along exactly the same lines as the one we used to prove the uniqueness of products and equalizers. This is of course no accident, given the similarity of the definitions via a unique mapping property.

Assume $[E, e]$ equalizes f and g, and suppose $e \circ h = e$. Then observe that the following diagram will commute

$$\begin{tikzcd}
E \arrow[dr, "e"] \arrow[dd, "h"'] & & \\
& X \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & Y \\
E \arrow[ur, "e"'] & &
\end{tikzcd}$$

Now obviously, $h = 1_E$ makes that diagram commute. But by hypothesis there is a unique arrow $E \to E$ which makes the diagram commute. So we can conclude that if $e \circ h = e$, then $h = 1_E$.

Now suppose $[E', e']$ is also an equalizer for f and g. Then $[E, e]$ must factor uniquely through it. That is to say, there is a (unique) mediating $j \colon E \to E'$ such that $e' \circ j = e$. And since $[E', e']$ must factor uniquely though $[E, e]$ there is a unique k such that $e \circ k = e'$. So $e \circ k \circ j = e$, and hence by our initial conclusion, $k \circ j = 1_E$.

A similar proof shows that $j \circ k = 1_{E'}$. Which makes the unique j an isomorphism. □

Proof using the alternative definition of equalizers. $[E, e]$ and $[E', e']$ are both terminal objects in the fork category $\mathscr{C}_{F(XY)}$. So by Theorem 26 there is a unique $\mathscr{C}_{F(XY)}$-isomorphism j between them. But, by definition, this has to be a $\mathscr{C}$-arrow $j \colon E \xrightarrow{\sim} E'$ commuting with the equalizing arrows. And j is easily seen to be an isomorphism in $\mathscr{C}$ too. □

Let's add two further general results about equalizers. First:

Theorem 49. *If* $[E, e]$ *constitute an equalizer, then* e *is a monomorphism.*

Proof. Assume $[E, e]$ equalizes $X \overset{f}{\underset{g}{\rightrightarrows}} Y$, and suppose $e \circ j = e \circ k$, where for some D, $D \overset{j}{\underset{k}{\rightrightarrows}} E$. Then the following diagram commutes,

$$\begin{tikzcd}
D \arrow[dd, shift right, "j"'] \arrow[dd, shift left, "k"] \arrow[dr, "e\circ j = e\circ k"] & & \\
& X \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & Y \\
E \arrow[ur, "e"'] & &
\end{tikzcd}$$

So $D \xrightarrow{e\circ j/e\circ k} X \overset{f}{\underset{g}{\rightrightarrows}} Y$ is a fork factoring through the equalizer. But by the definition of an equalizer, it has to factor uniquely, and hence $j = k$. In sum, e is left-cancellable in the equation $e \circ j = e \circ k$; i.e. e is monic. □

Second, in an obvious shorthand,

Theorem 50. *In any category, an epic equalizer is an isomorphism*

Proof. Assume again that $[E, e]$ equalizes $X \overset{f}{\underset{g}{\rightrightarrows}} Y$, so that $f \circ e = g \circ e$. So if e is epic, it follows that $f = g$. Then consider the following diagram

$$\begin{tikzcd}
X \arrow[dd, dashed, "u"'] \arrow[dr, "1_X"] & & \\
& X \arrow[r, "f", "g"'] & Y \\
E \arrow[ur, "e"'] & &
\end{tikzcd}$$

Because e equalizes, we know there is a unique u such that (i) $e \circ u = 1_X$.

But then also $e \circ u \circ e = 1_X \circ e = e = e \circ 1_E$. Hence, since equalizers are monic by the last theorem, (ii) $u \circ e = 1_E$.

Taken together, (i) and (ii) tell us that e has an inverse. Therefore e is an isomorphism. □

14.3 Co-equalizers

(a) Dualizing, we get the notion of a co-equalizer. First we say:

Definition 59. A *co-fork* (from X through Y to S) consists of parallel arrows $f\colon X \to Y$, $g\colon X \to Y$ and an arrow $k\colon Y \to S$, such that $k \circ f = k \circ g$. △

(Plain 'fork' is often used for the dual too: but the ugly 'co-fork' keeps things clear.) Diagrammatically, a co-fork looks like this: $X \overset{f}{\underset{g}{\rightrightarrows}} Y \xrightarrow{k} S$, with the composite arrows from X to S being equal. Then, as you would expect:

Definition 60. Let $\mathscr{C}$ be a category and $f\colon X \to Y$ and $g\colon X \to Y$ be a pair of parallel arrows in $\mathscr{C}$. The object C and arrow $c\colon Y \to S$ form a *co-equalizer* in $\mathscr{C}$ for those arrows iff $c \circ f = c \circ g$, and for any co-fork from X through Y to S there is a unique arrow $u\colon C \to S$ such the following diagram commutes:

$$\begin{tikzcd} & & S \\ X \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & Y \arrow[ur, "k"] \arrow[dr, "c"'] & \\ & & C \arrow[uu, dashed, "u"'] \end{tikzcd}$$

△

(b) We need not pause to spell out the dual arguments that co-equalizers are unique up to a unique isomorphism or that co-equalizers are epic. Instead, we turn immediately to consider one central example by asking: what do co-equalizers look like in Set?

Suppose we are given parallel arrows $f, g\colon X \to Y$ in Set. These arrows induce a relation R_{fg} (or R for short) on the members of Y, where yRy' holds when there is an $x \in X$ such that $f(x) = y \wedge g(x) = y'$. Now, given a co-fork $X \overset{f}{\underset{g}{\rightrightarrows}} Y \xrightarrow{k} S$, then yRy' implies $k(y) = k(y')$. And trivially, having equal k-values is an equivalence relation $\equiv_k$ on members of Y.

So, in sum, we've shown that given a co-fork via $k\colon Y \to S$ from the parallel arrows $f, g\colon X \to Y$, there is a corresponding equivalence relation $\equiv_k$ on Y such that if $yR_{fg}y'$ then $y \equiv_k y'$.

Now what's the limiting case of such an equivalence relation? It will have to be $R^{\sim}$, the smallest equivalence relation containing R_{fg}. So we'll expect that the limiting case of a cofork will comprise an arrow $c\colon Y \to C$ such that $\equiv_c \; = \; R^{\sim}$. In other words, we want c to be such that $c(y) = c(y')$ iff $yR^{\sim}y'$.

Which motivates the following:

Theorem 51. *Given functions $f, g\colon X \to Y$ in* Set*, let $R^{\sim}$ be the smallest equivalence relation containing R – where yRy' iff $(\exists x \in X)(f(x) = y \wedge g(x) = y')$.*

Let C be $Y/R^{\sim}$, i.e. the set of $R^{\sim}$-equivalence classes of Y; and let c map $y \in Y$ to the $R^{\sim}$-equivalence class containing y. Then $[C, c]$, so defined, is a co-equalizer for f and g.

Proof. We just have to do some routine checking. First we show $c \circ f = c \circ g$. But the left-hand side sends $x \in X$ to the $R^{\sim}$-equivalence class containing $f(x)$ and the right-hand side sends x to the $R^{\sim}$-equivalence class containing $g(x)$. However, $f(x)$ and $g(x)$ are by definition R-related, and hence are $R^{\sim}$-related: so by construction they belong to the same $R^{\sim}$-equivalence class. Hence $X \overset{f}{\underset{g}{\rightrightarrows}} Y \xrightarrow{c} C$ is indeed a co-fork.

Now suppose there is another co-fork $X \overset{f}{\underset{g}{\rightrightarrows}} Y \xrightarrow{k} S$. We need to show the co-fork ending with c factors through this via a unique mediating arrow u.

By assumption, $k \circ f = k \circ g$. And we first outline a proof that if $yR^{\sim}y'$ then $k(y) = k(y')$.

Start with R defined as before, and let R' be its reflexive closure. Obviously we'll still have that if $yR'y'$ then $k(y) = k(y')$. Now consider R'' the symmetric closure of R': again, we'll still have that if $yR''y'$ then $k(y) = k(y')$. Now note that if $yR''y'$ and $y'R''y''$, then $k(y) = k(y'')$. So if we take the transitive closure of R'', we'll still have a relation which, when it holds between some y and y'', implies that $k(y) = k(y'')$. But the transitive closure of R'' is $R^{\sim}$.

We have shown, then, that k is constant on members of a $R^{\sim}$-equivalence class, and so we can well-define a function $u\colon C \to S$ which sends an equivalence class to the value of k on a member of that class. This u is the desired mediating arrow which makes the diagram defining a co-equalizer commute. Moreover, since c is surjective and C only contains $R^{\sim}$-equivalence classes, u is the only function for which $u \circ c = k$. $\square$

In a slogan then: *in* Set*, quotienting by an equivalence relation is (up to unique isomorphism) the same as taking an associated co-equalizer*. In many other categories co-equalizers behave similarly, corresponding to 'naturally occurring' quotienting constructions. But we won't go into more detail here.

15 Limits and colimits defined

A terminal object is defined essentially in terms of how all the other objects in the category relate to it (by each sending it a unique arrow). A product wedge is defined in terms of how all the other wedges in a certain family relate to it (each factoring through it via a unique arrow). An equalizing fork is defined in terms of how all the other forks in a certain family relate to it (each factoring through it via a unique arrow). In an informal sense, terminal objects, products, and equalizers are limiting cases, defined in closely analogous ways using universal mapping properties. Likewise for their duals.

In this chapter, we now formally capture what's common to terminal objects, products and equalizers by defining a general class of *limits*, and confirming that terminal objects, products and equalizers are indeed examples. We also define a dual class of *co-limits*, which has initial objects, coproducts and co-equalizers as examples.

We then give a new pair of examples, one for each general class, the so-called pullbacks and pushouts.

15.1 Cones over diagrams

(a) We start by defining the notion of a cone over a diagram; then in the next section we can use this to define the key notion of a limit cone.

Way back in Defn. 17, we loosely characterized a diagram D in a category $\mathscr{C}$ as being what is represented by a representational diagram – i.e. as simply consisting in a bunch of objects with, possibly, some arrows between some of them. We now need some more systematic scheme for labelling the objects in a diagram. So henceforth we'll assume that the objects in D can be labelled by terms like 'D_j' where 'j' is an index from some suite of indices J. For convenience, we'll allow double counting, permitting the case where $D_j = D_k$ for different indices. We allow the limiting cases of diagrams where there are no arrows, and even the empty case where there are no objects either. So:

Definition 17* A (labelled) *diagram in a category* $\mathscr{C}$ is some (or no) objects D_j for indices j in the suite of indices J, and some (or no) $\mathscr{C}$-arrows between these objects. △

(We eventually, in §22.1, give a tauter definition of diagrams, but this will do to be getting on with.)

Definition 61. Let D be a diagram in category $\mathscr{C}$. Then a *cone over D* comprises a $\mathscr{C}$-object C, the *vertex* or *apex* the cone, together with $\mathscr{C}$-arrows $c_j \colon C \to D_j$ (often called the *legs* of the cone), one for each object D_j in D, such that whenever there is an arrow $d \colon D_k \to D_l$ in D, $c_l = d \circ c_k$, i.e. the following diagram commutes:

$$\begin{tikzcd} & C \arrow[dl, "c_k"'] \arrow[dr, "c_l"] & \\ D_k \arrow[rr, "d"'] & & D_l \end{tikzcd}$$

We use '$[C, c_j]$' as our notation for such a cone. △

Think of it diagrammatically(!) like this: arrange the objects in the diagram D in a plane, along with whatever arrows there are between them in D. Now sit the object C above the plane, with a quiverful of arrows from C zinging down, one to each object D_j in the plane. Those arrows form the 'legs' of a skeletal cone. And the key requirement is that any triangles thus formed with C at the apex must commute.

We should note, by way of aside, that some authors prefer to say more austerely that a cone is not a vertex-object-with-a-family-of-arrows-from-that-vertex but simply a family of arrows from the vertex. Since we can read off the vertex of a cone as the common source of all its arrows, it is very largely a matter of convenience whether we speak austerely or explicitly mention the vertex. But for the moment, we'll take the less austere line.

(b) For later use, but also to help check understanding now, here is another definition and then two theorems:

Definition 62. The (reflexive, transitive) *closure* of a diagram D in a category $\mathscr{C}$ is the smallest diagram which includes all the objects and arrows of D, but which also has an identity arrow on each object, and for any two of its composable arrows, it also contains their composition. △

In other words, the closure of a diagram D in $\mathscr{C}$ is what you get by adding identity arrows where necessary, forming composites of any composable arrows you now have, then forming composites of what you have at the next stage, and so on and so forth. Since the associativity of the composition operation will be inherited from $\mathscr{C}$, it is immediate that

Theorem 52. *The closure of a diagram D in $\mathscr{C}$ is a subcategory of $\mathscr{C}$.*

A little more interestingly, though almost equally easily, we have:

Theorem 53. *If $[C, c_j]$ is a cone over D, then it is a cone over the closure of D.*

Proof. The closure of D has no additional objects, so $[C, c_j]$ still has a leg from the vertex C to each object in the closure. It is trivial that, given an identity arrow $1_k \colon D_k \to D_k$, we have $c_k = 1_k \circ c_k$. So we just need to show a cone over composable arrows is still a cone when their composite is added. So suppose we have a cone over a diagram including the arrows $d \colon D_k \to D_l$ and $d' \colon D_l \to D_m$. That means $c_l = d \circ c_k$ and $c_m = d' \circ c_l$. Hence $c_m = (d' \circ d) \circ c_k$. So the cone is still a cone if we add the composite arrow $d' \circ d \colon D_k \to D_m$. □

15.2 Defining limit cones

(a) There can be many cones, with different vertices, over a given diagram D. But, in just the same spirit as our earlier definitions of products and equalizers, we can define a limiting case, by means of a universal mapping property:

Definition 63. A cone $[L, \lambda_j]$ over a diagram D in $\mathscr{C}$ is a *limit (cone) over* D iff any cone $[C, c_j]$ over D uniquely factors through it, so there is a unique mediating arrow $u \colon C \to L$ such that for each index j, $\lambda_j \circ u = c_j$. In other words, for each D_j in D, the corresponding triangle whose other vertices are C and L commutes:

$$\begin{tikzcd}
 & C \arrow[dl, "c_j"'] \arrow[d, dashed, "u"] \arrow[dr, "c_k"] \arrow[drr, "c_l"] & & \\
 & L \arrow[dl, "\lambda_j"] \arrow[dr, "\lambda_k"'] \arrow[drr, "\lambda_l"] & & \\
D_j & & D_k & D_l \quad \cdots
\end{tikzcd}$$

△

(b) Let's immediately confirm that our three announced examples of limits so far are indeed limit cones in the sense just defined.

(1) We start with the null case. Take the empty diagram in $\mathscr{C}$ – *zero* objects and so, necessarily, no arrows. Then a cone over the empty diagram is simply an object C, a lonely vertex (there is no further condition to fulfil), and an arrow between such minimal cones C, C' is just an arrow $C \to C'$. Hence L is a limit cone just if there is a unique arrow to it from any other object – i.e. just if L is a terminal object in $\mathscr{C}$!

(2) Consider now a diagram which is just *two* objects we'll call 'D_1', 'D_2', still with no arrow between them. Then a cone over such a diagram is just a wedge into D_1, D_2; and a limit cone is simply a product of D_1 with D_2. (We could equally have considered the reflexive transitive closure of this two object diagram, i.e. the discrete category with two objects plus their identity arrows: by our last theorem, it would make no difference.)

(3) Next consider a diagram which again has just two objects, but now with two parallel arrows between them, which we can represent $D_1 \overset{d}{\underset{d'}{\rightrightarrows}} D_2$. Then a cone over this diagram, or over its closure, is a commuting diagram like this:

$$\begin{tikzcd} & C \arrow[dl, "c_1"'] \arrow[dr, "c_2"] & \\ D_1 \arrow[rr, shift left, "d"] \arrow[rr, shift right, "d'"'] & & D_2 \end{tikzcd}$$

If there is such a diagram, then we must have $d \circ c_1 = d' \circ c_1$: and vice versa, if that identity holds, then we can put $c_2 = d \circ c_1 = d' \circ c_1$ to complete the commutative diagram. Hence we have a cone from the vertex C to our diagram iff $C \xrightarrow{c_1} D_1 \overset{d}{\underset{d'}{\rightrightarrows}} D_2$ is a fork. Since c_1 fixes what c_2 has to be to complete the cone, we can focus on the cut-down cone consisting of just $[C, c_1]$.

What is the corresponding cut-down limit cone? It consists in $[E, e]$ such there is a unique u such that $c_1 = e \circ u$. Hence $[E, e]$ is an equalizer of the parallel arrows $D_1 \overset{d}{\underset{d'}{\rightrightarrows}} D_2$.

(c) We can now give a direct proof, along now hopefully entirely familiar lines, for the predictable result

Theorem 54. *Limit cones over a given diagram D are unique up to a unique isomorphism commuting with the cones's arrows.*

Proof. As usual, we first note that a limit cone $[L, \lambda_j]$ factors through itself via the mediating identity $1_L \colon L \to L$. But by definition, a cone over D uniquely factors through the limit, so that means that

(i) if $\lambda_j \circ u = \lambda_j$ for all indices j, then $u = 1_L$.

Now suppose $[L', \lambda'_j]$ is another limit cone over D. Then $[L', \lambda'_j]$ uniquely factors through $[L, \lambda_j]$, via some f, so

(ii) $\lambda_j \circ f = \lambda'_j$ for all j.

And likewise $[L, \lambda_j]$ uniquely factors through $[L', \lambda'_j]$ via some g, so

(iii) $\lambda'_j \circ g = \lambda_j$ for all j.

Whence

(iv) $\lambda_j \circ f \circ g = \lambda_j$ for all j.

Therefore

(v) $f \circ g = 1_L$.

And symmetrically

(vi) $g \circ f = 1_{L'}$.

Whence f is not just unique (by hypothesis, the only way of completing the relevant diagrams to get the arrows to commute) but an isomorphism. □

15.3 Limit cones as terminal objects

We have already seen that

(1) A terminal object in $\mathscr{C}$ is ... wait for it! ... terminal in the given category $\mathscr{C}$.

(2) The product of X with Y in $\mathscr{C}$ is a terminal object in the derived category $\mathscr{C}_{W(X,Y)}$ of wedges to X and Y.

(3) The equalizer of parallel arrows through X to Y in $\mathscr{C}$ are (parts of) terminal objects in the derived category $\mathscr{C}_{F(XY)}$ of forks through X to Y.

Predictably, limit cones more generally are terminal objects in appropriate categories.

To spell this out, we first note that the cones $[C, c_j]$ over a given diagram D in $\mathscr{C}$ form a category in a very natural way:

Definition 64. Given a diagram D in category $\mathscr{C}$, the derived category $\mathscr{C}_{C(D)}$ – *the category of cones over D* – has the following data:

(1) Its objects are the cones $[C, c_j]$ over D.

(2) An arrow from $[C, c_j]$ to $[C', c'_j]$ is any $\mathscr{C}$-arrow $f\colon C \to C'$ such that $c'_j \circ f = c_j$ for all indices j. In other words, for each D_j, D_k, D_l, ..., in D, the corresponding triangle with remaining vertices C and C' commutes:

$$\begin{tikzcd}
& C \arrow[dl, "c_j"'] \arrow[d, dashed, "f"] \arrow[dr, "c_k"] \arrow[drr, "c_l"] & & & \\
& C' \arrow[dl, "c'_j"] \arrow[dr, "c'_k"'] \arrow[drr, "c'_l"] & & & \\
D_j & & D_k & D_l & \cdots
\end{tikzcd}$$

The identity arrow on a cone $[C, c_j]$ is the $\mathscr{C}$-arrow 1_C. And composition for arrows in $\mathscr{C}_{C(D)}$ is just composition of the corresponding $\mathscr{C}$-arrows. △

It is entirely routine to confirm that $\mathscr{C}_{C(D)}$ is indeed a category. We can then recast our earlier definition of a limit cone as follows:

Definition 65. A *limit cone for D in $\mathscr{C}$* is a cone which is a terminal object in $\mathscr{C}_{C(D)}$. △

And we now have an alternative proof of our last uniqueness result, Theorem 54:

Proof. Since a limit cone over D is terminal in $\mathscr{C}_{C(D)}$, it is unique in $\mathscr{C}_{C(D)}$ up to a unique isomorphism. But such an isomorphism in $\mathscr{C}_{C(D)}$ must be an isomorphism in $\mathscr{C}$ commuting with the cones's arrows. □

15.4 Results about limits

(a) Let's first prove two further simple theorems:

Theorem 55. *Suppose $[L, \lambda_j]$ is a limit cone over a diagram D in $\mathscr{C}$, and $[L', \lambda'_j]$ is another cone over D which factors through $[L, \lambda_j]$ via an isomorphism f. Then $[L', \lambda'_j]$ is also a limit cone.*

Proof. Take any cone $[C, c_j]$ over D. We need to show that (i) there is an arrow $v\colon C \to L'$ such that for all indices j for objects D_j in D, $c_j = \lambda'_j \circ v$, and (ii) v is unique.

But we know that there is a unique arrow $u\colon C \to L$ such that for j, $c_j = \lambda_j \circ u$. And we know that $f\colon L' \to L$ and $\lambda'_j = \lambda_j \circ f$ (so $\lambda_j = \lambda'_j \circ f^{-1}$).

Therefore put $v = f^{-1} \circ u$, and that satisfies (i).

Now suppose there is another arrow $v'\colon C \to L'$ such that $c_j = \lambda'_j \circ v'$. Then we have $f \circ v'\colon C \to L$, and also $c_j = \lambda_j \circ f \circ v'$. Therefore $[C, c_j]$ factors through $[L, \lambda_j]$ via $f \circ v'$, so $f \circ v' = u$. Whence $v' = f^{-1} \circ u = v$. Which proves (ii). □

Theorem 56. *Suppose $[L, \lambda_j]$ is a limit cone over a diagram D in $\mathscr{C}$. Then the cones over D with vertex C correspond one-to-one with $\mathscr{C}$-arrows from C to L.*

Proof. Take any arrow $u\colon C \to L$. If there is an arrow $d\colon D_k \to D_l$ in the diagram D, then (since $[L, \lambda_j]$ is a cone), $\lambda_l = d \circ \lambda_k$, whence $(\lambda_l \circ u) = d \circ (\lambda_k \circ u)$. Since this holds generally, $[C, \lambda_j \circ u]$ is a cone over D. But (again since $[L, \lambda_j]$ is a limit) every cone over D with vertex C is of the form $[C, \lambda_j \circ u]$ for unique u. Hence there is indeed a one-one correspondence between arrows $u\colon C \to L$ and cones over D with vertex C. (Moreover, the construction is a natural one, involving no arbitrary choices.) □

(b) We pause for a fun exercise and reality check, by remarking that the whole category $\mathscr{C}$ can be thought of as the limiting case of a diagram in itself, and then

Theorem 57. *A category $\mathscr{C}$ has an initial object if and only if $\mathscr{C}$, thought of as a diagram in $\mathscr{C}$, has a limit.*

Proof. Suppose $\mathscr{C}$ has an initial object I. Then for every $\mathscr{C}$-object C, there is a unique arrow $\lambda_C\colon I \to C$. $[I, \lambda_C]$ is a cone (since for any arrow $f\colon C \to D$, the composite $f \circ \lambda_C$ is an arrow from I to D and hence has to be equal to the unique λ_D. Further, $[I, \lambda_C]$ is a limit cone. For suppose $[A, a_C]$ is any other cone over the whole of $\mathscr{C}$. Then since it is a cone, the triangle

$$\begin{tikzcd} & A \arrow[dl, "a_I"'] \arrow[dr, "a_C"] & \\ I \arrow[rr, "\lambda_C"'] & & C \end{tikzcd}$$

has to commute for all C. But that's just the condition for $[A, a_C]$ factoring through $[I, \lambda_C]$ via a_I. And moreover, suppose $[A, a_C]$ also factors through by some u. Then in particular,

$$\begin{tikzcd} & A \arrow[dl, "u"'] \arrow[dr, "a_I"] & \\ I \arrow[rr, "1_I"'] & & I \end{tikzcd}$$

commutes, and so $u = a_I$. So the factoring is unique, and $[I, \lambda_C]$ is a limit cone.

Now suppose, conversely, that $[I, \lambda_C]$ is a limit cone over the whole of $\mathscr{C}$. Then there is an arrow $\lambda_C \colon I \to C$ for each C in $\mathscr{C}$. If we can show it is unique, I will indeed be initial.

Suppose then that there is an arrow $k \colon I \to C$ for a given C. Then since $[I, \lambda_C]$ is a cone, the diagram

$$\begin{tikzcd} & I \arrow[dl, "\lambda_I"'] \arrow[dr, "\lambda_C"] & \\ I \arrow[rr, "k"'] & & C \end{tikzcd}$$

has to commute. Considering the case where $k = \lambda_C$, we see that $[I, \lambda_C]$ factors through itself via λ_I; but it also factors via 1_C, so the uniqueness of factorization entails $\lambda_I = 1_C$. Hence the diagram shows that for any $k \colon I \to C$ has to be identical to λ_C. So I is initial. □

(c) Before proceeding further, let's introduce some standard notation:

Definition 66. We denote the limit object at the vertex of a given limit cone for the diagram D with objects D_j by '$\underset{\leftarrow j}{Lim}\, D_j$'. △

Do note, however, that since limit cones are only unique up to isomorphism, different but isomorphic objects can be denoted in different contexts by '$\underset{\leftarrow j}{Lim}\, D_j$'.

The projection arrows from this limit object to the various objects D_j will then naturally be denoted '$\lambda_i \colon \underset{\leftarrow j}{Lim}\, D_j \to D_i$', and the limit cone could therefore be represented by '$[\underset{\leftarrow j}{Lim}\, D_j, \lambda_j]$'. (The direction of the arrow under '*Lim*' in this notation is perhaps unexpected, but we just have to learn to live with it.)

15.5 Colimits defined

The headline, and thoroughly predictable, story about duals is: reverse the relevant arrows and you get a definition of colimits.

So, dualizing §15.2 and wrapping everything together, we get:

Definition 67. Let D be a diagram in category $\mathscr{C}$. Then a *cocone under* D is a $\mathscr{C}$-object C, together with an arrow $c_j : D_j \to C$ for each object D_j in D, such that whenever there is an arrow $d : D_k \to D_l$ in D, the following diagram commutes:

$$\begin{tikzcd} D_k \arrow[rr, "d"'] \arrow[dr, "c_k"'] & & D_l \arrow[dl, "c_l"] \\ & C & \end{tikzcd}$$

The cocones under D form a category with objects the cocones $[C, c_j]$ and an arrow from $[C, c_j]$ to $[C', c'_j]$ being any $\mathscr{C}$-arrow $f : C \to C'$ such that $c'_j = f \circ c_j$ for all indexes j. A colimit for D is an initial object in the category of cocones under D. It is standard to denote the object at the vertex of the colimit cocone for the diagram D by '$\underset{\to j}{Lim}\, D_j$'. △

It is now routine to confirm that our earlier examples of initial objects, coproducts and co-equalizers do count as colimits.

(1) The null case where we start with the empty diagram in $\mathscr{C}$ gives rise to a cocone which is simply an object in C. So the category of cocones over the empty diagram is just the category $\mathscr{C}$ we started with, and a limit cocone is just an initial object in $\mathscr{C}$!

(2) Consider now a diagram which is just *two* objects we'll call 'D_1', 'D_2', still with no arrow between them. Then a cocone over such a diagram is just a corner from D_1, D_2 (in the sense we met in §11.8); and a limit cocone in the category of such cocones is simply a coproduct.

(3) And if we start with the diagram $D_1 \underset{d'}{\overset{d}{\rightrightarrows}} D_2$ then a limit cocone over this diagram gives rise to a co-equalizer.

15.6 Pullbacks

(a) Let's illustrate all this by briefly exploring another kind of limit (in this section) and its dual (in the next section).

A co-wedge or, as I prefer to say (§11.8), a corner D in category $\mathscr{C}$ is a diagram which can be represented like this:

$$\begin{tikzcd} & D_2 \arrow[d, "e"] \\ D_1 \arrow[r, "d"] & D_3 \end{tikzcd}$$

Now, a cone over our corner diagram has a rather familiar shape, i.e. it is a commutative square:

$$\begin{tikzcd} C \arrow[r, "c_2"] \arrow[d, "c_1"] \arrow[dr, "c_3"] & D_2 \arrow[d, "e"] \\ D_1 \arrow[r, "d"] & D_3 \end{tikzcd}$$

Though note, we needn't really draw the diagonal here, for if the sides of the square commute thus ensuring $d \circ c_1 = e \circ c_2$, then we know a diagonal $c_3 = d \circ c_1$ exists making the triangles commute.

And a limit for this type of cone will be a cone with vertex $L = \underleftarrow{Lim}_j\, D_j$ and three projections $\lambda_j \colon L \to D_j$ such that for any cone $[C, c_j]$ over D, there is a unique $u \colon C \to L$ such that this diagram commutes:

$$\begin{tikzcd} C \arrow[dr, "u"] \arrow[drr, bend left, "c_2"] \arrow[ddr, bend right, "c_1"'] & & \\ & L \arrow[r, "\lambda_2"] \arrow[d, "\lambda_1"] & D_2 \arrow[d, "e"] \\ & D_1 \arrow[r, "d"] & D_3 \end{tikzcd}$$

And note that if this commutes, there's just one possible $\lambda_3 \colon L \to D_3$ and one possible $c_3 \colon C \to D_3$ which can add to make a diagram that still commutes.

Definition 68. A limit for a corner diagram is a *pullback*. The square formed by the original corner and its limit, with or without its diagonal, is a *pullback square*. △

(b) Let's immediately have a couple of examples of pullback squares living in the category Set.

(1) Changing the labelling, consider a corner comprising three sets X, Y, Z and a pair of functions which share the same codomain, thus:

$$\begin{tikzcd} & Y \arrow[d, "g"] \\ X \arrow[r, "f"] & Z \end{tikzcd}$$

We know from the previous diagram that the limit object L must be product-like (with any wedge over X, Y factoring through the wedge with vertex L). Hence to get the other part of the diagram to commute, the pullback square must have at its apex L something isomorphic to $\{\langle x, y\rangle \in X \times Y \mid f(x) = g(y)\}$ with the obvious projection maps to X and Y.

So suppose first that in fact both X and Y are subsets of Z, and the arrows into Z are both inclusion functions. And we then get a pullback square

$$\begin{tikzcd} L \arrow[r] \arrow[d] & Y \arrow[d, "i_2"] \\ X \arrow[r, "i_1"] & Z \end{tikzcd}$$

with $L \cong \{\langle x, y\rangle \in X \times Y \mid x = y\} = \{\langle z, z\rangle \mid z \in X \cap Y\} \cong X \cap Y$. Hence, in Set, the intersection of a pair of sets is their pullback object (fixed, as usual, up to isomorphism).

(2) Take another case in Set. Suppose we have a corner as before but with $Y = Z$ and $g = 1_Z$. Then

$$L \cong \{\langle x, z\rangle \in X \times Z \mid f(x) = z\} \cong \{x \mid \exists z f(x) = z\} \cong f^{-1}[Z],$$

i.e. a pullback object for this corner is, up to isomorphism, the inverse image of Z, and we have a pullback square

$$\begin{tikzcd} f^{-1}[Z] \arrow[r] \arrow[d] & Z \arrow[d, "1_Z"] \\ X \arrow[r, "f"] & Z \end{tikzcd}$$

Hence in Set, the inverse image of a function is also a pullback object.

We will meet another simple example of pullbacks in Set in §17.4

(c) Why 'pullback'? Look at e.g. the diagram in (2). We can say that we get to $f^{-1}[Z]$ from Z by pulling back along f – or more accurately, we get to the arrow $f^{-1}[Z] \to X$ by pulling back the identity arrow on Z along f.

In this sense,

Theorem 58. *Pulling back a monomorphism yields a monomorphism.*

In other words, if we start with the same corner $X \xrightarrow{f} Z \xleftarrow{g} Y$ with g monic, and can pullback g along f to give a pullback square

$$\begin{tikzcd} L \arrow[r, "b"] \arrow[d, "a"] & Y \arrow[d, tail, "g"] \\ X \arrow[r, "f"] & Z \end{tikzcd}$$

then the resulting arrow a is monic. (Note, this does not depend on the character of f.)

Proof. Suppose, for some arrows $C \overset{j}{\underset{k}{\rightrightarrows}} L$, $a \circ j = a \circ k$. Then $g \circ b \circ j = f \circ a \circ j = f \circ a \circ k = g \circ b \circ k$. Hence, given that g is monic, $b \circ j = b \circ k$.

It follows that the two cones over the original corner, $X \xleftarrow{a \circ j} C \xrightarrow{b \circ j} Y$ and $X \xleftarrow{a \circ k} C \xrightarrow{b \circ k} Y$ are in fact the *same* cone, and hence must factor through the limit L via the same unique arrow $C \to L$. Which means $j = k$.

In sum, $a \circ j = a \circ k$ implies $j = k$, so a is monic. □

Here's another result about monomorphisms and pullbacks:

Theorem 59. *The arrow $f \colon X \to Y$ is a monomorphism in $\mathscr{C}$ if and only if the following is a pullback square:*

$$\begin{tikzcd} X \arrow[r, "1_X"] \arrow[d, "1_X"'] & X \arrow[d, "f"] \\ X \arrow[r, "f"] & Y \end{tikzcd}$$

Proof. Suppose this is pullback diagram. Then any cone $X \xleftarrow{a} C \xrightarrow{b} X$ over the corner $X \xrightarrow{f} Y \xleftarrow{f} X$ must uniquely factor through the limit with vertex X. That is to say, if $f \circ a = f \circ b$, then there is a u such that $a = 1_X \circ u$ and $b = 1_X \circ u$, hence $a = b$ – so f is monic.

Conversely, if f is monic, then given any cone $X \xleftarrow{a} C \xrightarrow{b} X$ over the original corner, $f \circ a = f \circ b$, whence $a = b$. But that means the cone factors through the cone $X \xleftarrow{1_X} X \xrightarrow{1_X} X$ via the unique a, making that cone a limit and the square a pullback square. □

(d) We've explained, up to a point, the label 'pullback'. It should now be noted in passing that a pullback is sometimes called a *fibered product* (or fibre product) because of a construction of this kind on fibre bundles in topology. Those who know some topology can chase up the details.

But here's a way of getting products into the story, using an idea that we already know about. Remind yourself what slice categories are (Defn. 25). Then:

Theorem 60. *A pullback of a corner with vertex Z in a category $\mathscr{C}$ is a product in the slice category $\mathscr{C}/Z$.*

Proof. Recall, an object of $\mathscr{C}/Z$, on the economical definition, is a $\mathscr{C}$-arrow $f \colon C \to Z$, and an arrow of $\mathscr{C}/Z$ from $f \colon X \to Z$ to $g \colon Y \to Z$ is a $\mathscr{C}$-arrow $h \colon X \to Y$ such that $f = g \circ h$ in $\mathscr{C}$.

Now the pullback of the corner with vertex Z formed by f and g in $\mathscr{C}$ is a pair of arrows $a \colon L \to X$ and $b \colon L \to Y$ such that $f \circ a = g \circ b$ $(= k)$ and which form a wedge such that any other wedge $a' \colon L' \to X$, $b' \colon L' \to Y$ such that $f \circ a' = g \circ b'$ $(= k')$ factors uniquely through it.

Looked at as a construction in $\mathscr{C}/Z$, this means taking two $\mathscr{C}/Z$-objects f and g and getting a pair of $\mathscr{C}/Z$-arrows $a \colon k \to f$, $b \colon k \to g$. And this pair of arrows forms a wedge such that any other wedge $a' \colon k' \to f$, $b' \colon k' \to g$ factors uniquely through it. In other words, the pullback in $\mathscr{C}$ constitutes a product in $\mathscr{C}/Z$. □

(e) Because of that kind of connection, product notation is often used for pullbacks, thus:

$$\begin{tikzcd} X \times_Z Y \arrow[r] \arrow[d] \arrow[dr, phantom, "\lrcorner", very near start] & Y \arrow[d] \\ X \arrow[r] & Z \end{tikzcd}$$

with the subscript giving the vertex of the corner we are taking a limit over, and with the little corner-symbol in the diagram conventionally indicating it is indeed a pullback square.

15.7 Pushouts

Pullbacks are limits for corners. What is a colimit for a corner? Check the relevant diagram and it is obviously the corner itself. So the potentially interesting dualization of the notion of a pullback is when we take the colimit of 'co-corners', i.e. wedges.

Suppose then we take a wedge D, i.e. a diagram $D_1 \xleftarrow{d} D_3 \xrightarrow{e} D_2$. A cocone under this diagram is another commutative square (omitting again the diagonal arrow which is fixed by the others).

$$\begin{tikzcd} D_3 \arrow[r, "e"] \arrow[d, "d"] & D_2 \arrow[d, "c_2"] \\ D_1 \arrow[r, "c_1"] & C \end{tikzcd}$$

And a limit cocone of this type will be a cocone with apex $L = \underset{\to j}{Lim}\, D_j$ and projections $\lambda_j\colon D_j \to L$ such that for any cocone $[C, c_j]$ under D, there is a unique $u\colon L \to C$ such that the obvious dual of the whole pullback diagram above commutes.

Definition 69. A colimit for a wedge diagram is a *pushout*. △

Now, in Set, we get the limit object for a corner diagram $X \xrightarrow{f} Z \xleftarrow{g} Y$ by taking a certain *subset* of a *product* $X \times Y$. Likewise we get the colimit object for a wedge diagram $X \xleftarrow{f} Z \xrightarrow{g} Y$ by taking a certain *quotient* of a *coproduct* $X \amalg Y$. We won't, however, pause further over this now. Though it does again illustrate how taking colimits can tend to beget messier constructions than taking limits.

16 The existence of limits

We have seen that a whole range of very familiar constructions from various areas of ordinary mathematics can be regarded as instances of taking limits or colimits of (very small) diagrams in appropriate categories. Examples so far include: forming cartesian products or logical conjunctions, taking disjoint unions or free products, quotienting out by an equivalence relation, taking intersections, taking inverse images.

Not *every* familiar kind of construction in a category $\mathscr{C}$ involves taking a limit cone or cocone in $\mathscr{C}$: we'll meet a couple of important exceptions in the next two chapters. But plainly we are mining a very rich seam here – and we are already making good on our promise to show how category theory helps reveal recurring patterns across different areas of mathematics. So what more can we say about limits?

It would get tedious to explore case by case what it takes for a category to have limits for various further kinds of diagram, even if we just stick to considering limits over tiny diagrams. But fortunately we don't need to do such a case-by-case examination. In this chapter we show that if a category has certain basic limits of kinds that we have already met, then it has *all* finite limits (or more).

16.1 Pullbacks, products and equalizers related

(a) Here's an obvious definition:

Definition 70. The category $\mathscr{C}$ *has all finite limits* if for any finite diagram D – i.e. for any diagram whose objects are D_j for indices $j \in J$, where J is a finite set – $\mathscr{C}$ has a limit over D. A category with all finite limits is said to be *finitely complete.* △

Our main target theorems for this chapter are then as follows:

Theorem 61. *If $\mathscr{C}$ has a terminal object, and has all binary products and equalizers, it is finitely complete.*

Theorem 62. *If $\mathscr{C}$ has a terminal object, and has a pullback for any corner, it is finitely complete.*

(These theorems explain why we have chosen exactly our earlier examples of limits to explore!) Later, in §16.3, we will see how that we can very easily get an analogous result for limits over infinite diagrams; but it will help to fix ideas if we initially focus on the finite case. And of course, our theorems will have the predictable duals: we briefly mention them in §16.4.

We begin though, in this section, by proving the following much more restricted versions of our two stated theorems, versions which talk just about products, equalizers and pullbacks rather than about limits more generally:

Theorem 63. *If a category $\mathscr{C}$ has all binary products and equalizers, then it has a pullback for any corner.*

Theorem 64. *If $\mathscr{C}$ has a terminal object, and has a pullback for any corner, then it has all binary products and all equalizers.*

Proving these cut-down results first will have a double pay-off:

(1) We afterwards only need prove one of Theorems 61 and 62, since in the presence of the restricted theorems, the stronger theorems evidently imply each other. We will in fact later concentrate on proving Theorem 61 (leaving Theorem 62 as a simple corollary given Theorem 64).

(2) Our proof of the restricted Theorem 63 will provide an instructive guide to how to do establish the more general Theorem 61.

(b) For those rather nobly trying, as we go along, to prove stated theorems before looking at the proofs, the results in this chapter do require a little more thought than what's gone before. Even so, a little exploration should still reveal the only reasonable proof-strategies.

Proof for Theorem 63. Given an arbitrary corner $X \xrightarrow{f} Z \xleftarrow{g} Y$ we need to construct a pullback.

There is nothing to equalize yet. So our only option is to start by constructing some product. By assumption, $\mathscr{C}$ has binary products, so there will in particular be a product $X \times Y$ and also a triple product $X \times Y \times Z$. Now in fact, when we come to generalize our proof strategy for this theorem to prove Theorem 61, it will be the product of every object in sight that we'll need to work with. But because of special features of the present case, it is enough to consider the simpler product. So: take the product $X \times Y$ with the usual projections $\pi_1 \colon X \times Y \to X$ and $\pi_2 \colon X \times Y \to Y$.

This immediately gives us parallel arrows $X \times Y \underset{g\circ\pi_2}{\overset{f\circ\pi_1}{\rightrightarrows}} Z$. And because $\mathscr{C}$ has equalizers, this parallel pair must have an equalizer $[E, e]$, for which $f \circ \pi_1 \circ e = g \circ \pi_2 \circ e$. Which in turn means that the following diagram commutes:

$$\begin{tikzcd}
E \arrow[rr, "\pi_2\circ e"] \arrow[dr, "e"] \arrow[dd, "\pi_1\circ e"'] & & Y \arrow[dd, "g"] \\
& X\times Y \arrow[ur, "\pi_2"'] \arrow[dl, "\pi_1"] & \\
X \arrow[rr, "f"] & & Z
\end{tikzcd}$$

Claim: the wedge formed by E with the projections $\pi_1 \circ e$, $\pi_2 \circ e$ is indeed a pullback of the corner $X \xrightarrow{f} Z \xleftarrow{g} Y$.

From this point, the argument is just fairly routine checking. Consider any other cone over the original corner

$$\begin{tikzcd}
C \arrow[r, "c_2"] \arrow[d, "c_1"] \arrow[dr, "c_3"] & Y \arrow[d, "g"] \\
X \arrow[r, "f"] & Z
\end{tikzcd}$$

In other words, leaving the diagonals to take care of themselves, consider any wedge $X \xleftarrow{c_1} C \xrightarrow{c_2} Y$ with $fc_1 = gc_2$: we need to show that this factors uniquely through E.

$$\begin{tikzcd}
C \arrow[dr, "v"] \arrow[drr, bend left, "u"] \arrow[drrr, bend left, "c_2"] \arrow[dddr, bend right, "c_1"'] & & & \\
& E \arrow[rr, "\pi_2\circ e"] \arrow[dr, "e"] \arrow[dd, "\pi_1\circ e"'] & & Y \arrow[dd, "g"] \\
& & X\times Y \arrow[ur, "\pi_2"'] \arrow[dl, "\pi_1"] & \\
& X \arrow[rr, "f"] & & Z
\end{tikzcd}$$

Now, our wedge certainly uniquely factors through the product $X \times Y$, so there is a unique $u \colon C \to X \times Y$ such that $c_1 = \pi_1 \circ u$, $c_2 = \pi_2 \circ u$. Hence $f \circ \pi_1 \circ u = g \circ \pi_2 \circ u$. Therefore $C \xrightarrow{u} X \times Y \underset{g\circ\pi_2}{\overset{f\circ\pi_1}{\rightrightarrows}} Z$ is a fork, which must factor uniquely through the equalizer E via some v.

That is to say, there is a $v \colon C \to E$ such that $e \circ v = u$. Hence $\pi_1 \circ e \circ v = \pi_1 \circ u = c_1$. Similarly $\pi_2 \circ e \circ v = c_2$. Therefore the wedge with vertex C indeed factors through E, as we need.

To finish the proof, we have to establish the uniqueness of the mediating arrow v. Suppose then that $v' \colon C \to E$ also makes $\pi_1 \circ e \circ v' = c_1$, $\pi_2 \circ e \circ v' = c_2$.

Then the wedge $X \xleftarrow{c_1} C \xrightarrow{c_2} Y$ factors through $X \times Y$ via $e \circ v'$; but we know the wedge factors uniquely through the product $X \times Y$ by u. Therefore $e \circ v' = u = e \circ v$.

But equalizers are monic by Theorem 49, so $v' = v$, and we are done. $\square$

Proof for Theorem 64. Given that $\mathscr{C}$ has a terminal object, what corners are guaranteed to exist, for any given X, Y? Evidently $X \longrightarrow 1 \longleftarrow Y$. So take a pullback over this corner. Applying the definition, we immediately find that a pullback for such a corner is indeed just the product $X \times Y$ with its usual projection arrows.

To show that $\mathscr{C}$ has equalizers, given that it has pullbacks and hence products, start by thinking of the parallel arrows we want to equalize, say $X \overset{f}{\underset{g}{\rightrightarrows}} Y$, as a wedge $Y \xleftarrow{f} X \xrightarrow{g} Y$. This wedge will factor uniquely via an arrow $\langle f, g\rangle$ through the product $Y \times Y$ (which exists by hypothesis).

So now consider the corner $X \xrightarrow{\langle f,g\rangle} Y \times Y \xleftarrow{\delta_Y} Y$, where δ_Y is the 'diagonal' arrow (see Defn. 49). This is nice to think about since (to arm-wave a bit!) the first arrow is evidently related to the parallel arrows we want to equalize, and the second arrow does some equalizing.

Now take this corner's pullback (the only thing to do with it!):

$$\begin{tikzcd} E \arrow[r, "q"] \arrow[d, "e"'] \arrow[dr, phantom, "\lrcorner", very near start] & Y \arrow[d, "\delta_Y"] \\ X \arrow[r, "\langle f,g\rangle"] & Y \times Y \end{tikzcd}$$

Intuitively speaking, $E \xrightarrow{e} X \xrightarrow{\langle f,g\rangle} Y \times Y$ sends something in E to a pair of equals. So, morally, $[E, e]$ ought to be an equalizer for $X \overset{f}{\underset{g}{\rightrightarrows}} Y$. And, from this point on, it is a routine proof to check that it indeed is an equalizer. Here goes:

By the commutativity of the pullback square, $\delta_Y \circ q = \langle f, g\rangle \circ e$. Appealing to Theorems 37, 39 and 40, it follows that $\langle q, q\rangle = \langle f \circ e, g \circ e\rangle$, and hence $f \circ e = q = g \circ e$. Therefore $E \xrightarrow{e} X \overset{f}{\underset{g}{\rightrightarrows}} Y$ is a fork. It remains to show that it is a limit fork.

Take any other fork $C \xrightarrow{c} X \overset{f}{\underset{g}{\rightrightarrows}} Y$. The wedge $X \xleftarrow{c} C \overset{f\circ c}{\underset{g\circ c}{\longrightarrow}} Y$ must factor through E (because E is the vertex of the pullback) via a unique mediating arrow v:

$$\begin{tikzcd}
C \arrow[drr, bend left, "f\circ c", "g\circ c"'] \arrow[dr, dashed, "v"] \arrow[ddr, bend right, "c"'] & & \\
& E \arrow[r, "f\circ e", "g\circ e"'] \arrow[d, "e"] & Y \arrow[d, "\delta_Y"] \\
& X \arrow[r, "\langle f, g\rangle"] & Y \times Y
\end{tikzcd}$$

It follows that v makes this diagram commute:

$$\begin{tikzcd}
C \arrow[dr, "c"] \arrow[dd, "v"'] & & \\
& X \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & Y \\
E \arrow[ur, "e"'] & &
\end{tikzcd}$$

And any $v' : C \to E$ which makes the latter diagram commute will also be a mediating arrow making the previous diagram commute, so $v' = v$ by uniqueness of mediators in pullback diagrams. Hence $[E, e]$ is indeed an equalizer. $\square$

16.2 Categories with all finite limits

Our target now is to establish the promised main result:

Theorem 61. *If $\mathscr{C}$ has a terminal object, and has all binary products and equalizers, it is finitely complete.*

This is indeed our first Big Result. To prove it, we are going to generalize the strategy pursued in proving the cut-down result that having binary products and equalizers implies at least having pullbacks. So, the outline plan is this:

> Given a finite diagram D, we start by forming the product P of the objects from D (which we can do since $\mathscr{C}$ has all finite products). We then find some appropriate parallel arrows out of this product P. Then we take an equalizer $[E, e]$ of these arrows (which we can do since $\mathscr{C}$ has all equalizers). We then aim to use E as the vertex of the desired limit cone over the diagram D on the model of the proof of Theorem 63.

The devil, of course, is in the details! And to be frank, you won't lose much if you skip past them.

Consider again the proof of Theorem 63. There we started with a mini-diagram D, i.e. a corner with two arrows sharing a target, $f : X \to Z$, $g : Y \to Z$. We got parallel arrows which share a source as well as a target by taking a product, thereby getting $X \times Y \underset{g\circ\pi_2}{\overset{f\circ\pi_1}{\rightrightarrows}} Z$. And *then* we could look for an equalizer.

Now, in an arbitrary finite diagram D there could be lots of arrows of the kind $d\colon D_k \to D_l$ with a variety of different sources and targets. But we still want to end up by constructing out of them a pair of parallel arrows with the same source and same target so that we can then take an equalizer. To construct the single source and single target we use products again.

At the source end, we have two apparent options – we could take the product $[P, p_j]$ of *all* the objects in D, or we could take the product $[P', p'_j]$ of those objects in D which are sources of arrows in D. In turns out, after a bit of exploration, that in the general case the first is the one to go for. At the target end, the natural thing to do is to define $[Q, q_l]$ as the product of all the objects D_l which are targets for arrows in D. (We can make these constructions of course as we are assuming we are working in a category with all finite products).

So the name of the game is now to define a pair of parallel arrows

$$P \underset{w}{\overset{v}{\rightrightarrows}} Q$$

which we are going to equalize by some $[E, e]$.

However, there are in fact only two naturally arising arrows from P to Q.

(1) Consider first a certain cone over the objects D_l which contribute to the product Q – namely, the cone with vertex P and with an arrow $p_l\colon P \to D_l$ for each D_l. This cone (by definition of the product $[Q, q_l]$) must factor through the product by a unique mediating arrow v, so that $p_l = q_l \circ v$ for each l.

(2) Consider secondly the cone over the same objects with vertex P and an arrow $d \circ p_k\colon P \to D_l$ for each arrow $d\colon D_k \to D_l$ in D. This cone too must factor through the product $[Q, q_l]$ by a unique mediating arrow w, so that $d \circ p_k = q_l \circ w$ for each arrow $d\colon D_k \to D_l$.

Since we are assuming that all parallel arrows have equalizers in $\mathscr{C}$, we can take the equalizer of v and w, namely $[E, e]$.

And now the big claim, modelled exactly on the key claim in our proof of Theorem 63: $[E, p_j \circ e]$ will be a limit cone over D.

Let's state this as a theorem:

Theorem 65. *Let D be a finite diagram in a category $\mathscr{C}$ which has a terminal object, binary products and equalizers. Let $[P, p_j]$ be the product of the objects D_j in D, and $[Q, q_l]$ be the product of the objects D_l which are targets of arrows in D. Then there are arrows*

$$P \underset{w}{\overset{v}{\rightrightarrows}} Q$$

such that the following diagrams commute for each $d\colon D_k \to D_l$:

$$\begin{tikzcd} P \arrow[r, "v"] \arrow[dr, "p_l"'] & Q \arrow[d, "q_l"] \\ & D_l \end{tikzcd} \qquad \begin{tikzcd} P \arrow[r, "w"] \arrow[d, "p_k"'] & Q \arrow[d, "q_l"] \\ D_k \arrow[r, "d"] & D_l \end{tikzcd}$$

Let the equalizer of v and w be $[E, e]$. Then $[E, p_j \circ e]$ will be a limit cone over D in $\mathscr{C}$.

Proof. We have already shown that v and w exist such that the given diagrams commute and that an equalizer $[E, e]$ for them exists. So next we confirm $[E, p_j \circ e]$ is a cone. Suppose then that there is an arrow $d\colon D_k \to D_l$. For a cone, we require $d \circ p_k \circ e = p_l \circ e$.

But indeed $d \circ p_k \circ e = q_l \circ w \circ e = q_l \circ v \circ e = p_l \circ e$, where the inner equation holds because e is an equalizer of v and w and the outer equations are given by the commuting diagrams above.

Second we show that $[E, p_j \circ e]$ is a limit cone. So suppose $[C, c_j]$ is any other cone over D. Then there must be a unique $u\colon C \to P$ such that every c_j factors through the product and we have $c_j = p_j \circ u$.

Since $[C, c_j]$ is a cone, for any $d\colon D_k \to D_l$ in D we have $d \circ c_k = c_l$. Hence $d \circ p_k \circ u = p_l \circ u$, and hence for each q_l, $q_l \circ w \circ u = q_l \circ v \circ u$. But then we can apply the obvious generalized version of Theorem 43, and conclude that $w \circ u = v \circ u$. Which means that

$$\begin{tikzcd} C \arrow[r, "u"] & P \arrow[r, shift left, "v"] \arrow[r, shift right, "w"'] & Q \end{tikzcd}$$

is a fork, which must therefore uniquely factor through the equalizer $[E, e]$. That is to say, there is a unique $s\colon C \to E$ such that $u = e \circ s$, and hence for all j, $c_j = p_j \circ u = p_j \circ e \circ s$. That is to say, $[C, c_j]$ factors uniquely through $[E, p_j \circ e]$ via s. Therefore $[E, p_j \circ e]$ is indeed a limit cone. □

This more detailed result of course trivially implies the less specific Theorem 61. And that in turn, given Theorem 64, gives us Theorem 62. So we are done.

Given ingredients from our previous discussions, since the categories in question have terminal objects, binary products and equalizers,

Theorem 66. Set *and* FinSet *are finitely complete, as are categories of algebraic structured sets such as* Mon*,* Grp*,* Ab*,* Rng*. Similarly* Top *is finitely complete.*

While e.g. a poset-as-a-category may lack many products and hence not be finitely complete.

16.3 Infinite limits

Now we extend our key Theorem 61 to reach beyond the finite case. First, we need:

Definition 71. The category $\mathscr{C}$ *has all small limits* if for any diagram D whose objects are D_j for indices $j \in I$, for some set I, then $\mathscr{C}$ has a limit over D. A category with all small limits is also said to be *complete*. △

Again, as in talking of small products, small limits can be huge – we just mean no-bigger-than-set-sized. An easy inspection of the proof in the last section shows that, given our requirement that the objects in a diagram D can be indexed by a set, the argument will continue to go through just as before – assuming, that is, that we are still dealing with a category like Set which has products for all set-sized collections of objects (so we can still form the products $[P, p_j]$ and $[Q, q_l]$) and also all equalizers.

Hence, without further ado, we can state:

Theorem 67. *If $\mathscr{C}$ has all small products and has equalizers, then it has all small limits, i.e. is complete.*

We can similarly extend Theorem 66 to show that

Theorem 68. Set *is complete – as are the categories of structured sets* Mon*,* Grp*,* Ab*,* Rng*.* Top *too is complete.*

We have already met a category which, by contrast, is finitely complete but is evidently not complete, namely FinSet.

16.4 Dualizing again

Needless to say by this stage, our results in this chapter dualize in obvious ways. Thus we need not delay over the further explanations and proofs of

Theorem 69. *If $\mathscr{C}$ has initial objects, binary coproducts and co-equalizers, then it has all finite colimits, i.e. is finitely cocomplete. If $\mathscr{C}$ has all small coproducts and has co-equalizers, then it has all small colimits, i.e. is cocomplete.*

Theorem 70. Set *is cocomplete – as are the categories of structured sets* Mon*,* Grp*,* Ab*,* Rng*.* Top *too is cocomplete.*

But note that a category can of course be (finitely) complete without being (finitely) cocomplete and vice versa. For a generic source of examples, take again a poset $(P, \preccurlyeq)$ considered as a category. This automatically has all equalizers (and coequalizers) – see §14.1 Ex. (5). But it will have other limits (colimits) depending on which products (coproducts) exists, i.e. which sets of elements have suprema (infima). For a simple case, take a poset with a maximum element and such that every pair of elements has a supremum: then considered as a category it has all finite limits (but maybe not infinite ones). But it need not have a minimal element and/or infima for all pairs of objects: hence it can lack some finite colimits despite having all finite limits.

17 Subobjects

We have seen how to treat the results of various familiar operations, such as forming products or taking quotients, as limits or colimits. But as we said at the beginning of the last chapter, not every familiar kind of construction when treated categorically straightforwardly involves taking a limit or colimit. We'll consider a couple of examples. In the next chapter, we will look at exponentials. But first, in this chapter, we consider taking subobjects (as in subsets, subgroups, subspaces, etc.).

17.1 Subsets revisited

(a) We start though in familiar vein, still thinking about limits (or more particularly, equalizers). In §14.1, we saw that in Set, given two parallel arrows from an object X, a certain subset of X together with the trivial inclusion function provides an equalizer for those arrows – and §14.2 tells us that this is the unique equalizer, up to isomorphism.

We now note that a reverse result holds too:

Theorem 71. *In* Set*, any subset S of X, taken together with its natural inclusion map $i\colon S \to X$, forms an equalizer for certain parallel arrows from X.*

Proof. Let Ω be some *truth-value object*, i.e. a two-object set with members identified as *true* and *false*. Setting $\Omega = \{0, 1\}$, with 1 as *true* and 0 as *false* is of course the choice hallowed by tradition.

Then a subset $S \subseteq X$ has an associated *characteristic function* $s\colon X \to \Omega$ which sends $x \in X$ to *true* if $x \in S$ and sends x to *false* otherwise.

Let $t\colon 1 \to \Omega$ be the map which sends the sole object in the singleton 1 to *true*, and let $t!$ be the composite map $X \xrightarrow{!_X} 1 \xrightarrow{t} \Omega$.

We show that $[S, i]$ is an equalizer for the parallel arrows $s, t!\colon X \to \Omega$. First, it is trivial that $s \circ i = t! \circ i$, so as required $S \xrightarrow{i} X \overset{s}{\underset{t!}{\rightrightarrows}} \Omega$ is indeed a fork. It remains to confirm that any upper fork in this next diagram factors through the lower fork via a unique mediating u:

$$\begin{tikzcd}
R \arrow[dr, "f"] \arrow[dd, dashed, "u"'] & & \\
& X \arrow[r, shift left, "s"] \arrow[r, shift right, "t!"'] & \Omega \\
S \arrow[ur, "i"'] & &
\end{tikzcd}$$

Recycling an argument we've seen before, since $s \circ f = t! \circ f$ by assumption, it is immediate that $f[R] \subseteq S \subseteq X$. Hence, if we define $u\colon R \to S$ to agree with $f\colon R \to X$ on all inputs, then the diagram commutes. And this u is evidently the only arrow to give us a commuting diagram. $\square$

(b) Now, given these results relating subsets to certain equalizers, we might perhaps expect to meet at this point a general account of subobjects in terms of equalizers. And yes, we do indeed get a general connection, in appropriate categories, between subobjects and limits involving so-called truth-value objects like Ω. However, as we will later explain in §17.4, *this connection has to be read as fixing the general notion of a truth-value object in terms of the notion of a subobject rather than the other way around.* Hence we need a prior account of subobjects: we give it in the next section.

17.2 Subobjects as monic arrows

(a) Work in Set again. And note that any injective set-function $f\colon S \to X$ sets up a bijection $j\colon S \xrightarrow{\sim} f[S] \subseteq X$. In other words, any monic arrow $S \rightarrowtail X$ generates an isomorphism between S and a subset of X. So, if we only care about identifications up to isomorphism (the typical situation in category theory), then an object S together with a monic arrow $S \rightarrowtail X$ might as well be treated as a subobject of X in Set. And then noting that an arrow determines its source so we needn't really mention that separately, and generalizing to other categories, this suggests a very simple definition:

Definition 72. A *subobject*$_1$ of an object X in the category $\mathscr{C}$ is just a monomorphism $S \rightarrowtail X$. $\triangle$

(b) Subobjects are arrows and so we can't immediately talk about subobjects of subobjects. But there is a natural definition of subobject inclusion:

Definition 73. If $f\colon A \rightarrowtail X$ and $g\colon B \rightarrowtail X$ are subobjects$_1$ of X, then f *is included in* g, in symbols $f \subseteq g$ iff f factors through g, i.e. there is an arrow $h\colon A \to B$ such that $f = g \circ h$. $\triangle$

Question: Wouldn't it be more natural to also require the mediating arrow h to be monic too? Answer: We don't need to write that into the definition because h *is* monic by Theorem 16 (3).

It is then trivial to check that inclusion of subobjects, so defined, is reflexive and transitive. So far so good.

17.3 Subobjects as isomorphism classes

(a) However, if we adopt our first definitions of subobject and subobject-inclusion, we get some oddities.

(1) In Set, for example, the singleton set $\{1\}$ would have not two subobjects as you might expect (the empty set and itself) but infinitely many. Indeed it would have too many subobjects to form a set, since there are as many monic arrows $S \to \{1\}$ as there are singleton sets S, and there are too many singletons to form a set.

(2) Again in Set for example, two subobjects of X, $f\colon A \rightarrowtail X$ and $g\colon B \rightarrowtail X$, can be such that $f \subseteq g$ and $g \subseteq f$ even though $f \neq g$.

We know from Theorem 23 that if $f \subseteq g$ and $g \subseteq f$, i.e. if the two arrows factor through each other, then they factor via an isomorphism, so we'll have $A \cong B$. But we needn't have $A = B$ which would be required for the arrows f, g to be identical. So the subobjects of X ordered by inclusion needn't form a poset.

Arguably, neither is a happy consequence of our definitions so far.

(b) An obvious suggestion for keeping tallies of subobjects under control is to say that the monic arrows $f\colon S \rightarrowtail X$, $g\colon S' \rightarrowtail X$ should count as representing the same subobject of X iff $S \cong S'$. Or by Theorem 23 again, we could equivalently say:

Definition 74. A *subobject*$_2$ of X is a class of *subobjects*$_1$ of X which factor through each other. △

We can then show that

Theorem 72. *In* Set*, the subobjects*$_2$ *of* X *correspond one-to-one with the subsets of* X*.*

Proof. First, we remark that monic arrows $f\colon S \rightarrowtail X$, $g\colon S' \rightarrowtail X$ belong to the same subobject$_2$ of X if and only if f and g have the same image.

For suppose there is an isomorphism $i\colon S \to S'$ such that $f = g \circ i$. Therefore if $x \in f[S]$, then there is an $s \in S$ such that $x = f(s) = g(i(s))$ where $i(s) \in S'$, so $x \in g[S']$. Hence $f[S] \subseteq g[S']$. Likewise $g[S'] \subseteq f[S]$. Hence if f and g belong to the same subobject$_2$, they have the same image.

Conversely, suppose the monic arrows $f\colon S \rightarrowtail X$, $g\colon S' \rightarrowtail X$ have the same image. In Set monics are injections; so we can define a map i which sends s to the unique s' such that $g(s') = f(s)$, and then trivially $f = g \circ i$. Likewise g factors through f, and hence f and g belong to the same subobject$_2$ of X.

Now take any subset $S \subseteq X$. There is a corresponding monic inclusion function $f_S\colon S \rightarrowtail X$. So consider the map that sends a subset S to the subobject$_2$ which contains f_S. This is one-one and onto. It is one-one because if the subobject$_2$ which contains f_S is the subobject$_2$ which contains $f_{S'}$, then f_S has the same image as $f_{S'}$, and being inclusions it follows that $S = S'$. It is onto because the functions in any subobject$_2$ of X with the shared image $S \subseteq X$ will contain such an f_S. $\Box$

We get parallel results in other categories too. For example, subobjects$_2$ in the category Grp correspond one-to-one to subgroups, in the category $\mathsf{Vect_k}$ correspond to vector subspaces, and so on. (But topologists might like to work out why in Top the subobjects$_2$ don't straightforwardly correspond to subspaces.)

Suppose we now add

Definition 75. If $[\![f]\!]$ and $[\![g]\!]$ are subobjects$_2$ of X, respectively the isomorphism classes containing $f\colon A \to X$ and $g\colon B \to X$, then $[\![f]\!]$ *is included in* $[\![g]\!]$, in symbols $[\![f]\!] \Subset [\![g]\!]$ iff $f \subseteq g$. $\triangle$

It is routine to check that this definition of an order relation on isomorphism classes is independent of the chosen exemplar of the class. And then inclusion so defined is indeed reflexive, and $\Subset$ is a partial order – and hence the subobjects$_2$ of X, with this ordering, form a poset as intuitively we want.

(c) Given the way subobjects$_2$ more naturally line up with subsets, subgroups, etc., as normally conceived, many authors prefer Defn. 74 as their official categorial account of subobjects – see for example (Goldblatt 2006, p. 77), Leinster (2014, Ex. 5.1.40). But some authors prefer the first simple definition of subobject as monics as is given by e.g. Awodey (2006, §5.1). While Johnstone (2002, p. 18) says that 'like many writers on category theory' he will be deliberately ambiguous between the two definitions in his use of 'subobject', which sounds an unpromising line but in practice works quite well!

17.4 Subobjects, equalizers, and pullbacks

(a) How does our official account of subobjects in either form relate to our previous thought that we can treat subobjects, or at least subsets, as special equalizers?

Working in Set again, it is easily checked that if $i\colon S \rightarrowtail X$ is any monic arrow into X (not necessarily an inclusion map), and $s\colon X \to \Omega$ is now the map that sends $x \in X$ to *true* iff $x \in i[S]$, then $S \overset{i}{\rightarrowtail} X \underset{t!}{\overset{s}{\rightrightarrows}} \Omega$ is still a fork. Indeed, it is a limit fork such that any other fork through $s, t!$ factors uniquely through it. For take again the diagram

$$\begin{tikzcd}
R \arrow[dr, "f"] \arrow[dd, dashed, "u"'] & & \\
& X \arrow[r, shift left, "s"] \arrow[r, shift right, "t!"'] & \Omega \\
S \arrow[ur, tail, "i"'] & &
\end{tikzcd}$$

Since $s \circ f = t! \circ f$ by assumption, it is immediate that $f[R] \subseteq i[S] \subseteq X$. Hence, if we define u to send an object $r \in R$ to the pre-image of $f(r)$ under i (which is unique since i is monic), then the diagram commutes. And this u is evidently the only arrow to give us a commuting diagram. So, the subobject$_1$ $i\colon S \rightarrowtail X$ (together with its source) is still an equalizer in Set (and so a subobject$_2$ can be thought of as a class of equalizers).

(b) It is now interesting to note an equivalent way of putting the situation in Set. For note that the map $t! \circ i\colon S \to \Omega$, which sends everything in S to the value *true*, is of course trivially equal to the composite map $S \xrightarrow{!_S} 1 \xrightarrow{t} \Omega$ with 1 a terminal object in the category. Similarly for the map $t! \circ f\colon R \to \Omega$. Hence, the claim that $[S, i]$ equalizes $s, t!$ in Set is equivalent to the following. For any $f\colon R \to X$ such that $s \circ f = t! \circ 1_R$ there is a unique u which makes the whole diagram commute:

$$\begin{tikzcd}
R \arrow[dr, dashed, "u"] \arrow[drr, bend left, "!_R"] \arrow[ddr, bend right, "f"'] & & \\
& S \arrow[r, "!_S"] \arrow[d, tail, "i"] & 1 \arrow[d, "t"] \\
& X \arrow[r, "s"] & \Omega
\end{tikzcd}$$

And after our work in §15.6, we know a snappy way of putting that: *the lower square is a pullback square.*

(c) Now, we can indeed carry this last idea across to other categories. We can say that, in a category $\mathscr{C}$ with a terminal object, then given a truth-value object Ω and a *true*-selecting map $t\colon 1 \to \Omega$, then for any subobject$_1$ of X, i.e. for any monic $i\colon S \rightarrowtail X$, there is a unique 'characteristic function' $s\colon X \to \Omega$ which makes

$$\begin{tikzcd}
S \arrow[r, "!_S"] \arrow[d, tail, "i"] & 1 \arrow[d, "t"] \\
X \arrow[r, "s"] & \Omega
\end{tikzcd}$$

a pullback square.

However, now to pick up the thought we trailed at the end of §17.1, we *can't* regard this as an alternative definition of a subobject in terms of a limit – since that would presuppose we *already* have a handle on a general notion of truth-value object, and we don't. Rather, we need to look at things the other way about. What we have here is a general characterization of what can sensibly be counted as a 'truth-value object' Ω and an associated *true*-selecting map $t\colon 1 \to \Omega$ in a category $\mathscr{C}$ with a terminal object. We define such things across categories by requiring that they work as 'subobject classifiers', i.e. by requiring they together ensure the displayed square is a pullback for a unique s given any monic subobject arrow i. We will eventually return to this point.

17.5 Elements and subobjects

A final remark. Earlier we noted that, in Set, functions $\vec{x}\colon 1 \to X$ correspond one-to-one with elements of X, and so started treating arrows $\vec{x}$ as the categorial version of *set elements*. And inspired by that, we then called arrows $f\colon S \to X$ *generalized elements* of X. Yet now we have some of those same arrows, namely the monic ones, $i\colon S \rightarrowtail X$ being offered as the categorial version of *subsets*.

Now, one of the things that is drilled into us early is that we must very sharply distinguish the notion of element from the notion of subset. Yet here we seem to be categorially assimilating the notions – elements and subsets of X both get rendered by arrows in Set, and a subset-of-X-qua-subobject will count as a special kind of (generalized) element-of-X. Is this a worry? For the moment we just flag up the apparent anomaly: this is something else we will want to say more about later, in talking about the category theorist's view of sets more generally.

18 Exponentials

We will eventually have much more to say about limits, and in particular about how they can get 'carried over' from one category to another by maps *between* categories. For the moment, however, we pause to consider another categorial notion that applies *within* a category, one that is also defined in terms of a 'universal mapping property', but which isn't straightforwardly a limit – namely the notion of an exponential.

18.1 Two-place functions

First however, let's pause to revisit the issue of two-place functions in category theory which we shelved in §11.1 (b).

It might in fact be helpful to recall how a couple of other familiar frameworks manage to do without genuine multi-place functions by providing workable substitutes:

(1) Set-theoretic orthodoxy models a two-place total function from numbers to numbers (addition, say) as a function $f\colon \mathbb{N}^2 \to \mathbb{N}$. Here, $\mathbb{N}^2$ is the cartesian product of $\mathbb{N}$ with itself, i.e. is the set of ordered pairs of numbers. And an ordered pair is *one* thing not two things. So a function $f\colon \mathbb{N}^2 \to \mathbb{N}$ is in fact strictly speaking a *unary function*, a function that maps *one* argument, an ordered pair object, to a value, not a real binary function.

Of course, in set-theory, for any two things there is a pair-object that codes for them – we usually choose a Kuratowski pair – and so we can indeed trade in a function from two objects for a related function from the corresponding pair-object. And standard notational choices can make the trade quite invisible. Suppose we adopt, as we earlier did, the modern convention of using '(m, n)' as our notation for the ordered pair of m with n. Then '$f(m, n)$' invites being parsed either way, as representing a two-place function $f(\cdot\,, \cdot)$ with arguments m and n, or as a corresponding one-place function $f\cdot$ with the single argument, the pair (m, n). But note: the fact that the trade between the two-place and the one-place function is notationally glossed over doesn't mean that it isn't being made.

(2) Versions of type theory deal with two-place functions in a different way, by a type-shifting trick. Addition for example – naively a binary function

that just deals in numbers – is traded in for a function of the type $N \to (N \to N)$. This is a unary function which takes one number (of type N) and outputs something of a higher type, i.e. a unary *function* (of type $N \to N$). We then get from two numbers as input to a numerical output in two steps, by feeding the first number to a function which delivers another function as output and then feeding the second number to the second function.

This so-called 'currying' trick of course is also perfectly adequate for certain formal purposes. But again a trade is being made. Here's a revealing quote from *A Gentle Introduction to Haskell* on the haskell.org site (Haskell is one those programming languages where what we might think of naturally as binary functions are curried):

> Consider this definition of a function which adds its two arguments:
>
> add :: Integer → Integer → Integer
>
> add x y = x + y

So we have the declaration of type – we are told that add sends a number to a function from numbers to numbers. We are then told how this curried function acts ... but how? By appeal, of course, to our prior understanding of the familiar school-room two-place addition function! The binary function remains a rung on the ladder by which we climb to an understanding of what's going on in the likes of Haskell (even if we propose to throw away the ladder after we've climbed it).

So now back to categories. We don't have native binary morphisms in category theory. Nor do we get straightforward currying within a category, at least in the sense that we won't have an arrow inside a category whose target is another *arrow* of that category (though we will meet a reflection of the idea of currying in this chapter). Hence, as we have already seen, then, we need to use a version of the set-theoretic trick. We can in a noncircular way give a categorial treatment of pair-objects as ingredients of products. And with such objects now to hand, an arrow of the kind $f\colon X \times Y \to Z$ is indeed available to do duty for a two-place function from an object in X and an object in Y to a value in Z. So this, as already announced, will have to be our implementation device.

18.2 Exponentials defined

(a) It is standard to use the notation 'C^B' in set theory to denote the set of functions $f\colon B \to C$. But why is the exponential notation apt?

Here is one reason. 'C^n' is of course natural notation for the n-times Cartesian product of C with itself, i.e. the set of n-tuples of elements from C. But an n-tuple of C-elements can be regarded as equivalent to a function from an indexing set n, i.e. from the set $\{0, 1, 2, \ldots, n-1\}$, to C. Therefore C^n, the set of n-tuples, can indeed be thought of as equivalent to C^n, re-defined as the set of functions

$f\colon n \to C$. And is then natural to extend this notation to the case where the indexing set B is no longer a number n.

Four more observations, still in informal set-theory:

(1) For all sets B, C there is a set C^B.

(2) There is a *two*-place evaluation function $ev(\cdot\,,\cdot)$ which takes an element $f \in C^B$ and an element $b \in B$, evaluates the first argument f at the selected second argument b, and so returns the value $f(b) \in C$.

(3) Take any *two*-place function $g(\cdot\,,\cdot)$ that maps an element of A and an element of B to some value in C: informally notate that binary function $g\colon A, B \to C$. Then, fixing an element $a \in A$ determines a derived *one*-place function $g(a,\cdot)\colon B \to C$.

(4) So, for any such binary $g\colon A, B \to C$ there is a unique associated one-place function, its *exponential transpose* $\overline{g}\colon A \to C^B$, which sends $a \in A$ to $g(a,\cdot)\colon B \to C$. We then have $ev(\overline{g}(a), b) = g(a, b)$.

These elementary observations pretty much tell us how to characterize categorially an 'exponential object' C^B in Set. We simply need to remember that categorially we regiment two-place functions as arrows from products.

Hence, we can say this. In Set, for all B, C, there is an object C^B and an arrow $ev\colon C^B \times B \to C$ such that for any arrow $g\colon A \times B \to C$, there is a *unique* $\overline{g}\colon A \to C^B$ (g's exponential transpose) which makes the following diagram commute:

(Exp)
$$\begin{tikzcd}
A \times B \arrow[dr, "g"] \arrow[dd, dashed, "\overline{g}\times 1_B"'] & \\
& C \\
C^B \times B \arrow[ur, "ev"'] &
\end{tikzcd}$$

The product arrow $\overline{g} \times 1_B$ here, which acts componentwise on pairs in $A \times B$, is defined categorially in §12.3.

(b) Now generalize in the obvious way:

Definition 76. Assume $\mathscr{C}$ is a category with binary products. Then $[C^B, ev]$, with C^B an object and arrow $ev\colon C^B \times B \to C$, forms an *exponential* of C by B in $\mathscr{C}$ iff the following holds, with all the mentioned objects and arrows being in $\mathscr{C}$: for every object A and arrow $g\colon A \times B \to C$, there is a unique arrow $\overline{g}\colon A \to C^B$ (g's transpose) such that $ev \circ \overline{g} \times 1_B = g$, i.e. such that the diagram (Exp) commutes. △

Note that, as with products, the square-bracket notation here is once more just punctuation for readability's sake. More importantly, note that if we change the objects B, C the evaluation arrow $ev\colon C^B \times B \to C$ changes, since the source and/or target will change. (It might occasionally help to think of the notation 'ev' as really being lazy shorthand for something like '$ev_{C,B}$'.)

Definition 77. A category $\mathscr{C}$ *has all exponentials* iff for all $\mathscr{C}$-objects B, C, there is a corresponding exponential $[C^B, ev]$. △

(c) Exponentials in $\mathscr{C}$ aren't defined in terms of a type of cones (or cocone) in $\mathscr{C}$. But just as a limit cone over D is defined in terms of every cone over D 'factoring through' the limit via a unique arrow, so an exponential of C with B is defined in terms of every arrow from some $A \times B$ to C 'factoring through' the exponential via a unique arrow. In short, limits and exponentials alike are defined in terms of every relevant item factoring through via a unique map. That's why we can speak of both the properties of being a limit and being an exponential as examples of universal mapping properties.

18.3 Examples of exponentials

Let's immediately give three easy examples of categories which it is easy to see have exponentials:

(1) Defns. 76 and 77 were purpose-built to ensure that Set counts as having all exponentials – a categorial exponential of C by B is provided by the set C^B (in the standard set-theoretic sense) equipped with the set function ev as described before. But we can note now that this construction applies equally in FinSet, the category of finite sets, since the set C^B is finite if both B and C are finite, and hence C^B is also in FinSet. Therefore FinSet has all exponentials.

(2) In §11.1 (5) we met the category $\mathsf{Prop}_{\mathcal{L}}$ whose objects are wffs of a given first-order language $\mathcal{L}$, and where there is a unique arrow from A to B iff $A \vDash B$. Assuming $\mathcal{L}$ has the usual rules for conjunction and implication, then for any B, C, the conditional $B \to C$ provides an exponential object C^B, with the corresponding evaluation arrow $ev : C^B \times B \to C$ reflecting the modus ponens entailment $B \to C, B \vDash C$.

Why does this work? Recall that products in Prop_L are conjunctions. And note that, given $A \wedge B \vDash C$, then by the standard rules $A \vDash B \to C$ and hence – given the trivial $B \vDash B$ – we have $A \wedge B \vDash (B \to C) \wedge B$. We therefore get the required commuting diagram of this shape,

$$\begin{tikzcd}
A \wedge B \arrow[dr] \arrow[d, dashed] & \\
(B \to C) \wedge B \arrow[r] & C
\end{tikzcd}$$

where the down arrow is the product of the implication arrow from A to $B \to C$ and the trivial entailment from B to B.

(3) Relatedly, take a Boolean algebra $(B, \neg, \wedge, \vee, 0, 1)$, and put $a \leqslant b =_{\text{def}} (a \wedge b) = a$ for all $a, b \in B$. Then, treated as a partially ordered set with

that order, the Boolean algebra corresponds to a poset category, with a unique arrow between a and b when $a \leqslant b$. In this category, $a \wedge b$, with the only possible projection arrows, is the categorial product of a, b

Such a poset category based on a Boolean algebra has an exponential for each pair of objects, namely (to use a suggestive notation) the object $b \Rightarrow c =_{\mathrm{def}} \neg b \vee c$, together with the evaluation arrow ev the unique arrow corresponding to $(b \Rightarrow c) \wedge b \leqslant c$.

To check this claim, we need first to show that we have indeed well-defined the evaluation arrow ev for every b, c, i.e. show that we always have $(b \Rightarrow c) \wedge b \leqslant c$. However, as we want,

$$(\neg b \vee c) \wedge b = (\neg b \wedge b) \vee (c \wedge b) = 0 \vee (c \wedge b) = (c \wedge b) \leqslant c$$

by Boolean rules and the definition of $\leqslant$.

Second, we need to verify that the analogous diagram to the last one commutes, which crucially involves showing that if $a \wedge b \leqslant c$ then $a \wedge b \leqslant (b \Rightarrow c) \wedge b$. That's more Boolean algebra, which can perhaps be left as a brain-teaser.

So Boolean-algebras-treated-as-poset-categories have all exponentials. Working through the details, however, we find that the required proofs *don't* call on the Boolean principle $\neg\neg a = a$, so the claim about Boolean algebras can be strengthened to the claim that Heyting-algebras-treated-as-poset-categories have all exponentials (where a Heyting algebra is, in effect, what you get when you drop the 'double negation' rule from the Boolean case: we will return later to talk about this important case from logic).

Now these first examples are of categories which have *all* exponentials. But of course, a category may lack exponentials entirely (for example, take a poset category with no products). Or it may have just trivial exponentials (we'll see in the next section that, if a category has a terminal object 1, then it will automatically have at least the trivial exponentials X^1 and 1^X). And as we'll now see, it can also be the case that a category has *some* non-trivial exponentials, though not *all* exponentials.

(4) For an initial toy example, we might consider the poset category arising from a five-element non-distributive lattice, which has the following arrows (plus the necessary identity arrows and composites):

$$\begin{tikzcd} & X \arrow[rr] & & Y \arrow[dr] & \\ 0 \arrow[ur] \arrow[drr] & & & & 1 \\ & & Z \arrow[urr] & & \end{tikzcd}$$

In this category, X^Y doesn't exist, but $X^Z = Y$. It is perhaps a useful reality check to pause to show this:

Proof. Consider these two putative diagrams as imagined instances of (Exp):

$$\begin{tikzcd}
A \times Y \arrow[d, "\overline{g}\times 1_Y"'] \arrow[dr, "g"] & \\
X^Y \times Y \arrow[r, "ev"'] & X
\end{tikzcd}
\qquad
\begin{tikzcd}
A \times Z \arrow[d, "\overline{g}\times 1_Z"'] \arrow[dr, "g"] & \\
X^Z \times Z \arrow[r, "ev"'] & X
\end{tikzcd}$$

Suppose there is an exponential object X^Y. Then for every arrow $g\colon A \times Y \to X$ there must exist a unique $\overline{g}\colon A \to X^Y$ making the left-hand diagram commute. Since $Z \times Y = 0$, there is indeed an arrow $g_1\colon Z \times Y \to X$; and since $X \times Y = X$ there is an arrow $g_2\colon X \times Y \to X$. Therefore we need arrows $\overline{g_1}\colon Z \to X^Y$ and $\overline{g_2}\colon X \to X^Y$, which implies $X^Y = 1$. But $X^Y \times Y = Y$, and hence there is no possible arrow $ev\colon X^Y \times Y \to X$. Hence there is no exponential object X^Y, and the left-hand diagram is a mirage!

Now put $X^Z = Y$, with the arrow ev the sole arrow from 0 to X. Then it is easily checked that for each arrow $g\colon A\times Z \to X$ (that requires $A = 0, X$, or Y) there is a corresponding unique $\overline{g}\colon A \to Y$ making the diagram on the right commute. Just remember we are in a poset category so arrows with the same source and target are equal. □

(5) Consider next Count, the category of sets which are no larger than countably infinite, and of set-functions between them. If the Count-objects B and C are in fact finite sets, then there is another finite set C^B which, with the obvious function ev, will serve as an exponential. But if B is a countably infinite set, and C has at least two members, then the set C^B is uncountable, so won't be available to be an exponential in Count – and evidently, nothing smaller will so.

(6) The standard example, however, of an interesting category which has some but not all exponentials is Top. If X is a space living in Top, then it is 'exponentiable', meaning that Y^X exists for all Y, if and only if it is so-called *core-compact* – and not all spaces are core-compact. It would, however, take us far too far afield to explain and justify this example.

18.4 Exponentials are unique

(a) Defn. 76 talks of 'an' exponential of C with B. But exponentials – as we might expect by now, given that the definition is by a universal mapping property – are in fact unique, at least up to unique isomorphism:

Theorem 73. *In a category $\mathscr{C}$ with exponentiation, if given objects B, C have exponentials $[E, ev]$ and $[E', ev']$, then there is a unique isomorphism between E and E' compatible with the evaluation arrows.*

Proof. Two commuting diagrams encapsulate the core of the argument, which parallels the proof of Theorem 34:

$$\begin{tikzcd}
E \times B \arrow[dd, dashed, "e\times 1_B"] \arrow[dr, "ev"] & \\
& C \\
E \times B \arrow[ur, "ev"'] &
\end{tikzcd}
\qquad
\begin{tikzcd}
E \times B \arrow[d, dashed, "\overline{ev}\times 1_B"'] \arrow[dr, "ev"] & \\
E' \times B \arrow[r, "ev'"] \arrow[d, dashed, "\overline{ev'}\times 1_B"'] & C \\
E \times B \arrow[ur, "ev"'] &
\end{tikzcd}$$

By definition, if $[E, ev]$ is an exponential of C by B then there is a unique mediating arrow $e\colon E \to E$ such that $ev \circ e \times 1_B = ev$. But as the diagram on the left reminds us, 1_E will serve as a mediating arrow. Hence $e = 1_E$.

The diagram on the right then reminds us that $[E, ev]$ and $[E', ev']$ factor uniquely through each other, and putting the two commuting triangles together, we get

$$ev \circ (\overline{ev'} \times 1_B) \circ (\overline{ev} \times 1_B) = ev.$$

Applying Theorem 44, we know that $(\overline{ev'} \times 1_B) \circ (\overline{ev} \times 1_B) = (\overline{ev'} \circ \overline{ev}) \times 1_B$, and hence

$$ev \circ (\overline{ev'} \circ \overline{ev}) \times 1_B = ev.$$

And now applying the uniqueness result from the first diagram

$$\overline{ev'} \circ \overline{ev} = 1_E.$$

Similarly, by interchanging E and E' in the second diagram, we get

$$\overline{ev} \circ \overline{ev'} = 1_{E'}.$$

Whence $\overline{ev}\colon E \to E'$ is an isomorphism. ☐

(b) When we were talking about e.g. products and equalizers, we gave two types of proof for their uniqueness (up to unique isomorphism). One was a direct proof from the definitions. For the other proof, we noted that products are terminal objects in a category of wedges, equalizers terminal objects in a category of forks, and then appealed to the uniqueness of terminal objects.

We have now given a proof of the first type, a direct proof, of the uniqueness of exponentials. Can we give a proof of the second type? Well, consider:

Definition 78. Given objects B and C in the category $\mathscr{C}$, then the category $\mathscr{C}_{E(B,C)}$ of parametized maps from B to C has the following data:

1. Objects $[A, g]$ comprising a $\mathscr{C}$-object A, and a $\mathscr{C}$-arrow $g\colon A \times B \to C$,

2. An arrow from $[A, g]$ to $[A', g']$ is any arrow $\mathscr{C}$-arrow $h\colon A \to A'$ which makes the following diagram commute:

$$\begin{tikzcd} A \times B \arrow[d, dashed, "h \times 1_B"'] \arrow[dr, "g"] & \\ A' \times B \arrow[r, "g'"'] & C \end{tikzcd}$$

The identity arrows and composition are as in $\mathscr{C}$. △

It is easily checked that this indeed defines a category, and then we evidently have

Theorem 74. *An exponential* $[C^B, ev]$ *is a terminal object in the category* $\mathscr{C}_{E(B,C)}$.

Since exponentials are terminal in a suitable category that yields the second type of proof of their uniqueness.

So in summary the situation is this. Exponentials in $\mathscr{C}$ are *not* a type of limit in $\mathscr{C}$ as characterized in Defn. 63 (for that definition talks of limit cones over diagrams in that same category, and an exponential isn't such a thing). But exponentials *can* be thought of as limits in *another*, derived, category of the kind $\mathscr{C}_{E(B,C)}$.

18.5 Further results about exponentials

(a) We now show, as promised, that any category with a terminal object has at least trivial exponentials as follows:

Theorem 75. *If the category $\mathscr{C}$ has a terminal object 1, then for any $\mathscr{C}$-object B, C, we have (1) $1^B \cong 1$ and (2) $C^1 \cong C$.*

Perhaps we should put that more carefully. The claim (1) is that if there is a terminal object 1 then there exists an exponential $[1^B, ev]$; and for any such exponential object 1^B, $1^B \cong 1$. Similarly for (2).

Proof for (1). Using, as before, $!_X$ for the unique arrow from X to the terminal object 1, consider the following diagram:

$$\begin{tikzcd} A \times B \arrow[d, dashed, "!_A \times 1_B"'] \arrow[dr, "!_{A\times B}"] & \\ 1 \times B \arrow[r, "!_{1\times B}"'] & 1 \end{tikzcd}$$

This has to commute, whatever A is (because there is only one arrow from $A \times B$ to a terminal object). Since there is only one possible arrow from A to 1, this means that $[1, !_{1\times B}]$ can serve as an exponential for 1 by B. Hence there exists an exponential 1^B, and by the uniqueness theorem, for any such exponential object 1^B, $1^B \cong 1$. □

Proof for (2). Here's the natural proof-strategy. Suppose we are given an arrow $g\colon A \times 1 \to C$. Show that there is always a unique $\overline{g}$ making this commute,

$$\begin{tikzcd}
 & A \times 1 \arrow[dl, "g"'] \arrow[d, "\overline{g}\times 1"] \\
C & C \times 1 \arrow[l, "\pi"]
\end{tikzcd}$$

where π is the projection from the product. Then $[C, \pi]$ serves as an exponential of C by 1 and hence, by the uniqueness theorem, any $C^1 \cong C$.

But there's an isomorphism a' which sends A to $A \times 1$ (the inverse of the projection from the product); so put $\overline{g} = g \circ a'$, and then the diagram will commute. And that's the unique possibility, so we are done. ☐

If it isn't obvious why our definition of $\overline{g}$ does the trick in the last proof, perhaps we should expand the argument. So: the wedge $C \xleftarrow{g} A \times 1 \xrightarrow{!} 1$ must factor through the product wedge $C \xleftarrow{\pi} C \times 1 \xrightarrow{!} 1$ via a unique mediating u, making the lower triangles in the following diagram commute:

$$\begin{tikzcd}
A \arrow[d, "\overline{g}"'] & A \times 1 \arrow[l, "a"'] \arrow[r, "!"] \arrow[dl, "g"'] \arrow[d, "u"] \arrow[dr, "!"] & 1 \arrow[d, "1"] \\
C & C \times 1 \arrow[l, "\pi"] \arrow[r, "!"'] & 1
\end{tikzcd}$$

Complete the diagram with the product wedge $A \xleftarrow{a} A \times 1 \xrightarrow{1} 1$ as shown, and – recalling that a and π must be isomorphisms by Theorem 38 – put $\overline{g} = g \circ a'$ where a' is the inverse of a. Then the whole diagram commutes.

This means that $u = \overline{g} \times 1$ by definition of the operation $\times$ on arrows in §12.3. Hence for each $g\colon A \times 1 \to C$ there is indeed a corresponding $\overline{g}$ making our first diagram commute. Moreover $\overline{g}$ is unique. If $k \times 1$ makes the second diagram commute then (i) it must equal u, and so $k \times 1 = \pi^{-1} \circ g$, but also by its definition, $\pi \circ k \times 1 = k \circ a$. Hence $g = k \circ a$, so $k = g \circ a' = \overline{g}$.

(b) We next need to establish a crucial general result:

Theorem 76. *If there exists an exponential of C by B in the category $\mathscr{C}$, then, for any object A in the category, there is a one-one correlation between arrows $A \times B \to C$ and arrows $A \to C^B$.*

There is also a one-one correlation between arrows $A \to C^B$ and arrows $B \to C^A$.

Proof. By definition of the exponential $[C^B, ev]$, an arrow $g\colon A \times B \to C$ is associated with a unique 'transpose' $\overline{g}\colon A \to C^B$ making the diagram (Exp) commute.

The map $g \mapsto \overline{g}$ is injective. For suppose $\overline{g} = \overline{h}$. Then $g = ev \circ (\overline{g} \times 1_B) = ev \circ (\overline{h} \times 1_B) = h$.

The map $g \mapsto \overline{g}$ is also surjective. Take any $k \colon A \to C^B$; then if we put $g = ev \circ (k \times 1_B)$, $\overline{g}$ is the unique map such that $ev \circ (\overline{g} \times 1_B) = g$, so $k = \overline{g}$.

Hence $g \mapsto \overline{g}$ is the required bijection between arrows $A \times B \to C$ and arrows $A \to C^B$, giving us the first part of the theorem.

For the second part, we just note that arrows $A \times B \to C$ are in one-one correspondence with arrows $B \times A \to C$, in virtue of the isomorphism between $A \times B$ and $B \times A$ (see Theorems 25 and 35). We then apply the first part of the theorem. □

This last theorem gives us a categorial analogue of the idea of currying that we met in §18.1, where a two-place function of type $A, B \to C$ gets traded in for a one-place function of type $A \to (B \to C)$.

18.6 Cartesian closed categories

Categories like Set, Prop and Bool which have all exponentials (which presupposes having binary products) and which also have a terminal object (and hence *all* finite products) are important enough to deserve a standard label:

Definition 79. A category $\mathscr{C}$ is a *Cartesian closed category* iff it has all finite products and all exponentials.[1] △

Such categories have nice properties meaning that exponentials there indeed behave as exponentials 'ought' to behave. For a start:

Theorem 77. *If $\mathscr{C}$ is a Cartesian closed category, then for all $A, B, C \in \mathscr{C}$*

(1) If $B \cong C$, then $A^B \cong A^C$,

(2) $(A^B)^C \cong A^{B \times C}$,

(3) $(A \times B)^C \cong A^C \times B^C$.

Proof of (1). Here's the basic idea for a brute force proof. We know that there exists an arrow $ev \colon A^B \times B \to A$. Since $B \cong C$, there is a derived arrow $g \colon A^B \times C \to A$. This has a unique associated transpose, $\overline{g} \colon A^B \to A^C$. Similarly, there is an arrow $\overline{h} \colon A^C \to A^B$. It remains to confirm that these arrows are (as you'd expect) inverses of each other, whence $A^B \cong A^C$.

To spell that out, consider the following diagram (where $j \colon B \to C$ is an isomorphism witnessing that $B \cong C$):

[1] Terminological aside: some call a category with all finite products a *Cartesian* category – but this term is also used in other ways so is probably best avoided. By contrast, the notion of a *Cartesian closed* category has a settled usage.

$$
\begin{tikzcd}
A^B \times B \arrow[d, "1\times j"'] \arrow[ddr, "ev"] & \\
A^B \times C \arrow[d, "\overline{g}\times 1"'] \arrow[dr, "g"] & \\
A^C \times C \arrow[d, "1\times j^{-1}"'] \arrow[r, "ev'"] & A \\
A^C \times B \arrow[d, "\overline{h}\times 1"'] \arrow[ur, "h"] & \\
A^B \times B \arrow[uur, "ev"'] &
\end{tikzcd}
$$

Here we've omitted subscripts on labels for identity arrows to reduce clutter. It is easy to see that since 1 and j are isomorphisms, so is $1 \times j$, and then if we put $g = ev \circ (1 \times j)^{-1}$ the top triangle commutes. The next triangle commutes by definition of the transpose $\overline{g}$; the third commutes if we now put $h = ev' \circ (1 \times j^{-1})^{-1}$; and the bottom triangle commutes by the definition of the transpose $\overline{h}$.

Products of arrows compose componentwise, as shown in Theorem 44. Hence the composite vertical arrow reduces to $(\overline{h} \circ \overline{g}) \times 1$. However, by the definition of the exponential $[A^B, ev]$ we know that there is a unique mediating arrow, k such that this commutes:

$$
\begin{tikzcd}
A^B \times B \arrow[dd, "k\times 1"'] \arrow[dr, "ev"] & \\
& A \\
A^B \times B \arrow[ur, "ev"'] &
\end{tikzcd}
$$

We now have two candidates for k which make the diagram commute, the identity arrow and $\overline{h} \circ \overline{g}$. Hence by uniqueness, $\overline{h} \circ \overline{g} = 1$.

A similar argument shows that $\overline{g} \circ \overline{h} = 1$. We are therefore done. $\square$

Proofs of (2) and (3). We can given a similarly direct proof of (2), along the following lines. Start with the evaluation arrow $ev\colon A^{B\times C} \times (B \times C) \to A$. We can shuffle terms in the product to derive an arrow $(A^{B\times C} \times C) \times B \to A$. Transpose this once to get an arrow $A^{B\times C} \times C \to A^B$ and transpose again to get an arrow $A^{B\times C} \to (A^B)^C$. Then similarly find an arrow from $(A^B)^C \to A^{B\times C}$, and show the two arrows are inverses of each other.

We can, however, leave it as an exercise for enthusiasts to work out details here. That's because we will eventually be able to bring to bear some heavierduty general apparatus which will yield fast-track proofs of (2) and (3), and indeed of (1) again. ☐

Theorem 78. *If $\mathscr{C}$ is a Cartesian closed category with terminal object* 1*, then for all* $A, B, C \in \mathscr{C}$

(1) $1^B \cong 1$,

(2) $C^1 \cong C$,

And if $\mathscr{C}$ also has an initial object 0*, then*

(3) $A \times 0 \cong 0 \cong 0 \times A$,

(4) $A^0 \cong 1$,

(5) if there is an arrow $A \to 0$*, then* $A \cong 0$*,*

(6) there exists an arrow $1 \to 0$ *iff $\mathscr{C}$ is category whose objects are all isomorphic to each other.*

The first two results are just particular cases of Theorem 75. But it is worth noting that if we are assuming we are working in a Cartesian closed category, and hence assuming that 1^B exists, then we can instead use this slick argument:

Proof of (1). By the Theorem 76, for each A, there is a one-one correlation between arrows $A \to 1^B$ and arrows $A \times B \to 1$. But since 1 is terminal, there is exactly one arrow $A \times B \to 1$; hence, for each A, there is exactly one arrow $A \to 1^B$. Therefore 1^B is terminal, and hence $1^B \cong 1$. ☐

Proof of (3). Since $A \times 0$ and $0 \times A$ exist by hypothesis, and are isomorphic by Theorem 35 (2), we need only prove $0 \times A \cong 0$.

By Theorem 76, for all C, there is a one-one correspondence between arrows $0 \to C^A$ and arrows $0 \times A \to C$. But 0 is initial, so there is exactly one arrow $0 \to C^A$. Hence for all C there is exactly one arrow $0 \times A \to C$, making $0 \times A$ initial too. Whence $0 \times A \cong 0$. ☐

Proof of (4). By Theorem 76 again, for all C, there is a bijection between arrows $C \to A^0$ and arrows $C \times 0 \to A$. And by (3) and Theorem 25 there is a bijection between arrows $C \times 0 \to A$ and arrows $0 \to A$. Since 0 is initial there is exactly one arrow $0 \to A$, and hence for all C there is exactly one arrow $C \to A^0$, so A^0 is terminal and $A^0 \cong 1$. ☐

Proof of (5). By assumption, there exists a wedge $A \xleftarrow{1_A} A \xrightarrow{f} 0$, and this will factor uniquely through the product $A \times 0$, as in

$$\begin{tikzcd}
& A \arrow[dl, "1_A"'] \arrow[d, dashed, "\langle 1_A, f\rangle"] \arrow[dr, "f"] & \\
A & A \times 0 \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & 0
\end{tikzcd}$$

So $\pi_1 \circ \langle 1_A, f\rangle = 1_A$. But $A \times 0 \cong 0$, so $A \times 0$ is an initial object, so there is a unique arrow $A \times 0 \to A \times 0$, namely $1_{A\times 0}$. Hence (travelling round the left triangle) $\langle 1_A, f\rangle \circ \pi_1 = 1_{A\times 0}$. Therefore $\langle 1_A, f\rangle\colon A \to A \times 0$ has a two-sided inverse. Whence $A \cong A \times 0 \cong 0$. □

Proof of (6). One direction is trivial. For the other, suppose there is an arrow $f\colon 1 \to 0$. Then, for any A there must be a composite arrow $A \longrightarrow 1 \xrightarrow{f} 0$, hence by (5), $A \cong 0$. So every object in the category is isomorphic. □

Here's a quick application of the result (6), that in a Cartesian closed category with an arrow $1 \to 0$, all objects are isomorphic:

Theorem 79. *The category* Grp *is not Cartesian closed.*

Proof. The one-element group is both initial and terminal in Grp, so here $1 \cong 0$, and hence there is an arrow $1 \to 0$ in Grp. But trivially, not all groups are isomorphic! Therefore the category Grp cannot be Cartesian closed. □

19 Group objects, natural number objects

We have seen how to define categorially a variety of familiar constructions using universal mapping properties; in particular, we have defined products and exponentials (to mention just the two cases which will feature again most often in this chapter).

We will next see how to use the apparatus that we now have available to characterize two familiar kinds of mathematical structure in categorial terms. We first give a definition of so-called *group objects* living in categories, and explore these just a little. Then we turn to say something equally introductory about that most basic of structures, *the natural numbers*. We won't take these discussions very far for the moment: our aim here in each case is simply to illustrate how we can begin to explore types of well-known mathematical structures from inside category theory.

19.1 Groups in Set

We informally think of a group as a collection of objects equipped with a binary operation of group 'multiplication' and with a designated element which is an identity for the operation. The group operation is associative, and every element has a two-sided inverse.

So how can we characterize such a structure as living in the category Set? We need an object G to provide a collection of group-elements, and we need three arrows (which are functions in this category):

(i) $m\colon G \times G \to G$ (here, once again, we have to trade the informal two-place operation of 'multiplication' for an arrow from a corresponding single source, i.e. from a product);

(ii) $e\colon 1 \to G$ (this element-as-arrow from a terminal object picks out a particular group-element in G – we'll also call this distinguished member of the group 'e', allowing context to disambiguate);

(iii) $i\colon G \to G$ (this is the arrow which sends a group-element to its inverse).

We then need to impose constraints on these arrows corresponding to the usual group axioms:

(1) We require the group operation m to be associative. Categorially, consider the following diagram:

(G1)

$$\begin{tikzcd}
(G \times G) \times G \arrow[rr, "\cong"] \arrow[d, "m \times 1_G"] & & G \times (G \times G) \arrow[d, "1_G \times m"] \\
G \times G \arrow[r, "m"] & G & G \times G \arrow[l, "m"']
\end{tikzcd}$$

Here the arrow at the top represents the naturally arising isomorphism between the two triple products that is established by the proof of Theorem 35 (3) in §13.1.

Remembering that we are working in Set, take an element $((j,k),l) \in (G \times G) \times G$. Going round on the left, that gets sent to $(m(j,k),l)$ and then to $m(m(j,k),l)$. Going round the other direction we get to $m(j,m(k,l))$. So requiring the diagram to commute captures the associativity of m.

(2) Informally, we next require e to act like a multiplicative identity.

To characterize this condition categorically, start by defining the map $e!\colon G \to G$ by composing $G \xrightarrow{!} 1 \xrightarrow{e} G$. In Set we can think of $e!$ as the function which sends anything in the G to its designated identity element e. We then have the following product diagram:

$$\begin{tikzcd}
 & G \arrow[dl, "1_G"'] \arrow[d, dashed, "\langle 1_G, e! \rangle"] \arrow[dr, "e!"] & \\
G & G \times G \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & G
\end{tikzcd}$$

So we can think of the mediating arrow $\langle 1_G, e! \rangle$ as sending an element $g \in G$ to the pair (g, e).

The element e then behaves like a multiplicative identity on the right if m sends this pair (g, e) in turn back to g – i.e. if the top triangle in the following diagram commutes:

(G2)

$$\begin{tikzcd}
G \arrow[r, "\langle 1_G, e! \rangle"] \arrow[d, "\langle e!, 1_G \rangle"'] \arrow[dr, "1_G"] & G \times G \arrow[d, "m"] \\
G \times G \arrow[r, "m"'] & G
\end{tikzcd}$$

Similarly the lower triangle commutes just if e behaves as an identity on the left. So, for e to behave as a two-sided identity element, it is enough that the whole diagram commutes.

(3) Finally, we informally require that every element $g \in G$ has an inverse g^{-1} or $i(g)$ such that $m(g, i(g)) = e = m(i(g), g)$. Categorially, we can express this by requiring that the following commutes:

(G3)

$$\begin{tikzcd}
G \times G \arrow[d, "1_G \times i"] & G \arrow[l, "\delta_G"'] \arrow[r, "\delta_G"] \arrow[d, "e!"] & G \times G \arrow[d, "i \times 1_G"] \\
G \times G \arrow[r, "m"] & G & G \times G \arrow[l, "m"']
\end{tikzcd}$$

For take an element $g \in G$. Going left, the diagonal arrow δ_G (from Defn. 49) maps it to the pair (g, g), which is mapped in turn by $1_G \times i$ to $(g, i(g))$ and then by m to $m(g, i(g))$. The central vertical arrow meanwhile simply sends g to e. Therefore, the requirement that the left square commutes tells us, as we want, that $m(g, i(g)) = e$. Similarly the requirement that the right square commutes tells us that $m(i(g), g) = e$

In summary then, the informal group axioms correspond to the commutativity of our last three diagrams.

But note immediately that this categorial treatment of groups only requires that we are working in a category with binary products and a terminal object. So it is natural to generalize, as follows:

Definition 80. Suppose $\mathscr{C}$ is a category which has binary products and a terminal object. Let G be a $\mathscr{C}$-object, and $m\colon G \times G \to G$, $e\colon 1 \to G$ and $i\colon G \to G$ be $\mathscr{C}$-arrows. Then $[G, m, e, i]$ is a *group-object* in $\mathscr{C}$ iff the three diagrams (G1), (G2), (G3) commute, where $e!$ in the latter two diagrams is the composite map $G \xrightarrow{!} 1 \xrightarrow{e} G$. △

Here, 'group object' is the standard terminology (though some alternatively say 'internal group').

Then, if we don't fuss about the type-difference between an arrow $e\colon 1 \to G$ (in a group object) and a designated element e (in a group), we have established the summary result

Theorem 80. *In the category* Set*, a group object is a group.*

And conversely, every group – or to be really pernickety, every group which hasn't got too many elements to form a set – can be regarded as a group object in Set.

19.2 Groups in other categories

(a) Here are just a few more examples of group objects:

Theorem 81. *(1) In the category* Top*, which comprises topological spaces with continuous maps between them, a group object is a topological group in the standard sense.*

(2) In the category Man*, which comprises smooth manifolds with smooth maps between them, a group object is a Lie group.*

(3) In the category Grp*, a group object is an abelian group.*

The proofs of the first two claims are predictably straightforward if you know the usual definitions of topological groups and Lie groups, and we will not pause over them here. The third claim, by contrast, is probably unexpected. However, the proof is relatively straightforward, quite cute, and a rather useful reality-check:

Proof of (3). Suppose $[G, m, e, i]$ is a group-object in Grp. Then the object G is already a group, i.e. a set of objects $\dot{G}$ equipped with a group operation and an identity element. We'll use ordinary multiplication notation for that operation, as in '$x \cdot y$', and we'll dub the identity '$\dot{1}$' (so the innards of the group G are notated with dots!). The arrow $e\colon 1 \to G$ in the group object must also pick out a distinguished element of $\dot{G}$, call it '$\underline{1}$', an identity for m.

Now, each arrow in the group-object $[G, m, e, i]$ lives in Grp, so is a group homomorphism. That means in particular m is a homomorphism from $G \times G$ (the product group, with group operation $\times$) to G. So take the elements $x, y, z, w \in \dot{G}$. Then,

$$m(x \cdot z, y \cdot w) = m((x, y) \times (z, w)) = m(x, y) \cdot m(z, w)$$

The first equation holds because of how the operation $\times$ is defined for the product group; the second equation holds because m is a homomorphism.

For vividness, let's rewrite $m(x, y)$ as $x \star y$ (so $\underline{1}$ is the unit for $\star$). Then we have established the interchange law

$$(x \cdot z) \star (y \cdot w) = (x \star y) \cdot (z \star w).$$

We will now use this law twice over (the proof from this point on uses what is standardly called the Eckmann–Hilton argument, a general principle applying when we have such an interchange law between two binary operations with units). First, we have

$$\dot{1} = \dot{1} \cdot \dot{1} = (\underline{1} \star \dot{1}) \cdot (\dot{1} \star \underline{1}) = (\underline{1} \cdot \dot{1}) \star (\dot{1} \cdot \underline{1}) = \underline{1} \star \underline{1} = \underline{1}$$

We can therefore just write 1 for the shared unit, and show secondly that

$$\begin{aligned} x \cdot y &= (x \star 1) \cdot (1 \star y) = (x \cdot 1) \star (1 \cdot y) = x \star y \\ &= (1 \cdot x) \star (y \cdot 1) = (1 \star y) \cdot (x \star 1) = y \cdot x. \end{aligned}$$

We have shown, then, that if $[G, m, e, i]$ is a group object in Grp, G's own group operation commutes, and m is the same operation so that must also commute. Therefore the group object is indeed an abelian group. □

A similar argument, we might note, proves the reverse result: any abelian group can be regarded as a group object in Grp.

19.3 A very little more on groups

(a) We can continue the story, defining further group-theoretic notions in categorial terms.

(1) For a start, we can categorially define the idea of a homomorphism between group objects in a category.

Suppose $[G, m, e, i]$ and $[G', m', e', i']$ are group objects in Set. Then a homomorphism between them is a $\mathscr{C}$-arrow $h\colon G \to G'$ which 'preserves structure' by appropriately commuting with the group objects' arrows. More precisely, a moment's reflection shows that h is a homomorphism just if the following three diagrams commute:

$$\begin{tikzcd} G \times G \arrow[r, "h \times h"] \arrow[d, "m"'] & G' \times G' \arrow[d, "m'"] \\ G \arrow[r, "h"] & G' \end{tikzcd} \qquad \begin{tikzcd} & 1 \arrow[dl, "e"'] \arrow[dr, "e'"] & \\ G \arrow[rr, "h"] & & G' \end{tikzcd} \qquad \begin{tikzcd} G \arrow[r, "h"] \arrow[d, "i"'] & G' \arrow[d, "i'"] \\ G \arrow[r, "h"] & G' \end{tikzcd}$$

(2) Recall another group-theoretic idea, the key notion of the action of a group G on a set X. Informally, a (left) action is a two-place function $a\colon G, X \to X$ such that $a(e, x) = x$ where e is the group identity and $x \in X$, and $a(g \cdot h, x) = a(g, a(h, x))$ for any group elements g, h. This isn't the place to review the importance of the idea of a group action! Rather, we just note that we can categorially define e.g. the action of a group object $[G, m, e, i]$ on a set X in Set as an arrow $a\colon G \times X \to X$ which makes the following two diagrams commute:

$$\begin{tikzcd} 1 \times X \arrow[r, "e \times 1_X"] \arrow[dr, "\cong"'] & G \times X \arrow[d, "a"] \\ & X \end{tikzcd} \qquad \begin{tikzcd} (G \times G) \times X) \arrow[r, "m \times 1"] \arrow[d, "\cong"] & G \times X \arrow[dr, "a"] & \\ & & X \\ G \times (G \times X) \arrow[r, "1 \times a"] & G \times X \arrow[ur, "a"'] & \end{tikzcd}$$

And so it goes: along these lines, core group-theoretic ideas can be recast into a categorial framework.

(b) The explorations we have begun here could be continued in various directions. First, for example, we could similarly define other kinds of algebraic objects and their morphisms within categories. Second, noting that we can now define group-objects and group-homomorphisms inside a given category like Set, we could go on to categorially define categories of groups living in other categories. And then, generalizing that second idea, we can define the idea of internal categories. But in either of these directions, things begin to get pretty abstract (and not in a way that is particularly helpful for us at this stage in the proceedings). So in the rest of this chapter, we consider something much more basic and much more 'concrete', namely ...

19.4 Natural numbers

Our aim is to categorially characterize what are standardly called *natural number objects*. Like group objects in a category, natural number objects in a category aren't naked objects but rather objects-with-arrows. Which arrows? Intuitively, we need an arrow-as-element to pick out an initial object, a 'zero', and we need an arrow-as-operation which takes an element to its 'successor'. That will at least give us sequences – so we say:

Definition 81. If $\mathscr{C}$ is a category with a terminal object, then $[X, i, f]$ is a *sequence object* in $\mathscr{C}$ if X is a $\mathscr{C}$-object, and i, f are $\mathscr{C}$-arrows $i\colon 1 \to X$ and $f\colon X \to X$. △

If we are working in the category Set, for example, the arrow i picks out the initial element of a sequence, call this element i too; and f then generates a sequence i, $f(i)$, $f^2(i)$, $f^3(i)$, $\ldots$.

However, such a sequence could eventually repeat or cycle round; our task is therefore to categorially characterize the limiting case of sequence objects corresponding to non-repeating sequences $f^n(i)$ which look like the natural numbers (i.e. which are ω-sequences). To do this, we start with another definition:

Definition 82. If $\mathscr{C}$ is a category with a terminal object, then the derived category $\mathscr{C}_{Seq}$ has as objects all of $\mathscr{C}$'s sequence objects $[X, i, f]$, and an arrow $u\colon [X, i, f] \to [Y, j, g]$ is a $\mathscr{C}$-arrow u which makes the following diagram commute in $\mathscr{C}$:

$$\begin{tikzcd}
1 \arrow[r, "i"] \arrow[dr, "j"'] & X \arrow[r, "f"] \arrow[d, "u"] & X \arrow[d, "u"] \\
& Y \arrow[r, "g"] & Y
\end{tikzcd}$$

△

It is routine to check that this definition is in good order and $\mathscr{C}_{Seq}$ is indeed a category (with $\mathscr{C}_{Seq}$'s identity arrow on $[X, i, f]$ being $\mathscr{C}$'s identity arrow on X, and composition in $\mathscr{C}_{Seq}$ being composition in $\mathscr{C}$.)

Three observations about this:

(1) Suppose we have such a commuting diagram in Set. Then u sends a sequence $i, f(i), f^2(i), f^3(i), \ldots$ living in X to the sequence j, $g(j)$, $g^2(j)$, $g^3(j)$, $\ldots$ living in Y. And given u is functional, if $g^m(j) \neq g^n(j)$ then $f^m(i) \neq f^n(i)$. In other words, the sequence object $[X, i, f]$ can't be *more* constrained by equations of the form $f^m(i) = f^n(i)$ in the sequence than $[Y, j, g]$ is constrained by similar equations between *its* elements.

(2) So if Set_{Seq} has an initial object, call it $[N, 0, s]$, then this will have to be as unconstrained a sequence as possible, governed by no additional equations of the form $s^m(0) = s^n(0)$ (where $m \neq n$), and so never repeating. In other words, this initial object will have to correspond to an ω-sequence.

(3) Conversely, consider the standard implementation of the natural numbers $\mathbb{N} = \{\varnothing, \{\varnothing\}, \{\varnothing, \{\varnothing\}\}, \ldots\}$ in Set, together with the arrow $0\colon 1 \to \mathbb{N}$ which sends the object in the singleton to $\varnothing$, and the arrow $s\colon \mathbb{N} \to \mathbb{N}$ which sends a set $n \in \mathbb{N}$ to the set $n \cup \{n\}$. Then $[\mathbb{N}, 0, s]$ evidently form an initial object in $\mathscr{C}_{Seq}$. Given any other sequence object $[Y, j, g]$ in Set, setting u to be the arrow $n \mapsto g^n(j)$ makes the diagram commute, and evidently u is unique.

Which all goes to motivate the following general definition:

Definition 83. If $\mathscr{C}$ is a category with a terminal object, then a *natural number object* in $\mathscr{C}$ is an initial object of the derived category $\mathscr{C}_{Seq}$.

That is to say (with objects and arrows in $\mathscr{C}$) a natural number object $[N, 0, s]$ comprises an object N and two arrows $0\colon 1 \to N$ and $s\colon N \to N$ such that for any object Y and arrows $j\colon 1 \to Y$ and $g\colon Y \to Y$ there is a *unique* arrow u which makes the following diagram commute:

$$\begin{tikzcd} 1 \arrow[r, "0"] \arrow[dr, "j"'] & N \arrow[r, "s"] \arrow[d, dashed, "u"] & N \arrow[d, dashed, "u"] \\ & Y \arrow[r, "g"] & Y \end{tikzcd}$$

△

Being initial objects of the derived category $\mathscr{C}_{Seq}$, it follows that if $[N, 0, s]$ and $[N', 0', s']$ are natural number objects in $\mathscr{C}$ then $N \cong N'$ (and indeed there is a unique isomorphism commuting with the arrows in the obvious way).

19.5 The Peano postulates revisited

(a) Let's pause to recall the informal Peano postulates as presented to budding mathematicians. These postulates tell us that the natural numbers N include a distinguished zero object 0 and come equipped with a successor function s, and are such that:

(1) 0 is a number;

(2) If n is a number, so is its successor sn;

(3) 0 is not a successor of any number;

(4) Two numbers n, m with the same successor are equal;

(5) For any property P of natural numbers, if 0 has P, and if sn has P whenever n does, then P holds for all natural numbers.

Here, we should understand 'property' in the generous sense according to which any arbitrary subset A of numbers defines a property (the property of being a member of A). So we can take (5) as equivalent to

(5′) For any set A of natural numbers, if $0 \in A$, and if $n \in A \Rightarrow sn \in A$, then $A = N$.

A familiar informal set-theoretic argument now shows that the Peano postulates characterize the structure $N, 0, s$ up to isomorphism. And another familiar argument which we also won't repeat here shows that we can deduce the so-called Recursion Theorem:

> For any objects Y, selected object j among Y, and function g with Y as domain and codomain, there is a unique function $u\colon N \to Y$ such that $u(0) = j$ and $u(sn) = g(u(n))$.

Or in other words, definition by (simple) primitive recursion well-defines a function.

(b) That last observation tells us, of course, that if we take the arrow $0\colon 1 \to N$ to send the member of the singleton to the Peano zero, then the resulting $[N, 0, s]$ is a natural number object in Set.

What about the converse? Suppose $[N, 0, s]$ is a natural number object in Set. Then identifying the Peano zero with the image of the member of 1 under the arrow $0\colon 1 \to N$, we of course get (1) and (2) for free. As we noted before, $[N, 0, s]$ can't both be an initial object in the category of sequence objects and be constrained by equations of the form $s^m(0) = s^n(0)$ where $m \neq n$; and that gives us (3) and (4). Which just leaves the induction principle.

Suppose (i) there is an injection $i\colon A \to N$, (ii) $0 \in A$, (iii) $n \in A \Rightarrow sn \in A$. We need to show that $A = N$.

By the third supposition, s sends arguments in A to values in A and hence there is a function $s'\colon A \to A$ which is the restriction of $s\colon N \to N$ to A. So (iii) means the square in

$$\begin{tikzcd}
1 \arrow[r, "0'"] \arrow[dr, "0"'] & A \arrow[r, "s'"] \arrow[d, "i"] & A \arrow[d, "i"] \\
& N \arrow[r, "s"] & N
\end{tikzcd}$$

commutes. While (ii) tells us that there is an arrow $0'\colon 1 \to A$ which makes the triangle commute. Hence the following diagram commutes for some unique u (the top half by the universal property of the natural number object):

$$\begin{tikzcd}
& N \arrow[r, "s"] \arrow[d, dashed, "u"] & N \arrow[d, dashed, "u"] \\
1 \arrow[ur, "0"] \arrow[r, "0'"] \arrow[dr, "0"'] & A \arrow[r, "s'"] \arrow[d, "i"] & A \arrow[d, "i"] \\
& N \arrow[r, "s"] & N
\end{tikzcd}$$

Which means that the natural number object $[N, 0, s]$ factors through itself via the mediating arrow $i \circ u$. But trivially, it factors through itself by 1_N and hence, since the mediating arrow is unique, $i \circ u = 1_N$. Therefore i is a left inverse and so by Theorem 18 it is epic. Hence (since we are in Set) i is surjective. Which means that $A = N$, as we require.

19.6 More on recursion

(a) We have defined natural number objects in an intuitively appealing categorial way, and shown that at least in Set we thereby characterize a structure that satisfies the Peano postulates. So far, so good. But there's work still to be done.

For consider next the following pattern for the recursive definition of a *two*-place function $f\colon N, N \to N$ in terms of a couple of given one-place functions $g, h\colon N \to N$:

(1) $f(m, 0) = g(m)$

(2) $f(m, sn) = h(f(m, n))$.

Here's a very familiar example: if $g(m) = m$ and h is the successor function s again, then our equations give us a recursive definition of addition.

We can call this type of definition a *definition by parameterized recursion*, since there is a parameter m which we hold fixed as we run the recursion on n. And intuitively our equations do indeed well-define a determinate binary function f, given any determinate monadic functions g and h (and we can prove that from the Peano Postulates given enough ambient informal set theory).

Now, to characterize this kind of definition by parameterized recursion in a categorial framework, we will evidently have to replace the two-place function with an arrow f from a product. Suppose then that we are again working in some category $\mathscr{C}$ which has a natural number object $[N, 0, s]$. And now suppose too that (P): given any arrows $g\colon N \to N$ and $h\colon N \to N$, there is a unique arrow $f\colon N \times N \to N$ in $\mathscr{C}$ which makes this diagram commute

$$\begin{tikzcd}
N \arrow[r, "\langle 1_N, 0! \rangle"] \arrow[dr, "g"'] & N \times N \arrow[r, "1_N \times s"] \arrow[d, dashed, "f"] & N \times N \arrow[d, dashed, "f"] \\
& N \arrow[r, "h"] & N
\end{tikzcd}$$

where $0!$ is the composite map $N \xrightarrow{!} 1 \xrightarrow{0} N$. Saying the triangle commutes is the categorial equivalent of saying that (1) holds (since $\langle 1_N, 0! \rangle$ sends m to the pair $(m, 0)$ – cf. Theorem 42). And saying the square commutes is the equivalent of saying that (2) holds. Hence if a category $\mathscr{C}$ satisfies condition (P), then in effect parameterised recursion well-defines functions in $\mathscr{C}$. But it doesn't follow from a category's having a natural number object that it will automatically satisfy (P) as well. In other words, while having a natural number object in a category ensures that definitions by *simple* recursion work there, this does *not* automatically ensure that definitions by *parameterized* recursion are also allowed in $\mathscr{C}$.

(b) However, we do have the following general result:

Theorem 82. *If $\mathscr{C}$ is a Cartesian closed category with a natural number object $[N, 0, s]$, then given any objects A, C, and arrows $g\colon A \to C$ and $h\colon C \to C$, then there is a unique f which makes the following diagram commute:*

$$\begin{tikzcd}
A \arrow[r, "\langle 1_A, 0! \rangle"] \arrow[dr, "g"'] & A \times N \arrow[r, "1_A \times s"] \arrow[d, dashed, "f"] & A \times N \arrow[d, dashed, "f"] \\
& C \arrow[r, "h"] & C
\end{tikzcd}$$

Our previous diagram of course illustrates the special case where $A = C = N$. So in a Cartesian closed category with a natural number object we certainly can warrant the elementary kind of parameterized recursive definition we met at the beginning of the section. And in particular, since Set is Cartesian closed, such definitions will be permitted in Set-theoretic arithmetic (as we'd of course expect, having already noted that such an arithmetic will satisfy the full Peano postulates).

To prove our theorem we exploit the associations between arrows $A \times N \to C$ and arrows $N \times A \to C$ and between those and arrows $N \to C^A$ which are available in categories with exponentials. The idea is simple; the details are tiresome:

Proof. We suppose, then, that we working in a category $\mathscr{C}$ which has all exponentials (and hence binary products), which has a natural number object $[N, 0, s]$, and which also has two arrows $g\colon A \to C$ and $h\colon C \to C$.

By hypothesis, the exponential $[C^A, ev]$ exists. Let i be an isomorphism from $1 \times A$ to A. We now use g and h to define

$$g' = \overline{g \circ i}\colon 1 \to C^A, \quad h' = \overline{h \circ ev}\colon C^A \to C^A,$$

where, remember, overlining notates exponential transposes. These somewhat mysterious definitions can be explained by two commutative diagrams:

$$\begin{tikzcd}
1 \times A \arrow[dd, "\overline{g \circ i} \times 1_A"'] \arrow[dr, "i"] & \\
& A \arrow[d, "g"] \\
C^A \times A \arrow[r, "ev"] & C
\end{tikzcd}
\qquad
\begin{tikzcd}
C^A \times A \arrow[dd, "\overline{h \circ ev} \times 1_A"'] \arrow[dr, "ev"] & \\
& C \arrow[d, "h"] \\
C^A \times A \arrow[r, "ev"] & C
\end{tikzcd}$$

By the universal property of $\mathscr{C}$'s natural number object, we know that there is a unique map u which makes the following commute:

$$\begin{tikzcd}
1 \arrow[r, "0"] \arrow[dr, "g'"'] & N \arrow[r, "s"] \arrow[d, dashed, "u"] & N \arrow[d, dashed, "u"] \\
& C^A \arrow[r, "h'"] & C^A
\end{tikzcd}$$

So now the name of the game is to define an arrow $f\colon A \times N \to C$ in terms of $u\colon N \to C^A$ in such a way that the fact that our last diagram commutes will entail that the diagram in the statement of the theorem commutes.

The obvious way to start is to define an arrow $f^o\colon N \times A \to C$ by putting $f^o = ev \circ (u \times 1_A)$ so u is the exponential transpose of f^o. Which doesn't quite give us what we want. But there is an isomorphism $o\colon A \times N \to N \times A$, and we can put $f = f^o \circ o$.

We now need to show that (i) $f \circ \langle 1_A, 0!\rangle = g$, and (ii) $f \circ (1_A \times s) = h \circ f$. For (i), note first that the following diagram commutes (we've not labelled all the projection arrows, and compare the proof of Theorem 35 (2)):

$$\begin{tikzcd}
 & A \arrow[dl, "!"'] \arrow[d, "i^{-1}"] \arrow[dr, "1_A"] & \\
1 \arrow[d, "0"] & 1 \times A \arrow[l] \arrow[r] \arrow[d, "0\times 1_A"] & A \arrow[d, "1_A"] \\
N \arrow[d, "1_N"] & N \times A \arrow[l] \arrow[r] \arrow[d, "o^{-1}"] & A \arrow[d, "1_A"] \\
N & A \times N \arrow[l, "\pi_2"'] \arrow[r, "\pi_1"'] & A
\end{tikzcd}$$

So $A \xleftarrow{1_A} A \xrightarrow{0!} N$ factors through the product $A \xleftarrow{\pi_1} A \times N \xrightarrow{\pi_2} N$ via the composite of the vertical arrows. Hence $\langle 1_A, 0!\rangle = o^{-1} \circ (0 \times 1_A) \circ i^{-1}$. Therefore using Theorem 44 we can argue:

$$\begin{aligned}
f \circ \langle 1_A, 0!\rangle &= ev \circ (u \times 1_A) \circ o \circ o^{-1} \circ (0 \times 1_A) \circ i^{-1} \\
&= ev \circ (u \times 1_A) \circ (0 \times 1_A) \circ i^{-1} \\
&= ev \circ ((u \circ 0) \times (1_A \circ 1_A)) \circ i^{-1} \\
&= ev \circ (g' \times 1_A) \circ i^{-1} \\
&= ev \circ (\overline{g \circ i} \times 1_A) \circ i^{-1} \\
&= g \circ i \circ i^{-1} \\
&= g.
\end{aligned}$$

For (ii), we can appeal to Theorem 41 to show that $o \circ (1_A \times s) = (s \times 1_A) \times o$.

Then we can argue:

$$
\begin{aligned}
f \circ (1_A \times s) &= ev \circ (u \times 1_A) \circ o \circ (1_A \times s) \\
&= ev \circ (u \times 1_A) \circ (s \times 1_A) \circ o \\
&= ev \circ ((u \circ s) \times (1_A \times 1_A)) \circ o \\
&= ev \circ ((h' \circ u) \times (1_A \times 1_A)) \circ o \\
&= ev \circ (h' \times 1_A) \circ (u \times 1_A) \circ o \\
&= ev \circ (\overline{h \circ ev} \times 1_A) \circ (u \times 1_A) \circ o \\
&= h \circ ev \circ (u \times 1_A) \circ o \\
&= h \circ f.
\end{aligned}
$$

Finally, we need to confirm f's uniqueness. But perhaps, with all the ingredients to hand, we can leave that as an exercise! $\square$

Our theorem can now be extended in the same vein to cover not just definitions by recursion that carry along a single parameter but also the most general kind of definitions by primitive recursions. Therefore in a Cartesian closed category with a natural number object we can start doing some serious arithmetic. And this is just the beginning: Cartesian closed categories with extra features turn out to be suitable worlds in which to do great swathes of mathematics. About which a lot more in due course.

20 Functors introduced

We have so far been looking *inside* categories and characterizing various kinds of construction to be found there (products, equalizers, exponentials, and the like, and then even e.g. groups and natural number objects). We have seen the same constructions appearing and reappearing in various guises in different categories. An obvious next task is to develop some apparatus for relating categories by mapping such recurrent constructions from one category to another. After all, the spirit of category theory is to understand objects of a certain kind via the morphisms between them: so, in that spirit, we should surely now seek to understand more about categories by thinking about the maps or morphisms between *them*. The standard term for a structure-preserving map *between* categories is 'functor'. This chapter introduces such maps.

20.1 Functors defined

A category $\mathscr{C}$ has two kinds of data, its objects and its arrows. So a functor F from category $\mathscr{C}$ to category $\mathscr{D}$ will need to have two components, one that operates on objects, one that operates on arrows. Hence:

Definition 84. Given categories $\mathscr{C}$ and $\mathscr{D}$, a *functor* $F\colon \mathscr{C} \to \mathscr{D}$ comprises the following data:

(1) An operation or mapping F_{ob} whose value at the $\mathscr{C}$-object A is some $\mathscr{D}$-object we can represent as $F_{ob}(A)$ or, dropping the explicit subscript, as $F(A)$ or indeed simply as FA.

(2) An operation or mapping F_{arw} whose value at the $\mathscr{C}$-arrow $f\colon A \to B$ is a $\mathscr{D}$-arrow from $F(A)$ to $F(B)$ which, again dropping the explicit subscript, we can represent as $F(f)\colon F(A) \to F(B)$, or simply as $Ff\colon FA \to FB$.

But there's more. If a functor is to preserve at least the most basic categorial structure, its component mappings must obey two obvious conditions. First they must map identity arrows to identity arrows. Second they should respect composition. That is to say, since the commutative diagram

$$\begin{tikzcd} & B \arrow[dr, "g"] & \\ A \arrow[ur, "f"] \arrow[rr, "g\circ f"'] & & C \end{tikzcd}$$

gets sent by F to

$$\begin{tikzcd} & FB \arrow[dr, "F(g)"] & \\ FA \arrow[ur, "F(f)"] \arrow[rr, "F(g \circ f)"'] & & FC \end{tikzcd}$$

the second diagram should also commute. Hence we want:

Definition 84 (continued). The data in F must satisfy the following conditions:

Preserving identities: for any $\mathscr{C}$-object A, $F(1_A) = 1_{FA}$;

Respecting composition: for any $\mathscr{C}$-arrows f, g such that their composition $g \circ f$ exists, $F(g \circ f) = Fg \circ Ff$. △

These conditions on F are often called, simply, *functoriality*.

20.2 Some elementary examples of functors

Our first example illustrates a broad class of cases:

(F1) There is a functor $F : \mathsf{Mon} \to \mathsf{Set}$ with the following data:

i. F_{ob} sends the monoid $(M, \cdot, 1_M)$ to its carrier set M.

ii. F_{arw} sends $f\colon (M, \cdot, 1_M) \to (N, \times, 1_N)$, i.e. a monoid homomorphism acting on elements on M, to the same map thought of as a set-function $f\colon M \to N$.

So defined, F trivially obeys the axioms for being a functor. All it does is 'forget' about the structure carried by the collection of objects in a monoid. It's a *forgetful functor*, for short.

There are equally forgetful functors from other categories of structured sets to the bare underlying sets. For example, there is the functor $F\colon \mathsf{Grp} \to \mathsf{Set}$ that sends groups to their underlying carrier sets and sends group homorphisms to themselves as set function, forgetting about the group structure. Often, a forgetful functor such as this is called an *underlying* functor (and hence the common practice, which we shall occasionally adopt, of using the letter 'U' to denote such a functor).

Of course, these forgetful functors are not intrinsically very exciting! It will turn out, however, that they are the boring members of so-called adjoint pairs of functors where they are married to much more interesting companions. But that observation is for later chapters.

To continue just for a moment with the forgetful theme:

(F2) There is a functor $F : \mathsf{Set} \to \mathsf{Rel}$ which sends sets and triples (domain, graph, codomain) thought of as objects and arrows belonging to Set to the same items thought of as objects and arrows in Rel, forgetting that the arrows are functional.

(F3) There are also somewhat less forgetful functors, such as the functor from Rng to Grp that sends a ring to the additive group it contains, forgetting the rest of the ring structure. Or take the functor from Ab, the category of abelian groups, to Grp, that remembers about group structure but forgets about commutativity.

And now for some different kinds of functors:

(F4) The powerset functor $P\colon \mathsf{Set} \to \mathsf{Set}$ maps a set X to its powerset $\mathscr{P}(X)$ and maps a set-function $f\colon X \to Y$ to the function which sends $U \in \mathscr{P}(X)$ to its f-image $f[U] = \{f(x) \mid x \in U\} \in \mathscr{P}(Y)$.

(F5) Take monoids $(M, \cdot, 1_M)$ and $(N, \times, 1_N)$ and consider the corresponding categories $\mathscr{M}$ and $\mathscr{N}$ in the sense of §4.7.

So $\mathscr{M}$ has a single object $\star_{\mathscr{M}}$, and its arrows are elements of M, where the composition of the arrows m_1 and m_2 is just $m_1 \cdot m_2$, and the identity arrow is the identity element of the monoid, 1_M.

Likewise $\mathscr{N}$ has a single object $\star_{\mathscr{N}}$, and arrows are elements of N, where the composition of the arrows n_1 and n_2 is just $n_1 \times n_2$, and the identity arrow is the identity element of the monoid, 1_N.

So now we see that a functor $F : \mathscr{M} \to \mathscr{N}$ will need to do the following:

i. F must send $\star_{\mathscr{M}}$ to $\star_{\mathscr{N}}$.

ii. F must send the identity arrow 1_M to the identity arrow 1_N.

iii. F must send $m_1 \circ m_2$ (i.e. $m_1 \cdot m_2$) to $Fm_1 \circ Fm_2$ (i.e. $Fm_1 \times Fm_2$).

Apart from the trivial first condition, that just requires F to be a monoid homomorphism. So any homomorphism between two monoids induces a corresponding functor between the corresponding monoids-as-categories.

(F6) Take the posets $(S, \preccurlyeq)$ and $(T, \sqsubseteq)$ considered as categories $\mathscr{S}$ and $\mathscr{T}$. It is easy to check that a monotone function $f\colon S \to T$ (i.e. function such that $s \preccurlyeq s'$ implies $f(s) \sqsubseteq f(s')$) induces a functor $F\colon \mathscr{S} \to \mathscr{T}$ which sends an $\mathscr{S}$-object s to the $\mathscr{T}$-object $f(s)$, and sends an $\mathscr{S}$-arrow, i.e. a pair (s, s') where $s \preccurlyeq s'$, to the $\mathscr{T}$-arrow $(f(s), f(s'))$.

(F7) Next, take the group $G = (G, \cdot, e)$ and now consider it as a category $\mathcal{G}$ – see §7.27.3. Suppose $F\colon \mathcal{G} \to \mathsf{Set}$ is a functor. Then F must send $\mathcal{G}$'s unique object $\star$ to some set X. And F must send a $\mathcal{G}$-arrow $m\colon \star \to \star$ (that's just a member m of G) to a function $F(m)\colon X \to X$. Functoriality requires that $F(e) = 1_X$ and $F(m \cdot m') = F(m) \circ F(m')$. But those are just the conditions for F to constitute a group action of G on X. Conversely, a group action of G on X amounts to a functor from $\mathcal{G}$ to Set.

(F8) There is a list functor $List\colon \mathsf{Set} \to \mathsf{Set}$, where $List_{ob}$ sends a set X to $List(X)$, the set of all finite lists or sequences of elements of X, including the empty one. And $List_{arw}$ sends a function $f\colon X \to Y$ to the function

$List(f)\colon List(X) \to List(Y)$ which sends the list $x_0{}^\frown x_1{}^\frown x_2{}^\frown \ldots{}^\frown x_n$ to $fx_0{}^\frown fx_1{}^\frown fx_2{}^\frown \ldots{}^\frown fx_n$ (where $^\frown$ is concatenation).

Returning to the forgetful theme, we have seen cases of functors that simply forget (some of the) structure put on structured sets. We can also have a functor which obliterates some distinctions between objects or between arrows.

(F9) Suppose $\mathscr{S}$ is a thin, pre-order, category (so has just one arrow between any source and target), and let $\mathscr{C}$ be a fattened category which has the same objects as $\mathscr{S}$ but in addition to the arrows of $\mathscr{S}$ has perhaps extra arrows. Then there will be a functor F from $\mathscr{C}$ back to the slimmed-down $\mathscr{S}$ which takes objects to themselves, and maps every arrow from A to B in $\mathscr{C}$ to the unique such arrow in $\mathscr{S}$. We could call this F a 'thinning' functor.

(F10) A more extreme case: suppose $\mathscr{C}$ and $\mathscr{D}$ are any (non-empty!) categories, and D is any object in $\mathscr{D}$. Then there is a corresponding collapse-to-D constant functor $\Delta_D\colon \mathscr{C} \to \mathscr{D}$ which sends every $\mathscr{C}$-object to D and every $\mathscr{C}$-arrow to 1_D.

As a special case, there is a functor $\Delta_0\colon \mathscr{C} \to \mathsf{1}$ which sends every object of $\mathscr{C}$ to the sole object of one-object category $\mathsf{1}$, and sends every arrow in $\mathscr{C}$ to the sole arrow of $\mathsf{1}$.

Those last two functors take us from richer categories to more meagre ones. Now for a couple more that go in the other direction again:

(F11) For each object C in $\mathscr{C}$ there is a corresponding functor – overloading notation once more, we can usefully call it $C\colon \mathsf{1} \to \mathscr{C}$ – which sends the sole object of $\mathsf{1}$ to C, and sends the sole arrow of $\mathsf{1}$ to 1_C.

(F12) Suppose $\mathscr{S}$ is a subcategory of $\mathscr{C}$ (see §6.2). Then there is an inclusion functor $F\colon \mathscr{S} \to \mathscr{C}$ which sends objects and arrows in $\mathscr{S}$ to the same items in $\mathscr{C}$.

20.3 What do functors preserve and reflect?

Later in this chapter we will look at three more interesting examples of functors. But let's first make some general points.

A functor $F\colon \mathscr{C} \to \mathscr{D}$ sends each $\mathscr{C}$-object C to its image $F(C)$ and sends each $\mathscr{C}$-arrow $f\colon C \to C'$ to its image $F(f)\colon FC \to FC'$. These resulting images assemble into an overall image or representation of the category $\mathscr{C}$ living in the category $\mathscr{D}$. But how good a representation do we get in the general case? What features of $\mathscr{C}$ get carried over by a functor?

(a) First a general observation worth highlighting as a theorem as it is easy to go wrong about this:

Theorem 83. *The image of $\mathscr{C}$ in $\mathscr{D}$ assembled by a functor $F\colon \mathscr{C} \to \mathscr{D}$ need not be a subcategory of $\mathscr{D}$.*

Proof. A toy example establishes the point. Let $\mathscr{C}$ be the category we can diagram as

$$\begin{tikzcd} A \arrow[r] & B_1 & B_2 \arrow[r] & C \end{tikzcd}$$

and $\mathscr{D}$ be the category

$$\begin{tikzcd} A' \arrow[r] \arrow[rr, bend left] & B' \arrow[r] & C' \end{tikzcd}$$

(where we omit the identity arrows). Suppose F_{ob} sends A to A', both B_1, B_2 to B', and C to C'; and let F_{arw} send identity arrows to identity arrows, and send the arrows $A \to B_1$ and $B_2 \to C$ respectively to $A' \to B'$ and $B' \to C'$. Trivially F with those components is functorial. But the image of $\mathscr{C}$ under F is not a category (and so not a subcategory of $\mathscr{D}$), since it contains the arrows $A' \to B'$ and $B' \to C'$ but not their composition. $\square$

(b) We next introduce a pair of standard notions:

Definition 85. Suppose $F\colon \mathscr{C} \to \mathscr{D}$ and P is some property of arrows. Then

(1) F *preserves* P iff, for any $\mathscr{C}$-arrow f, if f has property P, so does $F(f)$.
(2) F *reflects* P iff, for any $\mathscr{C}$-arrow f, if $F(f)$ has property P, so does f.

We will say, for short, that F preserves (reflects) Xs if F preserves (reflects) the property of being an X. $\triangle$

One special case gets a special bit of terminology:

Definition 86. A functor F is *conservative* iff it reflects all isomorphisms. $\triangle$

So what properties of arrows get preserved or reflected by functors in general?

Theorem 84. *Functors do not necessarily preserve or reflect monomorphisms and epimorphisms.*

Proof. First, remember 2, the two-object category which we can diagram like this:

$$\begin{tikzcd} \bullet \arrow[loop left] \arrow[r] & \star \arrow[loop right] \end{tikzcd}$$

Trivially, the non-identity arrow m here is monic. And now consider a category $\mathscr{C}$ which adds to 2 another non-identity arrow n:

$$\begin{tikzcd} \bullet \arrow[loop left] \arrow[loop above, "n"] \arrow[r, "m"] & \star \arrow[loop right] \end{tikzcd}$$

In $\mathscr{C}$, we have $m \circ n = m \circ 1_\bullet$ but not $n = 1_\bullet$, so m is not monic in $\mathscr{C}$. Hence the obvious inclusion functor from 2 to $\mathscr{C}$ does not preserve monics.

Now consider the inclusion map $i_M \colon (\mathbb{N}, +, 0) \to (\mathbb{Z}, +, 0)$ in Mon. We saw in §8.1, Ex. (2) that this is epic. But plainly the inclusion map $i_S \colon \mathbb{N} \to \mathbb{Z}$ in Set is not epic (as it isn't surjective). Therefore the forgetful functor $F \colon \mathsf{Mon} \to \mathsf{Set}$ maps an epic map (i_M) to a non-epic one (i_S), so does not preserve epics.

For an example of a functor which need not reflect monics or epics, consider a collapse functor which maps $\mathscr{C}$ to 1, thereby sending arrows of all sorts to the trivially monic and epic identity arrow on the sole object of 1. □

Theorem 85. *Functors preserve right inverses, left inverses, and isomorphisms. But functors do not necessarily reflect those.*

Proof. We show functors preserve right inverses. Suppose $F \colon \mathscr{C} \to \mathscr{D}$ is a functor and the arrow $f \colon C \to D$ is a right inverse in the category $\mathscr{C}$. Then for some arrow g, $g \circ f = 1_C$. Hence $F(g \circ f) = F(1_C)$. By functoriality, that implies $F(g) \circ F(f) = 1_{FC}$. So $F(f)$ is a right inverse in the category $\mathscr{D}$.

Duality gives the result that left inverses are preserved. And putting the two results together shows that isomorphisms are preserved.

For the negative result, just consider again the collapse functor sending $\mathscr{C}$ to 1. The only arrow in 1, the identity arrow, is trivially an isomorphism (and so a left and right inverse). The $\mathscr{C}$-arrows sent to it will generally not be. □

20.4 Faithful, full, and essentially surjective functors

The moral of the previous section is that in general a functor's image of $\mathscr{C}$ inside another category $\mathscr{D}$ may not tell us very much about $\mathscr{C}$. We are obviously going to be interested, then, in looking at some special kinds of functor which *do* preserve and/or reflect more.

Let's start by defining analogues for the notions of injective and surjective functions. First, as far as their behaviour on arrows is concerned, the useful notions for functors turn out to be these:

Definition 87. A functor $F \colon \mathscr{C} \to \mathscr{D}$ is *faithful* iff given any $\mathscr{C}$-objects C, C', and any pair of parallel arrows $f, g \colon C \to C'$, then if $F(f) = F(g)$, then $f = g$.

F is *full* (that's the standard term) iff given any $\mathscr{C}$-objects C, C', then for any arrow $g \colon FC \to FC'$ there is an arrow $f \colon C \to C'$ such that $g = Ff$.

F is *fully faithful*, some say, iff it is full and faithful. △

Note, a faithful functor needn't be, overall, injective on arrows. For suppose $\mathscr{C}$ is in effect two copies of $\mathscr{D}$, and F sends each copy faithfully to $\mathscr{D}$: then F sends two copies of an arrow to the same image arrow. However, a faithful functor is, for each pair of objects C, C', injective from the arrows $C \to C'$ to the arrows $FC \to FC'$. Likewise, a full functor needn't be, overall, surjective on arrows: but it is locally surjective from the arrows $C \to C'$ to the arrows $FC \to FC'$.

Second, in connection with the way functors treat objects, the notion worth highlighting is this:

Definition 88. A functor $F\colon \mathscr{C} \to \mathscr{D}$ is *essentially surjective on objects* (e.s.o.) iff for any $\mathscr{D}$-object D, there is a $\mathscr{C}$-object C such that $FC \cong D$. △

Plain surjectivity (defined by requiring an object C such that $FC = D$) is less interesting, given that we don't usually care, categorially speaking, whether $\mathscr{D}$ has extra non-identical-but-isomorphic copies of objects. Injectivity on objects (defined in the obvious way by requiring $FC = FC'$ implies $C = C'$, for any $\mathscr{C}$-objects C and C') is not usually very exciting either.

Some examples:

(1) The forgetful functor F : $\mathsf{Mon} \to \mathsf{Set}$ is faithful, as F sends a set-function which happens to be a monoid homorphism to itself, so different arrows in Mon get sent to different arrows in Set. But the functor is not full: there will be lots of arrows in Set that don't correspond to a monoid homomorphism. Since any set can be trivially made into a monoid, F is essentially surjective on objects.

(2) The forgetful functor $F\colon \mathsf{Ab} \to \mathsf{Grp}$ is faithful, full but not e.s.o.

(3) The 'thinning' functor from §20.2 (F9), $F\colon \mathscr{C} \to \mathscr{S}$, is full but not faithful unless $\mathscr{C}$ is already a pre-order category. But it will be e.s.o.

(4) Suppose $\mathscr{M}$ and $\mathscr{N}$ are the categories that correspond to the monoids $(M, \cdot, 1_M)$ and $(N, \times, 1_N)$. And let f be a monoid homomorphism between those monoids which is surjective but not injective. Then the functor $F\colon \mathscr{M} \to \mathscr{N}$ corresponding to f is full but not faithful.

(5) You might be tempted to say that the 'total collapse' functor $\Delta_0\colon \mathsf{Set} \to \mathbb{1}$ is full but not faithful. But it isn't full. Take C, C' in Set to be respectively the singleton of the empty set and the empty set. There is a trivial identity map in $\mathbb{1}$, $1\colon \Delta_0 C \to \Delta_0 C'$; but there is no arrow in Set from C to C'.

(6) An inclusion functor $F\colon \mathscr{S} \to \mathscr{C}$ is faithful; if $\mathscr{S}$ is a full subcategory of $\mathscr{C}$, then the inclusion map is fully faithful, but usually not e.s.o.

How then do faithful or fully faithful functors behave?

Theorem 86. *A faithful functor $F\colon \mathscr{C} \to \mathscr{D}$ reflects monomorphisms and epimorphisms.*

Proof. Suppose Ff is monic, and suppose $f \circ g = f \circ h$. Then $F(f \circ g) = F(f \circ h)$, so by functoriality $Ff \circ Fg = Ff \circ Fh$, and since Ff is monic, $Fg = Fh$. Since F is faithful, $g = h$. Hence f is monic. Dually for epics. □

Theorem 87. *If a functor is fully faithful it reflects right inverses and left inverses, and hence is conservative.*

Proof. Suppose $F\colon \mathscr{C} \to \mathscr{D}$ is a fully faithful functor, and let Ff be a right inverse, with f an arrow in $\mathscr{C}$ with source A. Since F is full, Ff must be the right inverse of Fg for some arrow g in $\mathscr{C}$. So $Fg \circ Ff = 1_{FA}$, whence $F(g \circ f) = 1_{FA} = F(1_A)$. Since F is faithful, it follows that $g \circ f = 1_A$, and f is a right inverse.

Dually, F reflects left inverses, and combining the two results shows that F reflects isomorphisms, i.e. is conservative. □

Note, however, that the reverse of the last result is not true. A functor can reflect isomorphisms without being fully faithful. Example: consider the forgetful functor $F\colon \mathsf{Mon} \to \mathsf{Set}$. This is faithful but not full. But it is conservative because if the set function Ff is an isomorphism, so is the monoid homomorphism f – for a monoid homomorphism is an isomorphism if and only if its underlying function is.

20.5 A functor from Set to Mon

(a) For this and the next two sections we step back again from generalities to look at three more particular examples of functors. First, we define a functor going in the reverse direction to the forgetful functor in (F1), i.e. we construct a functor $F : \mathsf{Set} \to \mathsf{Mon}$.

There are trivial ways of doing this. For example just pick a monoid, any monoid, call it $\mathcal{M}$. Then there is a boring constant functor we could call $!_\mathcal{M}$: $\mathsf{Set} \to \mathsf{Mon}$ which sends every set X to $\mathcal{M}$ and sends every set-function $f\colon X \to Y$ to the identity arrow $1_\mathcal{M}\colon \mathcal{M} \to \mathcal{M}$ (the identity homomorphism).

But it is instructive to try to come up with something more interesting. So, consider again how we might send sets to monoids, but this time *making as few assumptions as we possibly can* about the monoid that a given set gets mapped to.

Start then with a set S. Since we are making no more assumptions than we need to, we'll have to take the objects in S as providing us with an initial supply of objects for building our monoid, the monoid's *generators*. We now need to equip our incipient monoid with a two-place associative function $*$. But we are assuming as little as we can about $*$ too, so we don't even yet know that applying it keeps us inside the original set of generators S. So S will need to be expanded to a set M that contains not only the original members of S, e.g. $x, y, z, \ldots$, but also all the possible 'products', i.e. everything like $x * x$, $x * y$, $y * x$, $y * z$, $x * y * x$, $x * y * x * z$, $x * x * y * y * z \ldots$, etc., etc. – we know, however, that since $*$ is associative, we needn't distinguish between e.g. $x * (y * z)$ and $(x * y) * z$.

But even taking all those products is not enough, for (in our assumption-free state) we don't know whether any of the resulting elements of M will act as an identity for the $*$-function. So to get a monoid, we need to throw into M^* some unit 1. However, since we are making no assumptions, we can't assume either that any of the products in M are equal, or that there are any other objects in M other than those generated from the unit and members of S.

Now, here's a neat way to model the resulting monoid 'freely' generated from the set S. Represent a monoid element (such as $x * x * y * y * z$) as a *finite list of members of* S, so M gets represented by $List(S)$ – see (F8) above. Correspondingly, model the $*$-function by simple concatenation $^\frown$. The identity element will then be modelled by the null list. The resulting $(List(S), ^\frown, 1)$ is often simply called *the free monoid* on S – though perhaps it is better to say it is a standard exemplar of a free monoid.

Which all goes to motivate the following construction:

(F13) There is a 'free' functor $F : \mathsf{Set} \to \mathsf{Mon}$ with the following data:

i. F_{ob} sends the set S to the monoid $(List(S), ^\frown, 1)$.
ii. F_{arw} sends the arrow $f\colon S \to S'$ to $List(f)$ (see (F8) again), where this is now treated as an arrow from $(List(S), ^\frown, 1)$ to $(List(S'), ^\frown, 1)$.

It is now trivial to check that F is indeed a functor.

(b) Note, different set functions $f, g\colon X \to Y$ get sent to different functions $Ff, Fg\colon List(X) \to List(Y)$ (if $fx \neq gx$, then $Ff(\langle x\rangle) \neq Fg(\langle x\rangle)$, where $\langle x\rangle$ is the list whose sole element is x). So F is faithful.

Now consider a singleton set 1. This gets sent by F to the free monoid with a single generator – which is tantamount to $\mathcal{N} = (\mathbb{N}, +, 0)$. The sole set-function from 1 to itself, the identity function, gets sent by F to the identity monoid homomorphism on $\mathcal{N}$. But there are other monoid homomorphisms from $\mathcal{N}$ to $\mathcal{N}$, e.g. $n \mapsto 2n$. So F is not full.

(c) We can generalize. There are similar functors that send sets to other *freely generated* structures on the set. For example there is a functor from Set to Ab which sends a set X to the freely generated abelian group on X (which is in fact the direct sum of X-many copies of $(\mathbb{Z}, +, 0)$ – the integers $\mathbb{Z}$ with addition forming the paradigm free abelian group on a single generator). But we need not concern ourselves with the further details of such cases.

20.6 Products, exponentials, and functors

To develop two examples of a different type, let's consider again first products and then exponentials.

(F14) Assume $\mathscr{C}$ has all products, and C is any object in the category. Then there is a functor $- \times C\colon \mathscr{C} \to \mathscr{C}$, which sends an object A to $A \times C$ and an arrow $f\colon A \to A'$ to $f \times 1_C\colon A \times C \to A' \times C$.

Similarly there is a functor $C \times -\colon \mathscr{C} \to \mathscr{C}$, which sends an object A to $C \times A$ and an arrow $f\colon A \to A'$ to $1_C \times f\colon C \times A \to C \times A'$.

Proof. Write $f \times C$ for $(- \times C)(f)$. To confirm functoriality the main thing is to show $(g \circ f) \times C = (g \times C) \circ (f \times C)$. But that is $g \circ f \times 1_C = (g \times 1_C) \circ (f \times 1_C)$, which follows from Theorem 44.

Similarly for the other functor. □

Suppose next that we are working in a category $\mathscr{C}$ which has all exponentials (and all binary products). And suppose we have an arrow $f\colon C \to C'$ between a couple of $\mathscr{C}$-objects. Now pick another object B in the category. Then there is a commuting diagram which looks like this:

$$\begin{tikzcd} C^B \times B \arrow[r, "ev"] \arrow[d, dashed, "\overline{(f \circ ev)} \times 1_B"'] & C \arrow[d, "f"] \\ C'^B \times B \arrow[r, "ev'"'] & C' \end{tikzcd}$$

Why so? Trivially, there is a composite arrow $f \circ ev\colon C^B \times B \to C'$. But then, since $[C'^B, ev']$ is an exponential, there is by definition a *unique* transpose $\overline{f \circ ev}\colon C^B \to C'^B$ which makes the diagram commute.

In this way, for fixed B, there is a natural association between the objects C and C^B and another between the arrows $f\colon C \to C'$ and $\overline{f \circ ev}\colon C^B \to C'^B$. And, as we might hope, the associations are indeed functorial. In other words, we hope that the following is true:

(F15) Assume $\mathscr{C}$ has all exponentials, and that B is a $\mathscr{C}$-object. Then there is a corresponding exponentiation functor $(-)^B\colon \mathscr{C} \to \mathscr{C}$ which sends an object C to C^B, and sends an arrow $f\colon C \to C'$ to $\overline{f \circ ev}\colon C^B \to C'^B$.

We need, however, to confirm that this is indeed correct:

Proof. We need to confirm that $(-)^B$ does indeed preserve identities and respect composition.

The first is easy. $(1_C)^B$ is by definition $\overline{1_C \circ ev}\colon C^B \to C^B$, so we have

$$\begin{tikzcd} C^B \times B \arrow[r, "ev"] \arrow[d, dashed, "(1_C)^B \times 1_B"'] & C \arrow[d, "1_C"] \\ C^B \times B \arrow[r, "ev"'] & C \end{tikzcd}$$

But evidently, the arrow $1_{C^B} \times 1_B$ on the left would also make the diagram commute. So by the requirement that there is a unique filling for $- \times 1_B$ which makes the square commute, $(1_C)^B = 1_{C^B}$, as required for functoriality.

Second, we need to show that given arrows $f\colon C \to C'$ and $g\colon C' \to C''$, $(g \circ f)^B = g^B \circ f^B$.

Consider the following diagram where the top square, bottom square, and (outer, bent) rectangle commute:

$$\begin{tikzcd}
C^B \times B \arrow[r, "ev"] \arrow[d, dashed, "f^B \times 1_B"] \arrow[dd, dashed, bend right=60, "(g \circ f)^B \times 1_B"'] & C \arrow[d, "f"'] \\
C'^B \times B \arrow[r, "ev'"] \arrow[d, dashed, "g^B \times 1_B"] & C' \arrow[d, "g"'] \\
C''^B \times B \arrow[r, "ev''"'] & C''
\end{tikzcd}$$

By Theorem 44, $(g^B \times 1_B) \circ (f^B \times 1_B) = (g^B \circ f^B) \times 1_B$. Hence $(g^B \circ f^B) \times 1_B$ is another arrow that makes a commuting rectangle. So again by the requirement that there is a unique filling for $- \times 1_B$ which makes the square commute, $(g \circ f)^B = g^B \circ f^B$. □

20.7 An example from algebraic topology

(a) Here's another particular example of a functor, this time a classic example from algebraic topology. This can readily be skipped if you don't know the setting. Though to get a glimmer of what's going on, you just need the idea of the fundamental group of a topological space (at a point), as follows.

Given a space and a chosen base point in it, consider all directed paths that start at this base point then wander around and eventually loop back to their starting point. Such directed loops can be "added" together in an obvious way: you traverse the "sum" of two loops by going round the first loop, then round the second. Every loop has an "inverse" (you go round the same path in the opposite direction). Two loops are considered 'homotopically' equivalent if one can be continuously deformed into the other. Consider, then, the set of all such equivalence classes of loops – so-called homotopy equivalence classes – and define "addition" for these classes in the obvious derived way. This set, when equipped with addition, evidently forms a group: it is the *fundamental group* for that particular space, with the given basepoint. (Though for many spaces, the group is independent of the basepoint.)

Suppose, therefore, that Top_* is the category of pointed topological spaces: an object in the category is a topological space X equipped with a distinguished base point x_0, and the arrows in the category are continuous maps that preserve basepoints. Then here's our new example of a functor:

(F16) There is a functor $\pi_1 \colon \mathsf{Top}_* \to \mathsf{Grp}$, to use its standard label, with the following data

i. π_1 sends a pointed topological space (X, x_0) – i.e. X with base point x_0 – to the fundamental group $\pi_1(X, x_0)$ of X at x_0.

ii. π_1 sends a basepoint-preserving continuous map $f \colon (X, x_0) \to (Y, y_0)$ to a corresponding group homomorphism $f_* \colon \pi_1(X, x_0) \to \pi_1(Y, y_0)$.

(For arm-waving motivation: f maps a continuous loop based at x_0 to a continuous loop based at y_0; and since f is continuous it can be used to send a continuous deformation of a loop in (X, x_0) to a continuous deformation of a loop in (Y, y_0) – and that induces a corresponding association f_* between the homotopy equivalence classes of (X, x_0) and (Y, y_0), and this will respect the group structure.)

We will suppose that we have done the work of checking that π_1 is indeed functorial.

(b) Here, then, is a nice application. We'll prove Brouwer's famed Fixed Point Theorem:

Theorem 88. *Any continuous map of the closed unit disc to itself has a fixed point.*

Proof. Suppose that there is a continuous map f on the two-dimensional disc D (considered as a topological space) without a fixed point, i.e. such that we always have $f(x) \neq x$.

Let the boundary of the disc be the circle S (again considered as a topological space). Then we can define a map that sends the point x in D to the point in S at which the ray from $f(x)$ through x intersects the boundary of the disc.

This map sends a point on the boundary to itself. Pick a boundary point to be the base point of the pointed space D_* and also of the pointed space S_*, then our map induces a map $r\colon D_* \to S_*$. Moreover, this map is evidently continuous (intuitively: nudge a point x and since f is continuous that just nudges $f(x)$, and hence the ray from $f(x)$ through x is only nudged, and the point of intersection with the boundary is only nudged). And r is a left inverse of the inclusion map $i\colon S_* \to D_*$ in Top_*, since $r \circ i = 1$.

Functors preserve left inverses by Theorem 85, so $\pi_1(r)$ will be a left inverse of $\pi_1(i)$, which means that $\pi_1(i)\colon \pi_1(S_*) \to \pi_1(D_*)$ is a right-inverse in Grp, hence by Theorem 18 is monic, and hence by Theorem **??** is an injection.

But that's impossible. $\pi_1(S_*)$, the fundamental group of S_*, is [equivalent to] the group $\mathbb{Z}$ of integers under addition (think of looping round a circle, one way or another, n times – each positive or negative integer corresponds to a different path); while $\pi_1(D_*)$, the fundamental group of D_*, is just a one element group (for every loop in the disk D_* can be smoothly shrunk to a point). And there is no injection between the integers and a one-element set! □

(c) What, if anything, do we gain from putting the proof in category theoretic terms? It might be said: the proof crucially depends on facts of algebraic topology – continuous maps preserve homotopic equivalences in a way that makes π_1 a functor, and the fundamental groups of S^* and D^* are respectively $\mathbb{Z}$ and the trivial group. And we could run the whole proof without actually mentioning categories at all. Still what we've done is, so to speak, very clearly demarcate those bits of the proof that depend on topic-specific facts of algebraic topology and those bits which depend on general proof-ideas about functoriality and about kinds of maps (inverses, monics, injections), ideas which are thoroughly *portable* to other contexts. And *that* surely counts as a gain in understanding.

20.8 Covariant vs contravariant functors

Here, finally, is another a very general question about functors. How do they interact with the operation of taking the opposite category?

Well, first we note:

Theorem 89. *A functor $F\colon \mathscr{C} \to \mathscr{D}$ induces a functor $F^{op}\colon \mathscr{C}^{op} \to \mathscr{D}^{op}$.*

Proof. Recall, the objects of $\mathscr{C}^{op}$ are exactly the same as the objects of $\mathscr{C}$. We can therefore define the object-mapping component of F^{op} as acting on $\mathscr{C}^{op}$-objects exactly as the object-mapping component of F acts on $\mathscr{C}$-objects. And then, allowing for the fact that taking opposites reverses arrows, we can define the arrow-mapping component of F^{op} as acting on the $\mathscr{C}^{op}$-arrow $f\colon C \to D$ exactly as the arrow-mapping component of F acts on the $\mathscr{C}$-arrow $f\colon D \to C$.

F^{op} will evidently obey the axioms for being a functor because F does. □

By the way, had we shown this before, we could have halved the work in our proof of Theorem 84 that functors do not necessarily preserve monics or epics. After we'd shown that $F\colon \mathsf{2} \to \mathscr{C}$ doesn't preserve monics, we could have just remarked that the $F^{op}\colon \mathsf{2}^{op} \to \mathscr{C}^{op}$ won't preserve epics!

Now for a new departure. We introduce a variant kind of functor:

Definition 89. $F\colon \mathscr{C} \to \mathscr{D}$ is a *contravariant* functor from $\mathscr{C}$ to $\mathscr{D}$ if $F\colon \mathscr{C}^{op} \to \mathscr{D}$ is a functor in the original sense. So it comprises the following data:

(1) A mapping F_{ob} whose value at the $\mathscr{C}$-object A is some $\mathscr{D}$-object $F(A)$.
(2) A mapping F_{arw} whose value at the $\mathscr{C}$-arrow $f\colon B \to A$ is a $\mathscr{D}$-arrow $F(f)\colon FA \to FB$. (NB the directions of the arrows!)

And this data satisfies the two axioms:

Preserving identities: for any $\mathscr{C}$-object A, $F(1_A) = 1_{F(A)}$;
Respecting composition: for any $\mathscr{C}$-arrows f, g such that their composition $g \circ f$ exists, $F(g \circ f) = Ff \circ Fg$. (NB the order of the two compositions!) △

Two comments. First, a functor in our *original* sense, when the contrast needs to be stressed, is also called a *covariant* functor. Second, it would of course be equivalent to define a contravariant functor from $\mathscr{C}$ to $\mathscr{D}$ to be a covariant functor from $\mathscr{C}$ to $\mathscr{D}^{op}$.

Let's finish the chapter, then, with a couple of examples of naturally arising contravariant functors.

(1) We have already met the covariant powerset functor. Its contravariant twin $\overline{P}\colon \mathsf{Set} \to \mathsf{Set}$ again maps a set to its powerset, and maps a set-function $f\colon Y \to X$ to the function which sends $U \in \mathscr{P}(X)$ to its inverse image $f^{-1}[U] \in \mathscr{P}(Y)$ (where $f^{-1}[U] = \{x \mid f(x) \in U\}$).

(2) Take Vect, the category whose objects are the finite dimension vector spaces over the reals, and whose arrows are linear maps between spaces.

Now recall, the dual space of given finite-dimensional vector space V over the reals is V^*, the set of all linear functions $f\colon V \to \mathbb{R}$ (where this set is equipped with vectorial structure in the obvious way). V^* has the same dimension as V (so, a fortiori, is also finite dimensional and belongs to Vect). We'll construct a dualizing functor $D\colon \mathsf{Vect} \to \mathsf{Vect}$, where D_{ob} sends a vector-space to its dual.

So how is our functor D going to act on arrows in the category Vect? Take spaces V, W and consider any linear map $g\colon W \to V$. Then, over on the dual spaces, there will be a naturally corresponding map $(-\circ g) : V^* \to W^*$ which maps $f\colon V \to \mathbb{R}$ to $f \circ g\colon W \to \mathbb{R}$. *But note the direction that the arrow g has to go in, if composition with f is to work.* This defines the action of a component D_{arw} for the dualizing functor D: it will send a linear map g to the map $(- \circ g)$.

And these components D_{ob} and D_{arw} evidently do give us a contravariant functor.

21 Categories of categories

We have seen how structured whatnots and structure-respecting maps between them can be assembled into categories. This gives us more structured data, the categories; and now we have also seen there are structure-respecting maps between *them*, i.e. functors. Can data of these last two sorts be assembled into further categories? Yes indeed. Quite unproblematically, there are at least some categories of categories.

However, just as we can have many sets of sets but arguably not, on pain of paradox, a set of *all* sets, so we can have many categories of categories but arguably not, on pain of paradox, a category of *all* categories. Some collections are, as the saying goes, 'too big to be sets'; there are similar worries about some assemblies of categories being 'too big'. We need then briefly to address these issues of size, which we have previously skated around once or twice.

21.1 Functors compose

Here are two simple theorems. In each case the proof is entirely straightforward from the definitions:

Theorem 90. *Given any category $\mathscr{C}$ there is an identity functor $1_{\mathscr{C}}\colon \mathscr{C} \to \mathscr{C}$ which sends objects and arrows alike to themselves.*

Theorem 91. *Suppose there exist functors $F\colon \mathscr{C} \to \mathscr{D}$, $G\colon \mathscr{D} \to \mathscr{E}$. Then there is also a composite functor $G \circ F\colon \mathscr{C} \to \mathscr{E}$ with the following data:*

(1) A mapping $(G \circ F)_{ob}$ which sends a $\mathscr{C}$-object A to the $\mathscr{E}$-object GFA – i.e., if you prefer that with brackets, to $G(F(A))$.

(2) A mapping $(G \circ F)_{arw}$ which sends a $\mathscr{C}$-arrow $f\colon A \to B$ to the $\mathscr{E}$-arrow $GFf\colon\ GFA \to GFB$ – i.e. to $G(F(f))$.

Further, such composition of functors is associative.

By the way, again to reduce clutter, we will later often allow ourselves to write simply 'GF' for the composite functor rather than '$G \circ F$'.

What happens if we compose two contravariant functors?

Theorem 92. *The composition of two contravariant functors, where defined, yields a covariant functor.*

That's immediate once we reflect that if the contravariant F and G compose, F sends an arrow $f\colon A \to B$ to $Ff\colon FB \to FA$ and then G sends that on to $GFf\colon GFA \to GFB$.

In other respects too, composition behaves just as you would expect on a moment's thought. For example:

Theorem 93. *The composition of full functors is full and the composition of faithful functors is faithful.*

Again the proof writes itself. Being full is being locally surjective, and compositions of surjective functions are surjective; similarly for faithfulness.

21.2 Categories of categories

The basic observations that there are identity functors, and that functors compose associatively now ensure that the following definition is in good order:

Definition 90. Suppose $\mathscr{X}$ comprises two sorts of data:

(1) *Objects*: some categories, $\mathscr{C}, \mathscr{D}, \mathscr{E}, \ldots,$

(2) *Arrows*: some functors, $F, G, H, \ldots,$ between those categories,

where the arrows (i) include the identity functor on each category, and (ii) also include $G \circ F$ for each included composable pair F and G (where F's target is G's source). Then $\mathscr{X}$ is a *category of categories.* △

Let's have some quick examples:

(1) Trivially, there is a category of categories whose sole object is the category $\mathscr{C}$ and whose sole arrow is identity functor $1_{\mathscr{C}}$.

(2) We noted that every monoid can be thought of as itself being a category. Hence the familiar category Mon can also be regarded as a category of categories.

(3) There is a category whose objects are the finite categories, and whose arrows are all the functors between finite categories.

So there certainly are *some* examples of categories of categories. But, as we have already indicated, there are limitations.

21.3 A universal category?

(a) Suppose we next say:

Definition 91. A category is *normal* iff it is not one of its own objects. △

The categories which we have met in previous chapters have all been normal. Now ask: can all the normal categories be gathered together as the objects of one really big category?

The answer is given by

Theorem 94. *There is no category of all normal categories.*

Proof. Suppose there is a category $\mathscr{N}$ whose objects are all the normal categories. Now ask, is $\mathscr{N}$ normal? If it is, then it is one of the objects of $\mathscr{N}$, so $\mathscr{N}$ is non-normal. So $\mathscr{N}$ must be non-normal. But then it is not one of the objects of $\mathscr{N}$, so $\mathscr{N}$ is normal after all. Contradiction. □

This argument of course just re-runs, in our new environment, the very familiar argument from Russell's Paradox to the conclusion that there is no set of all the normal sets (where a set is normal iff it is not a member of itself).

It is worth stressing that the Russellian argument is *not* especially to do with sets, for at its core is a simple, purely logical, observation. Thus, take *any* two-place relation R defined over some objects; then there can be no object r among them which is related by R to all and only those objects which are not R-related to themselves. In other words, it is a simple logical theorem that $\neg\exists r\forall x(Rxr \leftrightarrow \neg Rxx)$. Russell's original argument applies this elementary general result to the particular set-theoretic relation R_1, '… is a set which is a member of the set …', to show that there is no set of all normal (i.e. non-self-membered) sets. Our argument above now applies the same logical theorem to the analogous category-theoretic relation R_2, '… is a category which is an object of the category …', to show that there is no category of all normal categories.

(b) Russell's original argument that there is no set of all *normal* sets is usually taken to entail that, a fortiori, there is no universal set, no set of *all* sets. The reasoning being that if there were a universal set then we should be able carve out of it (via a separation principle) a subset containing just those sets which are normal, which we now know can't be done.

To keep ourselves honest, however, we should note that this *further* argument can be, and has been, resisted. There are cogent set theories on the market which allow universal sets. How can this possibly be? Well, we can motivate restricting separation and can thus block the argument that, if there is a universal set of all sets, we should in particular be able to carve out from it a set of all normal sets: see Forster (1995) for a classic discussion of set theories with a universal set which work this way. But we can't discuss this type of deviant theory here. Henceforth we'll have to just assume a standard line on sets at least in this respect – there are 'limitations of size', i.e. there are some entities (e.g. the sets themselves) which are too many to form a set.

Now, similarly to the argument about sets, the Russellian argument that there is no category of all normal categories might naturally be taken to entail that there is no universal category in the naive sense:

Definition 92. A category $\mathscr{U}$ is *universal* if it is a category of categories such that every category is an object of $\mathscr{U}$.

Theorem 95? *There is no universal category.*

The argument goes: suppose a universal category $\mathscr{U}$ exists. Then we could carve out from it a subcategory whose objects are just the normal categories, to get a category of all normal categories. But we have shown there can be no such category.

Can this line of argument be resisted? Could we justify saying that even if there is a category of *all* categories, we can't actually select out the normal categories and all the arrows between them to give us a subcategory of *normal* categories? Well, perhaps some themes in the debates about set theories with a universal set could be carried over to this case. But again, it would take us far too far away from mainstream concerns in category theory to try to explore this option any further here.

Let's not fuss about the possibility of a universal category any more but simply take it that, at least in the naive sense of Defn. 92, there is no such thing. Instead, we turn our attention to defining two much more useful notions of large-but-less-than-universal categories-of-categories.

21.4 'Small' and 'locally small' categories

(a) To repeat: when we talk here about sets, we assume we are working in a theory of sets which is standard at least in the respect of allowing that the sets are too many to themselves form a set.

We continue with a three new definitions:

Definition 93. A category $\mathscr{C}$ is *finite* iff it has overall only a finite number of arrows.

A category $\mathscr{C}$ is *small* iff it has overall only a 'set's worth' of arrows – i.e. the arrows of $\mathscr{C}$ can be put into one-one correspondence with the members of some set.

A category $\mathscr{C}$ is *large* iff it isn't small overall. But it counts as *locally small* iff for every pair of $\mathscr{C}$-objects C, D there is only a 'set's worth' of arrows from C to D, i.e. those arrows can be put into one-one correspondence with the members of some set. △

Some comments and examples:

(1) The terms 'small' and 'locally small' are standard.

(2) It would be more usual to say that in a small category the arrows themselves form a set. However, if our favoured set theory is a theory like pure ZFC where sets only have other sets as members, that would presuppose that arrows are themselves pure sets, and we might not necessarily want to make that assumption. So, for smallness, let's officially require only that

the arrows aren't too many to be indexed by a set. Similarly for local smallness.

(3) Since for every object in $\mathscr{C}$ there is at least one arrow, namely the identity arrow on $\mathscr{C}$, a finite category must have a finite number of objects. And if there are too many objects of $\mathscr{C}$ to be bijectively mapped to a set, then $\mathscr{C}$ has too many arrows to be small. Contraposing, if $\mathscr{C}$ is small, not only its arrows but its objects can be put into one-one correspondence with the members of some set (in fact, the set that indexes the identity arrows).

(4) Among our examples in §4.7, tiny finite categories like $\mathsf{1}$ and $\mathsf{2}$ are of course small. But so too are the categories corresponding to an infinite but set-sized monoid or to an infinite pre-ordered set. Categories such as Set or Mon, however, have too many objects (and hence too many arrows) to be small.

(5) While categories such as Set or Mon are not small, like all our other examples so far they are at least *locally* small. In Set, for example, the arrows between objects C to D are members of a certain subset of the powerset of $C \times D$: which makes Set locally small. (Indeed some authors build local smallness into their preferred definition of a category – see for example Schubert 1972, p. 1; Borceux 1994, p. 4; Adámek et al. 2009, p. 21.)

(b) Let's propose two more definitions:

Definition 94. Cat is the category whose objects are small categories and whose arrows are the functors between them.

Cat^* is the category whose objects are locally small categories and whose arrows are the functors between them. △

Are such definitions in good order?

Well, at least there aren't Russellian problems. First, a discrete category (with just identity arrows) only has as many arrows as objects. Which implies that the discrete category on any set is small. But that in turn implies that there are at least as many small categories as there are sets. Hence the category Cat of small categories has at least as many objects as there are sets, and hence is itself determinately *not* small. Since Cat is unproblematically not small, no paradox arises for Cat as it did for the putative category of normal categories.

Second, take a one-element category $\mathsf{1}$, which is certainly locally small. Then a functor from $\mathsf{1}$ to Set will just map the object of $\mathsf{1}$ to some particular set: and there will be as many distinct functors $F\colon \mathsf{1} \to \mathsf{Set}$ as there are sets. In other words, arrows from $\mathsf{1}$ to Set in Cat^* are too many to be mapped one-to-one to a set. Hence Cat^* is determinately *not* locally small. So again no Russellian paradox arises for Cat^*.

21.5 Isomorphisms between categories

(a) It seems, therefore, that we can legitimately talk of the category of small categories Cat. And if we don't build local smallness into the very definition of a category, as some do, then it seems that we can legitimately talk of the larger category of locally small categories Cat^*. Maybe we can countenance still more inclusive categories of categories.

It will be handy to have some flexible notation to use, in a given context, for a suitable category of categories that includes at least all the categories which are salient in that context: let's use CAT for this. We can then start applying familiar categorial definitions. For example,

Definition 95. A functor $F\colon \mathscr{C} \xrightarrow{\sim} \mathscr{D}$ is an *isomorphism* between categories in CAT iff it has an inverse, i.e. there is a functor $G\colon \mathscr{D} \to \mathscr{C}$ where $G \circ F = 1_{\mathscr{C}}$ and $F \circ G = 1_{\mathscr{D}}$. △

Here, $1_{\mathscr{C}}$ is of course the functor that sends every object to itself and every arrow to itself. And the definition makes the notion of being an isomorphism sensibly stable in the sense that if $F\colon \mathscr{C} \xrightarrow{\sim} \mathscr{D}$ is an isomorphism between categories in some CAT it remains an isomorphism in a more inclusive category.

As we would expect,

Theorem 96. *If $F\colon \mathscr{C} \xrightarrow{\sim} \mathscr{D}$ is an isomorphism, it is full and faithful.*

Proof. First suppose we have parallel arrows in $\mathscr{C}$, namely $f, g\colon A \to B$. Supposing $Ff = Fg$, then $GFf = GFg$ – where G is F's inverse (now supressing the clutter of explicit composition signs). So $1_{\mathscr{C}}f = 1_{\mathscr{C}}g$ and hence $f = g$. Therefore F is faithful.

Suppose we are given an arrow $h\colon FA \to FB$. Put $f = Gh$. Then $Ff = FGh = 1_{\mathscr{D}}h = h$. So every such h in $\mathscr{D}$ is the image under F of some arrow in $\mathscr{C}$. So F is full. □

The converse doesn't hold, however. We noted that the inclusion functor from a full subcategory $\mathscr{S}$ of $\mathscr{C}$ into $\mathscr{C}$ is fully faithful: but plainly that usually won't have an inverse.

(b) Just as we say that objects C and D inside a category are isomorphic iff there is an isomorphism $f\colon C \to D$, so we naturally say:

Definition 96. Categories $\mathscr{C}$ and $\mathscr{D}$ are *isomorphic* in CAT, in symbols $\mathscr{C} \cong \mathscr{D}$, iff there is an isomorphism $F\colon \mathscr{C} \xrightarrow{\sim} \mathscr{D}$. △

Let's have some examples:

(1) Take the toy two-object categories with different pairs of objects which we can diagram as

$$\circlearrowright \bullet \longrightarrow \star \circlearrowleft \qquad\qquad \circlearrowright a \longrightarrow b \circlearrowleft$$

Plainly they are isomorphic (and indeed there is a unique isomorphic functor that sends the first to the second)! If we don't care about distinguishing copies of structures that are related by a unique isomorphism, then we'll count these as the same in a strong sense. Which to that extent warrants our earlier talk about *the* category 2 (e.g. in §4.7, Ex. (C7)).

(2) Revisit the example in §6.3 of the coslice category $1/\mathsf{Set}$. This category has as objects all the arrows $\vec{x}\colon 1 \to X$ for any $X \in \mathsf{Set}$. And the arrows from $\vec{x}\colon 1 \to X$ to $\vec{y}\colon 1 \to Y$ are just the set-functions $j\colon X \to Y$ such that $j \circ \vec{x} = \vec{y}$.

Now we said before that this is in some strong sense 'the same as' the category Set_* of pointed sets. And indeed the categories are isomorphic. For take the function F_{ob} from objects in $1/\mathsf{Set}$ to objects Set_* which sends an object $\vec{x}\colon 1 \to X$ to the pointed set (X, x), i.e. X-equipped-with-the-basepoint-x, where x is the value of the function $\vec{x}$ for its sole argument. And take F_{arw} to send an arrow $j\colon X \to Y$ such that $j \circ \vec{x} = \vec{y}$ to an arrow $j'\colon (X, x) \to (Y, y)$ agreeing at every argument and preserving base points. Then it is trivial to check that F is a functor $F\colon 1/\mathsf{Set} \to \mathsf{Set}_*$.

In the other direction, we can define a functor $G\colon \mathsf{Set}_* \to 1/\mathsf{Set}$ which sends (X, x) to the function $\vec{x}\colon 1 \to X$ which sends the sole object in 1 to the point x, and sends a basepoint-preserving function from X to Y to itself.

And it is immediate that these two functors F and G are inverse to each other. Hence, as claimed, $\mathsf{Set}_* \cong 1/\mathsf{Set}$.

(3) For those who know just a little about Boolean algebras and the two alternative ways of presenting them: There is a category Bool whose objects are algebras $(B, \neg, \wedge, \vee, 0, 1)$ constrained by the familiar Boolean axioms, and whose arrows are homomorphisms that preserve algebraic structure. And there is a category BoolR whose objects are Boolean rings, i.e. rings $(R, +, \times, 0, 1)$ where $x^2 = x$ for all $x \in R$, and whose arrows are ring homomorphisms.

There is also a familiar way of marrying up Boolean algebras with corresponding rings, and vice versa. Thus if we start from $(B, \neg, \wedge, \vee, 0, 1)$, take the same carrier set and distinguished objects, put

(i) $x \times y =_{\text{def}} x \wedge y$,
(ii) $x + y =_{\text{def}} (x \vee y) \wedge \neg(x \wedge y)$ (exclusive 'or'),

then we get a Boolean ring. And if we apply the same process to two algebras B_1 and B_2, it is elementary to check that this will carry a homomorphism of algebras $f_a\colon B_1 \to B_2$ to a corresponding homomorphism of rings $f_r\colon R_1 \to R_2$. We can equally easily go from rings to algebras, by putting

(i) $x \wedge y =_{\text{def}} x \times y$,
(ii) $x \vee y =_{\text{def}} x + y + (x \times y)$
(iii) $\neg x =_{\text{def}} 1 + x$.

Note that going from a algebra to the associated ring and back again takes us back to where we started.

In summary, without going into any more details, we can in this way define a functor $F\colon \mathsf{Bool} \to \mathsf{BoolR}$, and a functor $G\colon \mathsf{BoolR} \to \mathsf{Bool}$ which are inverses to each other. So, as we'd surely have expected, the category Bool is isomorphic to the category BoolR.

(4) We will meet two more examples of isomorphic categories in §24.3.

So far, so good then. We have examples of pairs of categories which, intuitively, 'come to just the same' and are indeed isomorphic by our definition. Looking ahead to Chapter 28, however, it turns out that being isomorphic is not the notion of 'amounting to the same category' which is most useful. We in fact need a rather more relaxed notion of equivalence of categories. More about this later.

(c) For the moment, then, we just note that we can also carry over e.g. our categorial definition of initial and terminal objects and other limits to categories in CAT. We can check the following, for example:

Theorem 97. *The empty category is initial in* CAT*, and the trivial one-object category* 1 *is terminal.*

Theorem 98. *The category $\mathscr{C} \times \mathscr{D}$ (as defined in §6.2), equipped with the obvious projection functors $\Pi_1\colon \mathscr{C} \times \mathscr{D} \to \mathscr{C}$ and $\Pi_2\colon \mathscr{C} \times \mathscr{D} \to \mathscr{D}$ forms a categorial binary product of $\mathscr{C}$ with $\mathscr{D}$.*

21.6 An aside: other definitions of categories

(a) Having at long last explicitly highlighted the theme of categories with too many objects to form a set, now is the moment to pause to revisit our definition of the very idea of a category to explain its relation to other, sightly different, definitions. For issues of size crop up again.

Our own preferred definition began like this:

Definition 13 A category $\mathscr{C}$ comprises two kinds of things:

(1) *Objects* (which we will typically notate by 'A', 'B', 'C', ...).

(2) *Arrows* (which we typically notate by 'f', 'g', 'h', ...). ...

This accords with e.g. Awodey (2006, p. 4) and Lawvere and Schanuel (2009, p. 21). And this is given as a 'direct description' of categories by (Mac Lane 1997, p. 289). However, it is at least as common to put things as follows:

Definition 13* A category $\mathscr{C}$ consists of

(1) A collection *Obj* of entities called *objects*.

(2) A collection *Arw* of entities called *arrows*. ...

See (Goldblatt 2006, p. 24), and Simmons (2011, p. 2) for such definitions, and also e.g. Goedecke (2013).

Others prefer to talk of 'classes' here, but we probably shouldn't read very much into *that* choice of wording, 'collections' vs 'classes'. The real question is: what, if anything, is the difference between talking of a category as having as data some objects (plural) and some arrows (plural), and saying that a category consists in a collection/class (singular) of objects and a collection/class (singular) of arrows?

It obviously all depends what we mean here by 'collections'. Because many paradigm categories have too many objects for there to be a set of them, the notion of collection can't be just the standard notion of a set again. But that still leaves options. Is Defn. 13* in fact intended to involve only 'virtual classes', meaning that the apparent reference to classes is a useful fiction but can be translated away so that it ends up saying no more than is said by Defn. 13 which doesn't refer to collections-as-special-objects at all? Or is Defn. 13* to be read as buying into some overall two-layer theory of sets-plus-bigger-classes which in some way takes *large* collections, classes-which-aren't-sets, more seriously (and if so, then *how* seriously)?

Well, note that we have in fact been able to proceed quite far without making any clear assumption that categories are in some strong sense distinct entities over and above their objects and arrows (arguably, even talk of categories of categories doesn't commit us to that). In other words, it isn't obvious that we as yet *need* to buy in to a substantive theory of classes to get our theorizing about categories off the ground. For this reason, I prefer to stick to the overtly non-committal Defn. 13 as our initial definition, and thereby leave it as a separate question just when, and in what contexts, the category theorist eventually does make moves that require taking seriously collections-bigger-than-any-ordinary-set.

(b) While on the subject of variant definitions of category, here's another common one. It starts like this:

Definition 13** The data for a category $\mathscr{C}$ comprises:

(1) A collection $ob(\mathscr{C})$, whose elements we will call *objects*.
(2) For every $A, B \in ob(\mathscr{C})$, a collection $\mathscr{C}(A, B)$, whose elements f we will call *arrows* from A to B. We signify that the arrow f belongs to $\mathscr{C}(A, B)$ by writing $f\colon A \to B$ or $A \xrightarrow{f} B$.
(3) For every $A \in ob(\mathscr{C})$, an arrow $1_A \in \mathscr{C}(A, A)$ called the *identity* on A.
(4) For any $A, B, C \in ob(\mathscr{C})$, a two-place *composition* operation, which takes arrows f, g, where $f\colon A \to B$ and $g\colon B \to C$, to an arrow $g \circ f\colon A \to C$, the composite of f and g. ... △

This is essentially the definition given by Leinster (2014, p. 10). Relatedly, consider Borceux (1994, p. 4) and Adámek et al. (2009, p. 18) who have a category consisting of a class of objects but who insist that each collection of arrows between specific objects is to be a *set* – so they build local smallness into the very definition of a category.

Leaving aside the last point, the key difference is that Defn. 12* has one all-in class of arrows, Defn. 12** has lots of different classes (or sets) of arrows, one for every pair of objects in the category.

Obviously if we start from Defn. 12 or Defn. 12*, we can then augment it by defining the collection $\mathscr{C}(A, B)$ of arrows from A to B as containing the $\mathscr{C}$-arrows f such that $src(f) = A$ and $tar(f) = B$. Note, though, on Defn. 12 or Defn. 12* the arrows $f\colon A \to B$ and $f'\colon A' \to B'$ cannot be identical if $A \neq A'$ or $B \neq B'$. For if $src(f) \neq src(f')$, $f \neq f'$; likewise, of course, if $tar(f) \neq tar(f')$, $f \neq f'$. Hence, according to the now augmented Defn. 12 or Defn. 12*, if $A \neq A'$ or $B \neq B'$, $\mathscr{C}(A, B)$ and $\mathscr{C}(A', B')$ are disjoint. On the other hand, there's nothing in Defn. 12** which requires that. Which means that our two definitions don't quite line up. What to do?

The easy option is just to add to Defn. 12** the stipulation that the collections $\mathscr{C}(A, B)$ for different pairs of objects A, B are indeed disjoint. Adámek et al. (2009) adds just such a stipulation 'for technical convenience' and Leinster (2014) does the same. If we stick though to our original definition Defn. 12 (or to Defn. 1*, if you insist), then you get the same requirement for free.

22 Functors and limits

As we have seen, a functor $F\colon \mathscr{J} \to \mathscr{C}$ will, just in virtue of its functoriality, preserve/reflect some aspects of the categorial structure of $\mathscr{J}$ as it sends objects and arrows into $\mathscr{C}$. And if the functor has properties like being full or faithful it will preserve/reflect more.

We now want to ask: how do things stand with respect to preserving/reflecting limits and colimits?

22.1 Diagrams redefined as functors

(a) Now that we have the notion of a functor to hand, we can redefine the notion of a diagram, and hence the notion of a (co)limit over a diagram, in a particularly neat way.

We can think of a functor from one category to another as producing a kind of image or representation of the first category which lives in the second category – see the beginning of §20.3. Or, to say the same thing in other words, a functor $D\colon \mathscr{J} \to \mathscr{C}$ produces a sort of diagram of the category $\mathscr{J}$ inside $\mathscr{C}$. This thought in turn motivates overloading terminology in the following standard way:

Definition 97. Given a category $\mathscr{C}$, and a category J, we say that a functor $D\colon \mathsf{J} \to \mathscr{C}$, is a *diagram (of shape* J*) in* $\mathscr{C}$. △

(Here we start following what seems a rather common font-convention, and use e.g. 'J' rather than '$\mathscr{J}$' when a small – often *very* small – category is likely to be in focus: some indeed would build the requirement that $\mathscr{J}$ is small into our definition here of a diagram-as-functor.)

To go along with this definition of diagrams-as-functors, there are entirely predictable corresponding definitions of cones and limit cones (we just modify in obvious ways the definitions we met in §15.1, 15.2):

Definition 98. Suppose we are given a category $\mathscr{C}$, together with J a (possibly very small) category, and a diagram-as-functor $D\colon \mathsf{J} \to \mathscr{C}$. Then:

(1) *A cone over* D is an object $C \in \mathscr{C}$, together with an arrow $c_J\colon C \to D(J)$ for each J-object J, such that for any J-arrow $d\colon K \to L$, $c_L = D(d) \circ c_K$. We use $[C, c_J]$ (where 'J' is understood to run over objects in J) for such a cone.

(2) *A limit cone over* D is a cone we can notate $[\underset{\leftarrow \mathsf{J}}{Lim}\, D, \lambda_J]$ such that for every cone $[C, c_J]$ over D, there is a unique arrow $u\colon C \to \underset{\leftarrow \mathsf{J}}{Lim}\, D$ such that, for all J-objects J, $\lambda_J \circ u = c_J$. △

(b) How does our talk of diagrams and limits, old and new, interrelate? Three points:

(1) To repeat the motivating thought, a functor $D\colon \mathsf{J} \to \mathscr{C}$ will send the objects and arrows of J to a corresponding handful of objects and arrows sitting inside $\mathscr{C}$ and those latter objects will be indexed by the objects of J. So diagrams-as-functors of course generate diagrams-in-categories in the sense introduced rather loosely in §5.1 and then refined in §15.2.

(2) But on the other hand, not every diagram-in-$\mathscr{C}$ in the original sense corresponds to a diagram-as-functor. There's a trivial reason. A diagram of shape J in $\mathscr{C}$ will always carry over the required identity arrows on all the objects in J to identity arrows on all their images. But a diagram-in-a-category as we first defined it doesn't have to have identity arrows on all (or indeed any) of its objects.

(3) Still, the lack of a straight one-to-one correspondence between diagrams in the two senses makes no difference when thinking about limits. Limits over diagrams-as-functors will of course be limits in the old sense. And conversely, suppose $[L, \lambda_j]$ is a limit cone over some diagram D in $\mathscr{C}$ (diagram in the original sense). Then by Theorem 53, $[L, \lambda_j]$ is a limit over the (reflexive, transitive) closure of D (because *every* cone over D is equally a cone over its closure). By Theorem 52, we can think of this closure as a subcategory J of $\mathscr{C}$. So take the inclusion functor $D_i\colon \mathsf{J} \to \mathscr{C}$. Then, by our new definition, $[L, \lambda_j]$ is a limit cone over the diagram-as-functor $D_i\colon \mathsf{J} \to \mathscr{C}$. In short, limits old and new are just the same.

If our prime interest is in limits, then, we can in fact take the neat notion of diagram just introduced in Defn. 97 to be the primary one. And indeed, this line is widely, though not universally, adopted (compare e.g. Borceux 1994 and Leinster 2014). We too will think of diagrams this way from now on.

22.2 Preserving limits

(a) Start with a natural definition, extending the notion of preservation we met in §20.3: we say a functor preserves limits if it sends limits of a given shape to limits of the same shape and preserves colimits if it sends colimits to colimits. More carefully,

Definition 99. A functor $F\colon \mathscr{C} \to \mathscr{D}$ *preserves the limit* $[L, \lambda_J]$ over $D\colon \mathsf{J} \to \mathscr{C}$ iff $[FL, F\lambda_J]$ is a limit over $F \circ D\colon \mathsf{J} \to \mathscr{D}$.

More generally, a functor $F\colon \mathscr{C} \to \mathscr{D}$ *preserves limits of shape* J *in* $\mathscr{C}$ iff, for any diagram $D\colon \mathsf{J} \to \mathscr{C}$, if $[L, \lambda_J]$ is some limit cone over D, then F preserves it.

A functor which preserves limits of shape J in $\mathscr{C}$ for all finite (small) categories J is said to *preserve all finite (small) limits* (in $\mathscr{C}$).

Dually for preserving colimits. △

Preservation indeed behaves as you would expect in various respects. We will mention two:

Theorem 99. *If F preserves products, then $F(A \times B) \cong FA \times FB$.*

Proof. Assume F is a functor from $\mathscr{C}$ to $\mathscr{D}$. Suppose $\overline{2}$ is the discrete category with two objects, call them 0 and 1. Then, in terms of our new notion of a diagram, a product in $\mathscr{C}$ is a limit over some diagram $D\colon \overline{2} \to \mathscr{C}$. Take the diagram where $D(0) = A$ and $D(1) = B$. Then the product of course will be some $[A \times B, \pi_1, \pi_2]$.

By hypothesis, $[F(A \times B), F\pi_1, F\pi_2]$ is a limit over the diagram $F \circ D\colon \overline{2} \to \mathscr{D}$. That is to say it is a limit over the diagram in $\mathscr{D}$ (in our old sense of diagram) with just the objects FA and FB and their identity arrows. So it is a product; and another product over that diagram is $[FA \times FB, \pi_1', \pi_2']$ with appropriate projection arrows. By Theorem 34, these two products are isomorphic, hence $F(A \times B) \cong FA \times FB$. □

Theorem 100. *If $F\colon \mathscr{C} \to \mathscr{D}$ preserves some limit over the diagram $D\colon \mathsf{J} \to \mathscr{C}$, it preserves all limits over that diagram.*

Proof. Suppose $[L, \lambda_J]$ is a limit cone over $D\colon \mathsf{J} \to \mathscr{C}$. Then, by Theorem 54, if $[L', \lambda'_J]$ is another such cone, there is an isomorphism $f : L' \to L$ in $\mathscr{C}$ such that $\lambda'_J = \lambda_j \circ f$.

Suppose now that F preserves $[L, \lambda_J]$ so $[FL, F\lambda_J]$ is a limit cone over $F \circ D$. Then F will send $[L', \lambda'_J]$ to $[FL', F\lambda'_J] = [FL', F\lambda_J \circ Ff]$. But then this factors through $[FL, F\lambda_J]$ via the isomorphism $Ff\colon FL' \to FL$ (remember, functors preserve isomorphisms). Hence, by Theorem 55, $[FL', F\lambda'_J]$ is also a limit over $F \circ D$. In other words, F preserves $[L', \lambda'_J]$ too. □

But these general conditional claims don't tell us anything about which particular products or other limits actually do get preserved by which functors: now we need to get down to cases.

(b) Here is a first very simple example and then two further (rather artificial) toy examples, which together nicely illustrate some general points about how functors can *fail* to preserve limits.

(1) Take the posets $(\{0, 1, 2\}, \leqslant)$ and $(\mathbb{N}, \leqslant)$ thought of as categories. There is a trivial inclusion functor I from the first category to the second. Now, 2 is a terminal object in the first category, but $2 = I(2)$ is not terminal in the second. So I doesn't preserve that terminal object (the limit over the diagram-as-functor from the empty category).

I does, however, preserve products (recall the product of two elements in a poset, when it exists, is their greatest lower bound).

Two morals. First, since a functor need not preserve even terminal objects, functors certainly need not preserve limits generally. Second, a functor may preserve some limits and not others.

There is entertainment to be had in looking at a couple more illustrations of that second point:

(2) Take the functor $P\colon \mathsf{Set} \to \mathsf{Set}$ which sends the empty set $\varnothing$ to itself and sends every other set to the singleton 1, and acts on arrows in the only possible way if it is to be a functor (i.e. for $A \neq \varnothing$, it sends any arrow $\varnothing \to A$ to the unique arrow $\varnothing \to 1$, it sends the arrow $\varnothing \to \varnothing$ to itself and sends all other arrows to the identity arrow 1_1). Claim: *P preserves binary products but not equalizers* – i.e. it preserves all limits of the shape of the discrete two-object category but not all those of shape $\circlearrowright \bullet \rightrightarrows \star \circlearrowleft$.

Proving this claim is a routine exercise. For the first half, we simply consider cases. If neither A nor B is the empty set, then $A \times B$ is not empty either. P then sends the limit wedge $A \leftarrow A \times B \to B$ to $1 \leftarrow 1 \to 1$, and it is obvious that any other wedge $1 \leftarrow L \to 1$ factors uniquely through the latter. So P sends non-empty products to products.

If A is the empty set and B isn't, $A \times B$ is the empty set too. Then P sends the limit wedge $A \leftarrow A \times B \to B$ to $\varnothing \leftarrow \varnothing \to 1$. Since the only arrows in Set with the empty set as target have the empty set as source, the only wedges $\varnothing \leftarrow L \to 1$ have $L = \varnothing$, so trivially factor uniquely through $\varnothing \leftarrow \varnothing \to 1$. So P sends products of the empty set with non-empty sets to products.

Likewise, of course, for products of non-empty sets with the empty set, and the product of the empty set with itself. So, taking all the cases together, P sends products to products.

Now consider the equalizer in Set of two different maps $1 \overset{f}{\underset{g}{\rightrightarrows}} 2$, where 2 is a two-membered set. Since f and g never agree, their equalizer is the empty set (with the empty inclusion map). But since P sends both the maps f and g to the identity map on 1, the equalizer of Pf and Pg is the set 1 (with the identity map). Which means that the equalizer of $P(f)$ and $P(g)$ is *not* the result of applying P to the equalizer of f and g.

(3) Take the functor $Q\colon \mathsf{Set} \to \mathsf{Set}$ which sends any set X to the set $X \times 2$, and sends any arrow $f\colon X \to Y$ to $f \times 1_2\colon X \times 2 \to Y \times 2$ (the latter is of course the function which acts on a pair $\langle x, n\rangle \in X \times 2$ by sending it to $\langle fx, n\rangle$). Claim: *Q preserves equalizers but not binary products.*

Concerning products, if a functor F preserves binary products in Set, then by definition $F(X \times Y) \cong FX \times FY$. However, for X, Y finite, we have $Q(X \times Y) = (X \times Y) \times 2 \ncong (X \times 2) \times (Y \times 2) = QX \times QY$.

Now note that the equalizer of parallel arrows $X \overset{f}{\underset{g}{\rightrightarrows}} Y$ is essentially E, the subset of X on which f and g take the same value. And the equalizer of the parallel arrows $QX \overset{Qf}{\underset{Qg}{\rightrightarrows}} QY$ is the subset of $X \times 2$ on which $f \times 1_2$ and $g \times 1_2$ take the same value, which will be $E \times 2$, i.e. QE. So indeed Q preserves equalizers.

Moral, to repeat: a functor may preserve some but not all limits. Preservation isn't in general an all or nothing business.

(c) Now for an example of a functor that does preserve all limits:

(4) The forgetful functor $F\colon \mathsf{Mon} \to \mathsf{Set}$ sends a terminal object in Mon, a one-object monoid, to its underlying singleton set, which is terminal in Set. So F preserves limits of the empty shape.

The same functor sends a product $(M, \cdot) \times (N, *)$ in Mon to its underlying set of pairs of objects from M and N, which is a product in Set. So the forgetful F also preserves limits of the shape of the discrete two object category.

Likewise for equalizers. As we saw in §14.1, Ex. (2), the equalizer of two parallel monoid homomorphisms $(M, \cdot) \overset{f}{\underset{g}{\rightrightarrows}} (N, *)$ is $(E, \cdot)$ equipped with the inclusion map $E \to M$, where E is the set on which f and g agree. Which means that the forgetful functor takes the equalizer of f and g as monoid homomorphisms to their equalizer as set functions. So F preserves equalizers.

So the forgetful $F\colon \mathsf{Mon} \to \mathsf{Set}$ preserves terminal objects, binary products and equalizers – and hence, by appeal to the next theorem – this forgetful functor in fact preserves all finite limits.

At the last step we appeal to the fact that Mon is a finitely complete category, together with the following theorem:

Theorem 101. *If $\mathscr{C}$ is finitely complete, and a functor $F\colon \mathscr{C} \to \mathscr{D}$ preserves terminal objects, binary products and equalizers, then F preserves all finite limits.*

Proof. Suppose $\mathscr{C}$ is finitely complete. Then any limit cone $[C, c_J]$ over a diagram $D\colon \mathsf{J} \to \mathscr{C}$ is uniquely isomorphic to some limit cone $[C', c'_J]$ constructed from equalizers and finite products (see the proof of Theorem 65). Since F preserves terminal objects, binary products and equalizers, it sends the construction for $[C', c'_J]$ to a construction for a limit cone $[FC', Fc'_J]$ over $F \circ D\colon \mathsf{J} \to \mathscr{D}$. But F preserves isomorphisms, so $[FC, Fc_J]$ will be isomorphic to $[FC', Fc'_J]$ and hence is also a limit cone over $F \circ D\colon \mathsf{J} \to \mathscr{D}$. ☐

(d) Note that by contrast, however, the same forgetful $F\colon \mathsf{Mon} \to \mathsf{Set}$ does *not* preserve colimits with the 'shape' of the empty category, i.e. initial objects. For a one-object monoid is initial in Mon but its underlying singleton set is not initial in Set.

F does not preserve coproducts either – essentially because coproducts in Mon can be larger than coproducts in Set. Recall our discussion in §11.8 of coproducts in Grp : similarly, $F(M \oplus N)$, the underlying set of a coproduct of monoids M and N, is (isomorphic to) the set of finite sequences of alternating non-identity elements from M and N. Contrast $FM \oplus FN$, which is just the disjoint union of the underlying sets.

Our example generalizes, by the way. A forgetful functor from a category of structured sets to Set typically preserves finite limits but does not preserve all colimits.

(e) For the moment, we will finish on limit-preservation with a simple little result that we'll need to appeal to later:

Theorem 102. *If the functor $F\colon \mathscr{C} \to \mathscr{D}$ preserves pullbacks it preserves monomorphisms (i.e. sends monos to monos). Dually, if F preserves pushouts it preserves epimorphisms.*

Proof. We need only prove the first part. By Theorem 59, if $f\colon X \to Y$ in $\mathscr{C}$ is monic then it is part of the pullback square on the left:

$$\begin{tikzcd} X \arrow[r, "1_X"] \arrow[d, "1_X"'] & X \arrow[d, "f"] \\ X \arrow[r, "f"'] & Y \end{tikzcd} \quad \Rightarrow \quad \begin{tikzcd} FX \arrow[r, "1_{FX}"] \arrow[d, "1_{FX}"'] & FX \arrow[d, "Ff"] \\ FX \arrow[r, "Ff"'] & FY \end{tikzcd}$$

By assumption F sends a pullback squares to pullback squares, so the square on the right is also a pullback square. So by Theorem 59 again, Ff is monic too. ☐

22.3 Reflecting limits

(a) Here's a companion definition to set alongside the definition of preserving limits, together with a couple of general theorems:

Definition 100. A functor $F\colon \mathscr{C} \to \mathscr{D}$ *reflects limits of shape* J iff, given a cone $[C, c_J]$ over a diagram $D\colon \mathsf{J} \to \mathscr{C}$, then if $[FC, Fc_J]$ is a limit cone over $F \circ D\colon \mathsf{J} \to \mathscr{D}$, $[C, c_J]$ is already a limit cone over D.

Reflecting colimits is defined dually. △

Theorem 103. *Suppose $F\colon \mathscr{C} \to \mathscr{D}$ is fully faithful. Then F reflects limits.*

Proof. Suppose $[C, c_J]$ is a cone over a diagram $D\colon \mathsf{J} \to \mathscr{C}$, and $[FC, Fc_J]$ is a *limit* cone over $F \circ D\colon \mathsf{J} \to \mathscr{D}$. We need to show that $[C, c_J]$ must already be a limit cone too.

Now take any other cone $[B, b_J]$ over D. F sends this to a cone $[FB, Fb_J]$ which must uniquely factor through the limit cone $[FC, Fc_J]$ via some $u\colon FB \to FC$ which makes $Fb_J = Fc_J \circ u$ for each $J \in \mathsf{J}$. Since F is full and faithful, $u = Fv$ for some unique $v\colon B \to C$ such that $b_J = c_J \circ v$ for each J. So $[B, b_J]$ factors uniquely through $[C, c_J]$. Which shows that $[C, c_J]$ is a limit cone. ☐

Theorem 104. *Suppose $F\colon \mathscr{C} \to \mathscr{D}$ preserves limits. Then if $\mathscr{C}$ is complete and F reflects isomorphisms, then F reflects small limits.*

Proof. Since $\mathscr{C}$ is complete there exists a limit cone $[B, b_J]$ over any diagram $D\colon \mathsf{J} \to \mathscr{C}$ (where J is small), and so – since F preserves limits – $[FB, Fb]$ is a limit cone over $F \circ D\colon \mathsf{J} \to \mathscr{D}$.

Now suppose that there is a cone $[C, c_J]$ over D such that $[FC, Fc_J]$ is another limit cone over $F \circ D$. Now $[C, c_J]$ must uniquely factor through $[B, b_J]$ via a map $f\colon C \to B$. Which means that $[FC, Fc_J]$ factors through $[FB, Fb]$ via Ff. However, since these are by hypothesis both limit cones over $F \circ D$, Ff must be an isomorphism. Hence, since F reflects isomorphisms, f must be an isomorphism. So $[C, c_J]$ must be a limit cone by Theorem 55. □

(b) Since the forgetful functor $F\colon \mathsf{Mon} \to \mathsf{Set}$ preserves limits and reflects isomorphisms the last theorem shows that

(1) The forgetful functor $F\colon \mathsf{Mon} \to \mathsf{Set}$ reflects all limits. Similarly for some other forgetful functors from familiar categories of structured sets to Set.

However, be careful! For we also have ...

(2) The forgetful functor $F\colon \mathsf{Top} \to \mathsf{Set}$ which sends topological space to its underlying set *preserves* all limits but does not *reflect* all limits.

Here's a case involving binary products. Suppose X and Y are a couple of spaces with a coarse topology, and let Z be the space $FX \times FY$ equipped with a finer topology. Then, with the obvious arrows, $X \leftarrow Z \to Y$ is a wedge to X, Y but not the limit wedge in Top: but $FX \leftarrow FX \times FY \to FY$ is a limit wedge in Set.

Given the previous theorem, we can conclude that $F\colon \mathsf{Top} \to \mathsf{Set}$ doesn't reflect isomorphisms. (Which is also something we can show directly. Consider the continuous bijection from the half-open interval $[0, 1)$ to S^1. Think of this bijection as a topological map f; then f is not a homeomorphism in Top. However, treating the bijection as a set-function, i.e. as $F'f$, it *is* an isomorphism in Set.)

22.4 Creating limits

Alongside the natural notions of preserving and reflecting limits, we meet a related third notion which we should pause to explain:

Definition 101. A functor $F\colon \mathscr{C} \to \mathscr{D}$ *creates limits of shape* J iff, for any diagram $D\colon \mathsf{J} \to \mathscr{C}$, if $[M, m_J]$ is a limit cone over $F \circ D$, there is a unique cone $[C, c_J]$ over D such that $[FC, Fc_J] = [M, m_J]$, and moreover $[C, c_J]$ is a limit cone.

Creating colimits is defined dually. △

(Variant: some define creation of limits by only requiring that $[FC, Fc_J]$ is isomorphic to $[M, m_J]$ in the obvious sense.)

Why 'creation'? The picture is that every limit over $F \circ D$ in $\mathscr{D}$ is generated by F from a unique limit over D in $\mathscr{C}$. And while reflection is a condition on those limit cones over $F \circ D$ which take the form $[FC, Fc_J]$ for some cone $[C, c_J]$, creation is a similar condition on *any* limit cone over $F \circ D$. So as you would predict,

Theorem 105. *If the functor $F\colon \mathscr{C} \to \mathscr{D}$ creates limits of shape J, it reflects them.*

Proof. Suppose $[FC, Fc_J]$ is a limit cone over $F \circ D$: generated by the cone $[C, c_J]$ over D. Then, assuming F creates limits, $[C, c_J]$ has to be the unique cone over D such that $[FC, Fc_J]$ is generated by it, and has to be a limit cone. □

Theorem 106. *Suppose $F\colon \mathscr{C} \to \mathscr{D}$ is a functor, that $\mathscr{D}$ has limits of shape J and F creates such limits. Then $\mathscr{C}$ has limits of shape J and F preserves them.*

Proof. Take any diagram $D\colon \mathsf{J} \to \mathscr{C}$. Then there is a limit $[M, m_J]$ over $F \circ D$ (since $\mathscr{D}$ has all limits of shape J). Hence (since F creates limits), there is a limit cone $[C, c_J]$ over D where this is such that $[FC, Fc_J]$ is $[M, m_J]$ and hence is a limit cone too. □

23 Hom-functors

This chapter introduces the notion of a hom-functor, a type of functor which will turn out to play a rather special role in category theory. We show that, unlike the general run of functors, hom-functors do behave very nicely with (small) limits, always preserving them.

23.1 Hom-sets

(a) Suppose the category $\mathscr{C}$ is locally small. Then there is only a set's worth of arrows between any two $\mathscr{C}$-objects. Moreover, in many familiar locally small categories, these $\mathscr{C}$-arrows will be an appropriate kind of homomorphism. So this explains the terminology in the following conventional definition:

Definition 102. Given a locally small category $\mathscr{C}$, and $\mathscr{C}$-objects A and B, then the *hom-set* $\mathscr{C}(A, B)$ is the set of $\mathscr{C}$-arrows from A to B. △

The brusque but conventional notation we are using for collections of arrows between two objects has already made a fleeting appearance in §21.6: alternative and perhaps more reader-friendly notations are '$\mathrm{Hom}_{\mathscr{C}}(A, B)$' or just '$\mathrm{Hom}(A, B)$' when the relevant category is obvious.

(b) But although our definition is absolutely standard, it is not unproblematic. What kind of set is a hom-set? In categorial terms, in which category does a hom-set $\mathscr{C}(A, B)$ live? (We here return to a question already flagged-up in §21.4.)

The usual assumption, very often made with no comment at all, is that a hom-set lives in the category Set. "Where else?", you might reasonably ask. But what category is Set? Remember, we didn't fix this at the outset. We cheerfully said, just take your favourite universe of sets and functions between them, and the category Set can for now comprise *them*. But suppose – naturally enough – that you think of Set as containing just the sets you know and love from your basic set theory course in the delights of ZFC. In this case, Set is a category of *pure* sets, i.e. of sets whose members, if any, are sets whose members, if any, are sets ... all the way down. But if we think of $\mathscr{C}(A, B)$ as living in such a category of pure sets, then the arrows which are members of $\mathscr{C}(A, B)$ will themselves have to be pure sets too. Yet do we really want to suppose that categorial arrows are inevitably just more sets?

It seems that we have at least three options here. In headline terms, we can for a start ...

(i) Bite the bullet. Take Set to be a category of pure sets, and take $\mathscr{C}(A, B)$ to be a pure set living in Set. Then $\mathscr{C}$-arrows themselves have to be pure sets.

(ii) Backtrack. Take Set after all to be a category of possibly impure sets, where the non-set elements can, inter alia, be arrows in any category. So again we can endorse the standard view that $\mathscr{C}(A, B)$ lives in Set, but now without pre-supposing that all $\mathscr{C}$-arrows are sets.

(iii) Re-interpret. As in (i), take Set to be a category of pure sets. As in (ii), regard $\mathscr{C}(A, B)$ as, in general, an impure collection whose members are arrows (which needn't be themselves sets). But then we'll have to re-interpret the standard line that $\mathscr{C}(A, B)$ lives in Set. We will say it isn't strictly speaking the hom-set as originally defined which lives in Set but rather a pure set which represents or models or indexes it (that there can be such an indexing set is what we mean when we say that there is only a set's-worth of arrows in $\mathscr{C}(A, B)$).

We could even call this representing pure set $\mathscr{C}(A, B)$ too, with context deciding when we are talking about the 'true' impure hom-collection and when we are talking about its pure-set representation.

It is, to say the least, not entirely clear at the outset which of these options is the best way forward (or maybe we should be looking for a fourth way!).

Option (i) has weighty support. In his canonical 1997, Saunders Mac Lane initially gives a definition like our Defn. 13 as a definition of what *he* calls metacategories, and then for him a category proper "will mean any interpretation of the category axioms within set theory". So for Mac Lane, at least at the outset, all the gadgets of categories proper will unproblematically live in the universe of set theory, and that applies to hom-sets in particular. Presumably this is the standard universe of pure sets. Mac Lane doesn't, I think, make that explicit: but e.g. Horst Schubert does in §3.1 of his terse but very lucid (1972), writing "One has to be aware that the set theory used here has no 'primitive (ur-)elements'; elements of sets ... are always themselves sets." But, as we asked before, do we really want or need to suppose that categories are always and everywhere sets? Not if (as some do) we want to conceive of category theory as a more democratic way of organizing the mathematical universe, which provides an alternative to imperialistic set-theoretic reductionism. (Indeed, much later in his book, in the Appendix, Mac Lane suggests that we can perhaps after all use our Defn. 13, more or less, to describe categories directly, without going via set theory).

Option (ii), by contrast, avoids reducing everything to pure sets. But on the face of it, it is now quite unclear what *does* live in the universe of Set, if it is just a free-for-all at the level of urelements, and it is sheer mess at the bottom level of the hierarchy of sets. But maybe there is an option (ii′) where we re-think our story about the nature of sets in a way which still in some sense allows urelements but abstracts away from their nature. More about this in due course.

Option (iii) might seem to let us have our cake and eat it – we keep Set as a tidy category of pure sets without urelements, we keep collections of arrows as impure sets, and we model one by the other in a familiar enough way. But it adds a layer of complication which might not be welcome.

We won't try to judge which is the best option at this point. And after all, such verdicts are often best given rather late in the game, when we can look back to see what really are the essential requirements of the load-bearing parts of the theory we have been developing. So what to do? For the moment, we will take the path of least resistance and proceed conventionally, *as if* hom-sets do live in Set; and we'll have to return later to think more about how we really want to construe this.

23.2 Hom-functors

(a) Now to introduce the main notion of this chapter.

Assume $\mathscr{C}$ is locally small. So we can talk about $\mathscr{C}(A, B)$, the hom-set of $\mathscr{C}$-arrows from A to B. Keep A fixed. Then as we vary X through the objects in $\mathscr{C}$, we get varying $\mathscr{C}(A, X)$.

So: consider the resulting function which sends an object X in $\mathscr{C}$ to the set $\mathscr{C}(A, X)$, a set which we are following standard practice in taking as living in Set.

Can we now treat this function on $\mathscr{C}$-objects as the first component of a functor, call it $\mathscr{C}(A, –)$, from $\mathscr{C}$ to Set? Well, how could we find a component of the functor to deal with the $\mathscr{C}$-arrows? Such a component is going to need to send an arrow $f\colon X \to Y$ in $\mathscr{C}$ to a Set-function from $\mathscr{C}(A, X)$ to $\mathscr{C}(A, Y)$. The obvious candidate for the latter function is the one we can notate as $f \circ –$ that maps any $g\colon A \to X$ in $\mathscr{C}(A, X)$ to $f \circ g\colon A \to Y$ in $\mathscr{C}(A, Y)$. (Note, $f \circ g\colon A \to Y$ has to be in $\mathscr{C}(A, Y)$ because $\mathscr{C}$ is a category which by hypothesis contains $g\colon A \to X$ and $f\colon X \to Y$ and hence must contain their composition.)

It is easy to check that these components add up to a genuine covariant functor – in fact the functoriality in this case just reduces to the associativity of composition for arrows in a category and the basic laws for identity arrows.

Now, start again from the hom-set $\mathscr{C}(A, B)$ but this time keep B fixed: then as we vary X through the objects in $\mathscr{C}$, we again get varying hom-sets $\mathscr{C}(X, B)$. Which generates a function which sends an object X in $\mathscr{C}$ to an object $\mathscr{C}(X, B)$ in Set. To turn *this* into a functor $\mathscr{C}(–, B)$, we need again to add a component to deal with $\mathscr{C}$-arrows. That will need to send $f\colon X \to Y$ in $\mathscr{C}$ to some function between $\mathscr{C}(X, B)$ to $\mathscr{C}(Y, B)$. But this time, to get functions to compose properly, things will have to go the other way about, i.e. the associated functor will have to send a function $g\colon Y \to B$ in $\mathscr{C}(Y, B)$ to $g \circ f\colon X \to B$ in $\mathscr{C}(X, B)$. So this means that the resulting functor $\mathscr{C}(–, B)$ is a *contravariant* hom-functor.

(b) So, to summarize, we will say:

Definition 103. Given a locally small category $\mathscr{C}$, then the associated *covariant hom-functor* $\mathscr{C}(A, –): \mathscr{C} \to \mathsf{Set}$ is the functor with the following data:

(1) A mapping $\mathscr{C}(A, –)_{ob}$ whose value at the object X in $\mathscr{C}$ is the hom-set $\mathscr{C}(A, X)$.

(2) A mapping $\mathscr{C}(A, –)_{arw}$, whose value at the $\mathscr{C}$-arrow $f: X \to Y$ is the set function $f \circ –$ from $\mathscr{C}(A, X)$ to $\mathscr{C}(A, Y)$ which sends an element $g: A \to X$ to $f \circ g: A \to Y$.

And the associated *contravariant hom-functor* $\mathscr{C}(–, B): \mathscr{C} \to \mathsf{Set}$ is the functor with the following data:

(3) A mapping $\mathscr{C}(–, B)_{ob}$ whose value at the object X in $\mathscr{C}$ is the hom-set $\mathscr{C}(X, B)$.

(4) A mapping $\mathscr{C}(–, B)_{arw}$, whose value at the $\mathscr{C}$-arrow $f: Y \to X$ is the set function $– \circ f$ from $\mathscr{C}(X, B)$ to $\mathscr{C}(Y, B)$ which sends an element $g: X \to B$ to the map $g \circ f: Y \to B$.

The use of a blank in the notation '$\mathscr{C}(A, –)$' invites an obvious shorthand: instead of writing '$\mathscr{C}(A, –)_{arw}(f)$' to indicate the result of the component of the functor which acts on arrows applied to the function f, we will write simply '$\mathscr{C}(A, f)$'. Similarly for the dual. △

Alternative notations for hom-functors, to along with the alternative notations for hom-sets, are '$\mathrm{Hom}_{\mathscr{C}}(A, –)$' and '$\mathrm{Hom}_{\mathscr{C}}(–, B)$'.

(c) For the record, we can also define a related 'bi-functor' $\mathscr{C}(–, –): \mathscr{C}^{op} \times \mathscr{C} \to \mathsf{Set}$, which we can think of as contravariant in the first place and covariant in the second. This acts on the product category mapping the pair object (A, B) to the hom-set $\mathscr{C}(A, B)$, and the pair of morphisms $(f: X' \to X, g: Y \to Y')$ to the morphism between $\mathscr{C}(X, Y)$ and $\mathscr{C}(X', Y')$ that sends $h: X \to Y$ to $g \circ h \circ f: X' \to Y'$. We will return to this if/when we need to say more.

23.3 Hom-functors preserve limits

(a) As noted at the outset, hom-functors will play a special role in category theory, and we will meet them repeatedly. But in the rest of this chapter, we just consider how they interact with limits.

We start with a preliminary observation. If some functor F preserves products, it has to be the case that $F(C \times D) \cong FC \times FD$. So if a hom-functor $\mathscr{C}(A, –)$ is to preserve products, we need this to be true:

Theorem 107. *Assuming the product exists,* $\mathscr{C}(A, C \times D) \cong \mathscr{C}(A, C) \times \mathscr{C}(A, D)$.

However, this is easy to show:

Proof. An arrow $f\colon A \to C \times D$ factors into two arrows $c\colon A \to C$ and $d\colon A \to D$ via the projection arrows of the product $C \times D$. And two such arrows c, d form a wedge which factors uniquely through the product via f. This gives us a bijection between arrows f in $\mathscr{C}(A, C \times D)$ and pairs of arrows (c, d) in $\mathscr{C}(A, C) \times \mathscr{C}(A, D)$, an isomorphism in Set. $\square$

This observation can now be turned into a proof that hom-functors preserve any binary product which exists. They also preserve any terminal objects and equalizers. And then using the fact that if there is a limit cone over $D\colon \mathsf{J} \to \mathscr{C}$ (with J a small category), then it can be constructed from suitable products and equalizers (as indicated by the proof of Theorem 67), we can derive

Theorem 108. *Suppose that $\mathscr{C}$ is a small category. Then the covariant hom-functor $\mathscr{C}(A, -)\colon \mathscr{C} \to \mathsf{Set}$, for any A in the category $\mathscr{C}$, preserves all small limits that exist in $\mathscr{C}$.*

However, rather than officially prove this important theorem in the way just sketched, let's instead go for a brute-force just-apply-the-definitions-and-see-what-happens demonstration (for it is quite a useful reality check to run through the details):

Proof. We'll first check that $\mathscr{C}(A, -)\colon \mathscr{C} \to \mathsf{Set}$ sends a cone over the diagram $D\colon \mathsf{J} \to \mathscr{C}$ to a cone over $\mathscr{C}(A, -) \circ D\colon \mathsf{J} \to \mathsf{Set}$.

A cone has a vertex C, and arrows $C_J\colon C \to DJ$ for each $J \in \mathsf{J}$, where for any $f\colon J \to K$ in J, so for any $Df\colon DJ \to DK$, $c_K = Df \circ c_J$.

Now, acting on objects, $\mathscr{C}(A, -)$ sends C to $\mathscr{C}(A, C)$ and sends DJ to $\mathscr{C}(A, DJ)$. And acting on arrows, $\mathscr{C}(A, -)$ sends $c_J\colon C \to DJ$ to the set function $c_J \circ -$ which takes $g\colon A \to C$ and outputs $c_J \circ g\colon A \to DJ$; and it sends $Df\colon DJ \to DK$ to the set-function $Df \circ -$ which takes $h\colon A \to DJ$ and outputs $Df \circ h\colon A \to DK$.

Diagrammatically, then, the functor sends the triangle on the left to the one on the right:

$$\begin{tikzcd} & C \arrow[dl, "c_J"'] \arrow[dr, "c_K"] & \\ DJ \arrow[rr, "Df"'] & & DK \end{tikzcd} \quad \Rightarrow \quad \begin{tikzcd} & \mathscr{C}(A, C) \arrow[dl, "c_J \circ -"'] \arrow[dr, "c_K \circ -"] & \\ \mathscr{C}(A, DJ) \arrow[rr, "Df \circ -"'] & & \mathscr{C}(A, DK) \end{tikzcd}$$

And assuming $c_K = Df \circ c_J$, we have $c_K \circ - = (Df \circ c_J) \circ - = (Df \circ -) \circ (c_J \circ -)$: hence, if the triangle on the left commutes, so does the triangle on the right. Likewise for other such triangles. Which means that if $[C, c_J]$ is a cone over D, then $[\mathscr{C}(A, C), c_J \circ -]$ is indeed a cone over $\mathscr{C}(A, -) \circ D$.

So far, so good! It remains, then, to show that in particular $\mathscr{C}(A, -)$ sends limit cones to limit cones. So suppose that $[L, \lambda_J]$ is a limit cone in $\mathscr{C}$ over D. The functor $\mathscr{C}(A, -) : \mathscr{C} \to \mathsf{Set}$ sends the left-hand commuting diagram below to the commuting triangle at the bottom of the right-hand diagram. And we now suppose that $[M, m_j]$ is any other cone over the image of D:

$$\begin{tikzcd} & L \arrow[dl, "\lambda_J"'] \arrow[dr, "\lambda_K"] & \\ DJ \arrow[rr, "Df"'] & & DK \end{tikzcd} \qquad \begin{tikzcd} & M \arrow[ddl, "m_J"'] \arrow[ddr, "m_K"] & \\ & \mathscr{C}(A,L) \arrow[dl, "\lambda_J \circ -"] \arrow[dr, "\lambda_K \circ -"'] & \\ \mathscr{C}(A,DJ) \arrow[rr, "Df \circ -"'] & & \mathscr{C}(A,DK) \end{tikzcd}$$

Hence $m_K = (Df \circ -) \circ m_J$. Now remember that M lives in Set: so take a member x. Then $m_J(x)$ is a particular arrow in $\mathscr{C}(A, DJ)$, in other words $m_J(x)\colon A \to DJ$. Likewise we have $m_K(x)\colon A \to DK$. But $m_K(x) = Df \circ m_J(x)$. Which means that for all f the outer triangles on the left below commute and so $[A, m_J(x)]$ is a cone over D. And this must factor uniquely through an arrow $u(x)$ as follows:

$$\begin{tikzcd} & A \arrow[d, "u(x)"] \arrow[ddl, "m_J(x)"'] \arrow[ddr, "m_K(x)"] & \\ & L \arrow[dl, "\lambda_J"] \arrow[dr, "\lambda_K"'] & \\ DJ \arrow[rr, "Df"'] & & DK \end{tikzcd} \qquad \begin{tikzcd} & M \arrow[d, "u"] \arrow[ddl, "m_J"'] \arrow[ddr, "m_K"] & \\ & \mathscr{C}(A,L) \arrow[dl, "\lambda_J \circ -"] \arrow[dr, "\lambda_K \circ -"'] & \\ \mathscr{C}(A,DJ) \arrow[rr, "Df \circ -"'] & & \mathscr{C}(A,DK) \end{tikzcd}$$

Hence $u(x)$ is an arrow from A to L, i.e. an element of $\mathscr{C}(A, L)$. So consider the map $u\colon M \to \mathscr{C}(A, L)$ which sends x to $u(x)$. Since $m_J(x) = \lambda_J \circ u(x)$ for each x, $m_J = (\lambda_J \circ -) \circ u$. And since this applies for each J, So $[M, m_j]$ factors through the image of the cone $[L, \lambda_J]$ via u.

Suppose there is another map $v\colon M \to \mathscr{C}(A, L)$ such that we also have each $m_J = (\lambda_J \circ -) \circ v$. Then again take an element $x \in M$: then $m_J(x) = \lambda_J \circ v(x)$. So again, $[A, m_J(x)]$ factorizes through $[L, \lambda_J]$ via $v(x)$ – which, by the uniqueness of factorization through limits, means that $v(x) = u(x)$. Since that obtains for all $x \in M$, $v = u$. Hence $[M, m_j]$ factors uniquely through the image of $[L, \lambda_J]$. Since $[M, m_j]$ was an arbitrary cone, we have therefore proved that the image of the limit cone $[L, \lambda_J]$ is also a limit cone. □

(b) What is the dual of Theorem 108? We have two dualities to play with: limits vs colimits and covariant functors vs contravariant functors.

Two initial observations. First, a covariant hom-functor need not preserve colimits. For example, take the hom-functor $\mathsf{Grp}(A, -)$. In Grp the initial object 0 is also the terminal object, so for any group A, $\mathsf{Grp}(A, 0)$ is a singleton, which is not initial in Set. Second, contravariant hom-functors can't preserve either limits or colimits, because contravariant functors reverse arrows.

So the dual result we want is this:

Theorem 109. *Suppose that $\mathscr{C}$ is a small category. Then the contravariant hom-functor $\mathscr{C}(-, A)\colon \mathscr{C} \to \mathsf{Set}$, for any A in the category $\mathscr{C}$, sends a colimit of shape J (for small category J) to a limit of that shape.*

Yes, that's right: contravariant functors send colimits to limits (the two reversals of arrows involved in going from covariant to contravariant, and from limit to colimit, cancelling out). We can leave the proof as an exercise in dualizing.

24 Functors and comma categories

We have now introduced the notion of a functor as a map between categories, and seen how functors can e.g. preserve/reflect (or fail to preserve/reflect) various properties of arrows and various limit constructions. And we are about to move on to introduce the next Big Idea, i.e. the notion of maps between functors.

However, before we do that, this chapter pauses to use the notion of a functor to define the idea of a comma category. I'm afraid that this might initially seem to involve a rather contorted construction. But bear with me! We will in fact be repeatedly meeting instances of comma categories, so we ought to get to grips with this idea sooner or later.

24.1 Functors and slice categories

By way of a warm-up exercise, recall the notion of a slice category $\mathscr{C}/I$ (Defn. 25). If $\mathscr{C}$ is a category, and I is a $\mathscr{C}$-object, then $\mathscr{C}/I$'s objects, economically defined, are the arrows$_{\mathscr{C}}$ $f\colon A \to I$ (for any $\mathscr{C}$-object A), while $\mathscr{C}/I$'s arrows between these objects $f\colon A \to I$ and $g\colon B \to I$ are the arrows$_{\mathscr{C}}$ $j\colon A \to B$ such that $g \circ j = f$.

Here, then, are a couple of simple examples of functors operating on slice categories:

(1) There is functor, another kind of forgetful functor, $F\colon \mathscr{C}/I \to \mathscr{C}$, which sends a $\mathscr{C}/I$-object $f\colon A \to I$ back to A, and sends an arrow j in $\mathscr{C}/I$ back to the original arrow j in $\mathscr{C}$.

For example, recall the slice category FinSet/I_n which we met at the end of §6.4, which is the category of finite sets whose members are coloured from a palette of n colours. The forgetful functor $F\colon \mathsf{FinSet}/I_n \to \mathsf{FinSet}$ forgets about the colourings of a set S provided by functions $f\colon S \to I_n$.

(2) Next, let's show how we can use an arrow $k\colon I \to J$ (for $I, J \in \mathscr{C}$) to generate a corresponding functor $K\colon \mathscr{C}/I \to \mathscr{C}/J$.

The functor needs to act on *objects* in $\mathscr{C}/I$ and send them to objects in $\mathscr{C}/J$. That is to say, K_{ob} needs to send an arrow$_{\mathscr{C}}$ $f\colon X \to I$ to an arrow$_{\mathscr{C}}$ with codomain J. The obvious thing to do is to put $K_{ob}(f) = k \circ f$.

And how will a matching K_{arw} act on *arrows* of $\mathscr{C}/I$? Consider:

$$\begin{tikzcd}
A \arrow[dd, "j"'] \arrow[dr, "f"] \arrow[drr, "k\circ f"] & & \\
& I \arrow[r, "k"] & J \\
B \arrow[ur, "g"] \arrow[urr, "k\circ g"'] & &
\end{tikzcd}$$

Here, the $\mathscr{C}/I$-arrows from $f\colon A \to I$ to $g\colon B \to I$, by definition, include any j which makes the left-hand inner triangle commute. But then such a j will also make the outer triangle commute, i.e. j is an arrow from $k \circ f\colon A \to J$ to $k \circ g\colon B \to J$ (which is therefore an arrow from $K(f)$ to $K(g)$).

So we can simply put $K(j)$ (for $j\colon f \to g$ in $\mathscr{C}/I$) to be j (i.e. $j\colon K(f) \to K(g)$ in $\mathscr{C}/J$).

Claim: K is then a functor from $\mathscr{C}/I$ to $\mathscr{C}/J$.

It is a useful small reality check to confirm that (2) all makes sense, and that K is indeed a functor.

24.2 Comma categories

We have already met various ways of getting new categories from old, including the one we've just reminded ourselves about, namely constructing slice categories. Given that we now have the notion of a functor to hand, in this section we can introduce another way of defining new from old, this time deriving a category from three(!) categories and a pair of functors relating them.

Suppose, then, that we have a pair of functors sharing a target, say $S\colon \mathscr{A} \to \mathscr{C}$ and $T\colon \mathscr{B} \to \mathscr{C}$. Then we have a way of indirectly connecting an object A in $\mathscr{A}$ to an object B in $\mathscr{B}$, i.e. by looking at their respective images SA and TB and considering arrows $f\colon SA \to TB$ between them.

We are going to define a category of such connections. But if its objects are to comprise an $\mathscr{A}$-object A, a $\mathscr{B}$-object B, together with a $\mathscr{C}$-arrow $f\colon SA \to TB$, what could be the arrows in our new category? Suppose we have, then, two triples (A, f, B), (A', f', B'); an arrow between them will presumably involve arrows $a\colon A \to A'$ and $b\colon B \to B'$. But note that these two are sent respectively to arrows $Sa\colon SA \to SA'$ and $Tb\colon TB \to TB'$ in $\mathscr{C}$, and we will need these arrows to interact appropriately with the other $\mathscr{C}$-arrows f and f'.

All that prompts the following – seemingly rather esoteric – definition:

Definition 104. Given functors $S\colon \mathscr{A} \to \mathscr{C}$ and $T\colon \mathscr{B} \to \mathscr{C}$, then the '*comma category*' $(S \downarrow T)$ is the category with the following data:

(1) The objects of $(S \downarrow T)$ are triples (A, f, B) where A is an $\mathscr{A}$-object, B is a $\mathscr{B}$-object, and $f\colon SA \to TB$ is an arrow in $\mathscr{C}$.

(2) An arrow of $(S \downarrow T)$ from (A, f, B) to (A', f', B') is a pair (a, b), where $a\colon A \to A'$ is an $\mathscr{A}$-arrow, $b\colon B \to B'$ is an $\mathscr{B}$-arrow, and the following diagram commutes:

$$\begin{tikzcd} SA \arrow[r, "f"] \arrow[d, "Sa"] & TB \arrow[d, "Tb"] \\ SA' \arrow[r, "f'"] & TB' \end{tikzcd}$$

(3) The identity arrow on the object (A, f, B) is the pair $(1_A, 1_B)$.

(4) Composition in $(S \downarrow T)$ is induced by the composition laws of $\mathscr{A}$ and $\mathscr{B}$, thus: $(a', b') \circ (a, b) = (a' \circ_{\mathscr{A}} a, b' \circ_{\mathscr{B}} b)$. △

It is readily seen that, so defined, $(S \downarrow T)$ is indeed a category.

The standard label 'comma category' arises from an unhappy earlier notation '(S, T)': the notation has long been abandoned but the name has stuck. But why we should be bothering with such a construction? Well, the notion of a comma category in fact nicely generalizes a number of simpler constructions. And indeed, we have already met two comma categories in thin disguise. The next section reveals which they are.

24.3 Two (already familiar) types of comma category

(a) First take the minimal case where $\mathscr{A} = \mathscr{B} = \mathscr{C}$, and where both S and T are the identity functor on that category, $1_{\mathscr{C}}$.

Then the objects in this category $(1_{\mathscr{C}} \downarrow 1_{\mathscr{C}})$ are triples $(X, X \xrightarrow{f} Y, Y)$ for X, Y both $\mathscr{C}$-objects. And an arrow from $(X, X \xrightarrow{f} Y, Y)$ to $(X', X' \xrightarrow{f'} Y', Y')$ is a pair of $\mathscr{C}$-arrows $a\colon X \to X'$, $b\colon Y \to Y'$ such that the following diagram commutes:

$$\begin{tikzcd} X \arrow[r, "f"] \arrow[d, "a"] & Y \arrow[d, "b"] \\ X' \arrow[r, "f'"] & Y' \end{tikzcd}$$

So the only difference between $(1_{\mathscr{C}} \downarrow 1_{\mathscr{C}})$ and the arrow category $\mathscr{C}^{\rightarrow}$ is that we have now 'decorated' the objects of $\mathscr{C}^{\rightarrow}$, i.e. $\mathscr{C}$-arrows $f\colon X \to Y$, with explicit assignments of their sources and targets as $\mathscr{C}$-arrows, to give triples $(X, X \xrightarrow{f} Y, Y)$. Hence $(1_{\mathscr{C}} \downarrow 1_{\mathscr{C}})$ and $\mathscr{C}^{\rightarrow}$, although not strictly identical, come to the just same.

And of course, we can do better than limply say the two categories 'come to just the same'. Working in a big enough category CAT, consider the functor $F\colon \mathscr{C}^{\rightarrow} \to (1_{\mathscr{C}} \downarrow 1_{\mathscr{C}})$ which sends a $\mathscr{C}^{\rightarrow}$-object to the corresponding triple, and sends $\mathscr{C}^{\rightarrow}$-arrows (pairs of $\mathscr{C}$-arrows) to themselves. Then, F trivially has an inverse, and so the categories are isomorphic.

(b) Let's take secondly the special case where $\mathscr{A} = \mathscr{C}$ with S the identity functor $1_{\mathscr{C}}$, and where $\mathscr{B} = \mathbf{1}$ (the category with a single object $\star$ and the single arrow $1_\star$). And take the functor $I\colon \mathbf{1} \to \mathscr{C}$ which sends $\star$ to some individual $\mathscr{C}$-object which we'll also call I – see §20.2, Ex. (F11).

Applying the definition, the objects of the category $(1_{\mathscr{C}} \downarrow I)$ are therefore triples $(A, A \xrightarrow{f} I, \star)$, and an arrow between $(A, A \xrightarrow{f} I, \star)$ and $(B, B \xrightarrow{g} I, \star)$ will be a pair $(j, 1_\star)$, with $j\colon A \to B$ an arrow such the diagram on the left commutes:

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[d, "j"'] & I \arrow[d, "1_I"] \\ B \arrow[r, "g"] & I \end{tikzcd} \qquad \begin{tikzcd} A \arrow[dr, "f"] \arrow[dd, "j"'] & \\ & I \\ B \arrow[ur, "g"'] & \end{tikzcd}$$

The diagram on the left is trivially equivalent to that on the right – which should look very familiar! We've ended up with something tantamount to the slice category $\mathscr{C}/I$, the only differences being that (i) instead of the slice category's objects, i.e. pairs (A, f), we now have 'decorated' objects $(A, f, \star)$ which correspond one-to-one with them, and (ii) instead of the slice category's arrows $j\colon A \to B$ we have decorated arrows $(j, 1_\star)$ which correspond one-to-one with *them*.

Again the categories $(1_{\mathscr{C}} \downarrow I)$ and $\mathscr{C}/I$ are evidently isomorphic categories.

24.4 Another (new) type of comma category

(a) While we are looking at examples of comma categories, let's add for the record a third illustrative case (pretty similar to the case of slice categories). It will turn out to be useful, and we choose notation with an eye to a later application.

Suppose we have a functor $G\colon \mathscr{C} \to \mathscr{A}$ and an object $A \in \mathscr{A}$. There is a corresponding functor $A\colon \mathbf{1} \to \mathscr{A}$ (which sends the sole object $\star$ in the one-object category $\mathbf{1}$ to the object A in $\mathscr{A}$). Then what is the comma category $(A \downarrow G)$? Flat-footedly applying the definitions, we get:

(1) The objects of $(A \downarrow G)$ are triples $(\star, f, C)$ where C is a $\mathscr{C}$-object, and $f : A \to GC$ is an arrow in $\mathscr{A}$.

(2) An arrow of $(A \downarrow G)$ from $(\star, f, C)$ to $(\star, f', C')$ is a pair of arrows, $(1_\star, j)$ with $j\colon C \to C'$ such the following square commutes:

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[d, "1_A"'] & GC \arrow[d, "Gj"] \\ A \arrow[r, "f'"] & GC' \end{tikzcd}$$

However, since the $\star$-component in all the objects of $(A \downarrow G)$ is doing no real work, our comma category is tantamount to the stripped-down category such that

(1′) the objects are, more simply, pairs (C, f) where C is a $\mathscr{C}$-object and $f : A \to GC$ is an arrow in $\mathscr{A}$,

(2′) an arrow from (C, f) to (C', f') is, more simply, a $\mathscr{C}$-arrow $j \colon C \to C'$ making this commute:

$$\begin{tikzcd} & GC \arrow[dd, "Gj"] \\ A \arrow[ur, "f"] \arrow[dr, "f'"'] & \\ & GC' \end{tikzcd}$$

We add, of course, the obvious definitions for the identity arrows and for composition of arrows. And it is this stripped-down version which is in fact usually referred to by the label '$(A \downarrow G)$' (we can, incidentally, read 'A' in the label here as just referring to an object, not to the corresponding functor).

(b) Similarly, there is a category $(G \downarrow A)$. In its stripped down version,

(1″) its objects are pairs (C, f) where C is a $\mathscr{C}$-object and $f : GC \to A$ is an arrow in $\mathscr{A}$,

(2″) an arrow from (C, f) to (C', f') is a $\mathscr{C}$-arrow $j \colon C \to C'$ making this commute:

$$\begin{tikzcd} GC \arrow[dd, "Gj"'] \arrow[dr, "f"] & \\ & A \\ GC' \arrow[ur, "f'"'] & \end{tikzcd}$$

24.5 An application: free monoids again

We make a connection between the idea of a *free monoid* (which we met in §20.5) and the idea of a certain *comma category* (of the kind we met in the last section).

Take the two categories Mon and Set; let S be a set living in Set, and let $F \colon \mathsf{Mon} \to \mathsf{Set}$ be the forgetful functor. And now consider the comma category $(S \downarrow F)$. Unthinkingly applying the definition,

(1) the objects of this category $(S \downarrow F)$ are pairs $(\mathcal{N}, f)$ where $\mathcal{N}$ is a monoid $(N, \cdot, 1_N)$ and f is a set-function from S to $F(\mathcal{N})$, i.e. $f \colon S \to N$;

(2) an $(S \downarrow F)$-arrow from $(\mathcal{N}, f)$ to $(\mathcal{N}', f')$ is a monoid homomorphism $\underline{j} \colon \mathcal{N} \to \mathcal{N}'$, which treated as a set-function is $j = F\underline{j} \colon N \to N'$, such that $f' = j \circ f$.

But what does this mean, intuitively? We can think of a function $f\colon S \to N$ as *labelling* elements of N by members of S: N-elements can thereby receive zero, one, or many labels. So we can think of a pair $(\mathcal{N}, f)$ as a monoid with some S-labelled elements. And an arrow between these monoids-with-S-labelled-elements is a monoid homomorphism which sends labelled elements to elements with the same label(s).

Now suppose $(S \downarrow F)$ has an initial object $(\mathcal{M}, g)$. This is a monoid $\mathcal{M}$ with some elements labelled by $g\colon S \to M$ such that for *any* monoid $\mathcal{N}$ with S-labelled elements, there is a unique monoid homomorphism from $\mathcal{M}$ to $\mathcal{N}$ that preserves labels.

Since some labelled monoids have no objects with multiple labels, it follows that g also can't give the same object multiple labels. In other words, g is injective. Hence, without loss of generality, simply by swapping objects around, we can in fact choose $\mathcal{M}$ so that g is an inclusion.

So the situation is as follows. We can think of the monoid $\mathcal{M}$ as having objects M including the selected set S. And this monoid is such that, for any other monoid $\mathcal{N}$ and set-function $f\colon S \to N$, there is a *unique* homomorphism from $\mathcal{M}$ to $\mathcal{N}$ which sends members of S to their images under f.

A moment's reflection shows that $\mathcal{M}$ must be a free monoid with generators S, in the sense we initially characterized in §20.5. In other words, its objects M include a unit element, the members of S, all their possible products, products of products, etc., with no unnecessary identities between these elements, and with nothing else. Why so? Here's the argument:

1. Just because $\mathcal{M}$ is a monoid, it must contain a unit element, the members of S, all their possible products, products of products, and so on.

2. Suppose there were some unnecessary identity between two of those elements. Then take a monoid $\mathcal{M}'$ with the same generators (and the same labelling function g) but without that identity. Then a homomorphism from $\mathcal{M}$ to $\mathcal{M}'$ respecting labels will send generators to generators, and (being a homomorphism), will send their products to products, so enforcing the same identity to recur in $\mathcal{M}'$ contrary to hypothesis.

3. Suppose there were extra elements in $\mathcal{M}$ not generated from the unit and members of S. Then there could evidently be multiple homomorphisms from $\mathcal{M}$ to other monoids respecting labelled objects and their products but dealing with the 'junk' differently.

Which all goes to motivate an official categorial definition of the notion we previously only informally characterized:

Definition 105. A free monoid over the set S is an initial object of the comma category $(S \downarrow F)$, where $F\colon \mathsf{Mon} \to \mathsf{Set}$ is the forgetful functor. △

So here's another notion that we have defined in terms of a universal mapping property.

We should check that this tallies with our discussion back in §20.5:

Theorem 110. *Take the monoid $\mathcal{L} = (List(S), ^\frown, 1)$ and equip it with the function $g\colon S \to List(S)$ which sends an element s of S to the list with just that element. Then $(\mathcal{L}, g)$ is a free monoid over S.*

Proof. Suppose $\mathcal{N}$ is a monoid $(N, \cdot, 1_N)$ and $f\colon S \to N$ is a set function. We need to show that there is a unique monoid homomorphism from $\mathcal{L}$ to $\mathcal{N}$ which sends a list with the single element s to $f(s)$.

Let $j\colon List(S) \to N$ send the empty list to 1_N, and send a one-element list $s \in List(S)$ (with the single element $s \in S$) to $f(s)$. Extend the function to all members of $List(S)$ by putting $j(s_1^\frown s_2^\frown \ldots ^\frown s_n) = j(s_1) \cdot j(s_2) \cdot \ldots \cdot j(s_n)$. Then j is a monoid homomorphism.

Suppose k is another monoid homomorphism $j\colon List(S) \to N$ which sends a list with the single element s to $f(s)$, so j and k agree on unit lists. Hence

$$\begin{aligned} k(s_1^\frown s_2^\frown \ldots ^\frown s_n) &= k(s_1) \cdot k(s_2) \cdot \ldots \cdot k(s_n) \\ &= j(s_1) \cdot j(s_2) \cdot \ldots \cdot j(s_n) \\ &= j(s_1^\frown s_2^\frown \ldots ^\frown s_n). \end{aligned}$$

Whence j and k must agree on all members of $List(S)$. □

24.6 A theorem on comma categories and limits

We end this chapter with what you can consider for the moment to be a slightly tricky exercise to test understanding of various definitions: so by all means skip it for now. However, we will appeal to this result later, so we prove it now to avoid breaking up the flow later.

Theorem 111. *Suppose we have a functor $G\colon \mathscr{B} \to \mathscr{A}$ and an object $A \in \mathscr{A}$. Then if $\mathscr{B}$ has limits of shape J and G preserves them, then $(A \downarrow G)$ also has limits of shape J.*

Proof. Take any diagram $D\colon \mathsf{J} \to (A \downarrow G)$. By definition, for any J-object J, DJ is a pair (D_J, f_J), where D_J is a object in $\mathscr{B}$, and $f_J\colon A \to GD_J$ is an arrow in $\mathscr{A}$. And for any $d\colon J \to K$ in J, $Dd\colon D_J \to D_K$ is a $\mathscr{B}$-arrow such that $f_K = GDd \circ f_J$. The target is to show that, given our suppositions, D has a limit in $(A \downarrow G)$.

For convenience, we introduce the forgetful functor $U\colon (A \downarrow G) \to \mathscr{B}$ which acts in the obvious way, i.e. it sends an $(A \downarrow G)$-object (B, f) to B, and sends an $(A \downarrow G)$-arrow $j\colon B \to B'$ to itself.

Start with the functor $U \circ D\colon \mathsf{J} \to \mathscr{B}$. We know that *this* has a limit (by our hypothesis that $\mathscr{B}$ has all limits of shape J). Call this limit $[L, \pi_J]$. So L is a $\mathscr{B}$-object; and the π_J are $\mathscr{B}$-arrows such that any $d\colon J \to K$, $\pi_K = UDd \circ \pi_J$, i.e. $\pi_K = Dd \circ \pi_J$. And since G preserves limits, we also know that $[GL, G\pi_J]$ is a limit cone in $\mathscr{A}$ for $GUD\colon \mathsf{J} \to \mathscr{A}$.

Now take A and the arrows f_J. These comprise a cone $[A, f_J]$ over GUD in $\mathscr{A}$. Why? By definition, f_J is an arrow from A to GD_J i.e to $GUD(J)$. And we know that for each $d\colon J \to K$, $f_K = GUD(d) \circ f_J$.

This cone $[A, f_J]$ must therefore factor uniquely through the limit $[GL, G\pi_J]$: i.e. there is a unique $u\colon A \to GL$ such that for all J, $f_J = G\pi_J \circ u$. Which, by definition of arrows in the comma category, means that for each J, π_J is an arrow from (L, u) to (D_J, f_J) in $(A \downarrow G)$. And these arrows π_J give us a cone over D in $(A \downarrow G)$ with vertex (L, u), since as we have already seen, for any $d\colon J \to K$, $\pi_K = Dd \circ \pi_J$.

If we can show that this cone is indeed a limit cone, we are done. Suppose therefore that there is another cone over D in $(A \downarrow G)$ with vertex (B, v) and arrows $b_J\colon (B, v) \to (D_J, f_J)$ in $(A \downarrow G)$ where, given $d\colon J \to K$ in J, $b_K = Dd \circ b_J$. We need to show that there is a unique $k\colon (B, v) \to (L, u)$ in $(A \downarrow G)$, i.e. a unique $k\colon B \to B'$ in $\mathscr{B}$, such that for each J, $b_J = \pi_J \circ k$. However, our assumptions also make $[B, b_J]$ a cone over $U \circ D$. So $[B, b_J]$ must factor though the limit $[L, \pi_J]$ via a unique $k\colon B \to L$: so there is indeed a unique k such that, for each J, $b_J = \pi_J \circ k$. $\square$

25 Natural isomorphisms

> Category theory is an embodiment of Klein's dictum that it is the maps that count in mathematics. If the dictum is true, then it is the functors between categories that are important, not the categories. And such is the case. Indeed, the notion of category is best excused as that which is necessary in order to have the notion of functor. But the progression does not stop here. There are maps between functors, and they are called natural transformations.
> (Freyd 1965, quoted in Marquis 2008.)

Natural transformations – and more specifically, natural isomorphisms – were there from the very start. The founding document of category theory is the paper by Samuel Eilenberg and Saunders Mac Lane 'General theory of natural equivalences' (Eilenberg and Mac Lane 1945). But the key idea had already been introduced, three years previously, in a paper on 'Natural isomorphisms in group theory', before the categorial framework was invented in order to provide a general setting for the account (Eilenberg and Mac Lane 1942). Natural isomorphisms and natural transformations are now going to start to take centre stage in our story too.

25.1 Natural isomorphisms between functors defined

Suppose we have a pair of parallel functors $\mathscr{C} \underset{G}{\overset{F}{\rightrightarrows}} \mathscr{D}$; when do the two functors 'come to same', categorially speaking?

Each of F and G projects the objects and arrows of $\mathscr{C}$ into $\mathscr{D}$ giving two images of $\mathscr{C}$ within $\mathscr{D}$. Omitting identity arrows, we might have:

$$\begin{tikzcd}
 & & & FA \arrow[rr, "Ff"] \arrow[dr, "Fg"'] & & FB \\
A \arrow[rr, "f"] \arrow[dr, "g"'] & & B \arrow[ur, dashed, "F"] \arrow[dr, dashed, "G"'] & & FC \arrow[ur, "Fh"'] & \\
 & C \arrow[ur, "h"'] & & GA \arrow[rr, "Gf"] \arrow[dr, "Gg"'] & & GB \\
 & & & & GC \arrow[ur, "Gh"'] & \\
 & \mathscr{C} & & & \mathscr{D} &
\end{tikzcd}$$

In general these images of $\mathscr{C}$ can be significantly different. But at least we can guarantee that the results of applying F and G to objects will be the same (up to isomorphism) if there is a suite ψ of $\mathscr{D}$-isomorphisms $\psi_A \colon FA \xrightarrow{\sim} GA$, $\psi_B \colon FB \xrightarrow{\sim} GB$, etc., thus ensuring that $FA \cong GA$, $FB \cong GB$, etc.

Now, given such a suite of isomorphisms ψ and an arrow $f \colon A \to B$, there will be the following arrows from $FA \to FB$: Ff, of course, but also $\psi_B^{-1} \circ Gf \circ \psi_A$. If things are to fit together nicely, we should require these arrows to be the same (i.e. require that $\psi_B \circ Ff = Gf \circ \psi_A$). This ensures that when F and G are both applied to arrows $f, f', f'', \ldots : A \to B$, there is a tidy one-to-one correspondence between the arrows $Ff, Ff', Ff'', \ldots$ and $Gf, Gf', Gf'', \ldots$, so the results of applying F and G to arrows also stay in step.

Which all goes to motivate the following standard definition of an appropriate notion of isomorphism between parallel functors (or rather, it's a pair of definitions, one for each flavour of functor):

Definition 106. Let $\mathscr{C}$ and $\mathscr{D}$ be categories, let $\mathscr{C} \underset{G}{\overset{F}{\rightrightarrows}} \mathscr{D}$ be covariant functors (respectively, contravariant functors), and suppose that for each $\mathscr{C}$-object C there is a $\mathscr{D}$-isomorphism $\psi_C \colon FC \xrightarrow{\sim} GC$. Then ψ, the family of arrows ψ_C, is said to be a *natural isomorphism* between F and G if for every arrow $f \colon A \to B$ (respectively, $f \colon B \to A$, note the reversal!) in $\mathscr{C}$ the following *naturality square* commutes in $\mathscr{D}$:

$$\begin{array}{ccc} FA & \xrightarrow{Ff} & FB \\ \downarrow{\scriptstyle \psi_A} & & \downarrow{\scriptstyle \psi_B} \\ GA & \xrightarrow{Gf} & GB \end{array}$$

In this case, we write $\psi : F \xRightarrow{\sim} G$, and the ψ_C are said to be components of ψ. If there is such a natural isomorphism, F and G will be said to be naturally isomorphic, and we write $F \cong G$. △

25.2 Why 'natural'?

But why call this a *natural* isomorphism? There's a back-story which we mentioned in the preamble of the chapter and which we should now pause to explain, using one of Eilenberg and Mac Lane's own examples.

(a) Consider a finite dimensional vector space V over the reals $\mathbb{R}$, and the corresponding dual space V^* of linear functions $f \colon V \to \mathbb{R}$. It is elementary to show that V is isomorphic to V^* (there's a bijective linear map between the spaces).

Proof sketch: Take a basis $B = \{v_1, v_2, \ldots, v_n\}$ for V. Define the functions $v_i^* \colon V \to \mathbb{R}$ by putting $v_i^*(v_j) = 1$ if $i = j$ and $v_i^*(v_j) = 0$ otherwise. Then $B^* = \{v_1^*, v_2^*, \ldots, v_n^*\}$ is a basis for V^*, and the linear function $\varphi_B \colon V \to V^*$ generated by putting $\varphi_B(v_i) = v_i^*$ is an isomorphism.

Note, however, that the isomorphism we have arrived at here depends on the initial choice of basis B. And no choice of basis B is more 'natural' than any other. So no one of the isomorphisms from $\varphi_B : V \to V^*$ of the kind just defined is to be especially preferred.

To get a sharply contrasting case, now consider V^{**} the double dual of V, i.e. the space of functionals $g \colon V^* \to \mathbb{R}$. Suppose we select a basis B for V, define a derived basis B^* for V^* as we just did, and then use this new basis in turn to define a basis B^{**} for V^{**} by repeating the same construction. Then we can construct an isomorphism from V to V^{**} by mapping the elements of B to the corresponding elements of B^{**}. However, *we don't have to go through any such palaver of initially choosing a basis*. Suppose we simply define $\psi_V : V \to V^{**}$ as acting on an element $v \in V$ to give as output the functional $\psi_V(v) \colon V^* \to \mathbb{R}$ which sends a function $f \colon V \to \mathbb{R}$ to the value $f(v)$: in short, we set $\psi_V(v)(f) = f(v)$. It is readily checked that ψ_V is an isomorphism (we rely on the fact that V is finite-dimensional). And obviously we get *this* isomorphism independently of any arbitrary choice of basis.

Interim summary: it is very natural(!) to say that the isomorphisms of the kind we described between V and V^* are not intrinsic, are not 'natural' to the spaces involved. By contrast there *is* a 'natural' isomorphism between V and V^{**}, generated by a general procedure that applies to any suitable vector space.

Now, there are many other cases where we might similarly want to contrast intuitively 'natural' maps with more arbitrarily cooked-up maps between structured objects. The story goes that such talk was already bandied about quite a bit e.g. by topologists in the 1930s. So a question arises: can we give a clear general account of what makes for naturality here? Eilenberg and Mac Lane were aiming to provide such a story.

(b) To continue with our example, the isomorphism $\psi_V : V \xrightarrow{\sim} V^{**}$ which we constructed might be said to be natural *because the only information about V it relies on is that V is a finite dimensional vector space over the reals.*

That implies that our construction will work in exactly same way for any other such vector space W, so we get a corresponding isomorphism $\psi_W : W \xrightarrow{\sim} W^{**}$. Now, we will expect such naturally constructed isomorphisms to respect the relation between a structure-preserving map f between the spaces V and W and its double-dual correlate map between V^{**} to W^{**}. Putting that more carefully, we want the following informal diagram to commute, whatever vector spaces we take and for any linear map $f \colon V \to W$,

$$\begin{tikzcd}
V \arrow[r, "f"] \arrow[d, "\psi_V"] & W \arrow[d, "\psi_W"] \\
V^{**} \arrow[r, "DD(f)"] & W^{**}
\end{tikzcd}$$

where $DD(f)$ is the double-dual correlate of f.

Recall, back in §20.8, we saw that the correlate Df of $f \colon V \to W$ is the functional $(- \circ f) \colon W^* \to V^*$; and then moving to the double dual, the correlate

DDf will be the functional we can notate $(-\circ(-\circ f))\colon V^{**} \to W^{**}$. Our diagram can then indeed be seen to commute, both paths sending an element $v \in V$ to the functional that maps a function $k\colon W \to \mathbb{R}$ to the value $k(f(v))$. Think about it!

(c) So far, so good. Now let's pause to consider why there can't be a similarly 'natural' isomorphism from V to V^*. (The isomorphisms based on an arbitrary choice of basis aren't natural: but we want to show that there is no other 'natural' isomorphism either.)

Suppose then that there were a construction which gave us an isomorphism $\varphi_V : V \xrightarrow{\sim} V^*$ which again does not depend on information about V other than that it has the structure of a finite dimensional vector space. So again we will want the construction to work the same way on other such vector spaces, and to be preserved by structure-preserving maps between the spaces. This time, therefore, we will presumably want the following diagram to commute for any structure-preserving f between vector spaces (note, however, that we have to reverse an arrow for things to make any sense, given our definition of the contravariant functor D):

$$\begin{tikzcd}
V \arrow[r, "f"] \arrow[d, "\varphi_V"] & W \arrow[d, "\varphi_W"] \\
V^* & W^* \arrow[l, "D(f)"]
\end{tikzcd}$$

Hence $D(f) \circ \varphi_W \circ f = \varphi_V$. But by hypothesis, the φs are isomorphisms; so in particular φ_V has an inverse. So we have $(\varphi_V^{-1} \circ D(f) \circ \varphi_W) \circ f = 1_V$. Therefore f has a left inverse. But it is obvious that in general, a linear map $f\colon V \to W$ need not have a left inverse. Hence there can't in general be isomorphisms $\varphi_V, \varphi_W \colon V \to V^*$ making that diagram commute.

(d) We started off by saying that, intuitively, there's a 'natural', instrinsic, isomorphism between a (finite dimensional) vector space and its double dual, one that depends only on their structures as vector spaces. And we've now suggested that this intuitive idea can be reflected by saying that a certain diagram always commutes, for any choice of vector spaces and structure-preserving maps between them.

We have also seen that we can't get analogous always-commuting diagrams for the case of isomorphisms between a vector space and its dual – which chimes with the intuition that the obvious examples are *not* 'natural' isomorphisms.

So this gives us a promising way forward: characterize 'naturality' here in terms of the availability of a family of isomorphisms which make certain informal (non-categorial) diagrams commute. Note next, however, that the claim that the diagram

$$\begin{tikzcd}
V \arrow[r, "f"] \arrow[d, "\psi_V"] & W \arrow[d, "\psi_W"] \\
V^{**} \arrow[r, "DD(f)"] & W^{**}
\end{tikzcd}$$

always commutes can be indeed put a slightly different way, using category-speak.

For we have in effect been talking about the category we'll here call simply Vect (of finite-dimensional spaces over the reals and the structure-preserving maps between them), and about a functor we can call $DD\colon \mathsf{Vect} \to \mathsf{Vect}$ which takes a vector space to its double dual, and maps each arrow between vector spaces to its double-dual correlate as explained. There is also a trivial functor $1\colon \mathsf{Vect} \to \mathsf{Vect}$ that maps each vector space to itself and each Vect-arrow to itself. So we can re-express the claim that the last diagram commutes as follows. For every arrow $f\colon V \to W$ in Vect, there are isomorphisms ψ_V and ψ_W in Vect such that *this* diagram commutes:

$$\begin{tikzcd}
1(V) \arrow[r, "1(f)"] \arrow[d, "\psi_V"] & 1(W) \arrow[d, "\psi_W"] \\
DD(V) \arrow[r, "DD(f)"] & DD(W)
\end{tikzcd}$$

In other words, in the terms of the previous section, the suite of isomorphisms ψ_V provide a natural isomorphism $\psi\colon 1 \xRightarrow{\sim} DD$.

(e) In sum: our claim that there is an intuitively 'natural' isomorphism between two *spaces*, a vector space and its double dual, now becomes reflected in the claim that there is an isomorphism in our official sense between two *functors*, the identity and the double-dual functors from the category Vect to itself. Hence the aptness of calling the latter isomorphism between functors a *natural* isomorphism.

We will return at the end of the chapter to the thought that we can generalize from our example of vector spaces and claim that in many (most? all?) cases, intuitively 'natural' isomorphisms between widgets and wombats can be treated officially as natural isomorphisms between suitable functors.

25.3 More examples of natural isomorphormisms

We now have one case to hand. Let's next give some more simple examples of natural isomorphisms:

(1) We quickly mention the trivial case. Given any functor $F\colon \mathscr{C} \to \mathscr{D}$, then the following diagram of course commutes for every $f\colon A \to B$ in $\mathscr{C}$:

$$\begin{tikzcd} FA \arrow[r, "Ff"] \arrow[d, "1_{FA}"] & FB \arrow[d, "1_{FB}"] \\ FA \arrow[r, "Ff"] & FB \end{tikzcd}$$

So we have a natural isomorphism $1_F \colon F \xRightarrow{\sim} F$, where the components $(1_F)_A$ of the isomorphism are the identity arrows $1_{(FA)}$.

(2) Given a group $G = (G, *, e)$ we can define its opposite $G^{op} = (G, *^{op}, e)$, where $a *^{op} b = b * a$.

We can also define a functor $Op \colon \mathsf{Grp} \to \mathsf{Grp}$ which sends a group G to its opposite G^{op}, and sends an arrow f in the category, i.e. a group homomorphism $f \colon G \to H$, to $f^{op} \colon G^{op} \to H^{op}$ where $f^{op}(a) = f(a)$ for all a in G. f^{op} so defined is indeed a group homomorphism, since

$$f^{op}(a *^{op} a') = f(a' * a) = f(a') * f(a) = f^{op}(a) *^{op} f^{op}(a')$$

Claim: there is a natural isomorphism $\psi \colon 1 \xRightarrow{\sim} Op$ (where 1 is the trivial identity functor in Grp).

Proof. We need to find a family of isomorphisms $\psi_G, \psi_H, \ldots$ in Grp such that the following diagram always commutes for any homomorphism $f \colon G \to H$:

$$\begin{tikzcd} G \arrow[r, "f"] \arrow[d, "\psi_G"] & H \arrow[d, "\psi_H"] \\ G^{op} \arrow[r, "f^{op}"] & H^{op} \end{tikzcd}$$

(Careful: G, H are groups here, not functors!) Now, since taking the opposite *between* groups involves reversing the order of multiplication and taking inverses *inside* a group in effect does the same, let's put $\psi_G(a) = a^{-1}$ for any G-element a, and likewise for ψ_H, etc. It is easy to check that with this choice of components, ψ is a natural isomorphism. $\square$

(3) Recall from §20.2 the functor $List \colon \mathsf{Set} \to \mathsf{Set}$ which sends a set X to the set of finite lists of members of X. One natural isomorphism from this functor to itself is the identity isomorphism $1 \colon List \xRightarrow{\sim} List$. But there is also another natural isomorphism $\rho \colon List \xRightarrow{\sim} List$, whose component $\rho_X \colon List(X) \to List(X)$ acts on a list of X-elements to reverse their order.

(4) Now for an example involving contravariant functors from Set to Set.

First, recall the contravariant powerset functor $\overline{P} \colon \mathsf{Set} \to \mathsf{Set}$ which maps a set X to its powerset $\mathscr{P}(X)$, and maps a set-function $f \colon Y \to X$ to the function $Inv(f)$ which sends $U \subseteq X$ to its inverse image $f^{-1}[U] \subseteq Y$.

And let C be the hom-functor $\mathsf{Set}(-, 2)$, where 2 is some nice two-element set such as $\{\{\varnothing\}, \varnothing\}$ which we can think of as $\{true, false\}$. So C sends a set X to $\mathsf{Set}(X, 2)$, i.e. the set of functions from X to 2: and C sends an arrow $f\colon Y \to X$ to the function $- \circ f\colon \mathsf{Set}(X, 2) \to \mathsf{Set}(Y, 2)$ (i.e. the function which sends an arrow $g\colon X \to 2$ to the arrow $g \circ f\colon Y \to 2$).

Claim: $\overline{P} \cong C$.

Proof. We need to find a family of isomorphisms $\psi_X, \psi_Y, \ldots$ in Set such that the following diagram always commutes:

$$\begin{tikzcd} \overline{P}X \arrow[r, "\overline{P}f"] \arrow[d, "\psi_X"] & \overline{P}Y \arrow[d, "\psi_Y"] \\ CX \arrow[r, "Cf"] & CY \end{tikzcd} \quad \text{equivalently} \quad \begin{tikzcd} \mathcal{P}(X) \arrow[r, "Inv(f)"] \arrow[d, "\psi_X"] & \mathcal{P}(Y) \arrow[d, "\psi_Y"] \\ \mathsf{Set}(X,2) \arrow[r, "- \circ f"] & \mathsf{Set}(Y,2) \end{tikzcd}$$

Take any ψ_X to be the isomorphism which associates a set $U \subseteq X$ with its characteristic function (i.e the function which sends an element of X to *true* iff it is in U). Then it is easy to see that the diagram will always commute. Both routes sends a set $U \subseteq X$ to the function which sends y in Y to *true* iff $fy \in U$. □

(5) This time we take a certain pair of (covariant) functors $\mathsf{Grp} \underset{V}{\overset{U}{\rightrightarrows}} \mathsf{Set}$. Here U is simply the forgetful functor which sends a group G to its underlying set, and sends homomorphisms to themselves. While V is the hom-functor $\mathsf{Grp}(Z, -)$, where Z is the group of integers under addition. So, by definition, V sends an object, i.e. a group G, to the set of group homomorphisms from Z to G. And V sends an arrow $f\colon G \to G'$ to the function we notate $f \circ -$, i.e. the function which sends a homomorphism $h\colon Z \to G$ to the homomorphism $f \circ h\colon Z \to G'$. Claim: $U \cong V$.

Proof. Note first that a group homomorphism from $Z = (\mathbb{Z}, 0, +)$ to $G = (G, e, \cdot)$ is entirely fixed by fixing where 1 goes. For 0 has to go to the identity element e; and if 1 goes to the element a, every sum $1+1+1+\ldots+1$ has to go to the corresponding $a \cdot a \cdot a \cdot \ldots \cdot a$, with inverses going to inverses. Which means that there is a set-bijection ψ_G from elements of G to members of $\mathsf{Grp}(Z, -)$.

It is then immediate that the required naturality square commutes for any $f\colon G \to G'$:

$$\begin{tikzcd} UG \arrow[r, "f"] \arrow[d, "\psi_G"] & UG' \arrow[d, "\psi_{G'}"] \\ VG \arrow[r, "f \circ -"] & VG' \end{tikzcd}$$

with either route round the square taking us from an element $a \in G$ to the unique homomorphism from Z to G' which sends 1 to fa. □

Our next examples also involve hom-functors. For motivation, reflect on the natural one-to-one bijection between two-place set functions from A and B to C, and one-place functions from A to functions-from-B-to-C (see §18.1). Categorically, that gives us an isomorphism between the hom-sets $\mathsf{Set}(A \times B, C)$ and $\mathsf{Set}(A, C^B)$. And the intuitive naturality of the bijection means that this doesn't depend on particular choices of A, B or C. So we will expect, inter alia, that the hom-functors $\mathsf{Set}(A \times B, -)$ and $\mathsf{Set}(A, (-)^B)$ are isomorphic. Moreover, this should apply not just to the category Set but, generalizing,

(6) If $\mathscr{C}$ is a locally small category with exponentials, then $\mathscr{C}(A \times B, -) \cong \mathscr{C}(A, (-)^B)$.

Proof. Here $\mathscr{C}(A, (-)^B) = \mathscr{C}(A, -) \circ (-)^B$, where $(-)^B$ is the functor that we met in §20.6. Now, $(-)^B$ sends an arrow $f\colon C \to C'$ to $f^B = \overline{f \circ ev}$. Hence $\mathscr{C}(A, (-)^B)$ sends f to $\overline{f \circ ev} \circ -$.

To provide the announced natural isomorphism, we need to find a family of isomorphisms ψ_C such that for every $f\colon C \to C'$ in $\mathscr{C}$, the following diagram commutes in Set:

$$\begin{tikzcd}
\mathscr{C}(A \times B, C) \arrow[r, "\mathscr{C}(A\times B, f)\ =\ f\circ -"] \arrow[d, "\psi_C"] & \mathscr{C}(A \times B, C') \arrow[d, "\psi_{C'}"] \\
\mathscr{C}(A, C^B) \arrow[r, "\mathscr{C}(A, f^B)\ =\ \overline{(f\circ ev)}\circ -"] & \mathscr{C}(A, C'^B)
\end{tikzcd}$$

Suppose then that we take the component ψ_C to be the isomorphism which sends an arrow g in $\mathscr{C}(A \times B, C)$ to its exponential transpose $\overline{g}$ in $\mathscr{C}(A, C^B)$. Will that make the diagram commute?

Chase an arrow g in $\mathscr{C}(A \times B, C)$ round the diagram both ways. Then the diagram will commute if $\overline{f \circ ev} \circ \overline{g} = \overline{f \circ g}$. But consider:

$$\begin{tikzcd}
A \times B \arrow[d, dashed, "\overline{g}\times 1_B"] \arrow[dr, "g"] \arrow[dd, dashed, bend right=60, "\overline{f\circ g}\times 1_B"'] & \\
C^B \times B \arrow[r, "ev"] \arrow[d, dashed, "\overline{f\circ ev}\times 1_B"] & C \arrow[d, "f"] \\
C'^B \times B \arrow[r, "ev'"] & C'
\end{tikzcd}$$

Note the composite $f \circ g\colon A \times B \to C'$. By the definition of $[C'^B, ev']$ as an exponential, there is a unique arrow $\overline{f \circ g}$ such that

$$ev' \circ \overline{f \circ g} \times 1_B = f \circ g.$$

But since the top triangle and the bottom square also commute, we have

$$f \circ g = ev' \circ (\overline{f \circ ev} \times 1_B) \circ (\overline{g} \times 1_B) = ev' \circ (\overline{f \circ ev} \circ \overline{g}) \times 1_B.$$

Hence, by the uniqueness requirement, we get $\overline{f \circ ev} \circ \overline{g} = \overline{f \circ g}$, and we are done. ☐

(7) Similarly motivated, we see that if $\mathscr{C}$ is a locally small category with exponentials, then $\mathscr{C}(- \times B, C) \cong \mathscr{C}(-, C^B)$

Proof. Here, $\mathscr{C}(- \times B, C) = \mathscr{C}(-, C) \circ (- \times B)$, where the first is a contravariant hom-functor, and $- \times B$ is another functor that we met in §20.6. Now, $- \times B$ sends an arrow $f\colon A' \to A$ to $f \times 1_B$. Hence $\mathscr{C}(- \times B, C)$ sends f to $- \circ (f \times 1_B)\colon (A \times B, C) \to (A' \times B, C)$.

To provide the announced natural isomorphism, we need to find a family of isomorphisms ψ_A such that for every $f\colon A' \to A$ in $\mathscr{C}$, the following diagram commutes in Set:

$$\begin{tikzcd}
\mathscr{C}(A \times B, C) \arrow[r, "\mathscr{C}(f \times B, C) \,=\, - \circ (f \times 1_B)"] \arrow[d, "\psi_A"] & \mathscr{C}(A' \times B, C) \arrow[d, "\psi_{A'}"] \\
\mathscr{C}(A, C^B) \arrow[r, "\mathscr{C}(f, C^B) \,=\, - \circ f"] & \mathscr{C}(A', C^B)
\end{tikzcd}$$

As before, take the component ψ_A to be the isomorphism which sends an arrow g in $\mathscr{C}(A \times B, C)$ to its transpose $\overline{g}$ in $\mathscr{C}(A, C^B)$.

Chase an arrow g in $\mathscr{C}(A \times B, C)$ round the diagram both ways. Then the diagram will commute if $\overline{g} \circ f = \overline{g \circ (f \times 1_B)}$.

But now consider this further diagram:

$$\begin{tikzcd}
A' \times B \arrow[r, "f \times 1_B"] \arrow[dr, dashed, "\overline{g \circ (f \times 1_B)} \times 1_B"'] & A \times B \arrow[d, dashed, "\overline{g} \times 1_B"'] \arrow[dr, "g"] & \\
& C^B \times B \arrow[r, "ev"'] & C
\end{tikzcd}$$

By definition, $\overline{g \circ (f \times 1_B)}\colon A' \to C^B$ is the unique arrow that when plugged into $- \times 1_B$ makes the rhombus commute.

But the right-hand triangle commutes, so it follows that $(\overline{g} \times 1_B) \circ (f \times 1_B)$ is another arrow from $A' \times B$ to $C^B \times B$ which makes the rhombus commute. However, by Theorem 44, $(\overline{g} \times 1_B) \circ (f \times 1_B) = (\overline{g} \circ f) \times 1_B$,. Hence $\overline{g} \circ f$ plugged into $- \times 1_B$ also makes the rhombus commute. Which proves that $\overline{g} \circ f = \overline{g \circ (f \times 1_B)}$. ☐

These last two proofs show how confirming that two functors are indeed naturally isomorphic (even in simple cases where the result is entirely expected) can be fiddly. We will encounter this sort of annoyance again.

25.4 Natural/unnatural isomorphisms between objects

(a) Suppose we have functors $F, G\colon \mathscr{C} \to \mathscr{D}$; and let $A, A', A'', \ldots$ be objects in $\mathscr{C}$. Then there will be objects $FA, FA', FA'' \ldots$ and $GA, GA', GA'' \ldots$ in $\mathscr{D}$.

And in some cases these will be pairwise isomorphic, so that we have $FA \cong GA$, $FA' \cong GA'$, $FA'' \cong GA''$ ....

One way this can happen, as we have seen, is that there is a natural isomorphism between the functors F and G. But it is important to emphasize that it can happen in other, 'unnatural', ways. We've met unnaturalness before, but still let's have a couple more examples, one a toy example to make again the point of principle, then a standard illustrative case which is worth thinking through:

(1) Suppose $\mathscr{C}$ is a category with exactly one object A, and two arrows, the identity arrow 1_A, and distinct arrow f, where $f \circ f = f$. And now consider two functors, the identity functor $1_{\mathscr{C}} \colon \mathscr{C} \to \mathscr{C}$, and the functor $F \colon \mathscr{C} \to \mathscr{C}$ which sends the only object to itself, and sends both arrows to the identity arrow. Then, quite trivially, we have $1_{\mathscr{C}}(A) \cong F(A)$ for the one and only object in $\mathscr{C}$. But there isn't a natural isomorphism between the functors, because by hypothesis $1_A \neq f$, and hence the square

$$\begin{tikzcd} F(A) \arrow[r, "F(f)"] \arrow[d, "1_A"] & F(A) \arrow[d, "1_A"] \\ 1_{\mathscr{C}}(A) \arrow[r, "1_{\mathscr{C}}(f)"] & 1_{\mathscr{C}}(A) \end{tikzcd}, \text{ which is simply } \begin{tikzcd} A \arrow[r, "1_A"] \arrow[d, "1_A"] & A \arrow[d, "1_A"] \\ A \arrow[r, "f"] & A \end{tikzcd},$$

cannot commute.

(2) We'll work in the category $\mathscr{F}$ of finite sets and *bijections* between them.

There is a functor $Sym \colon \mathscr{F} \to \mathscr{F}$ which (i) sends a set A in $\mathscr{F}$ to the set of permutations on A (treating permutation functions as sets, this is a finite set), and (ii) sends a bijection $f \colon A \to B$ in $\mathscr{F}$ to the bijection that sends the permutation p on A to the permutation $f \circ p \circ f^{-1}$ on B. Note: if A has n members, there are $n!$ members of the set of permutations on A.

There is also a functor $Ord \colon \mathscr{F} \to \mathscr{F}$ which (i) sends a set A in $\mathscr{F}$ to the set of total linear orderings on A (you can identify an order-relation with a set, so we can think of this too as a finite set), and (ii) sends a bijection $f \colon A \to B$ in $\mathscr{F}$ to the bijection $Ord(f)$ which sends a total order on A to the total order on B where $x <_A y$ iff $f(x) <_B f(y)$. Again, if A has n members, there are also $n!$ members of the set of linear orderings on A.

Now, for any object A of $\mathscr{F}$, $Sym(A) \cong Ord(A)$ (since they are equinumerous finite sets). But there cannot be a natural isomorphism ψ between the functors Sym and Ord. For suppose otherwise, and consider the functors acting on a bijection $f \colon A \to A$. Then the following naturality square would have to commute:

$$\begin{tikzcd} Sym(A) \arrow[r, "Sym(f)"] \arrow[d, "\psi_A"] & Sym(A) \arrow[d, "\psi_A"] \\ Ord(A) \arrow[r, "Ord(f)"] & Ord(A) \end{tikzcd}$$

> Consider then what happens to the identity permutation i in $Sym(A)$: it gets sent by $Sym(f)$ to $f \circ i \circ f^{-1} = i$. So the naturality square would tell us that $\psi_A(i) = Ord(f)(\psi_A(i))$. But that in general won't be so – suppose f swaps around elements, so $Ord(f)$ is not the 'do nothing' identity map.

In a summary slogan, then: pointwise isomorphism doesn't entail natural isomorphism.

(b) We are, however, going mostly to be interested in cases where $FA \cong GA$ (and $FA' \cong GA'$, $FA'' \cong GA''$ ...) as a result of a natural isomorphism. There is standard terminology for such cases:

Definition 107. Given functors $F, G\colon \mathscr{C} \to \mathscr{D}$ and A an object in $\mathscr{C}$, we say that $FA \cong GA$ *naturally in* A (or *naturally in* A *in* $\mathscr{C}$) just if F and G are *naturally* isomorphic.

The definition mentions just a specific object A in $\mathscr{C}$; but there is an implicit generality here. For if $FA \cong GA$ naturally in A, then for some ψ we have $\psi\colon F \overset{\sim}{\Longrightarrow} G$. So as well as an isomorphism $\psi_A\colon FA \xrightarrow{\sim} GA$, there are other isomorphisms $\psi_{A'}\colon FA' \xrightarrow{\sim} GA', \psi_{A''}\colon FA'' \xrightarrow{\sim} GA''$, etc., for other objects $A', A'', \ldots$, making $FA' \cong GA'$ (naturally in A'), $FA'' \cong GA''$ (naturally in A''), etc.

To help fix ideas, let's note a useful little result about this notion of an isomorphism between objects holding naturally:

Theorem 112. *Given functors $F, G, H\colon \mathscr{C} \to \mathscr{D}$, an object A in $\mathscr{C}$, and a functor $K\colon \mathscr{B} \to \mathscr{C}$, then*

(1) if $FA \cong GA$ naturally in A, then for all A' in $\mathscr{C}$, $FA' \cong GA'$ naturally in A'.

(2) if $FA \cong GA$ and $GA \cong HA$, both naturally in A, then $FA \cong HA$ naturally in A.

(3) if $FA \cong GA$ naturally in A, then $FKB \cong GKB$ naturally in B in $\mathscr{B}$.

Proof. (1) is immediate, for if $FA \cong GA$ naturally in A, F is naturally isomorphic to G, so there is a component of the natural isomorphism at A' making $FA' \cong GA'$.

For (2), just note that natural isomorphisms vertically compose.

For (3), just note that, if there is a natural isomorphism α between F and G, then (by 'whiskering') there is a natural isomorphism between FK and GK, whose component at B is α_{KB}. □

(c) Let's mention just a few examples. We have seen that $V \cong DDV$ naturally in V in Vect: that was the message of §25.2.

Likewise, $UG \cong \mathsf{Grp}(Z, G)$ naturally in G in Grp: that was the message of §25.3 (5).

And from §25.3 (6) and (7) we get the following, which we will highlight as a theorem:

Theorem 113. *Given a category $\mathscr{C}$ with exponentials, $\mathscr{C}(A \times B, C) \cong \mathscr{C}(A, C^B)$ both naturally in A and naturally in C.*

25.5 An 'Eilenberg/Mac Lane Thesis'?

Let's return to the question we raised before. Can we generalize from e.g. our example of a vector space and its double dual, and say that whenever we have a 'natural' isomorphism between widgets and wombats (i.e. one that doesn't depend on arbitrary choices of co-ordinates, or the like), this can be regimented as a natural isomorphism between suitable associated functors? Let's call the claim that we *can* generalize like this the 'Eilenberg/Mac Lane Thesis'.

I choose the label to be reminiscent of the Church/Turing Thesis that we all know and love, which asserts that every algorithmically computable function (in an informally characterized sense) is in fact recursive/Turing computable/lambda computable. A certain intuitive concept, this Thesis claims, in fact picks out the same functions as certain (provably equivalent) sharply defined concepts.

What kind of evidence do we have for this thesis? Two sorts: (1) 'quasi-empirical', i.e. no unarguable clear exceptions have been found, and (2) conceptual, as in for example Turing's own efforts to show that when we reflect on what we mean by algorithmic computation we get down to the sort of operations that a Turing machine can emulate, so morally a computable function just ought to be Turing computable. The evidence in this case is so overwhelming that in fact we are allowed to appeal to the Church/Turing Thesis as a labour-saving device: if we can give an arm-waving sketch of an argument that a certain function is algorithmically computable, we are allowed to assume that it is indeed recursive/Turing computable/lambda computable without doing the hard work of e.g. defining a Turing maching to compute it.

We now seem to have on the table another Thesis of the same general type: an informal intuitive concept, the Eilenberg/Mac Lane Thesis claims, in fact picks out the same isomorphisms as a certain sharply defined categorial concept.

Evidence? We would expect two sorts. (1*) 'quasi-empirical', a lack of clear exceptions, and maybe (2*) conceptual, an explanation of why the Thesis just ought to be true.

It is, however, not clear exactly how things stand evidentially here, and the usual textbook discussions of natural isomorphisms oddly don't pause to do much more than give a few examples. More really needs to be said. We therefore can't suppose that the new Eilenberg/Mac Lane Thesis is so secure that we can cheerfully appeal to it in the same labour-saving way as the old Church/Turing Thesis. In other words, even if (i) intuitively an isomorphism between objects seems to be set up in a very 'natural' way, without appeal to arbitrary choices, and (ii) we can readily massage the claim of an isomorphism into a claim about at least pointwise isomorphism of relevant functors, we really need to pause to work through a proof if we are to conclude that in fact (iii) there is a natural isomorphism here in the official categorial sense. Annoying, as we said. For as we have already seen, such proofs can be a bit tedious.

26 Natural transformations

We think of isomorphisms categorially as special cases of some wider class of morphisms, namely those of the morphisms which have inverses. Thus isomorphisms inside categories are particular arrows, those with inverses; isomorphisms between categories are particular functors, those with inverses. And now natural isomorphisms between functors are special cases of .... What?

26.1 Natural transformations

(a) The generalized notion of morphisms between functors that we want is obvious enough. In fact, as before, the definition gives us two notions for the price of one:

Definition 108. Let $\mathscr{C}$ and $\mathscr{D}$ be categories, let $\mathscr{C} \underset{G}{\overset{F}{\rightrightarrows}} \mathscr{D}$ be covariant functors (respectively, contravariant functors), and suppose that for each $\mathscr{C}$-object C there is a $\mathscr{D}$-arrow $\alpha_C\colon FC \to GC$. Then α, the family of arrows α_C, is a *natural transformation* between F and G if for every $f\colon A \to B$ (respectively $f\colon B \to A$, note the reversal!) in $\mathscr{C}$ the following *naturality square* commutes in $\mathscr{D}$:

$$\begin{tikzcd} FA \arrow[r, "Ff"] \arrow[d, "\alpha_A"] & FB \arrow[d, "\alpha_B"] \\ GA \arrow[r, "Gf"] & GB \end{tikzcd}$$

In this case, we write $\alpha : F \Rightarrow G$. (A natural isomorphism is thus a natural transformation each of whose components is an isomorphism.) △

Note that while different styles of arrows can be found in use, Greek letters are almost universally used for names of natural transformations.

(b) In sum, a natural transformation between functors $\mathscr{C} \underset{G}{\overset{F}{\rightrightarrows}} \mathscr{D}$ sends an F-image of (some or all of) $\mathscr{C}$ to its G-image in a way which respects the internal structure of the original at least to the extent of preserving composition of arrows. Let's have a couple of initial toy examples of natural transformations which aren't isomorphisms:

(1) Suppose $\mathscr{D}$ has a terminal object 1, and let $F\colon \mathscr{C} \to \mathscr{D}$ be any functor. Then there is also a parallel functor $T\colon \mathscr{C} \to \mathscr{D}$ which sends every $\mathscr{C}$-object to the terminal object 1, and every $\mathscr{C}$-arrow to the identity arrow on the terminal object. Claim: there is a natural transformation $\alpha\colon F \Rightarrow T$.

Proof. We need a suite of $\mathscr{D}$-arrows α_A (one for each A in $\mathscr{C}$) which make the following commute for any $f\colon A \to B$ in $\mathscr{C}$:

$$\begin{tikzcd} FA \arrow[r, "Ff"] \arrow[d, "\alpha_A"] & FB \arrow[d, "\alpha_B"] \\ 1 \arrow[r, "1_1"] & 1 \end{tikzcd}$$

Put each component of α to be the unique arrow from its source to the terminal object: and the diagram must commute because all arrows from FA to 1 are equal. □

(2) Recall the functor $List\colon \mathsf{Set} \to \mathsf{Set}$ where $List_{ob}$ sends a set A to the set of all finite lists of members of A and $List_{arw}$ sends a set-function $f\colon A \to B$ to the map that sends a list $a_0{}^\frown a_1{}^\frown a_2{}^\frown \ldots{}^\frown a_n$ to $fa_0{}^\frown fa_1{}^\frown fa_2{}^\frown \ldots{}^\frown fa_n$. Claim: there is a natural transformation $\alpha\colon 1 \Rightarrow List$, where 1 is the trivial identity functor $1\colon \mathsf{Set} \to \mathsf{Set}$.

Proof. We need a suite of functions α_A which make the following commute for any $f\colon A \to B$ in $\mathscr{C}$:

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[d, "\alpha_A"] & B \arrow[d, "\alpha_B"] \\ List(A) \arrow[r, "List(f)"] & List(B) \end{tikzcd}$$

For any A, put α_A to be the function which sends an element of A to the length-one list containing just that element, and we are immediately done. □

Note, by the way, that we can think of $List$ as the composite functor GF where F is the 'free' functor from Set to Mon which we met in §20.5 and G is the forgetful functor in the other direction, from Mon to Set. We will find later that there are many important natural transformations which are significantly of the form $\alpha\colon 1_{\mathscr{C}} \Rightarrow GF$ (where $1_{\mathscr{C}}$ is the identity functor from $\mathscr{C}$ to itself, and for some $\mathscr{D}$, $\mathscr{C} \underset{G}{\overset{F}{\rightleftarrows}} \mathscr{D}$) and also many of the form $\alpha\colon FG \Rightarrow 1_{\mathscr{D}}$.

(c) Now for two cases of natural transformations which aren't isomorphisms and which have rather more mathematical significance (though we will only sketch them here):

(3) For those who know just a bit more group theory, consider the abelianization of a group G. Officially, this is the quotient of a group by its commutator subgroup $[G, G]$ (but you can think of it as the 'biggest' Abelian group A for which there is a surjective homomorphism from G onto A). There is then a functor Ab which sends a group G to its abelianization $Ab(G)$, and sends an arrow $f\colon G \to H$ to the arrow $Ab(f)\colon Ab(G) \to Ab(H)$ defined in a fairly obvious way.

We therefore have a pair of functors, $\mathsf{Grp} \underset{Ab}{\overset{1}{\rightrightarrows}} \mathsf{Grp}$, and we can then check that the following diagram always commutes,

$$\begin{tikzcd} G \arrow[r, "f"] \arrow[d, "\alpha_G"] & H \arrow[d, "\alpha_H"] \\ Ab(G) \arrow[r, "Ab(f)"] & Ab(H) \end{tikzcd}$$

where $\alpha_G = G/[G, G]$. So we have a natural transformation, but not usually a natural isomorphism, between the functors 1 and Ab.

(4) For those who know rather more topology, we can mention two important functors from topological spaces to groups. One we've met before in §20.7, namely the functor $\pi_1\colon \mathsf{Top}_* \to \mathsf{Grp}$ which sends a space with a basepoint to its fundamental group at the base point. The other functor $H_1\colon \mathsf{Top} \to \mathsf{AbGrp}$ sends a space to the abelian group which is its first homology group (we aren't going to try to explain that here!). Now these functors aren't yet parallel functors between the same categories. But we can define a functor $H_1'\colon \mathsf{Top}_* \to \mathsf{Grp}$ which first forgets base points of spaces, then applies H, and then forgets that the relevant groups are abelian. We simply record that it is a very important fact of topology that, in our categorial terms, there is natural transformation from π_1 to H_1'.

(d) A natural transformation is a suite of arrows from various sources, with each pair of arrows making certain diagrams commute. A cone is essentially a suite of arrows all from the same source, the apex of the cone, with each pair of arrows making certain diagrams commute. Which suggests that we should be able to treat cones as special cases of natural transformations. And we can.

(5) Suppose we have a diagram-as-functor $D\colon \mathsf{J} \to \mathscr{C}$ and also a collapse-to-C functor $\Delta_C\colon \mathsf{J} \to \mathscr{C}$, i.e. a constant functor which sends every J-object to C in $\mathscr{C}$ and every J-arrow to 1_C (see §20.2 (F10)). Let's ask: what does it take for there to be a natural transformation $\alpha\colon \Delta_C \to D$?

Given such an α, the following diagram must commute for any J-arrow $j\colon K \to L$:

$$
\begin{tikzcd}
\Delta_C K \arrow[r, "\Delta_C j"] \arrow[d, "\alpha_K"'] & \Delta_C L \arrow[d, "\alpha_L"] \\
DK \arrow[r, "Dj"] & DL
\end{tikzcd}
=
\begin{tikzcd}
C \arrow[r, "1_C"] \arrow[d, "\alpha_K"'] & C \arrow[d, "\alpha_L"] \\
DK \arrow[r, "Dj"] & DL
\end{tikzcd}
=
$$

$$
\begin{tikzcd}
& C \arrow[dl, "\alpha_K"'] \arrow[dr, "\alpha_L"] & \\
DK \arrow[rr, "Dj"] & & DL
\end{tikzcd}
$$

Which makes the α_J (where J runs over objects in J) the legs of a cone over D with a vertex C. Conversely, the legs of any cone over D with a vertex C can be assembled into a natural transformation $\alpha\colon \Delta_C \to D$. So that means that cones (thought of the austere way, as simply suites of arrows) are indeed certain natural transformations.

26.2 Vertical composition of natural transformations

Before continuing, a further bit of notation will prove useful. When we have functors $F\colon \mathscr{C} \to \mathscr{D}$, $G\colon \mathscr{C} \to \mathscr{D}$, together with a natural transformation $\alpha\colon F \Rightarrow G$, we can neatly represent the whole situation thus:

$$
\begin{tikzcd}
\mathscr{C} \arrow[r, bend left=50, "F"{name=U}] \arrow[r, bend right=50, "G"'{name=D}] & \mathscr{D}
\arrow[Rightarrow, from=U, to=D, "\alpha"]
\end{tikzcd}
$$

Now, arrows in a category can be composed to form new arrows (when targets and sources suitably mesh). Functors between categories can be composed to form new functors. Now we see that natural transformations between functors can be composed, in more than one way, to form new natural transformations. We'll run the discussion entirely in terms of transformations between covariant functors: but there will be parallel results about contravariant functors.

Suppose first that we have three functors $F\colon \mathscr{C} \to \mathscr{D}$, $G\colon \mathscr{C} \to \mathscr{D}$, $H\colon \mathscr{C} \to \mathscr{D}$, together with two natural transformations $\alpha\colon F \Rightarrow G$, and $\beta\colon G \Rightarrow H$.

We can evidently compose these two transformations to get a natural transformation $\beta \circ \alpha\colon F \Rightarrow H$, defined componentwise by putting $(\beta \circ \alpha)_A = \beta_A \circ \alpha_A$ for all objects A in $\mathscr{C}$. Vertically gluing together two commuting naturality squares which share a side gives us a bigger commuting square, meaning that for any $f\colon A \to B$ in $\mathscr{C}$, the following commutes in $\mathscr{D}$:

$$\begin{tikzcd}
FA \arrow[r, "Ff"] \arrow[d, "\alpha_A"] \arrow[dd, bend right, "\beta_A\circ\alpha_A"'] & FB \arrow[d, "\alpha_B"'] \arrow[dd, bend left, "\beta_B\circ\alpha_B"] \\
GA \arrow[r, "Gf"] \arrow[d, "\beta_A"] & GB \arrow[d, "\beta_B"'] \\
HA \arrow[r, "Hf"] & HB
\end{tikzcd}$$

Composing two transformations as in $\begin{tikzcd} \mathscr{C} \arrow[r, bend left=50, "F"{name=U}] \arrow[r, "G"{name=M, description}] \arrow[r, bend right=50, "H"'{name=D}] & \mathscr{D} \arrow[from=U, to=M, phantom, "\Downarrow\alpha"] \arrow[from=M, to=D, phantom, "\Downarrow\beta"] \end{tikzcd}$ to get $\begin{tikzcd} \mathscr{C} \arrow[r, bend left=50, "F"{name=U}] \arrow[r, bend right=50, "H"'{name=D}] & \mathscr{D} \arrow[from=U, to=D, Rightarrow, "\beta\circ\alpha"] \end{tikzcd}$ is rather predictably called *vertical composition*.

26.3 Horizontal composition of natural transformations

We can, however, also put things together *horizontally* in various ways. First, there is so-called *whiskering*(!) where we combine a functor with a natural transformation between functors to get a new natural transformation. Thus, what happens when we 'add a whisker' on the left of a diagram for a natural transformation?

The situation $\begin{tikzcd} \mathscr{C} \arrow[r, "F"] & \mathscr{D} \arrow[r, bend left=50, "J"{name=U}] \arrow[r, bend right=50, "K"'{name=D}] & \mathscr{E} \arrow[from=U, to=D, Rightarrow, "\beta"] \end{tikzcd}$ gives rise to $\begin{tikzcd} \mathscr{C} \arrow[r, bend left=50, "J\circ F"{name=U}] \arrow[r, bend right=50, "K\circ F"'{name=D}] & \mathscr{E} \arrow[from=U, to=D, Rightarrow, "\beta F"] \end{tikzcd}$

where the component of βF at A is the component of β at FA – i.e. $(\beta F)_A = \beta_{FA}$ (which is why the suggestive notation 'β_F' is quite often preferred to 'βF'). Why does this hold? Consider the function $Ff\colon FA \to FB$ in $\mathscr{D}$ (where $f\colon A \to B$ is in $\mathscr{C}$). Now apply the functors J and K, and since β is a natural transformation we get the commuting 'naturality square'

$$\begin{tikzcd}
J(FA) \arrow[r, "J(Ff)"] \arrow[d, "\beta_{FA}"] & J(FB) \arrow[d, "\beta_{FB}"] \\
K(FA) \arrow[r, "K(Ff)"] & K(FB)
\end{tikzcd}$$

and we can read that as giving a natural transformation between $J \circ F$ and $K \circ F$.

Likewise, adding a whisker on the right, the situation $\begin{tikzcd} \mathscr{C} \arrow[r, bend left=50, "F"{name=U}] \arrow[r, bend right=50, "G"'{name=D}] \arrow[from=U, to=D, Rightarrow, "\alpha"] & \mathscr{D} \arrow[r, "J"] & \mathscr{E} \end{tikzcd}$ gives rise to $\begin{tikzcd} \mathscr{C} \arrow[r, bend left=50, "J \circ F"{name=U}] \arrow[r, bend right=50, "J \circ G"'{name=D}] \arrow[from=U, to=D, Rightarrow, "J\alpha"] & \mathscr{E} \end{tikzcd}$

where the component of $J\alpha$ at X is $J(\alpha_X)$.

For future use, by the way, we should note the following mini-result:

Theorem 114. *Whiskering a natural isomorphism yields a natural isomorphism.*

Proof. Retaining the same notation as above, but now taking α and β to be isomorphisms, we saw that 'post-whiskering' α by J to get $J\alpha$ yields a transformation whose components are $J\alpha_X$, and since functors preserve isomorphisms, these components are all isomorphisms, hence so is $J\alpha$. 'Pre-whiskering' β by F to get βF yields a transformation whose components are (some of the) components of β and therefore are isomorphisms, hence again so is βF. $\square$

(a) Second, we can *horizontally compose* two natural transformations in the following way:

We take $\begin{tikzcd} \mathscr{C} \arrow[r, bend left=50, "F"{name=U1}] \arrow[r, bend right=50, "G"'{name=D1}] \arrow[from=U1, to=D1, Rightarrow, "\alpha"] & \mathscr{D} \arrow[r, bend left=50, "J"{name=U2}] \arrow[r, bend right=50, "K"'{name=D2}] \arrow[from=U2, to=D2, Rightarrow, "\beta"] & \mathscr{E} \end{tikzcd}$ and get $\begin{tikzcd} \mathscr{C} \arrow[r, bend left=50, "J \circ F"{name=U}] \arrow[r, bend right=50, "K \circ G"'{name=D}] \arrow[from=U, to=D, Rightarrow, "\beta * \alpha"] & \mathscr{E} \end{tikzcd}$.

How do we define $\beta * \alpha$? Take an arrow $f\colon A \to B$ and form this naturality square

for α: $\begin{tikzcd} FA \arrow[r, "Ff"] \arrow[d, "\alpha_A"] & FB \arrow[d, "\alpha_B"] \\ GA \arrow[r, "Gf"] & GB \end{tikzcd}$. Applying the functor J, $\begin{tikzcd} J(FA) \arrow[r, "J(Ff)"] \arrow[d, "J(\alpha_A)"] & J(FB) \arrow[d, "J(\alpha_B)"] \\ J(GA) \arrow[r, "J(Gf)"] & J(GB) \end{tikzcd}$

also commutes. And since $Gf\colon GA \to GB$ is a map in $\mathscr{D}$, and β is a natural transformation between $\mathscr{D} \overset{J}{\underset{K}{\rightrightarrows}} \mathscr{E}$, we have $\begin{tikzcd} J(GA) \arrow[r, "J(Gf)"] \arrow[d, "\beta_{GA}"] & J(GB) \arrow[d, "\beta_{GB}"] \\ K(GA) \arrow[r, "K(Gf)"] & K(GB) \end{tikzcd}$

commutes. Gluing together those last two commutative diagrams one above the other gives a natural transformation from $J \circ F$ to $K \circ G$, if we set the component of $\beta * \alpha$ at X to be $\beta_{GX} \circ J\alpha_X$.

Three remarks:

(1) That definition for $\beta * \alpha$ looks surprisingly asymmetric. But note that applying J to the initial naturality square for α and then pasting the result above a naturality square for β, we could have similarly applied K to the initial naturality square and pasted the result below another naturality square for β, thus showing that we can alternatively define the natural

transformation $J \circ F$ to $K \circ G$ as having the components $K\alpha_X \circ \beta_{FX}$. So symmetry is restored: we get equivalent accounts which mirror each other.

(2) We can think of whiskering as a special case of the horizontal composition of two natural transformations where one of them is the identity natural transformation. For example
$$\begin{tikzcd} \mathscr{C} \arrow[r, bend left=50, "F"{name=U}] \arrow[r, bend right=50, "G"'{name=D}] \arrow[Rightarrow, from=U, to=D, "\alpha"] & \mathscr{D} \arrow[r, bend left=50, "J"{name=U2}] \arrow[r, bend right=50, "J"'{name=D2}] \arrow[Rightarrow, from=U2, to=D2, "1_J"] & \mathscr{E} \end{tikzcd}$$
produces
$$\begin{tikzcd} \mathscr{C} \arrow[r, bend left=50, "J \circ F"{name=U}] \arrow[r, bend right=50, "J \circ G"'{name=D}] \arrow[Rightarrow, from=U, to=D, "1_J * \alpha"] & \mathscr{E} \end{tikzcd},$$
and the component of $1_J * \alpha$ at X is an identity composed with $J\alpha_X$. So this is the same as taking the left-hand natural transformation and simply whiskering with J on the right.

(3) We could now go on to consider the case of horizontally composing a couple of pairs of vertical compositions – and show that it comes to the same if we construe the resulting diagram as the result of vertically composing a couple of horizontal compositions. But we won't now pause over this, but return to the point if and when we ever need the construction. (Or see Leinster 2014, p. 38.)

27 Functor categories

In this chapter, we highlight the observation that the functors between two categories together with the natural transformations between the functors together give us the data for another sort of category!

27.1 Functor categories defined

We saw in §25.3 that for any functor $F\colon \mathscr{C} \to \mathscr{D}$, there is an identity natural transformation $1_F\colon F \Rightarrow F$.

We saw in §26.2 that given parallel functors F, $G, H\colon \mathscr{C} \to \mathscr{D}$, then if there are natural transformations $\alpha\colon F \Rightarrow G$ and $\beta\colon G \Rightarrow H$ then there is a composite natural transformation $\beta \circ \alpha\colon F \Rightarrow H$. Moreover, it is immediate from the definition of this 'vertical' composition of parallel functors, that composition is associative (that's because the composition of the arrows which are components of a transformation is associative).

So, lo and behold, the following definition must be in good order!

Definition 109. The *functor category* from $\mathscr{C}$ to $\mathscr{D}$, denoted $[\mathscr{C}, \mathscr{D}]$ is the category whose objects are all the (covariant) functors $F\colon \mathscr{C} \to \mathscr{D}$, with the natural transformations between them as arrows. △

The laconic notation here '$[\mathscr{C}, \mathscr{D}]$' is standard. An alternative is '$\mathscr{D}^{\mathscr{C}}$'. (We needn't worry about a category of contravariant functors as we can always talk about a category $[\mathscr{C}^{op}, \mathscr{D}]$ instead.)

We will see many instances of functor categories at work later. But let's pause now for a pair of simple examples:

(1) Recall the discrete category $\overline{2}$, which comprises just two objects $\bullet$ and $\star$ together with their identity arrows. Ask: what is the functor category $[\overline{2}, \mathscr{C}]$?

An object in this category is a functor $F\colon \overline{2} \to \mathscr{C}$, where (i) F_{ob} will send $\bullet$ to some $\mathscr{C}$-object X and send $\star$ to an object Y, and (ii) F_{arw} will map the identity arrows on $\bullet$ and $\star$ to the identity arrows on this X and Y. So (A) there is a simple bijection between such functors F, the objects of $[\overline{2}, \mathscr{C}]$, and pairs of $\mathscr{C}$-objects (X, Y).

What about the arrows of our functor category? By definition, each component of a natural transformation from F to the parallel functor F'

will be a $\mathscr{C}$-arrow between the F-image and the F'-image of some object in $\overline{2}$. And since there are no arrows between those objects in $\overline{2}$ there is no naturality square to impose additional constraints. Therefore (B) a natural transformation from F to F', an arrow of $[\overline{2}, \mathscr{C}]$, is simply any pair of $\mathscr{C}$-arrows $(j \colon X \to X', k \colon Y \to Y')$.

So in sum, by (A) and (B), our new category is (or strictly speaking, is isomorphic to) the product category $\mathscr{C} \times \mathscr{C}$ which we met in §6.2.

(2) Recall now the category 2. Omitting identity arrows, we can diagram this as $\bullet \longrightarrow \star$. Ask: what is the functor category $[2, \mathscr{C}]$?

An object in this category is a functor $F \colon 2 \to \mathscr{C}$, where (i) F_{ob} will send $\bullet$ to some $\mathscr{C}$-object X and send $\star$ to an object Y, and (ii) F_{arw} will map identity arrows to identity arrows and send the unique arrow from $\bullet$ to $\star$ to some $\mathscr{C}$-arrow $f \colon X \to Y$. This time, (A) there is therefore a simple bijection between the objects of $[2, \mathscr{C}]$ and $\mathscr{C}$-arrows.

And what about the arrows in our new category? A natural transformation from F to the parallel functor F' will have as components any two $\mathscr{C}$-arrows, j, k, which makes this a commutative square:

$$\begin{tikzcd} X \arrow[r, "f"] \arrow[d, "j"] & Y \arrow[d, "k"] \\ X' \arrow[r, "f'"] & Y' \end{tikzcd}$$

Thus (B) the arrows of the new category are exactly pairs of $\mathscr{C}$-arrows which make our relevant diagram commute.

So in sum, by (A) and (B), $[2, \mathscr{C}]$ is (or strictly speaking, is isomorphic to) the arrow category $\mathscr{C}^{\to}$ we met in §6.4).

27.2 Functor categories and natural isomorphisms

Suppose $[\mathscr{C}, \mathscr{D}]$ is a functor category. Then there will be isomorphisms in this category, in the usual categorial sense of 'isomorphism' – i.e. arrows which have inverses. Now, how do these isomorphisms in $[\mathscr{C}, \mathscr{D}]$ relate to the natural isomorphisms we defined between $\mathscr{C}$ and $\mathscr{D}$ as we defined them before?

Theorem 115. *The isomorphisms in the functor category $[\mathscr{C}, \mathscr{D}]$ are exactly the natural isomorphisms $\psi \colon F \Rightarrow G$, where $\mathscr{C} \underset{G}{\overset{F}{\rightrightarrows}} \mathscr{D}$.*

Proof. Suppose $\psi \colon F \xRightarrow{\sim} G$ is a natural isomorphism between the parallel functors $F, G \colon \mathscr{C} \to \mathscr{D}$, in the sense of Defn. 106. So for any $f \colon A \to B$, the naturality square

$$\begin{tikzcd} FA \arrow[r, "Ff"] \arrow[d, "\psi_A"] & FB \arrow[d, "\psi_B"] \\ GA \arrow[r, "Gf"] & GB \end{tikzcd}$$

commutes. But if $\psi_B \circ Ff = Gf \circ \psi_A$, then $Ff \circ \psi_A^{-1} = \psi_B^{-1} \circ Gf$ (relying on the fact that the components of ψ have inverses). Which makes this diagram always commute for any $f\colon A \to B$:

$$\begin{tikzcd} GA \arrow[r, "Gf"] \arrow[d, "\psi_A^{-1}"] & GB \arrow[d, "\psi_B^{-1}"] \\ FA \arrow[r, "Ff"] & FB \end{tikzcd}$$

Whence $\psi^{-1}\colon G \xRightarrow{\sim} F$ (where ψ^{-1} is assembled from the components ψ_A^{-1} etc. And trivially $\psi^{-1} \circ \psi = 1_F$ and $\psi \circ \psi^{-1} = 1_G$. Which makes ψ an isomorphism (an arrow with an inverse) in the functor category $[\mathscr{C}, \mathscr{D}]$.

Conversely, suppose the natural transformation $\psi\colon F \Rightarrow G$ has an inverse ψ^{-1} in the category $[\mathscr{C}, \mathscr{D}]$, i.e. $\psi^{-1} \circ \psi = 1_F$, and $\psi \circ \psi^{-1} = 1_G$ But vertical composition of natural transformations is defined component-wise, so this requires for each component that $\psi_X^{-1} \circ \psi_X = 1_{FX}$, $\psi_X \circ \psi_X^{-1} = 1_{GX}$. Therefore each component of ψ has an inverse, so is an isomorphism, and hence ψ is a natural isomorphism. □

27.3 Hom-functors from functor categories

We have now introduced a new kind of category – namely, functor categories $[\mathscr{C}, \mathscr{D}]$ whose objects are the functors from $\mathscr{C}$ to $\mathscr{D}$, and whose arrows are the natural transformations between those functors. As with any other category, there can be functors mapping to and from such categories to other categories. Some of these will later turn out to be of central importance in category theory. We start exploring in the rest of this chapter.

Suppose we have a functor category $[\mathscr{C}, \mathscr{D}]$. Its arrows, by definition, are natural transformations. And the collection of natural transformations from the functor $F\colon \mathscr{C} \to \mathscr{D}$ to $G\colon \mathscr{C} \to \mathscr{D}$, assuming it is set-sized, will be the hom-set $[\mathscr{C}, \mathscr{D}](F, G)$. We will repeatedly meet such hom-sets: it will therefore be handy to have a slightly more memorable alternative notation for them:

Definition 110. '$Nat(F, G)$' will denote the set of natural transformations from F to G (assuming it exists). △

Now, where there are hom-sets, there are hom-functors. Again we introduce some snappier notation for future use:

Definition 111. '$Nat(-, G)$' denotes the contravariant hom-functor $[\mathscr{C}, \mathscr{D}](-, G)\colon [\mathscr{C}, \mathscr{D}] \to \mathsf{Set}$; '$Nat(F, -)$' denotes the covariant hom-functor $[\mathscr{C}, \mathscr{D}](F, -)$. △

Let's pause to consider how such functors work. Take the first of them, for example. We simply apply the definition of a contravariant hom-functor. So $Nat(-, G)$ sends an object in the functor category $[\mathscr{C}, \mathscr{D}]$, i.e. a functor F, to the set $Nat(F, G)$. And it sends an arrow in the functor category, i.e. a natural transformation $\alpha\colon F' \Rightarrow F$, to a set-function from $Nat(F, G)$ to $Nat(F', G)$ – i.e. to the function that sends a natural transformation $\beta\colon F \Rightarrow G$ to $\beta \circ \alpha\colon F' \Rightarrow G$. (Note, if that latter function is indeed to live happily in Set, we must be officially thinking of natural transformations, defined as families of arrows, as themselves properly speaking sets.)

27.4 Evaluation and diagonal functors

(a) Start again with the functor category $[\mathscr{C}, \mathscr{D}]$ and this time also pick an object A in $\mathscr{C}$. Then there is a functor that looks at what is in $[\mathscr{C}, \mathscr{D}]$ and evaluates it at A:

Definition 112. The functor $ev_A\colon [\mathscr{C}, \mathscr{D}] \to \mathscr{D}$ sends a functor $F\colon \mathscr{C} \to \mathscr{D}$ to FA and sends a natural transformation $\alpha\colon F \Rightarrow G$ to $\alpha_A\colon FA \to GA$. △

It is trivial to check that ev_A really is functorial.

(b) Now let's consider a functor which goes in the opposite direction, i.e. one that maps *to* a functor category. We will suppose then that $\mathscr{C}$ is a category, and J is a small category. Then

Definition 113. The functor $\Delta_{\mathsf{J}}\colon \mathscr{C} \to [\mathsf{J}, \mathscr{C}]$ sends an object C to the functor $\Delta_C\colon \mathsf{J} \to \mathscr{C}$ and sends an arrow $f\colon C \to C'$ to the natural transformation from Δ_C to $\Delta_{C'}$ whose every component is simply f again. △

Recall, Δ_C is the constant collapse-to-C functor we first met in §20.2 (F10). To check that Δ_{J} is indeed a functor, the crucial thing is to show the last part of our definition does indeed characterize a natural transformation from Δ_C to $\Delta_{C'}$. For this, we just note that for every $d\colon K \to L$ in J, the required naturality square on the left is in fact none other than the trivially commuting square on the right:

$$\begin{tikzcd}
\Delta_C K \arrow[r, "\Delta_C d"] \arrow[d, "f"] & \Delta_C L \arrow[d, "f"] \\
\Delta_{C'} K \arrow[r, "\Delta_{C'} d"] & \Delta_{C'} L
\end{tikzcd}
\qquad\qquad
\begin{tikzcd}
C \arrow[r, "1_C"] \arrow[d, "f"] & C \arrow[d, "f"] \\
C' \arrow[r, "1_{C'}"] & C'
\end{tikzcd}$$

Such a functor Δ_{J} is often called a diagonal functor. Why? We are generalizing on the case where J is the discrete two-object category $\overline{2}$ with objects $0, 1$. Here, $\Delta_{\overline{2}}$ sends an object C in $\mathscr{C}$ to a functor that sends 0 to C and sends 1 to C. If we think of that latter functor as therefore representing a pair of outcomes (C, C), then the functor $\Delta_{\overline{2}}$ in effect sends C to (C, C). In other words, values of $\Delta_{\overline{2}}$ lie down the diagonal of pairs of $\mathscr{C}$-objects.

(c) Given the functor $\Delta_{\mathsf{J}} \colon \mathscr{C} \to [\mathsf{J}, \mathscr{C}]$, and an object D in $[\mathsf{J}, \mathscr{C}]$ (i.e. a diagram $D \colon \mathsf{J} \to \mathscr{C}$), there will be a comma category $(\Delta_{\mathsf{J}} \downarrow D)$. Applying the definition of such a category at the end of §24.4, we get the following:

(1) An object of $(\Delta_{\mathsf{J}} \downarrow D)$ is a pair of an object C in $\mathscr{C}$, and an arrow $c \colon \Delta_{\mathsf{J}} C \to D$ in $[\mathsf{J}, \mathscr{C}]$, i.e. a natural transformation from Δ_C to D. But the components of such a natural transformation we saw in §26.1 116 are just the legs c_J of a cone over D with vertex C. So an object (C, c) of our category are in effect just a cone $[C, c_J]$ over D, i.e. an object in the category of cones over D.

(2) An arrow of $(\Delta_{\mathsf{J}} \downarrow D)$ from (C, c) to (C', c') is a $\mathscr{C}$-arrow $f \colon C \to C'$ such that $c = c' \circ \Delta_{\mathsf{J}} f$, which says that for each J, $c_J = c'_J \circ f$. Which is just the condition for f to be an arrow between cones $[C, c_J]$ and $[C', c'_J]$ in the category of cones over D in Defn. 64.

Hence $(\Delta_{\mathsf{J}} \downarrow D)$ is just the category of cones over D! Which is neat. We can then say that a cone over D is just an object of the category $(\Delta_{\mathsf{J}} \downarrow D)$; and a limit over D is a terminal object of this category.

It will be no additional surprise to learn that $(D \downarrow \Delta_{\mathsf{J}})$ is the category of cocones under D.

27.5 Cones as natural transformations

(a) Fix on some small category $\mathbf{J}$. Consider the functor category $[\mathbf{J}, \mathscr{C}]$ whose objects are diagrams-as-functors $D \colon \mathbf{J} \to \mathscr{C}$ and whose arrows are natural transformations between such functors.

One particular kind of object in $[\mathbf{J}, \mathscr{C}]$ is a trivial constant functor such as $\Delta_C \colon \mathbf{J} \to \mathscr{C}$, i.e. the functor that sends every object in $\mathbf{J}$ to the object C and every arrow in $\mathbf{J}$ to 1_C.

Now, what would be a natural transformation from Δ_C to another diagram-as-functor D? Applying the definition, it would be a family α of $\mathbf{J}$ arrows $\alpha_J \colon \Delta_C(J) \to D(J)$ indexed by $J \in \mathbf{J}$, i.e. arrows $\alpha_J \colon C \to D(J)$, such that for every $d \colon K \to L$ in $\mathbf{J}$, the square below always commutes in $\mathscr{C}$. Hence, trivially, so does the triangle:

$$\begin{tikzcd} C \arrow[r, "1_C"] \arrow[d, "\alpha_K"'] & C \arrow[d, "\alpha_L"] \\ D(K) \arrow[r, "D(d)"] & D(L) \end{tikzcd} \quad \Rightarrow \quad \begin{tikzcd} & C \arrow[dl, "\alpha_K"'] \arrow[dr, "\alpha_L"] & \\ D(K) \arrow[rr, "D(d)"] & & D(L) \end{tikzcd}$$

But we recognize that! It means that C together with the α_J form a cone over D. And conversely, of course, the arrows α_J in any cone over D with vertex C form a natural transformation $\alpha \colon \Delta_C \Rightarrow D$. So in sum:

Theorem 116. *A cone over $D \colon \mathbf{J} \to \mathscr{C}$ with vertex C comprises C together with a natural transformation from the trivial functor $\Delta_C \colon \mathbf{J} \to \mathscr{C}$ to D.*

If we think of cones the more austere way (i.e. just as a family of arrows – see §15.1), then we can take $Cone(C, D)$, the collection of cones over D with vertex C, to be simply $[\mathbf{J}, \mathscr{C}](\Delta_C, D)$, i.e. the hom-set of arrows in the functor category $[\mathbf{J}, \mathscr{C}]$ from Δ_C to D.

(b) We can think of the functor Δ_C (living as an object in the functor category $[\mathbf{J}, \mathscr{C}]$) as itself the value at the object C of a functor $\Delta \colon \mathscr{C} \to [\mathbf{J}, \mathscr{C}]$.

To be functorial, how must Δ act on an arrow $f \colon C \to D$ in $\mathscr{C}$? It must send f to an arrow, i.e. a natural transformation, from $\Delta_C \colon \mathbf{J} \to \mathscr{C}$ to $\Delta_D \colon \mathbf{J} \to \mathscr{C}$. But just by definition, a natural transformation α from Δ_C to Δ_D is a suite of arrows α_J indexed by objects $J \in \mathbf{J}$ such that for any $j \colon K \to L$ in $\mathbf{J}$, these diagrams commute

$$\begin{tikzcd}
\Delta_C(K) \arrow[r, "\Delta_C(j)"] \arrow[d, "\alpha_K"] & \Delta_C(L) \arrow[d, "\alpha_L"] \\
\Delta_D(K) \arrow[r, "\Delta_D(j)"] & \Delta_D(L)
\end{tikzcd}
\quad \text{i.e.} \quad
\begin{tikzcd}
C \arrow[r, "1_C"] \arrow[d, "\alpha_K"] & C \arrow[d, "\alpha_L"] \\
D \arrow[r, "1_D"] & D
\end{tikzcd}$$

Therefore every component of α must be equal and we'll have to put all of them equal to f (the only arrow from C to D we are given!). The resulting action of Δ on f is easily checked to be functorial.

27.6 Limit functors

(a) Suppose every diagram D of shape J has a limit in $\mathscr{C}$. Then we can define a functor $\underset{\leftarrow \mathsf{J}}{Lim} \colon [\mathsf{J}, \mathscr{C}] \to \mathscr{C}$ which sends a diagram D living in the functor category $[\mathsf{J}, \mathscr{C}]$ to the vertex $\underset{\leftarrow \mathsf{J}}{Lim}\, D$ for some chosen limit cone over D in $\mathscr{C}$.

But note however that we do have to do some choosing here! This functor is not entirely 'naturally' or canonically defined: for recall, in the general case, limits over D are only unique up to isomorphism, so we will indeed have to select a particular limit object $\underset{\leftarrow \mathsf{J}}{Lim}\, D$ to be the value of our functor for input D.

And we need to say more. To get a functor, we now need suitably to define $\underset{\leftarrow \mathsf{J}}{Lim}$'s action on arrows. This must send an arrow in $[\mathsf{J}, \mathscr{C}]$, i.e. a natural transformation $\alpha \colon D \Rightarrow D'$ to an arrow in $\mathscr{C}$ from $\underset{\leftarrow \mathsf{J}}{Lim}\, D$ to $\underset{\leftarrow \mathsf{J}}{Lim}\, D'$. How can it do this in a, well, natural way? By hypothesis there are limit cones over D and D', respectively $[\underset{\leftarrow \mathsf{J}}{Lim}\, D, \pi_J]$ and $[\underset{\leftarrow \mathsf{J}}{Lim}\, D', \pi'_J]$. So now take any arrow $d \colon K \to L$ living in J and consider the following diagram:

$$\begin{tikzcd}
& \underset{\leftarrow \mathsf{J}}{Lim}\, D \arrow[dl, "\pi_K"'] \arrow[dr, "\pi_L"] \arrow[dd, dashed, "u_\alpha"] & \\
D(K) \arrow[rr, "D(d)"] \arrow[dd, "\alpha_K"'] & & D(L) \arrow[dd, "\alpha_L"] \\
& \underset{\leftarrow \mathsf{J}}{Lim}\, D' \arrow[dl, "\pi'_K"'] \arrow[dr, "\pi'_L"] & \\
D'(K) \arrow[rr, "D'(d)"] & & D'(L)
\end{tikzcd}$$

The top triangle commutes (because $[\underset{\leftarrow \mathsf{J}}{Lim}\, D, \pi_J]$ is a limit). The lower square commutes by the naturality of α. Therefore the outer pentangle commutes and so, generalizing over objects J in J, $[\underset{\leftarrow \mathsf{J}}{Lim}\, D, \alpha_J \circ \pi_J]$ is a cone over D'. But then *this* cone must factor uniquely through D''s limit cone $[\underset{\leftarrow \mathsf{J}}{Lim}\, D', \pi'_J]$ via some unique $u_\alpha \colon \underset{\leftarrow \mathsf{J}}{Lim}\, D \to \underset{\leftarrow \mathsf{J}}{Lim}\, D'$. The map $\alpha \mapsto u_\alpha$ is then a plausible candidate for $\underset{\leftarrow \mathsf{J}}{Lim}$'s action on arrows; and indeed this assignment is fairly easily checked to yield a functor.

In summary then:

Definition 114. Assuming every diagram D of shape J has a limit in $\mathscr{C}$, the functor $\underset{\leftarrow \mathsf{J}}{Lim} \colon [\mathsf{J}, \mathscr{C}] \to \mathscr{C}$ (or Lim for brief)

i. sends an object D in $[\mathsf{J}, \mathscr{C}]$ to the vertex $Lim\, D$ of some chosen limit cone $[Lim\, D, \pi_J]$ over D

ii. sends an arrow $\alpha \colon D \Rightarrow D'$ in $[\mathsf{J}, \mathscr{C}]$ to the arrow $u_\alpha \colon Lim\, D \to Lim\, D'$ where for all J in J, $\pi'_J \circ u_\alpha = \alpha_J \circ \pi_J$. △

The diagram above can be recycled, by the way, to show

Theorem 117. *Assuming limits of the relevant shape exist then, if we have a natural isomorphism $D \cong D'$, $Lim\, D \cong Lim\, D'$.*

Proof. Because we now have an natural isomorphism $D \cong D'$, we can show as above both that there is a unique $u \colon Lim\, D \to Lim\, D'$ and symmetrically that there is a unique $u' \colon Lim\, D' \to Lim\, D$. These compose to give us map $u' \circ u \colon Lim\, D \to Lim\, D$ which must be $1_{Lim\, D}$ by the now familiar argument (the limit cone with vertex $Lim\, D$ can factor through itself by both $u' \circ u$ and $1_{Lim\, D}$, but there is by hypothesis only one way for the limit cone to factor through itself). Likewise, $u \circ u' = 1_{Lim\, D'}$. So u is an isomorphism. □

(b) We now remark on the following simple theorem:

Theorem 118. *Suppose that $\mathscr{C}$ has all limits of shape J. Then for any $D\colon \mathsf{J} \to \mathscr{C}$ which the functor $F\colon \mathscr{C} \to \mathscr{D}$ preserves,*

$$(*) \qquad F(\underset{\leftarrow \mathsf{J}}{Lim}\, D) \cong \underset{\leftarrow \mathsf{J}}{Lim}(F \circ D).$$

In brief: F commutes with $\underset{\leftarrow \mathsf{J}}{Lim}$.

Proof. Since $\mathscr{C}$ has all limits of shape J, the limit functor Lim (for short) is well-defined.

Now, if F preserves a limit cone over $D\colon \mathsf{J} \to \mathscr{C}$ with vertex $Lim\, D$, then F sends that limit cone to a limit cone over $F \circ D$ with vertex $F(Lim\, D)$. But that vertex must be isomorphic to the vertex of any other limit cone over $F \circ D$. So in particular it must be isomorphic to whatever has been chosen to be $Lim(F \circ D)$. □

We will have occasion to return to consider the behaviour of limit functors at greater length. For the moment, however, we just recall a slogan from elementary analysis; 'continuous functions commute with limits'. Which explains a bit of standard terminology you might come across:

Definition 115. A functor which commutes with limits of shape J for all small categories J is said to be *continuous*. △

28 Equivalent categories

We defined what it is for categories to be isomorphic in §21.5, and gave a number of examples. However, as we announced at the time, there are cases of categories that surely 'come to just the same' (in some good intuitive sense) but which are not isomorphic. A weaker notion of equivalence of categories turns out to be more useful. It is defined using the notion of a natural transformation, which explains why we have had to wait to now to talk about equivalence.

28.1 The categories Pfn and $\mathsf{Set}_\star$ are 'equivalent'

In the general theory of computation, there is no getting away from the central importance of the notion of a partial function from $\mathbb{N}$ to $\mathbb{N}$ (for example, the function φ_e computed by the e-th Turing machine in a standard enumeration is typically partial).

But how should we treat partial functions in logic? Suppose the partial computable function $\varphi\colon \mathbb{N} \to \mathbb{N}$ takes no value for n (the algorithm defining φ doesn't terminate gracefully for input n). Then the term '$\varphi(n)$' apparently lacks a denotation. *But in standard first-order logic, all terms are assumed to denote.* Two-valued logic requires every sentence to be determinately either true or false and truth-value gaps are not allowed: but a sentence with a non-denoting term, on the standard semantics, will lack a truth-value. What to do?

Historically, there are a number of options on the market for dealing with empty terms in a regimented logical language, and hence for dealing with the partial functions which give rise to them. Here we mention just two. One strategy – due to the greatest nineteenth century logician, Gottlob Frege – is to stipulate that apparently empty terms are in fact not empty at all but denote some special object. Then there are no empty terms and no truth-value gaps, hence we can preserve standard logic. An alternative, less artificial, route forward is to bite the bullet and change our logic to allow non-denoting terms and then cope with the truth-value gaps which come along with them.

In just a bit more detail:

(1) *Frege's proposal* The idea, to repeat, is to provide apparently empty terms a default 'rogue' object for them to denote. Apparently empty terms are only superficially so: they are still genuine referring terms, but with a deviant denotation.

How does this work for functional terms? Well, given what we naively think of as a partial function $\varphi\colon \mathbb{N} \to \mathbb{N}$, we now treat this as officially being a *total* function $f\colon \mathbb{N} \cup \{\star\} \to \mathbb{N} \cup \{\star\}$, where $\star$ is any convenient non-number, and where $f(n) = \varphi(n)$ when $\varphi(n)$ takes a numerical value, and f takes the value $\star$ otherwise. If you like, you can think of $\star$ as coding 'not numerically defined'.

So, on this approach our functions are all total. What we *really* meet in a formalized theory of computation are total functions which are only partially numerical (not all their values are numbers). Because these total functions don't generate non-denoting terms, we can preserve our standard logic without truth-value gaps.

(2) *Logical revisionism* Alternatively, we can bite the bullet and live with truth-value gaps, as we surely already do in informal reasoning.

That means, when we come to adopt an official formalized logic, we'll want one which is free from the assumption that all terms denote; we will adopt a *free logic* for short. We will then have to give new accounts for the logical operators to tell us how they behave when they encounter truth-value gaps – for example, if P is truth-valueless because it contains a non-denoting term, is *not*-P also truth-valueless or is it true because P isn't true?

The details can get a little messy, and this logical revisionism has its costs and complications. But at least in a formalism with a free logic we can take at face value both partial functions and the apparently empty terms they give rise to.

There is a lot more to be said: and we could, for example, consider a third proposal due to Bertrand Russell which eliminates empty terms in a different way to Frege. But we won't continue the story any further now: the debate about the best logical treatment of partial functions is the sort of thing that might grip some philosophically-minded logicians but really seems of very little general mathematical interest.

And that's exactly the point of this section! From a mathematical point of view there surely isn't anything much to choose between logical revisionism and Frege's more artificial but more conservative proposal.

On the small scale, we can think of a world of genuinely *partial* numerical functions $\varphi\colon \mathbb{N} \to \mathbb{N}$ (genuinely partial because not everywhere defined, and hence giving rise to empty terms), or we can equally think of a corresponding world of *total* functions $f\colon \mathbb{N} \cup \{\star\} \to \mathbb{N} \cup \{\star\}$, with $\star \notin \mathbb{N}$, and $f(\star) = \star$. Take your pick! More generally, on the large scale, we can think of sets with partial functions between them, or of corresponding pointed sets (sets with a distinguished object as base point) and base-point preserving total functions between *them*. What's to choose, apart from familiarity? Mathematically, surely both approaches come to the same.

And so back to categories! There is a category Pfn whose objects are sets and whose arrows are (possibly) *partial* functions between them. And there is also the category $\mathsf{Set}_\star$ of pointed sets whose objects are sets with a distinguished base point, and whose arrows are (total) set-functions which preserve base points. We can work equally well in either category. So, putting the upshot of our reflections in this section in categorial terms, we get the following attractive

Desideratum *An account of what it is for two categories to be equivalent should surely count* $\mathsf{Set}_\star$ *and* Pfn *as being so, for mathematically they come to the same.*

28.2 Pfn and $\mathsf{Set}_\star$ are not isomorphic

In §21.5 we saw that some examples of categories which 'come to just the same' are in fact isomorphic. However, we can now show:

Theorem 119. $\mathsf{Set}_\star$ *is not isomorphic to* Pfn.

We can remark that there *is* an obvious functor $F\colon \mathsf{Set}_\star \to \mathsf{Pfn}$. F sends a pointed set (X, x) to the set $X\backslash\{x\}$, and sends a base-point preserving total function $f\colon (X, x) \to (Y, y)$ to the partial function $\varphi\colon X\backslash\{x\} \longrightarrow Y\backslash\{y\}$, where $\varphi(x) = f(x)$ if $f(x) \in Y\backslash\{y\}$, and is undefined otherwise. But, nice though this is, F isn't an isomorphism (it could send distinct (X, x) and (X', x') to the same target object).

Again, there is a whole family of functors from Pfn to $\mathsf{Set}_\star$ which take any set X and add an element not yet in X to give as an expanded set with the new object as a basepoint. Here's a way of doing this in a uniform way without making arbitrary choices for each X. Define $G\colon \mathsf{Pfn} \to \mathsf{Set}_\star$ as sending a X to the pointed set $X_* =_{\text{def}} (X \cup \{X\}, X)$, remembering that in standard set theories $X \notin X$! And then let G send a partial function $\varphi\colon X \to Y$ to the total basepoint-preserving function $f\colon X_* \to Y_*$, where $f(x) = \varphi(x)$ if $\varphi(x)$ is defined and $f(x) = \{y\}$ otherwise. G is a natural choice, but isn't an isomorphism (it isn't surjective on objects).

Still, those observations don't yet rule out there being *some* pair of functors between $\mathsf{Set}_\star$ and Pfn which are mutually inverse. But there can't be any such pair.

Proof. A functor which is an isomorphism from Pfn to $\mathsf{Set}_\star$ must send objects in Pfn one-to-one to objects in $\mathsf{Set}_\star$, and must send isomorphisms to isomorphisms, so should preserve the cardinality of isomorphism classes. But the isomorphism class of the empty set in Pfn has just one member, while there is no one-membered isomorphism class in $\mathsf{Set}_\star$. So there can't be an isomorphism between the categories. □

28.3 Equivalent categories

(a) The last two sections have together shown that there are categories Pfn and $\mathsf{Set}_\star$ which to all intents and purposes are mathematically equivalent but which aren't isomorphic (according to the natural definition of isomorphism for categories).

We did, however, note an obvious choice of functors $F\colon \mathsf{Set}_\star \to \mathsf{Pfn}$ and $G\colon \mathsf{Pfn} \to \mathsf{Set}_\star$. And while GF isn't the identity on $\mathsf{Set}_\star$ it *does* map $\mathsf{Set}_\star$ to itself in a rather natural way (without arbitrary choices).

Reflection on this case suggests the following weakening of the definition of isomorphism between categories:

Definition 116. Categories $\mathscr{C}$ and $\mathscr{D}$ are *equivalent*, in symbols $\mathscr{C} \simeq \mathscr{D}$, iff there are functors $F\colon \mathscr{C} \to \mathscr{D}$ and $G\colon \mathscr{D} \to \mathscr{C}$, together with a pair of natural isomorphisms $\alpha\colon 1_{\mathscr{C}} \Rightarrow GF$ and $\beta\colon FG \Rightarrow 1_{\mathscr{D}}$.

We can now give a direct proof that Pfn and $\mathsf{Set}_\star$ are indeed equivalent in this way (try it!).

But in fact we won't do this. Rather, we'll first prove a result which yields an alternative characterization of equivalence which is often much easier to apply:

Theorem 120. *Assuming a sufficiently strong choice principle, a functor $F\colon \mathscr{C} \to \mathscr{D}$ is part of an equivalence between $\mathscr{C}$ and $\mathscr{D}$ iff F is faithful, full and essentially surjective on objects.*

Proof. First suppose F is part of an equivalence between $\mathscr{C}$ and $\mathscr{D}$, so that there is a functor $G\colon \mathscr{D} \to \mathscr{C}$, where $GF \cong 1_{\mathscr{C}}$ and $FG \cong 1_{\mathscr{D}}$. Then:

(i) Given an arrows $f, g\colon A \to B$ in $\mathscr{C}$, then by hypothesis, the following square commutes for f (α is the required natural isomorphism between the identity functor and the composite GF),

$$\begin{array}{ccc} A & \xrightarrow{\quad f \quad} & B \\ \downarrow{\scriptstyle \alpha_A} & & \downarrow{\scriptstyle \alpha_B} \\ GFA & \xrightarrow{\quad GFf \quad} & GFB \end{array}$$

and hence $\alpha_B^{-1} \circ GFf \circ \alpha_A = f$. And of course $\alpha_B^{-1} \circ GFg \circ \alpha_A = g$. It immediately follows that if $Ff = Fg$ then $f = g$, i.e. F is faithful. A companion argument, interchanging the roles of $\mathscr{C}$ and $\mathscr{D}$, shows that G too is faithful.

(ii) Suppose we are given an arrow $h\colon FA \to FB$, then put $f = \alpha_B^{-1} \circ Gh \circ \alpha_A$. But we know that $f = \alpha_B^{-1} \circ GFf \circ \alpha_A$. So it follows that $GFf = Gh$, and since G is faithful, $h = Ff$. So every such h in $\mathscr{D}$ is the image under F of some arrow in $\mathscr{C}$. So F is full.

(iii) Recall, $F\colon \mathscr{C} \to \mathscr{D}$ is e.s.o. iff for any $D \in \mathscr{D}$ we can find some isomorphic object FC, for $C \in \mathscr{C}$. But we know that our natural isomorphism between $1_{\mathscr{D}}$ and FG means that that there is an isomorphism from D to FGD, so putting $C = GD$ gives the desired result that F is e.s.o.

Now for the argument in the other direction. Suppose, then, that $F\colon \mathscr{C} \to \mathscr{D}$ is faithful, full and e.s.o. We need to construct (iv) a corresponding functor $G\colon \mathscr{D} \to \mathscr{C}$, and then a pair of natural isomorphisms (v) $\beta\colon FG \Rightarrow 1_{\mathscr{D}}$ and (vi) $\alpha\colon 1_{\mathscr{C}} \Rightarrow GF$:

(iv) By hypothesis, F is e.s.o., so by definition every object $D \in \mathscr{D}$ is isomorphic in $\mathscr{D}$ to some object FC, for $C \in \mathscr{C}$. Hence – and here we are invoking an appropriate choice principle – for any given $D \in \mathscr{D}$, we can choose a pair (C, β_D), with $C \in \mathscr{C}$ and $\beta_D\colon FC \to D$ an isomorphism in $\mathscr{D}$. Now define G_{ob} as sending an object $D \in \mathscr{D}$ to the chosen $C \in \mathscr{C}$ (so $GD = C$, and $\beta_D\colon FGD \to D$).

To get a functor, we need the component G_{arw} to act suitably on an arrow $g\colon D \to E$. Now, note

$$FGD \xrightarrow{\beta_D} D \xrightarrow{g} E \xrightarrow{\beta_E^{-1}} FGE$$

and since F is full and faithful, there must be some unique $f\colon GD \to GE$ which F sends to the composite $\beta_E^{-1} \circ g \circ \beta_D$. Put $G_{arw}g = f$.

Claim: *G, with components G_{ob}, G_{arw}, is indeed a functor.* We need to show that G (a) preserves identities and (b) respects composition:

For (a), note that $G1_D = e$ where e is the unique arrow from GD to GD such that $Fe = \beta_D^{-1} \circ 1_D \circ \beta_D = 1_{FGD}$. So $e = 1_{GD}$.

For (b) we need to show that, given $\mathscr{D}$-arrows $g\colon D \to E$ and $h\colon E \to F$, $G(h \circ g) = Gh \circ Gg$. But note that

$$\begin{aligned} FG(h \circ g) = \beta_F^{-1} \circ h \circ g \circ \beta_D &= (\beta_F^{-1} \circ h \circ \beta_E) \circ (\beta_E^{-1} \circ g \circ \beta_D) \\ &= FG(h) \circ FG(g) = F(G(h) \circ G(g)) \end{aligned}$$

Hence, since $FG(h \circ g) = F(G(h) \circ G(g))$ and F is faithful, $G(h \circ g) = G(h) \circ G(g)$, so G is indeed a functor.

(v) By construction, β is natural isomorphism from FG to $1_{\mathscr{D}}$.

(vi) Note next that we have an isomorphism $\beta_{FA}^{-1}\colon FA \to FGFA$. As F is full and faithful, $\beta_{FA}^{-1} = F(\alpha_A)$ for some unique $\alpha_A\colon A \to GFA$. Since F is fully faithful it is conservative, i.e. reflects isomorphisms (by Theorem 87), hence α_A is also an isomorphism. Also, the naturality diagram

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[d, "\alpha_A"] & B \arrow[d, "\alpha_B"] \\ GFA \arrow[r, "GFf"] & GFB \end{tikzcd}$$

always commutes for any arrow $f\colon A \to B$ in $\mathscr{C}$. Why? Because [*needs checking*]

$$\begin{aligned} &F(\alpha_B \circ f) = F\alpha_B \circ Ff = \beta_{FB}^{-1} \circ Ff = \\ &\qquad FGFf \circ \beta_{FA}^{-1} = FGFf \circ F\alpha_A = F(GFf \circ F\alpha_A) \end{aligned}$$

relying on the naturality of β^{-1}. But if $F(\alpha_B \circ f) = F(GFf \circ F\alpha_A)$ then since F is faithful, $\alpha_B \circ f = GFf \circ F\alpha_A$. Hence the α_A are the components of our desired natural isomorphism $\alpha\colon 1_{\mathscr{C}} \Rightarrow GF$.

So we are done! $\square$

Our theorem enables us now to very quickly prove the following equivalence claim without any more hard work:

Theorem 121. $\mathsf{Pfn} \simeq \mathsf{Set}_\star$

Proof. Define the functor $G\colon \mathsf{Pfn} \to \mathsf{Set}_\star$ as before. It sends a set X to a set $X_* =_{\text{def}} X \cup \{X\}$ with basepoint X, and sends a partial function $f\colon X \to Y$ to the total function $f_*\colon X_* \to Y_*$, where for $f_*(x) = f(x)$ if $f(x)$ is defined and $f_*(x) = Y$ otherwise.

G is faithful, as it is easily checked that it sends distinct functions to distinct functions. And it is equally easy to check that G is full, i.e. given any basepoint preserving function between sets X_* and Y_*, there is a partial function f which G sends to it.

But G is essentially surjective on objects. For every pointed set in $\mathsf{Set}_\star$ – i.e. every set which can be thought of as the union of a set X with $\{*\}$ where $*$ is an additional basepoint element (not in X) – is isomorphic in $\mathsf{Set}_\star$ to the set $X \cup \{X\}$ with X as basepoint. Hence G is part of an equivalence between Pfn and $\mathsf{Set}_\star$. $\square$

(b) Now for another example. Recall FinSet is the category of finite sets and functions between them. Let $\mathsf{FinOrdn}$ be its full subcategory containing the empty set and all sets of the form $\{0, 1, 2, \ldots n-1\}$ and all functions between them. It doesn't really matter for present purposes how you think of the natural numbers; but to fix ideas, think of them set-theoretically as von Neumann ordinals, so the objects of $\mathsf{FinOrdn}$ are then the finite ordinals – hence the label for the category. We then have:

Theorem 122. $\mathsf{FinOrdn} \simeq \mathsf{FinSet}$

Proof. $\mathsf{FinOrdn}$ is a full subcategory of FinSet, so the inclusion functor F is fully faithful. F is also essentially surjective on objects: for take any object in FinSet, which is some n-membered set: that is in bijective correspondence (and hence isomorphic in FinSet) with the finite ordinal n. Hence F is part of an equivalence, and $\mathsf{FinOrdn} \simeq \mathsf{FinSet}$. $\square$

How should we regard this last result? We saw that defining equivalence of categories in terms of isomorphism would be too strong, as it rules out our treating Pfn and $\mathsf{Set}_\star$ as in effect equivalent. But now we've seen that defining equivalence of categories as in Defn. 28.3 makes the seemingly very sparse category $\mathsf{FinOrdn}$ equivalent to the seemingly much more abundant FinSet. Is that a strike against the definition of equivalence, showing it to be too weak?

It might help to think of a toy example. Consider the two categories which we can diagram respectively as

$$\begin{tikzcd} \bullet \arrow[loop right] \end{tikzcd} \qquad\qquad \begin{tikzcd} \bullet \arrow[loop left] \arrow[r, shift left] & \star \arrow[l, shift left] \arrow[loop right] \end{tikzcd}$$

On the left, we have the category $\mathsf{1}$; on the right we have a two-object category $\mathsf{2!}$ with arrows in *both* directions between the objects (in addition, of course, to the required identity arrows). These two categories are also equivalent. For the obvious inclusion functor $\mathsf{1} \hookrightarrow \mathsf{2!}$ is full and faithful, and it is trivially essentially surjective on objects as each object in the two-object category is isomorphic to the other.

What this toy example highlights is that our equivalence criterion counts categories as amounting to the same when (putting it very roughly) one is just the same as the other padded out with new objects and just enough arrows to make the new objects isomorphic to some old objects.

But on reflection that's fine. Taking a little bit of the mathematical world and bulking it out with copies of the structures it already contains and isomorphisms between the copies won't, for many (most? nearly all?) purposes, give us a real enrichment. Therefore a criterion of equivalence of categories-as-mathematical-universes that doesn't care about surplus isomorphic copies is what we typically need. Hence the results that $\mathsf{1} \simeq \mathsf{2!}$ and $\mathsf{Finord} \simeq \mathsf{FinSet}$ are arguably welcome features, not bugs, of our account of equivalence.

28.4 Skeletons and evil

(a) Even categories are regarded as being equivalent in an important sense even if one is bulked out with isomorphic extras, shouldn't the usual sort of concern for Bauhaus elegance and lack of redundancy lead us to privilege categories which are as skeletal as possible? Let's say:

Definition 117. The category $\mathscr{S}$ is a *skeleton* of the category $\mathscr{C}$ if $\mathscr{S}$ is a full subcategory of $\mathscr{C}$ which contains exactly one object from each class of isomorphic objects of $\mathscr{C}$. A category is *skeletal* if it is a skeleton of some category.

For a toy example, suppose $\mathscr{C}$ is a category arising from a pre-order – as in §4.4 (C4). Then any skeleton of $\mathscr{C}$ will be a poset category. (Check that!)

Theorem 123. *If $\mathscr{S}$ is a skeleton of the category $\mathscr{C}$ then $\mathscr{S} \simeq \mathscr{C}$.*

Proof. The inclusion functor $\mathscr{S} \hookrightarrow \mathscr{C}$ is fully faithful, and by the definition of $\mathscr{S}$ is essentially surjective on objects. So we can apply Theorem 120. ☐

So how about this for a programme? Take the usual universe of categories. But now slim it down by taking skeletons. Then work with these. And we can now forget bloated non-skeletal categories (and forget too about the notion of equivalence and revert to using the simpler notion of isomorphism, because equivalent skeletal categories are in fact isomorphic). What's not to like?

The trouble is that hardly any categories that occur in the wild (so to speak) are skeletal. And slimming down has to be done by appeal to an axiom of choice. Indeed the following statements are each equivalent to a version of the axiom of choice:

(1) Any category has a skeleton.

(2) A category is equivalent to any of its skeletons

(3) Any two skeletons of a given category are isomorphic.

The choice of a skeleton is usually quite artificial – there typically won't be a canonical choice. So any gain in simplicity from concentrating on skeletal categories would be bought at the cost of having to adopt 'unnatural', non-canonical, choices of skeletons. Given that category theory is supposed to be all about natural patterns already occurring in mathematics, this perhaps isn't going to be such a good trade-off after all.

(b) Categorial notions that are not invariant under equivalence are sometimes said to be 'evil'. So being skeletal is evil. So too is being small:

Theorem 124. *Smallness is not preserved by categorial equivalence.*

In other words, we can have $\mathscr{C}$ a small category, $\mathscr{C} \simeq \mathscr{D}$, yet $\mathscr{D}$ not small. This is a simple corollary of our observation in §28.3 that if we take a category, inflate it by adding lots of objects and just enough arrows to ensure that these objects are isomorphic to the original objects, then the augmented category is equivalent to the one we started with. For an extreme example, start with the one-object category $\mathbf{1}$, i.e. $\bullet \circlearrowleft$ (that's small)! Now add as new objects every set, and as new arrows an identity arrow for each set, and also for every set X a pair of arrows $\bullet \rightleftarrows X$ which composed to give identities. Then we get a new pumped-up category $\mathbf{1}^+$ (which is certainly not small). But $\mathbf{1}^+ \simeq \mathbf{1}$.

If you fuss about evil, you can highlight a neighbouring notion to smallness which evidently is virtuous:

Definition 118. A category is *essentially small* if it is equivalent to category with a set's worth of arrows.

But we aren't going to fuss here.

There is, by the way, a companion positive result

Theorem 125. *Local smallness* is *preserved by categorial equivalence.*

Proof. An equivalence $\mathscr{C} \underset{G}{\overset{F}{\rightleftarrows}} \mathscr{D}$ requires F and G to be full and faithful functors. So in particular, for any $\mathscr{D}$-objects D, D', there are the same number of arrows between them as between the $\mathscr{C}$-objects GD, GD'. So that ensures that if $\mathscr{C}$ has only a set's worth of arrows between any pair of objects, the same goes for $\mathscr{D}$. $\square$

29 The Yoneda embedding

We met hom-functors in Chapter 23: they have nice properties like preserving limits. We introduced natural transformations in Chapter 26. We now put things together and start talking about natural transformations between hom-functors.

This will quickly lead on to a proof of a preliminary, restricted, version of the important Yoneda Lemma, and we discover the related Yoneda embedding. These tell us how to find a category built from functors-into-Set-and-arrows-between-them which looks just like the category we start off with. This seems closely analogous to some classical representation theorems like e.g. Cayley's Theorem which tells us how, starting from any group, we can find a group built specifically from permutations-of-a-set which looks just the given group. So we will say something about this supposed analogy.

29.1 Natural transformations between hom-functors

(a) Take a locally small category $\mathscr{C}$: in fact, in this chapter, we assume all the relevant categories are locally small, so that we can unproblematically talk about the relevant hom-sets and hom-functors. Fix on a $\mathscr{C}$-arrow $f\colon B \to A$, noting the direction of the arrow here. And we now describe how to construct from f a corresponding natural transformation α from the hom-functor $\mathscr{C}(A,–)$ to the hom-functor $\mathscr{C}(B,–)$.

By definition, if α is to be a natural transformation, its components must be such that the following diagram commutes, given any arrow $j\colon X \to Y$:

$$\begin{tikzcd}
\mathscr{C}(A,X) \arrow[r, "\mathscr{C}(A,j)"] \arrow[d, "\alpha_X"] & \mathscr{C}(A,Y) \arrow[d, "\alpha_Y"] \\
\mathscr{C}(B,X) \arrow[r, "\mathscr{C}(B,j)"] & \mathscr{C}(B,Y)
\end{tikzcd}$$

where $\mathscr{C}(C,j)$, you will recall, is the map $j \circ -$ which sends an arrow $h\colon C \to X$ to the arrow $j \circ h\colon C \to Y$.

Suppose then that we set a component $\alpha_Z\colon \mathscr{C}(A,Z) \to \mathscr{C}(B,Z)$ to be the function $- \circ f$ that sends an arrow $k\colon A \to Z$ to the composite $k \circ f\colon B \to Z$ (the only obvious way to use f).

Then our diagram will indeed commute. For going round the top-route takes us from $g\colon A \to X$ to $j \circ g\colon A \to Y$ to $(j \circ g) \circ f\colon B \to Y$; and going round the bottom route takes us from $g\colon A \to X$ to $g \circ f\colon A \to Y$ to $j \circ (g \circ f)\colon B \to Y$.

So in sum, if there is a morphism $f\colon B \to A$, then there is a corresponding natural transformation $\alpha\colon \mathscr{C}(A,–) \Rightarrow \mathscr{C}(B,–)$ with components α_Z as defined.

And note too: if f is an isomorphism, then each component α_Z (i.e. $- \circ f$) has an inverse (i.e. $- \circ f^{-1}$), so is an isomorphism. Therefore the induced transformation α is a natural isomorphism.

To sum up this result and introduce some notation:

Theorem 126. *Suppose $\mathscr{C}$ is a locally small category, and $\mathscr{C}(A,–)$, $\mathscr{C}(B,–)$ are hom-functors (for objects A, B in $\mathscr{C}$). Then, given an arrow $f\colon B \to A$, there exists a corresponding natural transformation $\mathscr{C}(f,–)\colon \mathscr{C}(A,–) \Rightarrow \mathscr{C}(B,–)$, where for each Z, the component $\mathscr{C}(f,–)_Z\colon \mathscr{C}(A,Z) \to \mathscr{C}(B,Z)$ sends an arrow $k\colon A \to Z$ to $k \circ f\colon B \to Z$.*

Furthermore, if f is an isomorphism, then $\mathscr{C}(f,–)$ is a natural isomorphism.

(b) Both as a quick reality-check and for future use, let's pause to show:

Theorem 127. *Given a locally small category $\mathscr{C}$ including objects A, B, C, and arrows $f\colon B \to A$ and $g\colon C \to B$, then*

(1) $\mathscr{C}(f \circ g,–) = \mathscr{C}(g,–) \circ \mathscr{C}(f,–)$.

(2) $\mathscr{C}(f,–)_A 1_A = f$.

(3) $\mathscr{C}(1_A,–) = 1_{\mathscr{C}(A,–)}$.

Proof. (1) $\mathscr{C}(f \circ g,–)_Z$ sends any arrow $e\colon A \to Z$ to $e \circ (f \circ g)$. However, $(\mathscr{C}(f,–)_Z(e) = e \circ f$, so $\mathscr{C}(g,–)_Z(\mathscr{C}(f,–)_Z(e)) = (e \circ f) \circ g$. Which means that $\mathscr{C}(f \circ g,–)$ and $\mathscr{C}(g,–) \circ \mathscr{C}(f,–)$ agree on all components, so are identical natural transformations.

(2) $\mathscr{C}(f,–)_A$ sends any arrow $j\colon A \to A$ to $j \circ f\colon B \to A$. So in particular it sends 1_A to f.

(3) $\mathscr{C}(1_A,–)_Z$ sends any arrow $j\colon A \to Z$ to itself. While $1_{\mathscr{C}(A,–)}$ is the identity arrow on the object $\mathscr{C}(A,–)$ in the functor category $[\mathscr{C}, \mathsf{Set}]$. In other words it is natural transformation from $\mathscr{C}(A,–)$ to itself which in particular sends $j\colon A \to Z$ to itself. Which shows that $\mathscr{C}(1_A,–)$ and $1_{\mathscr{C}(A,–)}$ agree on all components so are identical. □

(c) The obvious next question to ask is: are *all* possible natural transformations between the hom-functors $\mathscr{C}(A,–)$ and $\mathscr{C}(B,–)$ generated from arrows $f\colon B \to A$ in the way described in Theorem 126?

Start from a natural transformation $\alpha\colon \mathscr{C}(A,–) \Rightarrow \mathscr{C}(B,–)$. If α is indeed of the form $\mathscr{C}(f,–)$ for some $f\colon B \to A$, then by the last theorem $\alpha_A 1_A = \mathscr{C}(f,–)_A 1_A = f$. So we already know one candidate for f, and we might naturally conjecture:

Theorem 128. *Suppose $\mathscr{C}$ is a locally small category, and consider the hom-functors $\mathscr{C}(A,–)$ and $\mathscr{C}(B,–)$, for objects A, B in $\mathscr{C}$. Then if there is a natural transformation $\alpha\colon \mathscr{C}(A,–) \Rightarrow \mathscr{C}(B,–)$, there is a unique arrow $f\colon B \to A$ such that $\alpha = \mathscr{C}(f,–)$, namely $f = \alpha_A(1_A)$.*

And this indeed is right:

Proof. Since α is a natural transformation, the following diagram in particular must commute, for any Z and any $g\colon A \to Z$,

$$\begin{tikzcd}
\mathscr{C}(A,A) \arrow[r, "\mathscr{C}(A,g)"] \arrow[d, "\alpha_A"] & \mathscr{C}(A,Z) \arrow[d, "\alpha_Z"] \\
\mathscr{C}(B,A) \arrow[r, "\mathscr{C}(B,g)"] & \mathscr{C}(B,Z)
\end{tikzcd}$$

We start with $\mathscr{C}(A,A)$ at the top left because we know that it is populated, at least by 1_A. Then, recalling the definitions, $\mathscr{C}(A,g)$ is the map that (among other things) sends an arrow $h\colon A \to A$ to the arrow $g \circ h\colon A \to Z$, and $\mathscr{C}(B,g)$ sends an arrow $k\colon B \to A$ to the arrow $g \circ k\colon B \to Z$.

Chase that identity arrow 1_A round the diagram from the top left to bottom right nodes. The top route sends it to $\alpha_Z(g)$. The bottom route sends it to $g \circ (\alpha_A(1_A))$, which equals $\mathscr{C}(\alpha_A(1_A),–)_Z(g)$ (check how we set up the notation in Theorem 126). Since our square always commutes we have

$$\text{for all objects } Z \text{ and arrows } g\colon A \to Z,\ \alpha_Z(g) = \mathscr{C}(\alpha_A(1_A),–)_Z(g).$$

Hence, since Z and g were arbitrary,

$$\alpha = \mathscr{C}(\alpha_A(1_A),–).$$

Putting $f\colon B \to A =_{\text{def}} \alpha_A(1_A)$ therefore proves the existence part of the theorem.

Now suppose both f and f' are such that $\alpha = \mathscr{C}(f,–) = \mathscr{C}(f',–)$. Then by Theorem 127 (2)

$$f = \mathscr{C}(f,–)_A(1_A) = \mathscr{C}(f',–)_A(1_A) = f'$$

which shows f's uniqueness. □

(d) The theorems so far in this section have been about covariant hom-functors. We have corresponding duals for contravariant hom-functors. Here's part of the story (proofs are routine exercises in dualization, paying attention to the direction of arrows):

Theorem 129. *Suppose $\mathscr{C}$ is a locally small category, and $\mathscr{C}(–,A)$, $\mathscr{C}(–,B)$ are contravariant hom-functors (for objects A, B in $\mathscr{C}$). Then (1) if there exists an arrow $f\colon A \to B$, there is a natural transformation $\mathscr{C}(–,f)\colon \mathscr{C}(–,A) \Rightarrow \mathscr{C}(–,B)$,* *where for each Z, the component $\mathscr{C}(-,f)_Z\colon \mathscr{C}(Z,A) \to \mathscr{C}(Z,B)$ sends an arrow $k\colon Z \to A$ to $f \circ k\colon Z \to B$.*

And (2) if there is a natural transformation $\alpha\colon \mathscr{C}(-,A) \Rightarrow \mathscr{C}(-,B)$, there is a unique arrow $f\colon A \to B$ such that $\alpha = \mathscr{C}(-,f)$, namely $f = \alpha_A(1_A)$.

(3) $\mathscr{C}(-,g \circ f) = \mathscr{C}(-,g) \circ \mathscr{C}(-,f)$.

29.2 The Restricted Yoneda Lemma

Sticking to the covariant case for the moment, we have been considering pairs of hom-functors such as $\mathscr{C}(A,–)\colon \mathscr{C} \to \mathsf{Set}$ and $\mathscr{C}(B,–)\colon \mathscr{C} \to \mathsf{Set}$, and the natural transformations between them. Theorem 128 tells us that there are no more such natural transformations than there are $\mathscr{C}$-arrows $f\colon B \to A$. Since we are assuming all along that $\mathscr{C}$ is locally small, that means there can be a set of such natural transformations. It is a hom-set for the functor category $[\mathscr{C}, \mathsf{Set}]$; in the notation of Defn. 110, we can denote it '$Nat(\mathscr{C}(A,–), \mathscr{C}(B,–))$'.

Now, a $\mathscr{C}$-arrow $f\colon B \to A$ is of course a member of the hom-set $\mathscr{C}(B,A)$. So, in the proofs of our Theorems 126 and 128 we have in effect defined two suites of functions $\mathcal{X}_{AB}$ and $\mathcal{E}_{AB}$ in Set (functions indexed by the $\mathscr{C}$-objects A, B), where

i) $\mathcal{X}_{AB}\colon \mathscr{C}(B,A) \to Nat(\mathscr{C}(A,–), \mathscr{C}(B,–))$ sends a function $f\colon B \to A$ to the natural transformation $\mathscr{C}(f,–)$.

ii) $\mathcal{E}_{AB}\colon Nat(\mathscr{C}(A,–), \mathscr{C}(B,–)) \to \mathscr{C}(B,A)$ sends a natural transformation $\alpha\colon \mathscr{C}(A,–) \Rightarrow \mathscr{C}(B,–)$ to $\alpha_A(1_A)$.

And again, the next thing to do is obvious: we check that $\mathcal{X}_{AB}$ and $\mathcal{E}_{AB}$ are inverses of each other in Set as they surely ought to be.

Let's fix on some particular A and B. Then we note:

(1) Given some $f\colon B \to A$,

$$(\mathcal{E}_{AB} \circ \mathcal{X}_{AB})f = \mathcal{E}_{AB}(\mathscr{C}(f,–)) = \mathscr{C}(f,–)_A(1_A) = f$$

with the last identity by Theorem 127 (2). But f was arbitrary. Whence $\mathcal{E}_{AB} \circ \mathcal{X}_{AB} = 1$.

(2) Given some $\alpha\colon \mathscr{C}(A,–) \Rightarrow \mathscr{C}(B,–)$,

$$(\mathcal{X}_{AB} \circ \mathcal{E}_{AB})\alpha = \mathcal{X}_{AB}(\alpha_A(1_A)) = \mathscr{C}(\alpha_A(1_A),–) = \alpha$$

where the last identity is as shown in the proof of Theorem 128. But α was arbitrary. Whence $\mathcal{X}_{AB} \circ \mathcal{E}_{AB} = 1$.

So $\mathcal{X}_{AB}$ and $\mathcal{E}_{AB}$ are mutual inverses, and hence isomorphisms. Therefore we have in summary:

Theorem 130 (Restricted Yoneda Lemma)**.** *Suppose $\mathscr{C}$ is a locally small category, and A, B are objects of $\mathscr{C}$. Then $Nat(\mathscr{C}(A,–), \mathscr{C}(B,–)) \cong \mathscr{C}(B,A)$.*

There is, needless to say, a dual version of all this. For each A, B in $\mathscr{C}$, there is an isomorphism $\mathcal{Y}_{AB} \colon \mathscr{C}(A, B) \to Nat(\mathscr{C}(-, A), \mathscr{C}(-, B))$ which sends a function $f \colon A \to B$ to the natural transformation $\mathscr{C}(-, f)$; and $\mathcal{Y}_{AB}$ has an inverse. Consequently,

Theorem 131 (Restricted Yoneda Lemma, continued)**.** *Suppose $\mathscr{C}$ is a locally small category, and A, B are objects of $\mathscr{C}$. Then $Nat(\mathscr{C}(-, A), \mathscr{C}(-, B)) \cong \mathscr{C}(A, B))$.*

The shared label we've given this dual pair of theorems is not standard, but the reason for it will become clear when we meet the full Yoneda Lemma in Ch. 30.

The future full version has a reputation for being the first result in category theory whose proof takes some real effort to understand. Be that as it may, at least the route up to our current cut-down version should seem entirely unproblematic. A simple observation established Theorem 126, that each $f \colon B \to A$ generates a natural transformation from $\mathscr{C}(A, –)$ to $\mathscr{C}(B, –)$. It was then very natural to ask if there is a converse result, and we get Theorem 128. In proving those simple theorems, we have set up maps each way between members of $\mathscr{C}(B, A)$ and of $Nat(\mathscr{C}(A, –), \mathscr{C}(B, –))$. Checking that those maps are indeed mutually inverse as we might expect gives us the Restricted Yoneda Lemma – which is all we need for the main result in this chapter, and for a number of other results which are often said to obtain 'by Yoneda'.

29.3 The Yoneda embedding

(a) Suppose, as always in this chapter, that the category $\mathscr{C}$ is locally small, then:

(i) we can define a map – let's call it $\mathcal{X}_{ob}$ – that takes any $\mathscr{C}$-object A (equivalently, any $\mathscr{C}^{op}$-object A) and sends it to the corresponding hom-functor $\mathscr{C}(A, –)$.

(ii) we can define another map – let's call it $\mathcal{X}_{arw}$ – that takes any $\mathscr{C}$-arrow $f \colon B \to A$ (equivalently, any $\mathscr{C}^{op}$-arrow $f \colon A \to B$) and sends it to $\mathcal{X}_{AB} f$, i.e. sends f to the natural transformation $\mathscr{C}(f, –) \colon \mathscr{C}(A, –) \Rightarrow \mathscr{C}(B, –)$.

Now, hom-functors like $\mathscr{C}(A, –)$ are objects of the functor category $[\mathscr{C}, \mathsf{Set}]$. And natural transformations like $\mathscr{C}(f, –) \colon \mathscr{C}(A, –) \Rightarrow \mathscr{C}(B, –)$ are arrows in that same category. So, we might hope that, as our labels for them prematurely suggest, the maps $\mathcal{X}_{ob}$ and $\mathcal{X}_{arw}$ can be put together as the components of a covariant functor $\mathcal{X} \colon \mathscr{C}^{op} \to [\mathscr{C}, \mathsf{Set}]$.

To confirm that they can be, we just need to check the two functorial axioms are indeed satisfied. First, identities are preserved:

$$\mathcal{X}(1_A) = \mathscr{C}(1_A, –) = 1_{\mathscr{C}(A,–)} = 1_{\mathcal{X}(A)}$$

where the central equation holds by Theorem 127 (3). And secondly, composition is respected. In other words, for any composable f, g in $\mathscr{C}^{op}$,

$$\mathcal{X}(g \circ^{\mathscr{C}^{op}} f) = \mathcal{X}(f \circ^{\mathscr{C}} g) = \mathscr{C}(f \circ^{\mathscr{C}} g, –) = \mathscr{C}(g, –) \circ^{[]} \mathscr{C}(f, –) = \mathcal{X}(g) \circ^{[]} \mathcal{X}(f)$$

where '$\circ^{[]}$' indicates composition in the functor category $[\mathscr{C}, \mathsf{Set}]$, and the third equation holds by Theorem 127 (1).

Let's summarize this important result, again along with its obvious dual companion where we similarly define a functor $\mathcal{Y}$ in terms of the maps $\mathcal{Y}_{AB}$:

Theorem 132. *For any locally small category $\mathscr{C}$, there is a functor we'll label simply $\mathcal{X}\colon \mathscr{C}^{op} \to [\mathscr{C}, \mathsf{Set}]$ with components $\mathcal{X}_{ob}$ and $\mathcal{X}_{arw}$ such that*

(1) for any $A \in ob(\mathscr{C}^{op})$, $\mathcal{X}_{ob}(A) = \mathscr{C}(A, –)$,

(2) for any arrow $f \in \mathscr{C}^{op}(A, B)$, i.e. arrow $f\colon B \to A$ in $\mathscr{C}$, $\mathcal{X}_{arw}(f) = \mathscr{C}(f, –)$.

And there is similarly a functor $\mathcal{Y}\colon \mathscr{C} \to [\mathscr{C}^{op}, \mathsf{Set}]$ with components $\mathcal{Y}_{ob}$ and $\mathcal{Y}_{arw}$ such that

(3) for any $A \in ob(\mathscr{C})$, $\mathcal{Y}_{ob}(A) = \mathscr{C}(–, A)$.

(4) For any arrow $f\colon A \to B$ in $\mathscr{C}$, $\mathcal{Y}_{arw}(f) = \mathscr{C}(–, f)$.

(b) It is immediate that the functors $\mathcal{X}$ and $\mathcal{Y}$ behave nicely in various ways. In particular:

Theorem 133. *$\mathcal{X}\colon \mathscr{C}^{op} \to [\mathscr{C}, \mathsf{Set}]$ and $\mathcal{Y}\colon \mathscr{C} \to [\mathscr{C}^{op}, \mathsf{Set}]$ are fully faithful functors which are injective on objects.*

Proof. By definition, $\mathcal{X}\colon \mathscr{C}^{op} \to [\mathscr{C}, \mathsf{Set}]$ is full just in case, for any $\mathscr{C}^{op}$-objects A, A', and any natural transformation $\alpha\colon \mathscr{C}(A,–) \to \mathscr{C}(A',–)$ there is an arrow $f\colon A \to A'$ in $\mathscr{C}^{op}$, i.e. an arrow $f\colon A' \to A$ in $\mathscr{C}$, such that $\alpha = \mathcal{X}f = \mathscr{C}(f,–)$. Which we have already proved as the existence claim in Theorem 128.

By definition, $\mathcal{X}\colon \mathscr{C}^{op} \to [\mathscr{C}, \mathsf{Set}]$ is faithful just in case, for any $\mathscr{C}^{op}$-objects A, A', and any pair of arrows $f, g\colon A \to A'$ in $\mathscr{C}^{op}$, i.e. any pair of arrows $f, g\colon A' \to A$ in $\mathscr{C}$, then if $\mathscr{C}(f,–) = \mathscr{C}(g,–)$ then $f = g$. But that follows immediately from the uniqueness claim in Theorem 128.

So the only new claim is that $\mathcal{X}$ is injective on objects, meaning that if $A \neq B$, then $\mathcal{X}(A) \neq \mathcal{X}(B)$. Suppose $\mathcal{X}(A) = \mathcal{X}(B)$, i.e. $\mathscr{C}(A,–) = \mathscr{C}(B,–)$. Then $\mathscr{C}(A,–)(C) = \mathscr{C}(B,–)(C)$, i.e. $\mathscr{C}(A,C) = \mathscr{C}(B,C)$. But that can't be so if $A \neq B$, since by our lights hom-sets on different pairs of objects must be disjoint (see the last sentence of §21.6).

The proof for $\mathcal{Y}\colon \mathscr{C} \to [\mathscr{C}^{op}, \mathsf{Set}]$ is straightforwardly dual. $\square$

As an important corollary, we now have

Theorem 134. *For any objects A, B in the locally small category $\mathscr{C}$, $A \cong B$ iff $\mathcal{X}A \cong \mathcal{X}B$, and likewise $A \cong B$ iff $\mathcal{Y}A \cong \mathcal{Y}B$.*

Proof. Suppose $A \cong B$. Then there is an isomorphism $f\colon B \xrightarrow{\sim} A$. So there is a natural transformation $\mathscr{C}(f,–)\colon \mathscr{C}(A,–) \Rightarrow \mathscr{C}(B,–)$, which by Theorem 126 is an isomorphism. So in our alternative notation, $\mathcal{X}f\colon \mathcal{X}A \overset{\sim}{\Longrightarrow} \mathcal{X}B$. Hence $\mathcal{X}A \cong \mathcal{X}B$.

Now suppose $\mathcal{X}A \cong \mathcal{X}B$. So there exists a natural isomorphism $\alpha\colon \mathscr{C}(A,-) \xRightarrow{\sim} \mathscr{C}(B,-)$. By Theorem 128, α is $\mathscr{C}(f,-)$ for some $f\colon B \to A$, i.e. is $\mathcal{X}f$. But $\mathcal{X}$ is fully faithful. So Theorem 87 tells us that since $\mathcal{X}f$ is an isomorphism, so is f. Hence $A \cong B$.

That shows $A \cong B$ iff $\mathcal{X}A \cong \mathcal{X}B$. The argument for the functor $\mathcal{Y}$ is dual. $\square$

(c) So the situation is this. The functor $\mathcal{Y}$, for example, injects a copy of the $\mathscr{C}$-objects one-to-one into the objects of the functor category $[\mathscr{C}^{op}, \mathsf{Set}]$; and then it fully and faithfully matches up the arrows between $\mathscr{C}$-objects with arrows between the corresponding objects in $[\mathscr{C}^{op}, \mathsf{Set}]$. In other words, $\mathcal{Y}$ *yields an isomorphic copy of* $\mathscr{C}$ *sitting inside the functor category as a full sub-category.*

So, in a phrase, $\mathcal{Y}$ *embeds* a copy of $\mathscr{C}$ in $[\mathscr{C}^{op}, \mathsf{Set}]$. Hence the terminology (in honour of its discoverer):

Definition 119. The full and faithful functor $\mathcal{Y}\colon \mathscr{C} \to [\mathscr{C}^{op}, \mathsf{Set}]$ is the *Yoneda embedding* of $\mathscr{C}$. $\triangle$

There was a reason, then, behind our use of '$\mathcal{Y}$' for this functor! And indeed the '$\mathcal{Y}$' notation – in upper or lower case, in one font or another – is pretty standard for the Yoneda embedding. However, '$\mathcal{X}$' is just our label for the dual embedding, which doesn't seem to have a standard name or notation, though we can usefully call it a Yoneda embedding too.

29.4 Yoneda meets Cayley

(a) Take any locally small category you like. Then the Yoneda embedding tells us how to find a category built from functors-into-Set-and-arrows-between-them which looks just like the category we started off with. Now, as we remarked in the preamble at the beginning of this chapter, this is surely reminiscent of some classical representation theorems which tell us how, given a mathematical structure of a certain type, we can find another structure which lives in the universe of sets and is isomorphic to it. At the simple end of the spectrum there is an observation that we can attribute to Dedekind: any given partially ordered objects are isomorphic to certain corresponding sets ordered by set-inclusion. A significantly more sophisticated result of the same flavour is the Stone Representation Theorem: any Boolean algebra is isomorphic to a field of sets (where a field of sets is a sub-algebra of a canonical power-set alegbra $(\mathscr{P}(X), \overline{}, \cap, \cup, \varnothing, X)$, where X is some set and of course $\overline{A}$ is $X - A$). Here, though, we'll concentrate on just one such classical representation theorem, namely Cayley's Theorem:

Theorem 135. *Any group* $(G, \cdot, e)$ *is isomorphic to a subgroup of the group* $Sym(G)$*, i.e. the group of permutations on the set* G.

Proof. (The usual one, rehearsed here in case you haven't seen it before, and to fix notation). Given any object $g \in G$, we define the set-function $\underline{g}\colon G \to G$ by setting $\underline{g}(x) = g \cdot x$ (i.e. $\underline{g} = \{(x,y) \mid x, y \in G \wedge y = g \cdot x\}$).

Evidently any such $\underline{g}$ is surjective: for any $x \in G$, there's an object which $\underline{g}$ sends to x, namely $g^{-1} \cdot x$. And if $\underline{g}(x) = \underline{g}(y)$, then $g \cdot x = g \cdot y$ whence $g^{-1} \cdot g \cdot x = g^{-1} \cdot g \cdot y$, therefore $x = y$. Hence $\underline{g}$ is also injective and is therefore a bijection on G, i.e. is a permutation of the group objects.

Put $K = \{\underline{g} \mid g \in G\}$ It is now routine to confirm $(K, \circ, \underline{e})$ is a group, and hence a subgroup of $Sym(G)$, where the group operation is composition of functions:

i. Any two functions $\underline{f}$, $\underline{g}$ have a product $\underline{f} \circ \underline{g}$, where $(\underline{f} \circ \underline{g})(x) = f \cdot g \cdot x$.
ii. The function $\underline{e}$ is a group identity.
iii. $\underline{f} \circ (\underline{g} \circ \underline{h}) = (\underline{f} \circ \underline{g}) \circ \underline{h}$ because $f \cdot (g \cdot h) = (f \cdot g) \cdot h$.
iv. We note that $(\underline{g^{-1}} \circ \underline{g})(x) = \underline{g^{-1}}(g \cdot x) = g^{-1} \cdot g \cdot x = x = \underline{e}(x)$. So $\underline{g^{-1}} \circ \underline{g} = \underline{e}$, and similarly $\underline{g} \circ \underline{g^{-1}} = \underline{e}$. So each $\underline{g}$ has an inverse.

It remains to check that the map F defined by $g \mapsto \overline{g}$ is a group isomorphism from $(G, \cdot, e) \to (K, \circ, \underline{e})$. F is injective. For if $\underline{f} = \underline{g}$, then $\underline{f}(e) = \underline{g}(e)$, so $f \cdot e = g \cdot e$, so $f = g$. Since F is also a surjection just by the definition of K, F (as a map on the carrier sets) is an isomorphism.

Also, for any x, $F(f \cdot g)(x) = \underline{(f \cdot g)}(x) = f \cdot g \cdot x = \underline{f}(g \cdot x) = \underline{f}(\underline{g}(x)) = (\underline{f} \circ \underline{g})(x) = (Ff \circ Fg)(x)$, so F indeed respects group structure. □

(b) Now, the modern way is – at least officially – to think of a group $(G, \cdot, e)$ as a set-theoretic structure from the outset; so Cayley's theorem might seem just to tell us that, given one set theoretic structure, we can find another isomorphic one. Big deal! However, that rather disguises what's actually going on.

For various reasons – some good, some rather disreputable – it has become absolutely standard in mathematics to trade in a lot of plural talk (referring to many objects at once) for singular talk (referring to a set of those many objects). For example, we've learnt to slide easily e.g. from talk of the natural numbers (plural) to talk of the set $\mathbb{N}$ (singular). So instead of stating the Least Number Principle as e.g. 'Given any natural numbers, one of them will be the least' we say 'Any set S, where $S \subseteq \mathbb{N}$, has a least member'. But note that the singular talk about a set here is not yet doing any real work. And indeed, quite a lot of informal set talk is in fact similarly low-level, non-committal stuff which can however be readily translated away, most naturally into a plural idiom. That applies here, to part of the statement of Cayley's Theorem. Instead of starting 'Any group $(G, \cdot, e)$, ...' and thinking of this as already referring to a set-theoretic object (e.g. an ordered triple of a set, a set-function and a set-member), we can capture the core of the theorem like this:

> Suppose we are given some objects and a group operation on them with a unit for that operation. Then there will always also be some *sets* (in particular, some set-functions) with a group structure on *them* which form a group isomorphic to the one we started with.

Put this way, stripped of one layer of unnecessary set-idiom, we have (in an intuitive sense) a 'cross-category' result which says that objects with a group structure on them (whatever objects they are) can always be represented by an isomorphic structure living in the world of sets.

(c) Recall from §7.2 that a group can be considered as a category in its own right, a one-object category all of whose arrows are isomorphisms. If we take a group $(G, \cdot, e)$ then the corresponding category $\mathscr{G}$ has the following data:

(i) the sole object of $\mathscr{G}$: choose whatever object you like, and dub it '$\star$'.

(ii) the arrows of $\mathscr{G}$ are the *elements* of the group $(G, \cdot, e)$.

(iii) the identity arrow $1_\star$ of $\mathscr{G}$ is the identity element e of the group G.

(iv) the composite $g \circ f \colon \star \to \star$ of the two arrows $g, f \colon \star \to \star$ is just $g \cdot f$.

Moreover, $\mathscr{G}$ is locally small since its sole potential hom-set $\mathscr{G}(\star, \star)$ is none other than G, which we assume is indeed set-sized.

We can therefore apply the Restricted Yoneda Lemma in one version or the other. And there's only one possible application of each version. Consider then the version which tells us that

$$Nat(\mathscr{G}(-, \star), \mathscr{G}(-, \star)) \cong \mathscr{G}(\star, \star).$$

So: what are the natural transformations $\alpha \colon \mathscr{G}(-, \star) \Rightarrow \mathscr{G}(-, \star)$? We can apply Theorem 129: every such α is $\mathscr{G}(-, g)$ for some arrow g in $\mathscr{G}$.

Now, by definition, $\mathscr{G}(-, g)$ sends an arrow $x \colon \star \to \star$ to $g \circ x \colon \star \to \star$. But $\mathscr{G}(\star, \star)$ is just G, and arrows are G-elements, so $\mathscr{G}(-, g)$ acts on G by sending an element x to the element $g \cdot x$. Hence $\mathscr{G}(-, g)$ is the function we earlier called $\underline{g}$. As before, that's a bijective map on G, i.e. a permutation on G.

Therefore the Restricted Yoneda Lemma tells us that some set of permutations on the set G is in bijection with the members of G.

Moreover, our proof of the Lemma gives us the isomorphism $\mathcal{Y}$, which sends the arrow $g \colon \star \to \star$ to $\mathscr{G}(-, g)$. By Theorem 129,

$$\mathcal{Y}(g \cdot g') = \mathscr{G}(-, g \cdot g') = \mathscr{G}(-, g) \circ \mathscr{G}(-, g') = \mathcal{Y}(g) \circ \mathcal{Y}(g').$$

So if as before we put a group structure on the natural transformations $\mathscr{G}(-, g)$, i.e. the functions $\underline{g}$, by again defining multiplication as composition, our isomorphism $\mathcal{Y}$ preserves group structure.

So in short, we can more or less immediately read off from the proof of the Restricted Yoneda Lemma that a group $(G, \cdot, e)$ is isomorphic to a group of permutations on G with composition as the group operation.

Which is why it is often said that the Yoneda Lemma is a generalization of Cayley's Theorem.

30 The Yoneda Lemma

In Chapter 29 we showed that a couple of easy preliminary theorems were enough to establish what we called the Restricted Yoneda Lemma, and also that the Yoneda embedding is indeed an embedding. For many purposes, this is all we need to know about Yoneda. Still, talking about a Restricted Lemma invites an obvious question: what's the full-power *unrestricted* Yoneda Lemma? This chapter explains.

30.1 Towards the full Yoneda Lemma

Let F be the functor $\mathscr{C}(B, -)$. Then one half of the Restricted version of the Yoneda Lemma, Theorem 130, tells us that there is an isomorphism between $Nat(\mathscr{C}(A, -), F)$ and FA. The other half of the Restricted Lemma is of course the dual, but for the moment we'll let it look after itself.

Now, to get from where we are to the Yoneda Lemma proper we need two steps:

(1) We look again at the ingredients of the proof of the restricted version and ask 'Where did we essentially depend on the fact that the second functor, now notated simply 'F', actually was a hom-functor $\mathscr{C}(B, -)$ for some B?' Close inspection reveals that we didn't. So we in fact have the more general result that for any locally small category $\mathscr{C}$, *any* functor $F\colon \mathscr{C} \to \mathsf{Set}$, and any $\mathscr{C}$-object A, there is an isomorphism $\mathcal{E}$ between $Nat(\mathscr{C}(A, -), F)$ and FA.

(2) Next we note that our proof of this generalization (like the proof of the original Restricted Lemma) provides a *general recipe* for constructing the required isomorphism. Take a locally small category $\mathscr{C}$ and any $\mathscr{C}$-object A, then, without having to invoke any arbitrary choices, our proof fixes inverse isomorphisms $\mathcal{X}_{AF}$ and $\mathcal{E}_{AF}$ between $Nat(\mathscr{C}(A, -), F)$ and FA. In an intuitive sense, we've constructed a *natural* isomorphism. And so we should be able to show that there is a *natural isomorphism* in the official, categorial, sense between some relevant functors.

In sum, we will get from the Restricted Yoneda Lemma to the full-dress Yoneda Lemma by generalizing a construction, and then recasting in category-theoretic terms an intuitive judgement of the naturality of our construction. Neither step involves anything conceptually very difficult: we just need to nail down all the details. (Some of these proof details are fiddly. By all means skim over them on a first reading, since they are just a matter of checking that the announced steps do go through.)

30.2 The generalizing move

We continue working in a locally small category $\mathscr{C}$. Let's restate some of what we already know, still using 'F' to abbreviate '$\mathscr{C}(B,-)$':

(i) There is a bijection between arrows in FA and natural transformations $\mathscr{C}(A,-) \Rightarrow F$, which sends f in FA to the transformation whose Z-component maps an arrow $g\colon A \to Z$ to $g \circ f\colon B \to Z$.

(ii) By definition, the functor F maps an arrow $g\colon A \to Z$ to a function Fg which sends an arrow $f\colon B \to A$ to the arrow $g \circ f\colon B \to Z$. In other words, $Fg(f) = g \circ f$.

(iii) Hence, putting (i) and (ii) together, we have: there's a bijection which sends an element f in FA to the natural transformation whose Z-component maps $g\colon A \to Z$ to $Fg(f)$.

We next want to redeploy this last idea to prove the following generalization of the Restricted Lemma (where we now free up the interpretation of F to allow it to be any functor from $\mathscr{C}$ to Set):

Theorem 136. *For any locally small category $\mathscr{C}$, object $A \in \mathscr{C}$ and functor $F\colon \mathscr{C} \to \mathsf{Set}$, $Nat(\mathscr{C}(A,-),F) \cong FA$.*

Proof. Following the constructions in the proof leading up to Restricted Lemma, Theorem 130, first we generalize on $\mathcal{X}_{AB}$ and we'll introduce a map we'll call $\mathcal{X}_{AF}$:

(1) $\mathcal{X}_{AF}$ sends f in FA to a natural transformation $\chi = \mathcal{X}_{AF}f\colon\ \mathcal{C}(A,-) \Rightarrow F$. We define χ by requiring its Z-component to be the map which takes $g\colon A \to Z$ to $Fg(f)$.

We had better pause to check that this definition indeed defines a natural transformation. But that's easy. For χ is a natural transformation if the following square commutes for any $u\colon Z \to Z'$:

$$\begin{tikzcd}
\mathscr{C}(A,Z) \arrow[r, "\mathscr{C}(A,u)"] \arrow[d, "\chi_Z"] & \mathscr{C}(A,Z') \arrow[d, "\chi_{Z'}"] \\
FZ \arrow[r, "Fu"] & FZ'
\end{tikzcd}$$

The upper route takes some $j\colon A \to Z$ to $u \circ j$ to $F(u \circ j)(f)$. The lower route takes j to $Fj(f)$ to $Fu \circ Fj(f)$. The functoriality of F ensures these are equal.

Now, to prove our theorem, we show that $\mathcal{X}_{AF}$ is an isomorphism by providing it with a two-sided inverse. Again, we follow the pattern in the proof of the Restricted Lemma, this time generalizing on $\mathcal{E}_{AB}$. So we introduce a map we'll call $\mathcal{E}_{AF}$:

(2) $\mathcal{E}_{AF}$ sends a natural transformation $\alpha\colon \mathcal{C}(A,–) \Rightarrow F$ to the element $\alpha_A(1_A)$ in FA.

And now we check that $\mathcal{E}_{AF}$ is indeed a two-sided inverse of $\mathcal{X}_{AF}$.

First, given an arbitrary element f in FA,

$$\mathcal{E}_{AF} \circ \mathcal{X}_{AF}(f) = \mathcal{E}_{AF} \circ \chi = \chi_A(1_A) = F1_A(f) = 1_{FA}(f)$$

and therefore $\mathcal{E}_{AF} \circ \mathcal{X}_{AF} = 1$.

Secondly, for $\alpha\colon \mathscr{C}(A,–) \Rightarrow F$, we have $\mathcal{X}_{AF} \circ \mathcal{E}_{AF}(\alpha) = \mathcal{X}_{AF}(\alpha_A(1_A))$. The Z-component of that sends a map $g\colon A \to Z$ to $Fg(\alpha_A(1_A))$. But since α is a natural transformation, this next diagram must commute:

$$\begin{array}{ccc} \mathscr{C}(A,A) & \xrightarrow{\mathscr{C}(A,g)} & \mathscr{C}(A,Z) \\ \downarrow{\scriptstyle \alpha_A} & & \downarrow{\scriptstyle \alpha_Z} \\ FA & \xrightarrow{\quad Fg \quad} & FZ \end{array}$$

So chasing the arrow 1_A round the diagram by each route, we get $Fg(\alpha_A(1_A)) = \alpha_Z(\mathscr{C}(A,g)(1_A)) = \alpha_Z(g)$.

In other words, for any given Z, the Z-component of $\mathcal{X}_{AF} \circ \mathcal{E}_{AF}(\alpha)$ acts on g just like the Z-component of α. Hence $\mathcal{X}_{AF} \circ \mathcal{E}_{AF}(\alpha) = \alpha$ and, since α too was arbitrary, $\mathcal{X}_{AF} \circ \mathcal{E}_{AF} = 1$. (Reality check: what object is that last identity arrow on?). □

30.3 Making it all natural

One further step takes us to the full Yoneda Lemma. Not only is there an isomorphism $\mathcal{E}_{AF}$ from $Nat(\mathscr{C}(A,–), F)$ to FA, but $\mathcal{E}_{AF}$ is intuitively 'natural' in the sense of constructed in a uniform way given A and F, without arbitrary choices. We now want to capture this intuitive remark using our official categorial account of a natural isomorphism.

Here's a reminder:

Definition 107 Given functors $F, G\colon \mathscr{C} \to \mathscr{D}$, we say that $FA \cong GA$ *naturally in A* just if F and G are *naturally* isomorphic.

And what we want to prove first, keeping F fixed, is that $Nat(\mathscr{C}(A,–), F) \cong FA$ naturally in A. Which, by our definition, means we have to establish that the functor $Nat(\mathscr{C}(\,\cdot\,,–), F)$ (using the dot as a place-holder marking where we have abstracted from A) is naturally isomorphic to F. The first functor is in fact just the composite functor [*check this*]

$$\mathscr{C}^{op} \xrightarrow{\mathcal{X}} [\mathscr{C}, \mathsf{Set}] \xrightarrow{Nat(-,F)} \mathsf{Set}$$

where $\mathcal{X}$ is as in Theorem 132, and $Nat(-, F)$ is the sort of contravariant functor we met in Defn. 111. Note, since $\mathcal{X}$ can also be thought of as a contravariant functor from $\mathscr{C}$ and contravariant functors compose to give a covariant functor, we do indeed end up with a covariant functor from $\mathscr{C}$!

So we want to show the following:

Theorem 137. *Let $\mathscr{C}$ be a locally small category, and F a functor $F\colon \mathscr{C} \to \mathsf{Set}$. Then the functors $N = Nat(-, F) \circ \mathcal{X}$ and F are naturally isomorphic.*

Proof. Working through the definition of N

(i) N sends any $\mathscr{C}$-object A to the set $Nat(\mathscr{C}(A,-), F)$.

(ii) N sends any $\mathscr{C}$-arrow $f\colon A \to B$ to an arrow between $Nat(\mathscr{C}(A,-), F)$ and $Nat(\mathscr{C}(B,-), F)$, namely the arrow that sends any $\alpha\colon \mathscr{C}(A,-) \Rightarrow F$ to the corresponding $\alpha \circ \mathscr{C}(f,-)\colon \mathscr{C}(B,-) \Rightarrow F$.

So now, given any $f\colon A \to B$, consider the following diagram,

$$\begin{tikzcd}
Nat(\mathscr{C}(A,-), F) \arrow[r, "Nf"] \arrow[d, "\mathcal{E}_{AF}"] & Nat(\mathscr{C}(B,-), F) \arrow[d, "\mathcal{E}_{BF}"] \\
FA \arrow[r, "F(f)"] & F(B)
\end{tikzcd}$$

Take any $\alpha\colon \mathscr{C}(A,-) \Rightarrow F$. Then we have:

(1) $\mathcal{E}_{BF} \circ Nf(\alpha) = \mathcal{E}_{BF}(\alpha \circ \mathscr{C}(f,-)) = (\alpha \circ \mathscr{C}(f,-))_B(1_B) = \alpha_B \circ \mathscr{C}(f,-)_B(1_B) = \alpha_B(f)$ (for the last equation, compare the end of the proof of Theorem 128).

(2) But also $F(f) \circ \mathcal{E}_{AF}(\alpha) = F(f)(\alpha_A(1_A)) = \alpha_B \circ \mathscr{C}(A, f)(1_A) = \alpha_B(f)$ (for the middle equation we note that $F(f) \circ \alpha_A = \alpha_B \circ \mathscr{C}(A, f)$ by a naturality square for α).

So our diagram will always commute, and hence there is a natural isomorphism $\mathcal{E}_F\colon N \Rightarrow F$ with components $(\mathcal{E}_F)_A = \mathcal{E}_{AF}$ for each $A \in \mathscr{C}$, and our theorem is proved. $\square$

That captures in categorial terms the intuition that the construction of $\mathcal{E}_{AF}$ depends in a natural way on A; now for the companion intuition that it depends in a natural way on F too.

Keeping A fixed, we want to prove $Nat(\mathscr{C}(A,-), F) \cong FA$ naturally in F. This means showing the following:

Theorem 138. *Let $\mathscr{C}$ be a locally small category. Then $Nat(\mathscr{C}(A,-),-)$ and ev_A are naturally isomorphic.*

Here $Nat(\mathscr{C}(A,-),-)$ is a covariant hom-functor of the kind we met in Defn. 111, and ev_A is the evaluation-at-A functor which sends F to FA and which we met in Defn. 112.

Proof. Given any $\gamma\colon F \Rightarrow G$, consider the following diagram,

$$\begin{tikzcd}
Nat(\mathscr{C}(A,-),F) \arrow[r, "{Nat(\mathscr{C}(A,-),\gamma)}"] \arrow[d, "\mathcal{E}_{AF}"] & Nat(\mathscr{C}(A,-),G) \arrow[d, "\mathcal{E}_{AG}"] \\
ev_A(F) = FA \arrow[r, "ev_A(\gamma)"] & ev_A(G) = GA
\end{tikzcd}$$

Take any $\alpha\colon \mathscr{C}(A,-) \Rightarrow F$, and recall that $Nat(\mathscr{C}(A,-),\gamma)$ sends α to $\gamma \circ \alpha$. Then we have [*check this!*]

(1) $\mathcal{E}_{AG} \circ Nat(\mathscr{C}(A,-),\gamma)(\alpha) = \mathcal{E}_{AG}(\gamma \circ \alpha) = (\gamma \circ \alpha)_A(1_A) = \gamma_A(\alpha_A(1_A))$.

(2) But also $ev_A(\gamma) \circ \mathcal{E}_{AF}(\alpha) = \gamma_A(\alpha_A(1_A))$.

Hence the diagram always commutes. Therefore there is a natural isomorphism $\mathcal{E}_A\colon K \Rightarrow ev_A$ with components $(\mathcal{E}_A)_F = \mathcal{E}_{AF}$ for each $F \in [\mathscr{C}, \mathsf{Set}]$. So we are done. ☐

30.4 Putting everything together

So now combine all the ingredients from the last three theorems ...

Cue drum-roll!

... and we at last have the full-dress result:

Theorem 139 (Yoneda Lemma). *For any locally small category $\mathscr{C}$, object $A \in \mathscr{C}$, and functor $F\colon \mathscr{C} \to \mathsf{Set}$, $Nat(\mathscr{C}(A,-),F) \cong FA$, both naturally in $A \in \mathscr{C}$ and naturally in $F \in [\mathscr{C}, \mathsf{Set}]$.*

There will evidently be a dual version too (involving contravariant functors in $\mathscr{C}$, i.e. functors in $\mathscr{C}^{op}$):

Theorem 140 (Yoneda Lemma). *For any locally small category $\mathscr{C}$, object $A \in \mathscr{C}$, and functor $F\colon \mathscr{C}^{op} \to \mathsf{Set}$, $Nat(\mathscr{C}(-,A),F) \cong FA$, both naturally in $A \in \mathscr{C}$ and naturally in $F \in [\mathscr{C}^{op}, \mathsf{Set}]$.*

Some authors call only the second version the Yoneda Lemma: we'll use the label for both, talking of the covariant and contravariant versions if we need to mark the distinction.

And having done all this work, we see as an afterword that a further generalization is in principle possible. We've so far been working with locally small categories, i.e. categories whose classes of arrows between pairs of objects are indeed sets which live in Set. Suppose we turn our attention to larger categories whose hom-classes (as we could naturally call them) are some bigger collections which live in a suitably well-behaved category, call it SET, which allows bigger collections. Then we can re-run our arguments to show that for a category $\mathscr{C}$ with hom-classes in SET, $A \in \mathscr{C}$, and a functor $F\colon \mathscr{C} \to$ SET, then the SET of natural transformations from $\mathscr{C}(A,-)$ to F is in natural isomorphism with FA.

But we won't delay over this further generalization – indeed, will we have occasion to use it?

30.5 A brief afterword on 'presheaves'

We pause for a footnote on some jargon that you might well encounter in treatments of the Yoneda Lemma: you ought to know about it, even though we will not adopt it here.

Recall our earlier talk of diagrams. In these terms, a set-valued (covariant) functor $F\colon \mathscr{C} \to \mathsf{Set}$ counts as diagram of shape $\mathscr{C}$ in Set. Unpredictably, the corresponding term for a set-valued contravariant functor is this:

Definition 120. A contravariant functor from $\mathscr{C}$ to Set, i.e. a functor $F\colon \mathscr{C}^{op} \to \mathsf{Set}$, is a *presheaf* on $\mathscr{C}$. △

The terminology 'presheaf' comes from an example in topology. But we will have to just take it as an arbitrary, though widely used, label.

Definition 121. The presheaves on $\mathscr{C}$ (as objects) together with the natural transformations between them (as arrows) form *the presheaf category on* $\mathscr{C}$, denoted $\widehat{\mathscr{C}}$. △

But note, $\widehat{\mathscr{C}}$ is just a relabelling of the functor category we met in §29.3 and called $[\mathscr{C}^{op}, \mathsf{Set}]$. And so the Yoneda embedding $\mathcal{Y}$ we met there is a functor $\mathcal{Y}\colon \mathscr{C} \to \widehat{\mathscr{C}}$; and in our new notation we can say that $\mathscr{C}$ is isomorphic to a full subcategory of $\widehat{\mathscr{C}}$.

Recall $\mathcal{Y}A = \mathscr{C}(-,A)$. Hence $\widehat{\mathscr{C}}(\mathcal{Y}A, F)$ is the hom-class of the presheaf category $\widehat{\mathscr{C}}$ which comprises the arrows of that functor category from $\mathscr{C}(-,A)$ to F, i.e. it is $Nat(\mathscr{C}(-,A), F)$. That's why (one version) of the Yoneda Lemma can also be presented like this: on the usual assumptions, $\widehat{\mathscr{C}}(\mathcal{Y}A, F) \cong FA$, naturally in both $A \in \mathscr{C}$ and F in $\widehat{\mathscr{C}}$.

31 Representables and universal elements

We saw in §23.3 that covariant hom-functors $\mathscr{C}(A,-)$ have the key property of preserving whatever (small) limits exist in $\mathscr{C}$. We will show in a moment that isomorphic functors preserve the same limits. So we are naturally going to be interested too in the functors which are isomorphic to hom-functors, as they will also preserve limits. These are the *representable* functors.

This chapter, then, discusses representable functors, their so-called representations, and the associated notion of universal elements. The definitions and theorems are easy: but the wider significance of these notions will perhaps only become clear when we discuss them in relation to adjunctions in later chapters.

31.1 Isomorphic functors preserve the same limits

We start with the intuitive thought that naturally isomorphic functors ought to behave in essentially the same way. In particular, we ought to have the following theorem:

Theorem 141. *Suppose the parallel functors $F, G\colon \mathscr{C} \to \mathscr{D}$ are naturally isomorphic. Then if F preserves a given limit so does G.*

We confirm this by a pedestrian apply-the-definitions proof. The argument would look simpler if we could wave our hands at diagrams drawn with different coloured chalks and growing in real time on a blackboard! But in monochrome, we have:

Proof. Let $[L, \pi_J]$ be a limit cone for $D\colon \mathsf{J} \to \mathscr{C}$. Then for any $f\colon J \to K$ in J, this diagram commutes in $\mathscr{C}$:

$$\begin{tikzcd} & L \arrow[dl, "\pi_J"'] \arrow[dr, "\pi_K"] & \\ DJ \arrow[rr, "Df"'] & & DK \end{tikzcd}$$

The actions of F and G now send this triangle to the two commuting triangles in the next diagram, and the assumumed natural isomorphism $\alpha\colon F \xRightarrow{\sim} G$ gives us *three* naturality squares, giving us the sides of a commuting prism in $\mathscr{D}$:

$$\begin{tikzcd}
& FL \arrow[drr, "\alpha_L"] \arrow[ddl, "F\pi_J"'] \arrow[ddr, "F\pi_K"] & & & \\
& & & GL \arrow[ddl, "G\pi_J"'] \arrow[ddr, "G\pi_K"] & \\
FDJ \arrow[rr, "FDf"'] \arrow[drr, "\alpha_{DJ}"'] & & FDK \arrow[drr, "\alpha_{DK}"] & & \\
& & GDJ \arrow[rr, "GDf"'] & & GDK
\end{tikzcd}$$

So now consider any cone $[C, c_J]$ over GD with vertex C. Being part of a cone, each tall triangle such as the one below commutes:

$$\begin{tikzcd}
& & & C \arrow[dll, dashed, "u"'] \arrow[dd, dashed, "v"] \arrow[ddddl, "c_J"'] \arrow[ddddr, "c_K"] & \\
& FL \arrow[drr, dotted] \arrow[ddl, dotted] \arrow[ddr, dotted] & & & \\
& & & GL \arrow[ddl, dotted] \arrow[ddr, dotted] & \\
FDJ \arrow[rr, dotted] \arrow[drr, dotted] & & FDK \arrow[drr, dotted] & & \\
& & GDJ \arrow[rr, "GDf"'] & & GDK
\end{tikzcd}$$

Further, using the commuting base square of the prism, we can extend each leg c_J of the cone by composition with α_{DJ}^{-1} to get a cone $[C, \alpha_{DJ}^{-1} \circ c_J]$ over FD.

Now *suppose for the sake of argument that F preserves the limit* $[L, \pi_J]$. Then $[FL, F\pi_J]$ must be a limit cone over FD. Which means that our cone $[C, \alpha_{DJ}^{-1} \circ c_J]$ over FD must factor through this limit cone via a unique $u\colon C \to FL$.

But it is easy to check – chasing arrows round the diagram, using the sloping sides of the prism – that this implies in turn that $[C, c_J]$ over GD factors through $[GL, G\pi_J]$ via $v = \alpha_L \circ u$.

And $[C, c_J]$ can't factor through a distinct v': or else there would be a distinct $u' = \alpha_L^{-1} \circ v'$ which makes everything commute, which is impossible by the uniqueness of u.

Hence, in sum, any $[C, c_J]$ factors through $[GL, G\pi_J]$ via a unique v, and therefore $[GL, G\pi_J]$ is a limit cone. *So G also preserves the limit* $[L, \pi_J]$. □

31.2 Representable functors

(a) As we remarked at the outset, covariant hom-functors preserve limits. Isomorphisms between functors carry over this property. Similarly contravariant hom-functors preserve colimits as limits of the same shape (see Theorem 109):

and, by duality, isomorphisms between contravariant functors similarly carry over this property.

This makes the following concept an evidently interesting one:

Definition 122. A set-valued functor $F : \mathscr{C} \to \mathsf{Set}$ which is naturally isomorphic to some hom-functor $\mathscr{C}(A,-)\colon \mathscr{C} \to \mathsf{Set}$ is said to be *representable*.

Likewise, a set-valued contravariant functor $F : \mathscr{C} \to \mathsf{Set}$ which is naturally isomorphic to some hom-functor $\mathscr{C}(-,A)\colon \mathscr{C} \to \mathsf{Set}$ is also said to be representable. △

And it is immediate that

Theorem 142. *A covariant representable functor $F : \mathscr{C} \to \mathsf{Set}$ preserves all (small) limits that exist in $\mathscr{C}$. Similarly, a contravariant representable functor preserves colimits as limits of the same shape.*

Now, it would perhaps seem most natural to describe the hom-functor that gives us an isomorphic copy of the representable functor $F : \mathscr{C} \to \mathsf{Set}$ as being a representation of F. But that isn't how the standard jargon goes. Rather:

Definition 123. If there is a natural isomorphism $\psi\colon \mathscr{C}(A,-) \xRightarrow{\sim} F$, then the object A in $\mathscr{C}$, is said to be a *representation* of the representable functor F. Similarly for the contravariant case. △

This way of talking does make some claims about representations initially sound slightly odd: we just have to live with that.

Representations need not be strictly unique. However, we do have

Theorem 143. *If the functor $F\colon \mathscr{C} \to \mathsf{Set}$ is represented by both A and B, then $A \cong B$.*

Proof. If we have $\mathscr{C}(A,-) \cong F \cong \mathscr{C}(B,-)$ then, in the notation of Theorem 134, $\mathscr{X}A \cong \mathscr{X}B$ and hence $A \cong B$. □

31.3 A first example

Quite trivially, hom-functors themselves are representables. But are there other kinds of example?

Let's return to the very first functor we met back in §20.2, the forgetful functor $F\colon \mathsf{Mon} \to \mathsf{Set}$ which sends any monoid $\mathcal{M} = (M, \cdot, 1_M)$ to its underlying set M, and sends a monoid homomorphism $f\colon \mathcal{M} \to \mathcal{M}'$ to the same function thought of as an arrow $f\colon M \to M'$ in Set. And let's ask: is there a representing object, i.e. a monoid $\mathcal{R}$, such that the hom-functor $\mathsf{Mon}(\mathcal{R},-)$ is naturally isomorphic to the forgetful F?

Applying the usual definition, the hom-functor $\mathsf{Mon}(\mathcal{R},-)$ sends a monoid $\mathcal{M}$ in Mon to $\mathsf{Mon}(\mathcal{R},\mathcal{M})$. And it sends a monoid homomorphism $f\colon \mathcal{M} \to \mathcal{M}'$ to the set-function $f \circ -$ which sends an arrow $g\colon R \to M$ in $\mathsf{Mon}(\mathcal{R},\mathcal{M})$ to the arrow $f \circ g\colon R \to M'$ in $\mathsf{Mon}(\mathcal{R},\mathcal{M}')$.

And if this functor $\mathsf{Mon}(\mathcal{R}, -)$ is to be naturally isomorphic with the forgetful functor F, there will have to be an isomorphism ψ with a component at each monoid $\mathcal{M}$ such that, for any $f\colon \mathcal{M} \to \mathcal{M}'$ in Mon, the following diagram commutes in Set:

$$\begin{tikzcd}
M \arrow[r, "f"] \arrow[d, "\psi_{\mathcal{M}}"] & M' \arrow[d, "\psi_{\mathcal{M}'}"] \\
\mathsf{Mon}(\mathcal{R}, \mathcal{M}) \arrow[r, "f \circ -"] & \mathsf{Mon}(\mathcal{R}, \mathcal{M}')
\end{tikzcd}$$

For this to work, we certainly need to choose a representing monoid $\mathcal{R}$ such that (for any monoid $\mathcal{M}$) there is a bijection between M and $\mathsf{Mon}(\mathcal{R}, \mathcal{M})$. And presumably, for the needed generality, $\mathcal{R}$ will have to be a monoid without too much distinctive structure. That severely limits the possible options.

First shot: take the simplest such 'boring' monoid, the one-element monoid 1. But a moment's reflection shows that this can't work as a candidate for $\mathcal{R}$ (typically M has many members, $\mathsf{Mon}(1, \mathcal{M})$ can have only one, so there won't be an isomorphism between them).

Second shot: take the next simplest unstructured monoid, the free monoid with a single generator. We can think of this monoid as $\mathcal{N} = (\mathbb{N}, +, 0)$ whose generator is 1, and whose every element is a sum of 1s. Now consider a homomorphism from $\mathcal{N}$ to $\mathcal{M}$. $0 \in \mathbb{N}$ has to be sent to the identity element 1_M in M. And once we also fix that $1 \in \mathbb{N}$ gets sent to some $m \in M$, that determines where every element of $\mathbb{N}$ goes (since every non-zero $\mathbb{N}$ element $1+1+1+\ldots+1$ will be sent to a corresponding M-element $m \cdot m \cdot m \cdot \ldots \cdot m$).

So consider $\psi_{\mathcal{M}}\colon M \to \mathsf{Mon}(\mathcal{N}, \mathcal{M})$ which maps m to the unique homomorphism $\overline{m}\colon \mathcal{N} \to \mathcal{M}$ which sends $1 \in \mathbb{N}$ to m. $\psi_{\mathcal{M}}$ is evidently bijective – each homomorphism from $\mathcal{N}$ to $\mathcal{M}$ is some $\overline{m}$ for one and only one m in M. Hence $\psi_{\mathcal{M}}$ is an isomorphism in Set.

And now it is easily seen that our diagram always commutes. Chase an element $m \in M$ round the diagram. The route via the north-east node gives us $m \longmapsto fm \longmapsto \overline{fm}$, the other route gives us $m \longmapsto \overline{m} \longmapsto f \circ \overline{m}$. But $f \circ \overline{m} = \overline{fm}$ (consider how each acts e.g. on the number 3).

Since the diagram always commutes, this means in turn that the maps $\psi_{\mathcal{M}}$ assemble into a natural isomorphism $\psi\colon F \xRightarrow{\sim} \mathsf{Mon}(\mathcal{N}, -)$. Hence, in summary:

Theorem 144. *The forgetful functor $F\colon \mathsf{Mon} \to \mathsf{Set}$ is representable, and is represented by $\mathcal{N}$, the free monoid on one generator.*

Being representable, it follows that the forgetful F preserves limits: but we knew that already.

31.4 More examples of representables

Unsurprisingly, there are analogous representation theorems for other forgetful functors. For instance, although we won't pause over the proofs, we have:

Theorem 145. *(1) The forgetful functor $F\colon \mathsf{Grp} \to \mathsf{Set}$ is representable, and is represented by $\mathcal{Z}$, the group of integers under addition.*

(2) The forgetful functor $F\colon \mathsf{Ab} \to \mathsf{Set}$ is representable, and is also represented by $\mathcal{Z}$.

(3) The forgetful functor $F\colon \mathsf{Vect} \to \mathsf{Set}$ (where Vect is the category of vector spaces over the reals) is representable, and is represented by $\mathcal{R}$, the reals treated as a vector-space.

(4) The forgetful functor $F\colon \mathsf{Top} \to \mathsf{Set}$ is representable, and is represented by the one-point topological space, call it S_0.

To comment on the only last of these, we simply note that a trivial continuous function with domain S_0 into a space S in effect picks out a single point of S, so the set of arrows $\mathsf{Top}(S_0, S)$ is indeed in bijective correspondence with the set of points FS.

Given such examples, you might be tempted to conjecture that *all* such forgetful functors into Set are representable. But not so. Consider FinGrp, the category of finite groups. Then

Theorem 146. *The forgetful functor $F\colon \mathsf{FinGrp} \to \mathsf{Set}$ is not representable,*

Proof. Suppose a putative representing group $\mathcal{R}$ has r members, and take any group $\mathcal{G}$ with $g > 1$ members, where g is coprime with r. Then it is well known that the only group homomorphism from $\mathcal{R}$ to $\mathcal{G}$ is the trivial one that sends everything to the identity in G. But then the underlying set of $\mathcal{G}$ can't be in bijective correspondence with $\mathsf{FinGrp}(\mathcal{R}, \mathcal{G})$ as would be required for a naturality square proving that $\mathcal{R}$ represented F. ☐

Let's take another pair of examples. We first need to recall definitions from Chapter 20:

(i) The (covariant) *powerset functor* $P\colon \mathsf{Set} \to \mathsf{Set}$ maps a set X to its powerset $\mathscr{P}(X)$ and maps a set-function $f\colon X \to Y$ to the function which sends $U \in \mathscr{P}(X)$ to its image $f[U] \in \mathscr{P}(Y)$.

(ii) The *contravariant powerset functor* $\overline{P}\colon \mathsf{Set}^{op} \to \mathsf{Set}$ again maps a set to its powerset, and maps a set-function $f\colon Y \to X$ to the function which sends $U \in \mathscr{P}(X)$ to its inverse image $f^{-1}[U] \in \mathscr{P}(Y)$.

Theorem 147. *The contravariant powerset functor $\overline{P}$ is represented by the set $2 = \{0, 1\}$; but the covariant powerset functor P is not representable.*

Proof. As yet, we don't have any general principles about representables and non-representables which we can invoke to prove theorems such us this. So again we just need to labour through by applying definitions and seeing what we get.

If the contravariant functor $\overline{P}$ is to be representable, then there must be a representing set R and a natural isomorphism ψ with components such that, for all set functions $f\colon Y \to X$, the following diagram always commutes:

$$\begin{tikzcd}
\overline{P}X \arrow[r, "\overline{P}f"] \arrow[d, "\psi_X"] & \overline{P}Y \arrow[d, "\psi_Y"] \\
\mathsf{Set}(X,R) \arrow[r, "\mathsf{Set}(f,R)"] & \mathsf{Set}(Y,R)
\end{tikzcd}$$

Now $\mathsf{Set}(X,R)$ is the set of set-functions from X to R, whose cardinality is $|R|^{|X|}$; and the cardinality of $\overline{P}X$, i.e. $\mathscr{P}(X)$, is $2^{|X|}$. So that forces R to be a two-membered set: so we pick the set $2 = \{0,1\}$.

$\mathsf{Set}(X,2)$ is then the set of characteristic functions for subsets of X, i.e. the set of functions $c_U\colon X \to \{0,1\}$ where $c_U(x) = 1$ iff $x \in U \subseteq X$. So the obvious next move is to take $\psi_X\colon \overline{P}X \to \mathsf{Set}(X,R)$ to be the isomorphism that sends a set $U \in \mathscr{P}(X)$ to its characteristic function c_U.

With this choice, the diagram always commutes. Chase the element $U \in \overline{P}X$ around. The route via the north-east node takes us from $U \subseteq X$ to $f^{-1}[U] \subseteq Y$ to its characteristic function. i.e. the function which maps $y \in Y$ to 1 iff $f(y) \in U$. Meanwhile, the route via the south-west node takes us first from $U \subseteq X$ to c_U, and then we apply $\mathsf{Set}(f,2)$, which maps $c_U\colon X \to 2$ to $c_U \circ f\colon Y \to 2$, which again is the function which maps $y \in Y$ to 1 iff $f(y) \in U$. Which establishes the first half of the theorem.

For the second half of the theorem, we just note that if we try to run a similar argument for the covariant functor P, we'd need to find a representing set R' such that PX and $\mathsf{Set}(R',X)$ are always in bijective correspondence. But $\mathsf{Set}(R',X)$ is the set of set-functions from R' to X, whose cardinality is $|X|^{|R'|}$, while the cardinality of PX is $2^{|X|}$. And there is no choice of R' which will make these equal for varying X. □

31.5 Universal elements

Back, though, to the basic idea. Concentrate on the covariant functors (we will mostly do this for a couple of sections, letting duality take care of contravariant cases). We say that a functor $F\colon \mathscr{C} \to \mathsf{Set}$ is representable iff there is some hom-functor $\mathscr{C}(A,-)\colon \mathscr{C} \to \mathsf{Set}$ such that $F \cong \mathscr{C}(A,-)$. And then A is said to be a representation of F.

We might prefer to say, however, that a full certificate for the representability of F comprises not just the object A such that $F \cong \mathscr{C}(A,-)$ but also the required natural isomorphism $\psi\colon \mathscr{C}(A,-) \xRightarrow{\sim} F$. In this spirit we might call the pair (A,ψ) the *full* representation of F.

Now, the Yoneda Lemma – or more exactly, Theorem 136 proved en route to the full Lemma – tells us more about natural transformations from $\mathscr{C}(A,-) \to F$. We can picture the situation like this:

$$\begin{aligned} FA &\xrightarrow{\quad \mathcal{X}_{AF} \quad} Nat(\mathscr{C}(A,-),F) \\ a &\longmapsto \alpha\colon \mathscr{C}(A,-) \to F \\ & \qquad\quad \alpha_Z\colon \mathscr{C}(A,Z) \to FZ \\ & \qquad\qquad\quad g \longmapsto Fg(a) \end{aligned}$$

That is to say, there is a bijection $\mathcal{X}_{AF}$ between the members of FA and the members of $Nat(\mathscr{C}(A,-),F)$. This bijection matches up $a \in FA$ with the natural transformation $\alpha = \mathcal{X}_{AF}(a)\colon \mathscr{C}(A,-) \to F$. And this is the transformation whose Z-component α_Z sends a map $g\colon A \to Z$ to $Fg(a)$.

Therefore, instead of saying that a full certificate for the representability of F is a pair (A,ψ), with $A \in \mathscr{C}$ and $\psi\colon \mathscr{C}(A,-) \xRightarrow{\sim} F$, we could equivalently invoke the pair (A,a), with $A \in \mathscr{C}$ and $a \in FA$, where $\mathcal{X}_{AF}(a) = \psi$.

Now note that, since ψ is an isomorphism, each Z-component of $\mathcal{X}_{AF}(a)$ has to be an isomorphism; which means that for each $z \in FZ$ there must be a unique $g\colon A \to Z$ such that $Fg(a) = z$.

Which all goes to motivate introducing the following concept (even if it doesn't yet explain the label for the notion):

Definition 124. A *universal element* of the functor $F\colon \mathscr{C} \to \mathsf{Set}$ is a pair (A,a), where $A \in \mathscr{C}$ and $a \in FA$, and where for each $Z \in \mathscr{C}$ and $z \in FZ$, there is a unique map $g\colon A \to Z$ such that $Fg(a) = z$. △

The story for contravariant functors, by the way, will be exactly the same, except that the map g will go the other way about, $g\colon Z \to A$.

Theorem 148. *A functor $F\colon \mathscr{C} \to \mathsf{Set}$ is representable by A iff it has a universal element (A,a).*

Proof. Our motivating remarks have already established the 'only if' direction; so we only have to prove the converse.

Suppose, therefore, that (A,a) is a universal element for F. Then, $a \in FA$, and there is a natural transformation $\chi = \mathcal{X}_{AF}(a)\colon \mathscr{C}(A,-) \to F$ whose component $\chi_Z\colon \mathscr{C}(A,Z) \to FZ$ sends a map $g\colon A \to Z$ to $Fg(a)$.

We need to show χ_Z has an inverse. But the definition of a universal element tells in effect that there's a function δ_Z which sends $z \in FZ$ to the unique $g\colon A \to Z$ in $\mathscr{C}(A,Z)$ where $Fg(a) = z$. And we can immediately see that χ_Z and δ_Z are inverses.

So each component χ_Z is an isomorphism, and hence $\chi\colon \mathscr{C}(A,-) \xRightarrow{\sim} F$, witnessing that F is representable by A. □

The proof of Theorem 136 also shows that the bijection $\mathcal{X}_{AF}$ associates $\alpha_A(1_A)$ in FA with the natural transformation $\alpha\colon \mathscr{C}(A,-) \to F$ Hence,

Theorem 149. *If the functor* $F\colon \mathscr{C} \to \mathsf{Set}$ *has the full representation* (A, α), *then* F *has the universal element* $(A, \alpha_A(1_A))$.

31.6 Categories of elements

(a) Why '*universal* element'? Because the definition invokes a universal mapping property: (A, a) is a universal element iff for every ... there is a unique map such that .... As in other cases, then, we might expect to be able to define a wider category in which universal elements appear as special cases picked out by this universal mapping property. So here goes:

Definition 125. $\mathsf{Elts}_{\mathscr{C}}(F)$, the *category of elements of the functor* $F\colon \mathscr{C} \to \mathsf{Set}$, has the following data:

(1) Objects are the pairs (A, a), where $A \in \mathscr{C}$ and $a \in FA$.
(2) An arrow from (A, a) to (B, b) is a $\mathscr{C}$-arrow $f\colon A \to B$ such that $Ff(a) = b$.
(3) The identity arrow on (A, a) is 1_A.
(4) Composition of arrows is induced by composition of $\mathscr{C}$-arrows.

It is easily checked that this *is* a category. (Alternative symbolism for the category includes variations on '$\int_{\mathscr{C}} F$'.)

(b) Why 'category of *elements*'? After all, functors don't in a straightforward sense have elements. But we can perhaps throw some light on the name as follows.

(i) Suppose we are given a category $\mathscr{C}$ whose objects *are* sets (perhaps with some additional structure on them) and whose arrows are functions between sets. Then there will be some derived categories whose objects are (or involve) *elements* of $\mathscr{C}$'s objects, and whose arrows between these elements are induced by the arrows between the containing sets.

Now such a category can be constructed in more than one way. But if we don't want the derived category to forget about which elements belong to which sets, then a natural way to go would be to say that the objects of the derived category – which could be called the category of elements of $\mathscr{C}$ – are all the pairs (A, a) for $A \in \mathscr{C}$, $a \in A$. And then given elements $a \in A$, $b \in B$, whenever there is a $\mathscr{C}$-arrow $f\colon A \to B$ such that $f(a) = b$, we'll say that f is also an arrow from (A, a) to (B, b) in our new category. This derived category of elements in a sense unpacks what's going on inside the original category $\mathscr{C}$.

(ii) However, in the general case, $\mathscr{C}$'s objects need not be sets so need not have elements. But a functor $F\colon \mathscr{C} \to \mathsf{Set}$ gives us a diagram of $\mathscr{C}$ inside Set, and of course the objects in the resulting diagram of $\mathscr{C}$ *do* have elements. So we can consider the category of elements of F's-diagram-of-$\mathscr{C}$, which – following the template in (i) – has as objects all the pairs (FA, a) for $A \in \mathscr{C}$, $a \in FA$. And then given elements $a \in FA$, $b \in FB$, whenever there is a Set-arrow $Ff\colon FA \to FB$ such that $Ff(a) = b$, we'll say that Ff is also an arrow from (FA, a) to (FB, b) in our new category.

Now, we can streamline that. Instead of taking the objects to be pairs (FA, a) take them simply to be pairs (A, a) (but where, still, $a \in FA$). And instead of talking of the arrow $Ff\colon FA \to FB$ we can instead talk more simply of $f\colon A \to B$ (but where, still, $Ff(a) = b$). And with that streamlining – lo and behold! – we are back with the category $\mathsf{Elts}_{\mathscr{C}}(F)$, which is isomorphic to category of elements of F's-diagram-of-$\mathscr{C}$, and which – as convention has it – we'll call the category of elements of F, for short.

So the construction of $\mathsf{Elts}_{\mathscr{C}}(F)$ is tolerably natural.

(c) Here is another way of thinking of this category. Let 1 be some singleton in Set. Then what is the comma category $(1 \downarrow F)$? Applying the definition of such categories given in §24.4, the objects of this category are pairs (A, a) where $A \in \mathscr{C}$ and $a\colon 1 \to FA$ is an arrow in Set. And the arrows of the category from (A, a) to (B, b) is a $\mathscr{C}$-arrow $f\colon A \to B$ such that $b = Ff \circ a$.

But *that* is just the definition of $\mathsf{Elts}_{\mathscr{C}}(F)$ except that we have traded in the requirement that a is *member* of FA for the requirement that a is an *arrow* $1 \to FA$. But as we well know by now, members of a set are in bijective correspondence with such arrows from a fixed singleton, and from a categorial perspective we can treat members as such arrows (hence our using the same label 'a' here for both). Hence

Theorem 150. *For a given functor $F\colon \mathscr{C} \to$ Set, the category $\mathsf{Elts}_{\mathscr{C}}(F)$ is (isomorphic to) the comma category $(1 \downarrow F)$ where 1 is terminal in Set.*

(d) Having defined a category $\mathsf{Elts}_{\mathscr{C}}(F)$ for universal elements of $F\colon \mathscr{C} \to$ Set to live in, we can finish by asking: how do we distinguish universal elements from other elements categorially? The answer is immediate from Defn. 124, which in our new terminology says:

Theorem 151. *An object $I = (A, a)$ in $\mathsf{Elts}_{\mathscr{C}}(F)$ is a universal element iff, for every object E in $\mathsf{Elts}_{\mathscr{C}}(F)$ there is exactly one morphism $f\colon I \to E$, so I is initial in $\mathsf{Elts}_{\mathscr{C}}(F)$.*

But initial objects are unique up to unique isomorphism. Which, recalling what isomorphisms in $\mathsf{Elts}_{\mathscr{C}}(F)$ are, implies

Theorem 152. *If (A, a) and (A', a') are universal elements for $F\colon \mathscr{C} \to$ Set, then there is a unique $\mathscr{C}$-isomorphism $f\colon A \to A'$ such that $Ff(a) = a'$.*

31.7 Limits and exponentials as universal elements

(a) Let $Cone(C, D)$ be the set of cones over some diagram D with vertex C in some given category $\mathscr{C}$ – and we will assume that $\mathscr{C}$ is small enough for $Cone(C, D)$ indeed to be a set living in Set.

We can now define a contravariant functor $Cone(-, D)\colon \mathscr{C}^{op} \to$ Set as follows.

(i) $Cone(-, D)$ sends an object C to $Cone(C, D)$.

(ii) $Cone(-, D)$ sends an arrow $f\colon C' \to C$ to $Cone(f, D)\colon Cone(C, D) \to Cone(C', D)$, which takes a cone $[C, c_j]$ and sends it to $[C', c_j \circ f]$.

It is easily checked that this is indeed a functor.

We now apply the definition of universal elements, tweaked for the contravariant case. Then a universal element of the functor $Cone(-, D)$ is a pair $(L, [L, l_J])$, where L is in $\mathscr{C}$ and $[L, l_J]$ is in $Cone(L, D)$, the set of cones over D with vertex L. And moreover, we require that for each $C \in \mathscr{C}$ and each cone $[C, c_J]$, there is a unique map $f\colon C \to L$ such that $Cone(f)[L, l_J] = [C, c_J]$, which requires $l_J \circ f = c_J$ for each J. But that's just to say that $[L, l_J]$ is a limit cone! Hence

Theorem 153. *In small enough categories, a limit cone over a diagram D is a universal element for $Cone(-, D)$.*

Since limits are therefore initial objects in an associated category of elements, they have to be unique up to a unique appropriate isomorphism, giving us another proof of Theorem 54.

(b) Consider the contravariant functor $\mathscr{C}(-\times B, C)$ which we met in §25.3 Ex. (7). This sends an object A in $\mathscr{C}$ to the hom-set of arrows from $A \times B$ to C. And it sends an arrow $f\colon A' \to A$ to the map $- \circ f \times 1_B$ (i.e. to the map which takes an arrow $j\colon A \times B \to C$ and yields the arrow $j \circ f \times 1_B\colon A' \times B \to C$).

Now apply the definition of universal element for the contravariant case. Then a universal element of $\mathscr{C}(- \times B, C)$ is a pair (E, ev), with E in $\mathscr{C}$ and ev in $\mathscr{C}(E \times B, C)$, such that for every A and every $g \in \mathscr{C}(A \times B, C)$, there is a unique $\overline{g}\colon A \to E$ such that $\mathscr{C}(- \times B, C)(\overline{g})(ev) = g$, i.e. $ev \circ \overline{g} \times 1_B = g$.

But, trivially squaring up the brackets, a pair $[E, ev]$ with those properties is exactly the exponential $[C^B, ev]$. Hence

Theorem 154. *The exponential $[C^B, ev]$, when it exists in $\mathscr{C}$, is a universal element of $\mathscr{C}(- \times B, C)$.*

Since exponentials are therefore also initial objects in an associated category of elements, they too have to be unique up to a unique appropriate isomorphism, giving us this time another proof of Theorem 73.

32 Galois connections

We will have quite a lot more to say about functors, limits and representables and about how they interrelate after we have introduced the next really important Big Idea from category theory – namely, the idea of pairs of *adjoint functors* and the *adjunctions* they form.

Now, one option would be to dive straight into the general story about adjoints. But that multi-faceted story can initially seem rather complex, and it is quite easy to get lost in the details. So the plan here is to start by looking first at a very restricted class of cases. These are the so-called Galois connections, which are in effect adjunctions between two categories which are posets. In this chapter, then, we discuss these Galois connections in an elementary way, as a way of introducing us to some key themes. And for the moment, we largely suppress the categorial context.

32.1 (Probably unnecessary) reminders about posets

Recall: The set C equipped with the binary relation $\preccurlyeq$, which we denote $(C, \preccurlyeq)$, is a poset just in case $\preccurlyeq$ is a partial order – i.e., for all $x, y, z \in C$, (i) $x \preccurlyeq x$, (ii) if $x \preccurlyeq y$ and $y \preccurlyeq z$ then $x \preccurlyeq z$, (iii) if $x \preccurlyeq y$ and $y \preccurlyeq x$ then $x = y$. (We will, as appropriate, recruit '$\sqsubseteq$', '$\leqslant$', '$\subseteq$' as other symbols for partial orders.)

Reversing a partial order gives us another partial order. Hence reversing the order in a poset $\mathscr{C} = (C, \preccurlyeq)$ gives us a dual poset $\mathscr{C}^{op} = (C, \succcurlyeq)$ defined in the obvious way.

There is a related notion of a strict poset defined in terms of a strict partial order $\prec$, where $x \prec y$ iff $x \preccurlyeq y \wedge x \neq y$ for some partial order $\preccurlyeq$. It is just a matter of convenience whether we concentrate on the one flavour of poset or the other, and you will already be familiar with a variety of examples of 'naturally occurring' posets of both flavours.

The following notions will also be entirely familiar, in one terminology or another:

Definition 126. Suppose that $\mathscr{C} = (C, \preccurlyeq)$ and $\mathscr{D} = (D, \sqsubseteq)$ are two posets. Let the map $F : \mathscr{C} \to \mathscr{D}$ be a function between the carrier sets C and D. Then

(1) F is monotone just in case, for all $x, y \in C$, if $x \preccurlyeq y$ then $Fx \sqsubseteq Fy$;

(2) F is an order-embedding just in case, for all $x, y \in C$, $x \preccurlyeq y$ iff $Fx \sqsubseteq Fy$;

(3) F is an order-isomorphism iff F is a surjective order-embedding. △

Some obvious remarks about these notions:

i. Monotone maps compose to give monotone maps and composition is associative. Likewise for order-embeddings and order-isomorphisms.

ii. Order-embeddings are injective. Keeping the same notation, suppose $Fx = Fy$ and hence both $Fx \sqsubseteq Fy$ and $Fy \sqsubseteq Fx$. Then, if F is an embedding, $x \preccurlyeq y$ and $y \preccurlyeq x$, and hence $x = y$.

iii. If $F[C]$ is C's image under F, an order-embedding $F\colon (C, \preccurlyeq) \to (D, \sqsubseteq)$ is an order-isomorphism from $(C, \preccurlyeq)$ to $(F[C], \sqsubseteq)$.

iv. An order-isomorphism is bijective, and therefore is an isomorphism as a set-function. Order-isomorphisms have unique inverses which are also order-isomorphisms.

v. Posets are deemed isomorphic if there is an order-isomorphism between them.

If $(C, \preccurlyeq)$ is a poset and $X \subseteq C$, then a maximum of X (with respect to the inherited order $\preccurlyeq$) is defined in the obvious way: m is a maximum of X iff $m \in X \wedge (\forall x \in X) x \preccurlyeq m$. Maxima are unique when they exist – for if $m, m' \in X$ are both maxima, $m' \preccurlyeq m$ and similarly $m \preccurlyeq m'$ and hence $m = m'$.

If $X \subseteq C$ we say that $(X, \preccurlyeq)$ is a sub-poset of $(C, \preccurlyeq)$; and note here that we will not routinely fuss to distinguish a relation defined over C from the restriction of that relation to X.

Definition 127. Suppose Π is a collection of sets. Then Π ordered by inclusion, i.e. $(\Pi, \subseteq)$, is an *inclusion poset.* △

Theorem 155. *Every poset is isomorphic to an inclusion poset.*

Proof. Take the poset $(C, \preccurlyeq)$. For each $y \in C$, now form the set containing it and its $\preccurlyeq$-predecessors $\pi_y = \{x \in C \mid x \preccurlyeq y\}$. Let Π the set of all π_y for $y \in C$. Then $(\Pi, \subseteq)$ is an inclusion poset.

Define $F\colon (C, \preccurlyeq) \to (\Pi, \subseteq)$ by putting $Fx = \pi_x$. Then F is very easily seen to be a bijection, and also $x \preccurlyeq y$ iff $\pi_x \subseteq \pi_y$. So F is an order-isomorphism. □

32.2 An introductory example

We rather informally describe what will turn out to be an important instance of a Galois connection: we choose notation with an eye to smoothing the transitions to later generalizations.

Suppose, then, that we have a poset $\mathscr{C} = (C, \preccurlyeq)$ where the members of C are sets of sentences from some suitable formal language $\mathcal{L}$ (the details of $\mathcal{L}$ won't matter too much), and $\preccurlyeq$ is simply set-inclusion. We can think of the members of C as *theories* couched in the language $\mathcal{L}$; these theories are then partially ordered from less specific (saying less) to more specific (saying more).

There is a corresponding poset $\mathscr{D} = (D, \sqsubseteq)$ where the members of D are collections of $\mathcal{L}$-structures, i.e. sets of potential models for theories couched in $\mathcal{L}$; and we will take $\sqsubseteq$ to be the *converse* of inclusion. A member of D can be thought of as a set of alternative model 'worlds' a theory could be true of; these sets of models are then also partially ordered from less specific (more alternatives) to more specific (a narrower range).

There are then two very natural maps between these posets.

i. $F\colon \mathscr{C} \to \mathscr{D}$ sends a theory $c \in C$ to $d \in D$, where d is the set of models of c (i.e. d is the set containing each model on which all the sentences in c are true).

ii. $G\colon \mathscr{D} \to \mathscr{C}$ sends a set of models d to the set c containing each sentence which is true on every model in d.

Put it this way: F is the 'find the models' function. It takes a bunch of sentences and returns all its models, the set of structures where the sentences in the bunch are all true. In the other direction, G is the equally natural 'find all the true sentences' function. It takes a bunch of structures and returns the set of sentences that are true in all of those structures.

In general F and G will not be inverse to each other. But the mapping functions do interrelate in the following nice ways:

(1) F and G are monotone.

And for all $c \in C$, $d \in D$,

(2) $c \preccurlyeq GFc$ and $FGd \sqsubseteq d$,

(3) $Fc \sqsubseteq d$ iff $c \preccurlyeq Gd$.

And further

(4) $FGF = F$ and $GFG = G$.

Why so? For (1) we note that if the theory c' is more informative than c, then it will be true of a narrow range of possible models. And conversely, if d' is a narrower range of models than d, then more sentences will be true of everything in d' than are true of everything in d.

For the first half of (2) we note that if we start with a bunch of sentences c, look at the models where they are all true together, and then look at the sentences true in all those models together, we'll get back original sentences in c plus all their consequences (where consequence is defined in the obvious way in terms of preservation of truth in the relevant set of structures).

For the other half of (2) we note that if we start from a collection of models d, find the sentences true in all of them, and then look at the models for those sentences, we must get back at least the models we started with, maybe more. (Remember, $\sqsubseteq$ is the converse of inclusion!)

For (3) we note that if the models where all the sentences of c are true include all those in d then the theory c must be included in the set of sentences true in all the models in d, and vice versa.

For the first half of (4) we note that the models of a set of sentences c together with their consequences are just the models of the original c. Similarly for the other half.

So in summary: we have here a pair of posets $\mathscr{C} = (C, \preccurlyeq)$, $\mathscr{D} = (D, \sqsubseteq)$ and a pair of functions $F\colon \mathscr{C} \to \mathscr{D}$ and $G\colon \mathscr{D} \to \mathscr{C}$ for which conditions (1) to (4) hold. We will see in the next section that this situation is repeatedly realized in different contexts.

32.3 Galois connections defined

We now generalize. However, as we'll see in the next section, conditions (1) to (4) are not independent. The first two together imply the third and fourth, and the third implies the rest. Simply because it is prettier, then, we plump in this section for a general definition just in terms of the third condition (which we relabel):

Definition 128. Suppose that $\mathscr{C} = (C, \preccurlyeq)$ and $\mathscr{D} = (D, \sqsubseteq)$ are two posets, and let $F\colon \mathscr{C} \to \mathscr{D}$ and $G\colon \mathscr{D} \to \mathscr{C}$ be a pair of functions such that for all $c \in C$, $d \in D$,

$$\text{(G)} \qquad Fc \sqsubseteq d \text{ iff } c \preccurlyeq Gd.$$

Then F and G form a *Galois connection* between $\mathscr{C}$ and $\mathscr{D}$. When this holds, we write $F \dashv G$, and F is said to be the *left adjoint* of G, and G the right adjoint of F.[1] $\triangle$

The first discussion of a version of such a connection $F \dashv G$ – and hence the name – is to be found in Evariste Galois's work in what has come to be known as Galois theory, a topic beyond our purview here. And there are plenty of other serious mathematical examples (e.g. from number theory, abstract algebra and topology) of two posets with a Galois connection between them. But we really don't want to get bogged down in unnecessary mathematics at this early stage; so for the moment let's just give some simple cases, to add to our informally described motivating example in the last section:

(1) Suppose F is an order-isomorphism between $(C, \preccurlyeq)$ and $(D, \sqsubseteq)$: then F^{-1} is an order-isomorphism in the reverse direction. Take $c \in C, d \in D$: then trivially $Fc \sqsubseteq d$ iff $F^{-1}Fc \preccurlyeq F^{-1}d$ iff $c \preccurlyeq F^{-1}d$. Hence $F \dashv F^{-1}$.

[1] Talk of adjoints here seems to have been originally borrowed from the old theory of Hermitian operators, where in e.g. a Hilbert space with inner product $\langle\cdot,\cdot\rangle$ the operators A and A^* are said to be adjoint when we have, generally, $\langle Ax, y\rangle = \langle x, A^*y\rangle$. The formal analogy is evident.

(2) Take $\mathscr{N} = (\mathbb{N}, \leqslant)$ and $\mathscr{Q}^+ = (\mathbb{Q}^+, \leqslant)$, i.e. the naturals and the non-negative rationals in their standard orders. Let $I\colon \mathscr{N} \to \mathscr{Q}^+$ be the injection function which maps a natural number to the corresponding rational integer, and let $F\colon \mathscr{Q}^+ \to \mathscr{N}$ be the 'floor' function which maps a rational to the natural corresponding to its integral part. Then $I \dashv F$ is a Galois connection from $\mathscr{N}$ to $\mathscr{Q}$· Likewise if $C\colon \mathscr{Q}^+ \to \mathscr{N}$ is the 'ceiling' function which maps a rational to the smallest integer which is at least as big, then $C \dashv I$ is a Galois connection going in the opposite direction.

(3) Let $f\colon X \to Y$ be some function between two sets X and Y. It induces a function $F\colon \mathcal{P}(X) \to \mathcal{P}(Y)$ between their powersets which sends $A \subseteq X$ to $f[A]$, and another function $F^{-1}\colon \mathcal{P}(Y) \to \mathcal{P}(X)$ which sends $B \subseteq Y$ to its pre-image under f, $F^{-1}[B] = \{x \in X \mid f(x) \in B\}$. Then $F \dashv F^{-1}$ is a Galois connection between the inclusion posets $(\mathcal{P}(X), \subseteq)$ and $(\mathcal{P}(Y), \subseteq)$.

(4) Take any poset $\mathscr{C} = (C, \leqslant)$, and let 1 be a one object poset, i.e. of the form $(\{0\}, =)$. Let $F\colon \mathscr{C} \to 1$ be the only possible function, the trivial one which sends everything to 0. Then F has a right adjoint $G\colon 1 \to \mathscr{C}$ just if it is the case that, for any $c \in C$, $Fc = 0$ iff $c \leqslant G0$. So F has a right adjoint just in case $\mathscr{C}$ has a maximum, and then G sends 1's only element to it. Dually, F has a left adjoint just in case $\mathscr{C}$ has a minimum, and then the left adjoint G' sends 1's only element to *that*.

(5) Our next example is from elementary logic. Choose a favourite logical proof-system $\mathcal{L}$ – it could be classical or intuitionistic, or indeed any other logic, so long as it has a normally-behaved conjunction and conditional connectives and a sensible deducibility relation. Let $\alpha \vdash \beta$ notate, as usual, that there is a formal $\mathcal{L}$-proof from premiss α to conclusion β. Then let $|\alpha|$ be the equivalence class of wffs of the system interderivable with α. Take E to be set of all such equivalence classes, and put $|\alpha| \leqslant |\beta|$ in E iff $\alpha \vdash \beta$. Then it is easily checked that $(E, \leqslant)$ is a poset.

Now consider the following two functions between $(E, \leqslant)$ and itself. Fix γ to be some $\mathcal{L}$-wff. Then let F send the equivalence class $|\alpha|$ to the class $|(\gamma \wedge \alpha)|$, and let G send $|\alpha|$ to the class $|(\gamma \to \alpha)|$.

Given our normality assumption, $\gamma \wedge \alpha \vdash \beta$ if and only if $\alpha \vdash \gamma \to \beta$. Hence $|\gamma \wedge \alpha| \leqslant |\beta|$ iff $|\alpha| \leqslant |\gamma \to \beta|$. That is to say $F|\alpha| \leqslant |\beta|$ iff $|\alpha| \leqslant G|\beta|$. Hence we have a Galois connection $F \dashv G$ between $(E, \leqslant)$ and itself, and in a slogan, 'Conjunction is left adjoint to conditionalization'.

(6) Our last example for the moment is another example from elementary logic. Let $\mathcal{L}$ now be a first-order logic, and consider the set of $\mathcal{L}$-wffs with at most the variables $\vec{x}$ free.

We will write $\varphi(\vec{x})$ for a formula in this class, $|\varphi(\vec{x})|$ for the class of formulae interderivable with $\varphi(\vec{x})$, and $E_{\vec{x}}$ for the set of such equivalence classes of formulae with at most $\vec{x}$ free. Using $\leqslant$ as in the last example, $(E_{\vec{x}}, \leqslant)$ is a poset for any choice of variables $\vec{x}$.

We now consider two maps between the posets $(E_{\vec{x}}, \leqslant)$ and $(E_{\vec{x},y}, \leqslant)$. In other words, we are going to be moving between (equivalence classes of) formulae with at most $\vec{x}$ free, and (equivalence classes of) formulae with at most $\vec{x}, y$ free – where y is a new variable not among the $\vec{x}$.

First, since every wff with at most the variables $\vec{x}$ free also has at most the variables $\vec{x}, y$ free, there is a trivial map $F\colon E_{\vec{x}} \to E_{\vec{x},y}$ that sends the class of formulas $|\varphi(\vec{x})|$ in $E_{\vec{x}}$ to the same class of formulas which is also in $E_{\vec{x},y}$.

Second, we define the companion map $G\colon E_{\vec{x},y} \to E_{\vec{x}}$ that sends $|\varphi(\vec{x}, y)|$ in $E_{\vec{x},y}$ to $|\forall y \varphi(\vec{x}, y)|$ in $E_{\vec{x}}$.

Then $F \dashv G$, i.e. we have another Galois connection. For that is just to say

$$F(|\varphi(\vec{x})|) \leqslant |\psi(\vec{x}, y)| \quad \text{iff} \quad |\varphi(\vec{x})| \leqslant G(|\psi(\vec{x}, y)|).$$

Which just reflects the familiar logical rule that

$$\varphi(\vec{x}) \vdash \psi(\vec{x}, y) \text{ iff } \varphi(\vec{x}) \vdash \forall y \psi(\vec{x}, y),$$

so long as y is not free in $\varphi(\vec{x})$. Hence universal quantification is right-adjoint to a certain trivial inclusion operation.

And we can exactly similarly show that existential quantification is left-adjoint to the same operation.

Some morals. Our first example shows that Galois connections are at least as plentiful as order-isomorphisms: and such an isomorphism will have a right adjoint and left adjoint which are the same (i.e. both are the isomorphism's inverse). The second and fourth cases show that posets that aren't order-isomorphic can in fact still be Galois connected. The third case shows that posets can have many Galois connections between them (as any $f\colon X \to Y$ generates a connection between the inclusion posets on the powersets of X and Y). The fourth example gives a case where a function has both a left and a right adjoint which are different. The fourth and sixth cases give a couple of illustrations of how a significant construction (taking maxima, forming a universal quantification respectively) can be regarded as adjoint to some quite trivial operation. The fifth example, like the third, shows that even when the Galois-connected posets are isomorphic (in the fifth case trivially so, because they are identical!), there can be a pair of functions which aren't isomorphisms but which also go to make up a connection between the posets. And the fifth and sixth examples, like the motivating example in the previous section, illustrate why Galois connections are of interest to logicians.

32.4 Galois connections re-defined

The following theorem is basic:

Theorem 156. *Suppose that $\mathscr{C} = (C, \leqslant)$ and $\mathscr{D} = (D, \sqsubseteq)$ are posets with maps $F\colon \mathscr{C} \to \mathscr{D}$ and $G\colon \mathscr{D} \to \mathscr{C}$ between them. Then $F \dashv G$ iff and only if*

(1) F and G are both monotone, and

(2) for all $c \in C$, $d \in D$, $c \preccurlyeq GFc$ and $FGd \sqsubseteq d$, and

(3) $FGF = F$ and $GFG = G$.

Proof. (*If*) Assume conditions (1) and (2) both hold. And suppose $Fc \sqsubseteq d$. Since by (1) G is monotone, $GFc \preccurlyeq Gd$. But by (2) $c \preccurlyeq GFc$. Hence by transitivity $c \preccurlyeq Gd$. That establishes one half of the biconditional (G). We don't need (3) here. The proof of the other half is dual.

(*Only if*) Suppose (G) is true. Then in particular, $Fc \sqsubseteq Fc$ iff $c \preccurlyeq GFc$. Since $\sqsubseteq$ is reflexive, $c \preccurlyeq GFc$. Similarly for the other half of (2).

Now, suppose also that $c \preccurlyeq c'$. Then since we've just shown $c' \preccurlyeq GFc'$, we have $c \preccurlyeq GFc'$. But by (G) we have $Fc \sqsubseteq Fc'$ iff $c \preccurlyeq GFc'$. Whence, $Fc \sqsubseteq Fc'$ and F is monotone. Similarly for the other half of (1).

For (3), since for any $c \in C$, $c \preccurlyeq GFc$, and also F is monotone, it follows that $Fc \sqsubseteq FGFc$.

But the fundamental condition (G) yields $FGFc \sqsubseteq Fc$ iff $GFc \preccurlyeq GFc$. The r.h.s. is trivially true, so $FGFc \sqsubseteq Fc$.

By the antisymmetry of $\sqsubseteq$, then, $FGFc = Fc$. Since c was arbitrary, $FGF = F$. Similarly for the other half of (3). □

This theorem means that, as already intimated at the end of §32.2, we could equally well have defined a Galois connection like this:

Definition 129 (Alternative)**.** Suppose that $\mathscr{C} = (C, \preccurlyeq)$ and $\mathscr{D} = (D, \sqsubseteq)$ are two posets, and let $F\colon \mathscr{C} \to \mathscr{D}$ and $G\colon \mathscr{D} \to \mathscr{C}$ be a pair of functions such that for all $c \in C$, $d \in D$,

(1) F and G are both monotone, and

(2) for all $c \in C$, $d \in D$, $c \preccurlyeq GFc$ and $FGd \sqsubseteq d$, and

(3) $FGF = F$ and $GFG = G$.

Then F and G form a Galois connection between $\mathscr{C}$ and $\mathscr{D}$. △

Two comments about this. First, our proof of Theorem 156 shows that we needn't have explicitly given clause (3) in our alternative definition as it follows from the other two. We include it because when we move on from the case of Galois connections to discuss adjunctions more generally, again giving two definitions, we will need to explicitly mention the analogue of clause (3).

Second, note that we could replace clause (2) with the equivalent clause

(2′) (i) if $c \preccurlyeq c'$, then both $c \preccurlyeq c' \preccurlyeq GFc'$ and $c \preccurlyeq GFc \preccurlyeq GFc'$; and
(ii) if $d \sqsubseteq d'$, then both $FGd \sqsubseteq d \sqsubseteq d'$ and $FGd \sqsubseteq FGd' \sqsubseteq d'$.

For trivially (2′) implies (2); conversely (1) and (2) imply (2′). Again, we mention this variant on our alternative definition of Galois connections for later use when we come to generalize.

32.5 Some basic results about Galois connections

(a) We now have a pair of equivalent definitions of Galois connections, and a small range of elementary examples. In this section we start by proving a couple of theorems that show that such connections behave just as you would hope, in two different respects. First, if there is a connection between $\mathscr{C}$ and $\mathscr{D}$ and a connection between $\mathscr{D}$ and $\mathscr{E}$ then they can be composed to give a connection between $\mathscr{C}$ and $\mathscr{E}$. And second, inside a Galois connection, a left adjoint uniquely fixes its right adjoint, and vice versa. Thus:

Theorem 157. *Suppose there is a Galois connection $F \dashv G$ between the posets $\mathscr{C} = (C, \preccurlyeq)$ and $\mathscr{D} = (D, \sqsubseteq)$, and a connection $H \dashv K$ between the posets $\mathscr{D}$ and $\mathscr{E} = (E, \subseteq)$. Then there is a Galois connection $HF \dashv GK$ between $\mathscr{C}$ and $\mathscr{E}$.*

Proof. Take any for any $c \in C, e \in E$. Then, using the first connection, we have $Fc \sqsubseteq Ke$ iff $c \preccurlyeq GKe$. And by the second connection, we have $HFc \subseteq e$ iff $Fc \sqsubseteq Ke$.

Hence $HFc \subseteq e$ iff $c \preccurlyeq GKe$. Therefore $HF \dashv GK$. ☐

Theorem 158. *If we have Galois connections $F \dashv G$, $F \dashv G'$ between the posets $(C, \preccurlyeq)$ and $(D, \sqsubseteq)$, then $G = G'$. Likewise, if $F \dashv G$, $F' \dashv G$ are both Galois connections between the same posets, then $F = F'$.*

Proof. We prove the first part. $F \dashv G'$ implies, in particular, that for any $d \in D$, $FGd \sqsubseteq d$ iff $Gd \preccurlyeq G'd$.

But by Theorem 156, applied to the connection $F \dashv G$, we have $FGd \sqsubseteq d$. So we can infer that, indeed, $Gd \preccurlyeq G'd$.

By symmetry, $G'd \preccurlyeq Gd$. But d was arbitrary, so indeed $G = G'$. ☐

Careful, though! This second theorem does not say that, for any F which maps between $(C, \preccurlyeq)$ and $(D, \sqsubseteq)$, there must actually exist a unique corresponding G in the reverse direction such that $F \dashv G$ (this isn't true as we saw in §32.3 Ex. (4)). Nor does it say that when there is a Galois connection between the posets, it is unique (our toy examples have already shown that that is false too). The claim is only that, if you are given a possible left adjoint – or a possible right adjoint – there can be at most one candidate for its companion to complete a connection.

(b) Given that adjoint functions determine each other, we naturally seek an explicit definition of one in terms of the other. Here it is:

Theorem 159. *If $F \dashv G$ is a Galois connection between the posets $(C, \preccurlyeq)$ and $(D, \sqsubseteq)$, then*

(1) $Gd =$ the maximum of $\{c \in C \mid Fc \sqsubseteq d\}$,

(2) $Fc =$ the minimum of $\{d \in D \mid c \preccurlyeq Gd\}$.

Proof. We argue for (1), leaving the dual (2) to take care of itself. Fix on an arbitrary $d \in D$ and for brevity, put $\Sigma = \{c \in C \mid Fc \sqsubseteq d\}$.

Theorem 156 tells us that (i) for any $u \in C$, $u \preccurlyeq GFu$, (ii) $FGd \sqsubseteq d$, and (iii) G is monotone. So by (ii), $Gd \in \Sigma$.

Now suppose $u \in \Sigma \subseteq C$. Then $Fu \sqsubseteq d$. By (iii), $GFu \preccurlyeq Gd$. Whence from (i), $u \preccurlyeq Gd$.

That shows Gd is both a member of and an upper bound for Σ, i.e. is a maximum for Σ. □

Recall the posets $\mathscr{N} = (\mathbb{N}, \leqslant)$ and $\mathscr{Q}^+ = (\mathbb{Q}^+, \leqslant)$ with the injection map $I\colon \mathscr{N} \to \mathscr{Q}^+$ and floor function $F\colon \mathscr{Q}^+ \to \mathscr{N}$ which maps a rational to the natural corresponding to its integral part. Then we remarked before that $I \dashv F$. Now we note that $F \dashv I$ is false. Indeed, there can be no connection of the form $F \dashv G$ from $\mathscr{Q}^+$ to $\mathscr{N}$. For $Fq = 1$ iff $1 \leqslant q < 2$, and hence $\{q \in \mathbb{Q}^+ \mid Fq \leqslant 1\}$ has no maximum, and so there can be no right adjoint to F.

Generalizing, we have the following:

Theorem 160. *Galois connections are not necessarily symmetric. That is to say, given $F \dashv G$ is a Galois connection between the posets $\mathscr{C}$ and $\mathscr{D}$, it does not follow that $G \dashv F$ is a connection between $\mathscr{D}$ and $\mathscr{C}$.*

32.6 Fixed points, isomorphisms, and closures

Theorems 157 and 158 tell us that Galois connections are rather nicely behaved. This section now explores some of the consequences of there being a Galois connection $F \dashv G$ between two posets.

(a) Theorem 156 tells us, in particular, where to find the fixed points of the composite maps GF and FG:

Theorem 161. *Given a Galois connection $F \dashv G$ between the posets $(C, \preccurlyeq)$ and $(D, \sqsubseteq)$, then*

(1) $c \in G[D]$ iff c is a fixed point of GF; $d \in F[C]$ iff d is a fixed point of FG.

(2) $G[D] = (GF)[C]$; $F[C] = (FG)[D]$.

Proof. (1) Suppose $c \in G[D]$. Then for some $d \in D$, $c = Gd$ and hence $GFc = GFGd = Gd = c$, so c is a fixed point of GF. Conversely suppose $GFc = c$. Then c is the value of Gd for $d = Fc$, and therefore $c \in G[D]$.

Hence $c \in G[D]$ iff c is a fixed point of GF. The other half of (1) is dual.

(2) We have just seen that if $c \in G[D]$ then $c = GFc$ so $c \in (GF)[C]$. Therefore $G[D] \subseteq (GF)[C]$. Conversely, suppose $c \in (GF)[C]$, then for some $c' \in C$, $c = GFc'$; but $Fc' \in D$ so $c \in G[D]$. Therefore $(GF)[C] \subseteq G[D]$.

Hence $G[D] = (GF)[C]$. The other half of (2) is dual. □

(b) We know that a pair of posets which have a Galois connection between them needn't be isomorphic overall. But this next theorem says that they will typically contain an interesting pair of isomorphic sub-posets (alongside the trivially isomorphic one-object posets!).

Definition 130. Suppose $F \dashv G$ is a Galois connection between the posets $\mathscr{C} = (C, \preccurlyeq)$ and $\mathscr{D} = (D, \sqsubseteq)$. Put $C^{\dashv} = G[D]$ and $D^{\dashv} = F[C]$. Then we define $\mathscr{C}^{\dashv} = (C^{\dashv}, \preccurlyeq)$ and $\mathscr{D}^{\dashv} = (D^{\dashv}, \sqsubseteq)$. △

Theorem 162. *If $F \dashv G$ is a Galois connection between the posets $\mathscr{C} = (C, \preccurlyeq)$ and $\mathscr{D} = (D, \sqsubseteq)$, then $\mathscr{C}^{\dashv}$ and $\mathscr{D}^{\dashv}$ are order-isomorphic.*

Proof. We show that F restricted to $C^{\dashv}$ provides the desired order isomorphism.

Note first that if $c \in C^{\dashv}$, then $Fc \in F[C] = D^{\dashv}$. So F as required sends elements of $C^{\dashv}$ to elements of $D^{\dashv}$. Moreover every element of $D^{\dashv}$ is Fu for some $u \in C^{\dashv}$. For if $d \in F[C]$, then for some c, $d = Fc = FGFc = Fu$ where $u = GFc \in G[D] = C^{\dashv}$.

So F restricted to $C^{\dashv}$ is onto $D^{\dashv}$. It remains to show that it is an order-embedding. We know that F will be monotone, so what we need to prove is that, if $c, c' \in C^{\dashv}$ and $Fc \sqsubseteq Fc'$, then $c \preccurlyeq c'$.

But if $Fc \sqsubseteq Fc'$, then by the monotonicity of G, $GFc \preccurlyeq GFc'$. Recall, though, that $c, c' \in C^{\dashv} = G[D]$ are fixed points of GF. Hence $c \preccurlyeq c'$ as we want. □

(c) Finally, we want the idea of a closure function K on a poset which, roughly speaking, maps a poset 'upwards' to a subposet which then stays fixed under further applications of K:

Definition 131. Suppose $\mathscr{C} = (C, \preccurlyeq\rangle)$ is a poset; then a *closure function* on $\mathscr{C}$ is a function $K\colon \mathscr{C} \to \mathscr{C}$ such that, for all $c, c' \in C$,

(1) $c \preccurlyeq Kc$;

(2) if $c \preccurlyeq c'$, then $Kc \preccurlyeq Kc'$, i.e. K is monotone;

(3) $KKc = Kc$, i.e. K is idempotent. △

Theorem 163. *If $F \vdash G$ is a Galois connection between $\mathscr{C}$ and another poset, then GF is a closure function for $\mathscr{C}$.*

Proof. We quickly check that the three conditions for closure apply. (i) is given by Theorem 156. (ii) is immediate as GF is a composition of monotone functions. And for (iii), we know that $FGF = F$, and hence $GFGF = GF$. □

32.7 One way a Galois connection can arise

The last three sections have been about Galois connections in general, and reveal that they have a perhaps surprisingly rich structure. In this section, we now note one characteristic way in which connections can arise.

Theorem 164. *Let R be a binary relation between members of X and members of Y. We define posets on the powersets, $\mathscr{C} = (\mathcal{P}(X), \subseteq)$, $\mathscr{D} = (\mathcal{P}(Y), \supseteq)$ – note the order reversal.*

Define $F\colon \mathscr{C} \to \mathscr{D}$ by putting $FA = \{b \mid (\forall a \in A)aRb\}$ for $A \subseteq X$. Similarly define $G\colon \mathscr{D} \to \mathscr{C}$ by putting $GB = \{a \mid (\forall b \in B)aRb\}$ for $B \subseteq Y$.

Then $F \dashv G$.

Proof. We just have to prove that principle (G) holds, i.e. for any $A \subseteq X$, $B \subseteq Y$, $FA \supseteq B$ iff $A \subseteq GB$.

But simply by applying definitions we see $FA \supseteq B$ iff $(\forall b \in B)(\forall a \in A)aRb$ iff $(\forall a \in A)(\forall b \in B)aRb$ iff $A \subseteq GB$. ☐

Let's say that Galois connection produced in this way is *relation-generated*. Galois's original classic example was of this kind. And our original motivating example, which we return to in the next section, is relation-generated too.

32.8 Syntax and semantics briefly revisited

(a) In his famous Dialectica paper 'Adjointness in foundations' (1969), F. William Lawvere writes of 'the familiar Galois connection between sets of axioms and classes of models, for a fixed [signature]'. This is in fact the motivating example which we presented very informally in §32.2. We will very briefly revisit it.

Let $\mathcal{L}$ be a formal language. Then a set of $\mathcal{L}$-axioms in the wide sense that Lawvere is using is just any old set of $\mathcal{L}$-sentences. And by talk of 'models', Lawvere means structures apt for interpreting $\mathcal{L}$'s. (We'll cheerfully sidestep issues of size by assuming that there's only a set's-worth of such structures.)

We defined two posets. First, $\mathscr{C} = (C, \preccurlyeq)$, where C is a collection of sets of $\mathcal{L}$-sentences, and the ordering is set-inclusion. Second, $\mathscr{D} = (D, \sqsubseteq)$, where D is a collection of sets of $\mathcal{L}$-structures, and the ordering is the inverse of set-inclusion. Then we met two functions which we can define like this (using φ, σ as variables over sentences and structures respectively)

(1) $F\colon \mathscr{C} \to \mathscr{D}$ is such that $Fc = \{\sigma \mid (\forall \varphi \in c)\, \sigma \vDash \varphi\}$,

(2) $G\colon \mathscr{D} \to \mathscr{C}$ is such that $Gd = \{\varphi \mid (\forall \sigma \in d)\, \sigma \vDash \varphi\}$,

where $\sigma \vDash \varphi$ if φ is true interpreted in the structure σ.

Put like that, Theorem 164 (with the generating relation R between a sentence and a structure the converse of $\vDash$) immediately gives us

Theorem 165. *$F \dashv G$ is a Galois connection between $\mathscr{C}$ and $\mathscr{D}$.*

(b) Now we can just turn the handle, and apply all those general theorems about Galois connections from the preceding sections to our special case of the connection between the 'syntax' $\mathscr{C}$ and 'semantics' $\mathscr{D}$, recovering the sorts of results listed at the end of §32.2 and more. Of course, we get no exciting new logical news this way. But that's not the name of the game. The point rather is this. We take the fundamental *true-of* relation which can obtain between an $\mathcal{L}$-sentence and an $\mathcal{L}$-structure: this immediately generates a certain Galois connection $F \dashv G$ between two naturally ordered 'syntactic' and 'semantic' posets, and this in turn already dictates that e.g. the composite maps GF and FG will have special significance as closure operations. So we come to see some familiar old logical ideas as exemplifying essentially general order-theoretic patterns which recur elsewhere. And that's illuminating.

33 Adjoints introduced

NB: This chapter, and the next two, are taken, unrevised, from an earlier set of Notes on Category Theory. They continue the story without, I hope, too many jarring discontinuities. These chapters are less gentle than what's gone before and need a great deal of rewriting, not to mention checking for bad errors! However, if you have got this far then they should still be manageable and will hopefully be useful as a Rough Guide to adjunctions.

Recall that quotation from Tom Leinster which we gave at the very outset:

> Category theory takes a bird's eye view of mathematics. From high in the sky, details become invisible, but we can spot patterns that were impossible to detect from ground level. (Leinster 2014, p. 1)

Perhaps the most dramatic patterns that category theory newly reveals are those which involve *adjunctions*. As Mac Lane famously puts it (1997, p. vii) the slogan is "Adjoint functors arise everywhere." In the last two chapters, we have seen a restricted version of the phenomenon (well known before category theory). But category theory enables us to generalize radically.

33.1 Adjoint functors: a first definition

(a) Let $\mathscr{P}$ now be (not the poset itself but) the category corresponding to the poset $(P, \preccurlyeq)$. So the objects of $\mathscr{P}$ are the members of P, and there is a $\mathscr{P}$-arrow $p \to p'$ (for $p, p' \in P$), which we can identify with the pair $\langle p, p' \rangle$, if and only if $p \preccurlyeq p'$. Similarly let $\mathscr{Q}$ be the category corresponding to the poset $(Q, \sqsubseteq)$.

Now, changing symbolism just a little, a Galois connection between the posets $(P, \preccurlyeq)$ and $(Q, \sqsubseteq)$ is a pair of functions $f\colon P \to Q$ and $g\colon Q \to P$ such that

(i) f and g are monotone, and

(ii) $f(p) \sqsubseteq q$ iff $p \preccurlyeq g(q)$ for all $p \in P, q \in Q$.

(Well, we know condition (ii) implies condition (i), but it is helpful now to make it explicit.) However, monotone functions f, g between posets give rise to functors F, G between the corresponding categories – see §20.2, Ex. (F6). Thus the monotone function $f\colon P \to Q$ gives rise to the functor $F\colon \mathscr{P} \to \mathscr{Q}$ which sends the object p in $\mathscr{P}$ to $f(p)$ in $\mathscr{Q}$, and sends an arrow $p \to p'$ in $\mathscr{P}$, i.e. the pair $\langle p, p' \rangle$, to the pair $\langle f(p), f(p') \rangle$ which is an arrow in $\mathscr{Q}$. Similarly, $g\colon Q \to P$ gives rise to a functor $G\colon \mathscr{Q} \to \mathscr{P}$.

So (ii) means that our adjoint *functions*, i.e. the Galois connection (f, g) between the posets $(P, \preccurlyeq)$ and $(Q, \sqsubseteq)$, gives rise to a pair of *functors* (F, G) between the poset categories $\mathscr{P}$ and $\mathscr{Q}$, one in each direction, such that there is a (unique) arrow $Fp \to q$ in $\mathscr{Q}$ iff there is a corresponding (unique) arrow $p \to Gq$ in $\mathscr{P}$. This sets up an isomorphism between the hom-sets $\mathscr{Q}(Fp, q)$ and $\mathscr{P}(p, Gq)$, for each $p \in \mathscr{P}, q \in \mathscr{Q}$.

Of course, for a particular choice of p, q, this will be a rather trivial isomorphism, as the homsets in this case are either both empty or both single-membered. But what isn't trivial is that the existence of the isomorphism arises *systematically* from the Galois connection, in a uniform and natural way. And we now know how to put that informal claim into more formal category-theoretic terms: we have a *natural isomorphism* here, i.e. $\mathscr{Q}(Fp, q) \cong \mathscr{P}(p, Gq)$ *naturally* in $p \in \mathscr{P}, q \in \mathscr{Q}$.

(b) Now we generalize this last idea in the obvious way, and also introduce some absolutely standard notation:

Definition 132. Suppose $\mathscr{A}$ and $\mathscr{B}$ are categories and $F\colon \mathscr{A} \to \mathscr{B}$ and $G\colon \mathscr{B} \to \mathscr{A}$ are functors. Then F *is left adjoint to* G *and* G *is right adjoint to* F, notated $F \dashv G$, iff

$$\mathscr{B}(F(A), B) \cong \mathscr{A}(A, G(B))$$

naturally in $A \in \mathscr{A}$, $B \in \mathscr{B}$. We also write $\begin{tikzcd} \mathscr{A} \arrow[r, shift left, "F"] \arrow[r, phantom, "\perp"] & \mathscr{B} \arrow[l, shift left, "G"] \end{tikzcd}$ when this situation obtains, or $F \dashv G\colon \mathscr{A} \to \mathscr{B}$, and we say that F and G (together with the associated isomorphism between the relevant hom-sets) form an *adjunction*.

Here, and onwards through our discussions of adjunctions, we'll take it that there is no problem in talking about the relevant hom-*sets* (either because the categories are small enough, or because we are taking a relaxedly inclusive line on what counts as 'sets').

There is an additional fairly standard bit of notation to indicate the action of the natural isomorphism between the hom-sets in an adjunction:

Definition 133. Given the situation just described, and an arrow $f\colon F(A) \to B$, then one direction of the natural bijection between the hom-sets sends that arrow to its *transpose* $\overline{f}\colon A \to G(B)$; likewise the inverse bijection associates an arrow $g\colon A \to G(B)$ to its transpose $\overline{g}\colon F(A) \to B$.

(Another common notation distinguishes $f^\flat$ for our $\overline{f}$ and $g^\sharp$ for our $\overline{g}$, and this notation might be preferable in principle since transposing by 'sharpening' and 'flattening' are indeed different operations. But the double use of the overlining notation is standard, and is slick.)

Evidently, transposing twice takes us back to where we started: $\overline{\overline{f}} = f$ and $\overline{\overline{g}} = g$.

33.2 Examples

As we'd expect from our discussion of Galois connections, given the existence of an adjoint connection $F \dashv G$ we can deduce a range of additional properties of the adjoint functors and of the operation of transposition. But before exploring this any further in the abstract, let's have some more examples of adjunctions (to add to those generated by Galois connections).

For a warm-up exercise, we start with a particularly easy case:

(1) Consider any (non-empty!) category $\mathscr{A}$ and the one object category $\mathbf{1}$ (comprising just the object $\bullet$ and its identity arrow). There is a unique functor $F\colon \mathscr{A} \to \mathbf{1}$. Questions: when does F have a right adjoint $G\colon \mathbf{1} \to \mathscr{A}$? what about a left adjoint?

If G is to be a right adjoint, remembering that $FA = \bullet$ for any $A \in \mathscr{A}$, we require

$$\mathbf{1}(\bullet, \bullet) \cong \mathscr{A}(A, G\bullet),$$

for any A. The hom-set on the left contains just the identity arrow. So that can only be in bijection to the hom-set on the right, for each A, if there is always a *unique* arrow $A \to G\bullet$, i.e. if $G\bullet$ is terminal in $\mathscr{A}$.

In sum, F has a right adjoint $G\colon \mathbf{1} \to \mathscr{A}$ just in case G sends $\mathbf{1}$'s unique object to $\mathscr{A}$'s terminal object: no terminal object, no right adjoint.

Dually, F has a left adjoint if and only if $\mathscr{A}$ has an initial object.

This toy example reminds of what we have already seen in the special case of Galois connections, namely that a functor may or may not have a right adjoint, and independently may or may not have a left adjoint, and if both adjoints exist they may be different. But let's also note that we have here a first indication that adjunctions and limits can interact in interesting way: in this case, indeed, we could *define* terminal and initial objects for a category $\mathscr{A}$ in terms of the existence of right and left adjoints to the functor $F\colon \mathscr{A} \to \mathbf{1}$. We will return to this theme.

Now for a couple of more substantive examples. And to speed things along, we will procede informally: we won't in this section actually prove that the relevant hom-sets in our various examples are naturally isomorphic in the official formal sense, but rather we will take it as enough to find a bijection which can be evidently set up in a systematic and intuitively natural way, without arbitrary choices.

(2) Let's next consider the forgetful functor $U\colon \mathbf{Top} \to \mathbf{Set}$ which sends each topological space to its underlying set of points, and sends any continuous function between topological spaces to the same function thought of as a set-function. Questions: does this have a left adjoint? a right adjoint?

If U is to have a left adjoint $F\colon \mathbf{Set} \to \mathbf{Top}$, then for any set S and for any topological space (T, O) – with T a set of points and O a topology (a suitable collection of open sets) – we require

$$\mathbf{Top}(F(S), (T, O)) \cong \mathbf{Set}(S, U(T, O)) = \mathbf{Set}(S, T),$$

where the bijection here needs to be a natural one.

Now, on the right we have the set of *all* functions $f\colon S \to T$. So that needs to be in bijection with the set of all *continuous* functions from FS to (T, O). How can we ensure this holds in a systematic way, for any S and (T, O)? Well, suppose that for any S, F sends S to the topological space (S, D) which has the discrete topology (i.e. all subsets of S count as open). It is a simple exercise to show that *every* function $f\colon S \to T$ then counts as a continuous function $f\colon (S, D) \to (T, O)$. So the functor F which assigns a set the discrete topology will indeed be left adjoint to the forgetful functor.

Similarly, the functor $G\colon \mathbf{Set} \to \mathbf{Top}$ which assigns a set the indiscrete topology (the only open sets are the empty set and S itself) is right adjoint to the forgetful functor U.

(3) Let's now take another case of a forgetful functor, this time the functor $U\colon \mathbf{Mon} \to \mathbf{Set}$ which forgets about monoidal structure. Does U have a left adjoint $F\colon \mathbf{Set} \to \mathbf{Mon}$. If $(M, \cdot)$ is a monoid and S some set, we need

$$\mathbf{Mon}(FS, (M, \cdot)) \cong \mathbf{Set}(S, U(M, \cdot)) = \mathbf{Set}(S, M).$$

The hom-set on the right contains all possible functions $f\colon S \to M$. How can these be in one-one correspondence with the monoid homomorphisms from FS to $(M, \cdot)$?

Arm-waving for a moment, suppose FS is some monoid with a lot of structure (over and above the minimum required to be a monoid). Then there may be few if any monoid homomorphisms from FS to $(M, \cdot)$. Therefore, if there are potentially to be *lots* of such monoid homorphisms, one for each $f\colon S \to M$, then FS will surely need to have minimal structure. Which suggests going for broke and considering the limiting case, i.e. the functor F which sends a set S to $(S^*, *)$, the *free* monoid on S which we met back in §20.5, Ex. (F13). Recall, the objects of $(S^*, *)$ are sequences of S-elements (including the null sequence) and its monoid operation is concatenation.

There is an obvious map α which takes an arrow $f\colon S \to M$ and sends it to $\overline{f}\colon (S^*, *) \to (M, \cdot)$, where $\overline{f}$ sends the empty sequence of S-elements to the unit of M, and sends the finite sequence $x_1 * x_2 * x_3 * \ldots * x_n$ to the M-element $fx_1 \cdot fx_2 \cdot fx_3 \cdot \ldots \cdot fx_n$. So defined, $\overline{f}$ respects the unit and the monoid operation and so is a monoid homomorphism.

There is an equally obvious map β which takes an arrow $g\colon (S^*, *) \to (M, \cdot)$ to the function $\overline{g}\colon S \to M$ which sends an element $x \in S$ to $g\langle x\rangle$ (i.e. to g applied to the one-element list containing x).

Evidently α and β are inverses, so form a bijection, and their construction is quite general (i.e. can be applied to any set S and monoid $(M, \cdot)$). Which establishes that, as required $\mathbf{Mon}(FS, (M, \cdot)) \cong \mathbf{Set}(S, M)$.

So in sum, the free functor F which takes a set to the free monoid on that set is left adjoint to the forgetful functor U which sends a monoid to its underlying set.

Now recall Theorem 158: if a function f has a left or right adjoint to make up a Galois connection, then that adjoint is unique. An analogous uniqueness result applies to adjoints more generally: if a functor has a left adjoint, then it is unique up to isomorphism, and likewise right adjoints (when they exist) are unique up to isomorphism. So we can say that the functor which assigns a set the indiscrete topology is in fact *the* right adjoint to the functor which forgets topological structure, and we can say that the functor sending a set to the free monoid on that set is *the* left adjoint of the forgetful functor on monoids. However, the uniqueness theorem for adjoints takes a bit of work; so we'll delay the proof until the next chapter, §34.4. For the moment, then, we'll officially continue simply to talk of one functor being left (right) adjoint to another without making explicit uniqueness claims.

Our example involving monoids is actually typical of a whole cluster of cases. A left adjoint of the trivial forgetful functor from some class of algebraic structures to their underlying sets is characteristically provided by the non-trivial functor that takes us from a set to a free structure of that algebraic kind. Thus we have, for example,

(4) The forgetful functor $U\colon \mathbf{Grp} \to \mathbf{Set}$ has as a left adjoint the functor $F\colon \mathbf{Set} \to \mathbf{Grp}$ which sends a set to the free group on that set (i.e. the group obtained from a set S by adding just enough elements for it to become a group while imposing no constraints other than those required to ensure we indeed have a group).

What about *right* adjoints to our last two forgetful functors?

(5) We will later show that the forgetful functor $U\colon \mathbf{Mon} \to \mathbf{Set}$ has no right adjoint by a neat proof in §35.3. But here's a more arm-waving argument. U would have a right adjoint $G\colon \mathbf{Set} \to \mathbf{Mon}$ just in case $\mathbf{Set}(M, S) = \mathbf{Set}(U(M, \cdot), S) \cong \mathbf{Mon}((M, \cdot), GS)$, for all monoids $(M, \cdot)$ and sets S. But this requires the monoid homomorphisms from $(M, \cdot)$ to GS always to be in bijection with the set-functions from M to S. But that's not possible (consider keeping the sets M and S fixed, but changing the possible monoid operations with which M is equipped).

Similarly the forgetful functor $U\colon \mathbf{Grp} \to \mathbf{Set}$ has no right adjoint.

(6) There are however examples of 'less forgetful' algebraic functors which have both left and right adjoints. Take the functor $U\colon \mathbf{Grp} \to \mathbf{Mon}$ which forgets about group inverses but keeps the monoidal structure. This has a left adjoint $F\colon \mathbf{Mon} \to \mathbf{Grp}$ which converts a monoid to a group by adding inverses for elements (and otherwise making no more assumptions that are needed to get a group). U also has a right adjoint $G\colon \mathbf{Mon} \to \mathbf{Grp}$ which rather than adding elements subtracts them by mapping a monoid to the submonoid of its invertible elements (which can be interpreted as a group).

Let's quickly check just the second of those claims. We have $U \dashv G$ so long as

$$\mathbf{Mon}(U(K,\times),(M,\cdot)) \cong \mathbf{Grp}((K,\times),G(M,\cdot)),$$

for any monoid $(M,\cdot)$ and group $(K,\times)$. Now we just remark that every element of $(K,\times)$-as-a-monoid is invertible and a monoid homomorphism sends invertible elements to invertible elements. Hence a monoid homomorphism from $(K,\times)$-as-a-monoid to $(M,\cdot)$ will in fact also be a group homomorphism from $(K,\times)$ to the submonoid-as-a-group $G(M,\cdot)$.

(7) Recall the functor $F\colon \mathbf{Set} \to \mathbf{Rel}$ which 'forgets' that arrows are functional (see §20.2, Ex. (F2)). And now we introduce a powerset functor $P\colon \mathbf{Rel} \to \mathbf{Set}$ defined as follows:

a) P sends a set A to its powerset $\mathcal{P}(A)$, and

b) P sends a relation R in $A \times B$ to the function $f_R\colon \mathcal{P}(A) \to \mathcal{P}(B)$ which sends $X \subseteq A$ to $Y = \{b \mid (\exists x \in X)Rxb\} \subseteq B$.

Claim: $F \dashv P$.

We observe that there is a (natural!) isomorphism which correlates a relation R in $A \times B$ with a function $f\colon A \to \mathcal{P}(B)$ where $f(x) = \{y \mid Rxy\}$ and so Rxy iff $y \in f(x)$. This gives us a isomorphism $\mathbf{Rel}(FA,B) \cong \mathbf{Set}(A,PB)$ which can be checked to be natural in $A \in \mathbf{Set}$ and $B \in \mathbf{Rel}$.

And now for some cases not involving forgetful functors:

(8) Suppose $\mathscr{C}$ is a category with exponentiation (and hence with products). Then, in a slogan, exponentiation by B is right adjoint to taking the product with B.

To see this, we define a pair of functors from $\mathscr{C}$ to itself. First, there is the functor $- \times B\colon \mathscr{C} \to \mathscr{C}$ which sends an object A to the product $A \times B$, and sends an arrow $f\colon A \to A'$ to $f \times 1_B\colon A \times B \to A' \times B$.

Second there is the functor $(-)^B\colon \mathscr{C} \to \mathscr{C}$ which sends an object C to C^B, and sends an arrow $f\colon C \to C'$ to $\overline{f \circ ev}\colon C^B \to C'^B$ as defined in the proof that (F15) is functor in §20.6. It is easily checked that $(-)^B$ satisfies the conditions for functoriality.

By the theorem just mentioned, $\mathscr{C}(A \times B, C) \cong \mathscr{C}(A, C^B)$ naturally in A and C. Hence $(- \times B) \dashv (-)^B$.

(9) Recall Defn. 23 which defined the product of two categories. Given a category $\mathscr{C}$ there is a trivial diagonal functor $\Delta\colon \mathscr{C} \to \mathscr{C} \times \mathscr{C}$ which sends a $\mathscr{C}$-object A to the pair $\langle A, A\rangle$, and sends a $\mathscr{C}$-arrow f to the pair of arrows $\langle f, f\rangle$. What would it take for this functor to have a right adjoint $G\colon \mathscr{C} \times \mathscr{C} \to \mathscr{C}$? We'd need

$$(\mathscr{C} \times \mathscr{C})(\langle A, A\rangle, \langle B, C\rangle) \cong \mathscr{C}(A, G\langle B, C\rangle)$$

naturally in $A \in \mathscr{C}$ and in $\langle B, C\rangle \in \mathscr{C} \times \mathscr{C}$. But by definition the left hand hom-set is $\mathscr{C}(A,B) \times \mathscr{C}(A,C)$. But then if we can take G to be the product functor that sends $\langle B, C\rangle$ to the product object $B \times C$ in $\mathscr{C}$ we'll get an obvious natural isomorphism

$$\mathscr{C}(A, B) \times \mathscr{C}(A, C) \cong \mathscr{C}(A, B \times C).$$

So in sum, $\Delta\colon \mathscr{C} \to \mathscr{C} \times \mathscr{C}$ has a right adjoint if $\mathscr{C}$ has binary products.

(10) For topologists, let's simply mention another example of a case where the adjoint of a trivial functor is something much more substantial. The inclusion functor from **KHaus**, the category of compact Hausdorff spaces, into **Top** has a left adjoint, namely the Stone-Čech compactification functor.

33.3 Naturality

We said: $F \dashv G\colon \mathscr{A} \to \mathscr{B}$ just in case

$$\mathscr{B}(F(A), B) \cong \mathscr{A}(A, G(B))$$

holds naturally in $A \in \mathscr{A}$, $B \in \mathscr{B}$. Let's now be more explicit about what the official naturality requirement comes to.

By Defn. 107, the required bijection holds naturally in B (to take that case first) just if the two hom-functors $\mathscr{B}(F(A), -)$ and $\mathscr{A}(A, G(-))$ are naturally isomorphic. By Defn. 106, that means there have to be isomorphisms $\varphi_B : \mathscr{B}(F(A), B) \to \mathscr{A}(A, G(B))$, one for each B, such that for every $h\colon B \to B'$, the usual naturality square always commutes:

$$\begin{tikzcd}
\mathscr{B}(F(A), B) \arrow[r, "\mathscr{B}(F(A),h)"] \arrow[d, "\varphi_B"] & \mathscr{B}(F(A), B') \arrow[d, "\varphi_{B'}"] \\
\mathscr{A}(A, G(B)) \arrow[r, "\mathscr{A}(A,G(h))"] & \mathscr{A}(A, G(B'))
\end{tikzcd}$$

But how does the covariant hom-functor $\mathscr{B}(F(A), -)$ operate on $h\colon B \to B'$? As we saw in §23.2, it sends h to $h \circ -$, i.e. to that function which composes h with an arrow from $\mathscr{B}(F(A), B)$ to give an arrow in $\mathscr{B}(F(A), B')$. Similarly, $\mathscr{A}(A, G(-))$ will send h to $Gh \circ -$.

So consider an arrow $f\colon F(A) \to B$ living in $\mathscr{B}(F(A), B)$. The naturality square now tells us that for any $h\colon B \to B'$, $\varphi_{B'}(h \circ f) = Gh \circ \varphi_B(f)$.

But (by the definition of transposition!), the components of φ send an arrow to its transpose. So we have shown the first part of the following theorem. And the second part of this theorem follows by a dual argument, in which some arrows get reversed because the relevant hom-functors in this case are contravariant.

Theorem 166. *Given $F \dashv G\colon \mathscr{A} \to \mathscr{B}$, then*

(1) for any $f\colon F(A) \to B$ and $h\colon B \to B'$, $\overline{h \circ f} = Gh \circ \overline{f}$,

(2) for any $g\colon A \to G(B)$ and $k\colon A' \to A$, $\overline{g \circ k} = \overline{g} \circ Fk$, i.e. $\overline{\overline{g} \circ Fk} = g \circ k$.

Inspecting the proof, we see that there is an obvious converse to this theorem. Given functors $F\colon \mathscr{A} \to \mathscr{B}$ and $G\colon \mathscr{B} \to \mathscr{A}$ such that there is always a bijection between $\mathscr{B}(F(A), B)$ and $\mathscr{A}(A, G(B))$ then, if conditions (1) and (2) hold, the bijections (for different As and Bs) will assemble into natural transformations, so that $\mathscr{B}(F(A), B) \cong \mathscr{A}(A, G(B))$ holds naturally in $A \in \mathscr{A}$, $B \in \mathscr{B}$, and hence $F \dashv G$.

33.4 An alternative definition

We now know what it takes for a pair of functors to be adjoint to each other, and we have given various examples of adjoint pairs (to add to the special cases from the previous two chapters where the adjunctions are Galois connections).

Now, our first definition of adjunctions was inspired by our original definition of Galois connections in §32.3. But we gave an alternative definition of such connections in §32.4. This too can be generalized to give a second definition of adjunctions. In this section we show how, and prove that the new definition is equivalent to our first one. (This alternative definition will turn out to look somewhat more complicated, but it is useful in practice – though for the moment our prime aim is to bring out something of the structural richness of adjunctions.)

A Galois connection between the posets $(P, \preccurlyeq)$, $(Q, \sqsubseteq)$, according to the alternative definition, comprises a pair of functions $f\colon P \to Q$ and $g\colon Q \to P$ such that

(i) f and g are monotone,

(ii) $p \preccurlyeq g(f(p))$ for all $p \in P$, and

(iii) $f(g(q)) \sqsubseteq q$ for all $q \in Q$.

Since the composition of monotone functions is monotone, (ii) and (iii) are in fact easily seen to be equivalent to

(ii′) if $p \preccurlyeq p'$, then $p \preccurlyeq p' \preccurlyeq g(f(p'))$ and $p \preccurlyeq g(f(p)) \preccurlyeq g(f(p'))$,

(iii′) if $q \sqsubseteq q'$, then $f(g(q)) \sqsubseteq q \sqsubseteq q'$ and $f(g(q)) \sqsubseteq f(g(q')) \sqsubseteq q'$.

As before, let $\mathscr{P}$ be the category corresponding to the poset $(P, \preccurlyeq)$, and recall that there is an arrow $p \to p'$ in $\mathscr{P}$ just when $p \preccurlyeq p'$ in the poset $(P, \preccurlyeq)$. Likewise for $\mathscr{Q}$ corresponding to $(Q, \sqsubseteq)$. And again as before, note that the monotone functions f, g between the posets give rise to functors F, G between the corresponding categories. Hence, in particular, the composite monotone function $g \circ f$ gives rise to a functor $G \circ F\colon \mathscr{P} \to \mathscr{P}$, and likewise $f \circ g$ gives rise to a functor $F \circ G\colon \mathscr{Q} \to \mathscr{Q}$.

Now, (ii′) corresponds in $\mathscr{P}$ to the claim that the following diagram always commutes:

$$\begin{array}{ccc} p & \longrightarrow & p' \\ \downarrow & & \downarrow \\ (G \circ F)p & \longrightarrow & (G \circ F)p' \end{array}$$

(We needn't label the arrows as in the poset category $\mathscr{P}$ arrows between objects are unique when they exist.)

Dropping the explicit sign for composition of functors for brevity's sake, let's define $\eta_p \colon 1_{\mathscr{P}} \Rightarrow GF$ to be the arrows $p \to GFp$, one for each $p \in \mathscr{P}$. Then our commutative diagram version of (ii′) can be revealingly redrawn as follows:

$$\begin{tikzcd} 1_{\mathscr{P}}\, p \arrow[r] \arrow[d, "\eta_p"] & 1_{\mathscr{P}}\, p' \arrow[d, "\eta_{p'}"] \\ GFp \arrow[r] & GFp' \end{tikzcd}$$

This commutes for all p, p'. So applying Defn. 108, this is just to say that the η_p assemble into a natural transformation $\eta \colon 1_{\mathscr{P}} \Rightarrow GF$ in $\mathscr{P}$.

Likewise, (iii) and hence (iii′) correspond to the claim that there is a natural transformation $\varepsilon \colon FG \Rightarrow 1_{\mathscr{Q}}$ in $\mathscr{Q}$.

(a) So far so good. We have here the initial ingredients for an alternative definition for an adjunction between functors $F \colon \mathscr{A} \to \mathscr{B}$ and $G \colon \mathscr{B} \to \mathscr{A}$: we will require there to be a pair of natural transformations $\eta \colon 1_{\mathscr{A}} \Rightarrow GF$ and $\varepsilon \colon FG \Rightarrow 1_{\mathscr{B}}$.

However, as we'll see, this isn't yet quite enough. But the additional ingredients we want are again suggested by our earlier treatment of Galois connections. Recall from Theorem 156 that if (f, g) is a Galois connection between $(P, \preccurlyeq)$ and $(Q, \sqsubseteq)$, then we immediately have the key identities

(iv) $f \circ g \circ f = f$, and

(v) $g \circ f \circ g = g$.

By (iv), $fp \preccurlyeq (f \circ g \circ f)p \preccurlyeq fp$, for p in $(P, \preccurlyeq)$. Hence in $\mathscr{P}$ the following diagram commutes for each p:

$$\begin{tikzcd} Fp \arrow[r] \arrow[dr] & FGFp \arrow[d] \\ & Fp \end{tikzcd}$$

Here, the diagonal arrow is the identity 1_{Fp}. The downward arrow is ε_{Fp} (the component of ε at Fp). And the horizontal arrow is $F\eta_p$. So we have $\varepsilon_{Fp} \circ F\eta_p = 1_{Fp}$ for each p.

Or what comes to the same, in the functor category $[\mathscr{P}, \mathscr{Q}]$ this diagram commutes[1]

$$\begin{tikzcd} F \arrow[r, "F\eta"] \arrow[dr, "1_F"'] & FGF \arrow[d, "\varepsilon F"] \\ & F \end{tikzcd}$$

For remember whiskering(!), discussed in §26.3: the components $F(\eta_p)$ assemble into the natural transformation we symbolized '$F\eta$', and the components ε_{Fp} assemble into the natural transformation we symbolized 'ε_F'. And then recall from §27.1 that 'vertical' composition of natural transformations between e.g. the functors $F\colon \mathscr{A} \to \mathscr{B}$ and $FGF\colon \mathscr{A} \to \mathscr{B}$ is defined component-wise. So, for each p,

$$(\varepsilon F \circ F\eta)_p = \varepsilon_{Fp} \circ F\eta_p = 1_{Fp} = (1_F)_p,$$

where 1_F is the natural transformation whose component at p is 1_{Fp}. Since all components are equal, the left-most and right-most natural transformations in that equation are equal and our diagram commutes.

Exactly similarly, from (v) we infer that $G\varepsilon_q \circ \eta_{Gq} = 1_{Gq}$. In other words, the next diagram commutes in $[\mathscr{Q}, \mathscr{P}]$:

$$\begin{tikzcd} G \arrow[r, "\eta G"] \arrow[dr, "1_G"'] & GFG \arrow[d, "G\varepsilon"] \\ & G \end{tikzcd}$$

(b) And *now* we can put everything together to give us our second definition for adjoint functors:

Definition 134 (Alternative). Suppose $\mathscr{A}$ and $\mathscr{B}$ are categories and $F\colon \mathscr{A} \to \mathscr{B}$ and $G\colon \mathscr{B} \to \mathscr{A}$ are functors. Then F *is left adjoint to* G *and* G *is right adjoint to* F, notated $F \dashv G$, iff

(i) there are natural transformations $\eta\colon 1_{\mathscr{A}} \Rightarrow GF$ and $\varepsilon\colon FG \Rightarrow 1_{\mathscr{B}}$ such that

(ii) $\varepsilon_{FA} \circ F\eta_A = 1_{FA}$ for all $A \in \mathscr{A}$, and $G\varepsilon_B \circ \eta_{GB} = 1_{GB}$ for all $B \in \mathscr{B}$; or equivalently

(ii′) the following *triangle identities* hold in the functor categories $[\mathscr{A}, \mathscr{B}]$ and $[\mathscr{B}, \mathscr{A}]$ respectively:

[1]Notational fine print: our convention has been to use single arrows to represent arrows inside particular categories, and double arrows to represent natural transformations between functors across categories. We are now dealing with natural-transformations-thought-of-as-arrows-within-a-particular-functor-category. Some use double arrows for diagrams in a functor category, to remind us these are natural transformations (between functors relating some other categories); some use single arrows because these are being treated as arrows (in the functor category). I'm jumping the second way, following the majority and also getting slightly cleaner diagrams.

$$\begin{tikzcd} F \arrow[r, "F\eta"] \arrow[dr, "1_F"'] & FGF \arrow[d, "\varepsilon F"] \\ & F \end{tikzcd} \qquad \begin{tikzcd} G \arrow[r, "\eta G"] \arrow[dr, "1_G"'] & GFG \arrow[d, "G\varepsilon"] \\ & G \end{tikzcd}$$

Note, η and ε are standardly called the *unit* and *counit* of the adjunction.

It remains to show that Defn. 132 and Defn. 134 are equivalent:

Theorem 167. *For given functors $F\colon \mathscr{A} \to \mathscr{B}$ and $G\colon \mathscr{B} \to \mathscr{A}$, $F \dashv G$ holds by our original definition iff it holds by the alternative definition.*

Proof (If). Suppose there are natural transformations $\eta\colon 1_{\mathscr{A}} \Rightarrow GF$ and $\varepsilon\colon FG \Rightarrow 1_{\mathscr{B}}$ for which the triangle identities hold.

Take any f in $\mathscr{B}(F(A), B)$. Then $\eta_A\colon A \to GF(A)$ and $G(f)\colon GF(A) \to GB$ compose. And so we can define $\varphi_{AB}\colon \mathscr{B}(F(A), B) \to \mathscr{A}(A, G(B))$ by putting $\varphi_{AB}(f) = G(f) \circ \eta_A$.

Likewise, we can define $\psi_{AB}\colon \mathscr{A}(A, G(B)) \to \mathscr{B}(F(A), B)$ by putting $\psi_{AB}(g) = \varepsilon_B \circ F(g)$ for any $g\colon A \to G(B)$.

Keep A fixed: then, as we vary B, the various components φ_{AB} assemble into a natural transformation $\varphi_A\colon \mathscr{B}(F(A), -) \Rightarrow \mathscr{A}(A, G(-))$. That's because the naturality square

$$\begin{tikzcd} \mathscr{B}(F(A), B) \arrow[r, "h\,\circ\,-"] \arrow[d, "\varphi_{AB}"] & \mathscr{B}(F(A), B') \arrow[d, "\varphi_{AB'}"] \\ \mathscr{A}(A, G(B)) \arrow[r, "Gh\,\circ\,-"] & \mathscr{A}(A, G(B')) \end{tikzcd}$$

commutes for every $h\colon B \to B'$, i.e. for every f in $\mathscr{B}(F(A), B)$ we have

$$\varphi_{AB'}(h \circ f) = G(h \circ f) \circ \eta_A = Gh \circ (Gf \circ \eta_A) = Gh \circ \varphi_{AB}(f)$$

which holds by the functoriality of G.

Now keep B fixed: then by a parallel argument, as we vary A, the various components φ_{AB} assemble into a natural transformation $\varphi_B\colon \mathscr{B}(F(-), B) \Rightarrow \mathscr{A}(-, G(B))$ between the two *contravariant* functors.

Similarly if we keep A fixed, the various ψ_{AB} assemble into a natural transformation $\psi_A\colon \mathscr{A}(A, G(-)) \Rightarrow \mathscr{B}(F(A), -)$; and if we keep B fixed, the various ψ_{AB} assemble into $\psi_B\colon \mathscr{A}(-, G(B)) \Rightarrow \mathscr{B}(F(-), B)$.

We now need to show that these natural transformations are isomorphisms, from which the desired result will follow: i.e. $\mathscr{B}(F(A), B) \cong \mathscr{A}(A, G(B))$ naturally in $A \in \mathscr{A}$ and in $B \in \mathscr{B}$.

We show each φ_{AB} and ψ_{AB} are mutually inverse. Take any $f\colon FA \to B$. Then

$$\begin{aligned}
\psi_{AB}(\varphi_{AB}(f)) &= \psi_{AB}(G(f)\circ\eta_A) && \text{by definition of } \varphi\\
&= \varepsilon_B \circ F(G(f)\circ\eta_A)) && \text{by definition of } \psi\\
&= \varepsilon_B \circ FGf \circ F\eta_A && \text{by functoriality of } F\\
&= f \circ \varepsilon_{FA} \circ F\eta_A && \text{by naturality square for } \varepsilon\\
&= f \circ 1_{FA} && \text{by triangle equality}\\
&= f
\end{aligned}$$

Hence $\psi_{AB} \circ \varphi_{AB} = 1$ (note how we did need to appeal to the added triangle equality, not just functoriality and the naturality of ε). Likewise $\varphi_{AB} \circ \psi_{AB} = 1$. □

Proof (Only if). Suppose $\mathscr{B}(F(A), B) \cong \mathscr{A}(A, G(B))$ naturally in $A \in \mathscr{A}$ and in $B \in \mathscr{B}$. We need to define a unit and counit for the adjunction, and show they satisfy the triangle equalities.

Take the identity arrow 1_{FA} in $\mathscr{B}(FA, FA)$. The natural isomorphism defining the adjunction sends 1_{FA} to an arrow we will hopefully call $\eta_A\colon A \to GF(A)$.

We first show that the components η_A do indeed assemble into a natural transformation from $1_{\mathscr{A}}$ to GF. So consider the following two diagrams:

$$\begin{tikzcd}
FA \arrow[r, "Ff"] \arrow[d, "1_{FA}"] & FA' \arrow[d, "1_{FA'}"] \\
FA \arrow[r, "Ff"] & FA'
\end{tikzcd}
\qquad
\begin{tikzcd}
A \arrow[r, "f"] \arrow[d, "\eta_A"] & A' \arrow[d, "\eta_{A'}"] \\
GFA \arrow[r, "GFf"] & GFA'
\end{tikzcd}$$

Trivially, the diagram on the left commutes for all $f\colon A \to A'$. That is to say, $Ff \circ 1_{FA} = 1_{FA'} \circ Ff$. Transposition must evidently preserve identities. So $\overline{Ff \circ 1_{FA}} = \overline{1_{FA'} \circ Ff}$. But by the first of the naturality requirements in §33.3, $\overline{Ff \circ 1_{FA}} = GFf \circ \overline{1_{FA}} = GFf \circ \eta_A$. And by the other naturality requirement, $\overline{1_{FA'} \circ Ff} = \overline{\overline{\eta_{A'}} \circ Ff} = \eta_{A'} \circ f$. So we have $GFf \circ \eta_A = \eta_{A'} \circ f$ and the diagram on the right commutes for all f. Hence the components η_A do indeed assemble into a natural transformation.

Similarly the same natural isomorphism in the opposite direction sends 1_{GB} to its transpose $\varepsilon_B\colon FG(B) \to B$, and the components ε_B assemble into a natural transformation from FG to $1_{\mathscr{B}}$.

Now consider these two diagrams:

$$\begin{tikzcd}
A \arrow[r, "\eta_A"] \arrow[d, "\eta_A"] & GFA \arrow[d, "1_{GFA}"] \\
GFA \arrow[r, "1_{GFA}"] & GFA
\end{tikzcd}
\qquad
\begin{tikzcd}
FA \arrow[r, "F\eta_A"] \arrow[d, "1_{FA}"] & FGFA \arrow[d, "\varepsilon_{FA}"] \\
FA \arrow[r, "1_{FA}"] & FA
\end{tikzcd}$$

The diagram on the left trivially commutes. Transpose it via the natural isomorphism that defines the adjunction and use the naturality requirements again; we find that the diagram on the right must also commute. So $\varepsilon_{FA} \circ F\eta_A = 1_{FA}$ for all $A \in \mathscr{A}$ – which gives us one of the triangle identities. The other identity we get dually. □

We are done. But although the strategies for proving the equivalence of our definitions are entirely straightforward, checking the details was a bit tedious and required keeping our wits about us. So let's pause before resuming in the next chapter the exploration of adjunctions.

33.5 Adjoints and equivalent categories

Our second definition of an adjunction should remind you strongly of our earlier characterization of what it takes for categories to be equivalent. We should pause to say something about this.

We can slightly recast our definitions to highlight the parallelism:

Definition 116* An equivalence between categories $\mathscr{A}$ and $\mathscr{B}$ is a pair of functors $F\colon \mathscr{A} \to \mathscr{B}$ and $G\colon \mathscr{B} \to \mathscr{A}$ and a pair of natural *isomorphisms* $\eta\colon 1_{\mathscr{A}} \Rightarrow GF$ and $\varepsilon\colon FG \Rightarrow 1_{\mathscr{B}}$.

Definition 134* An adjunction between categories $\mathscr{A}$ and $\mathscr{B}$ is a pair of functors $F\colon \mathscr{A} \to \mathscr{B}$ and $G\colon \mathscr{B} \to \mathscr{A}$ and a pair of natural *transformations* $\eta\colon 1_{\mathscr{A}} \Rightarrow GF$ and $\varepsilon\colon FG \Rightarrow 1_{\mathscr{B}}$ such that $\varepsilon_{FA} \circ F\eta_A = 1_{FA}$ for all $A \in \mathscr{A}$, and $G\varepsilon_B \circ \eta_{GB} = 1_{GB}$ for all $B \in \mathscr{B}$.

Since transformations need not be isomorphisms, an adjunction needn't be an equivalence (and indeed we have met lots of examples of adjunctions between non-equivalent categories). In the other direction, an isomorphism needn't satisfy the triangle identities, so an equivalence needn't be an adjunction either. However, we *do* have the following result:

Theorem 168. *If there is an equivalence between $\mathscr{A}$ and $\mathscr{B}$ constituted by a pair of functors $F\colon \mathscr{A} \to \mathscr{B}$ and $G\colon \mathscr{B} \to \mathscr{A}$ and a pair of natural isomorphisms $\eta\colon 1_{\mathscr{A}} \Rightarrow GF$ and $\gamma\colon FG \Rightarrow 1_{\mathscr{B}}$, then there is an adjunction $F \dashv G$ with unit η and counit ε (defined in terms of γ and η), and further there is also an adjunction $G \dashv F$.*

In other words, take an equivalence, fix one of the natural transformations, but tinker (if necessary) with the other, and we get an adjunction. Further we can construct an adjunction in the opposite direction.

Proof. Define the natural transformation ε by composition as follows:

$$\varepsilon\colon FG \xRightarrow{FG\gamma^{-1}} FGFG \xRightarrow{(F\eta G)^{-1}} FG \xRightarrow{\gamma} 1_{\mathscr{B}}$$

Since η and γ are isomorphisms, and by Theorem 114 whiskering natural isomorphisms yields another natural isomorphism, the inverses mentioned here must exist.

So we just need to establish that, with ε so defined, we get the usual triangle identities $\varepsilon_{FA} \circ F\eta_A = 1_{FA}$ for all $A \in \mathscr{A}$, and $G\varepsilon_B \circ \eta_{GB} = 1_{GB}$ for all $B \in \mathscr{B}$.

So, firstly, for any A, we need the composite arrow (*)

$$FA \xrightarrow{F\eta_A} FGFA \xrightarrow{(FG\gamma^{-1})_{FA}} FGFGFA \xrightarrow{(F\eta G)^{-1})_{FA}} FGFA \xrightarrow{\gamma_{FA}} FA$$

to equal the identity arrow on FA (recall, the component of a 'vertical' composite of natural transformations for FA is the composite of the components of the individual transformations).

We begin be noting that, for any $A \in \mathscr{A}$, the following square commutes by the naturality of η:

$$\begin{tikzcd}
A \arrow[r, "\eta_A"] \arrow[d, "\eta_A"] & GFA \arrow[d, "\eta_{GFA}"] \\
GFA \arrow[r, "GF\eta_A"] & GFGFA
\end{tikzcd}$$

So we have $\eta_{GFA} \circ \eta_A = GF\eta_A \circ \eta_A$. But since η_A is an isomorphism, it is epic (right-cancellable), so we have $\eta_{GFA} = GF\eta_A$ for all A. Similarly, we have $\gamma^{-1}_{FGB} = (FG\gamma^{-1})_B$ for all $B \in \mathscr{B}$.

So now consider the following diagram:

$$\begin{tikzcd}
FA \arrow[r, "F\eta_A"] \arrow[d, "(\gamma^{-1})_{FA}"] & FGFA \arrow[d, "(\gamma^{-1})_{FGFA}=(FG\gamma^{-1})_{FA}"] \\
FGFA \arrow[r, "FGF\eta_A"] \arrow[d, "1_{FGFA}"] & FGFGFA \arrow[dl, "(F\eta G)^{-1}_{FA}"] \\
FGFA \arrow[d, "\gamma_{FA}"] & \\
FA &
\end{tikzcd}$$

The top square commutes, being a standard naturality square. (Fill in the schema of Defn. 108 by putting the natural transformation $\alpha = \gamma^{-1} \colon 1_{\mathscr{B}} \to FG$, and put f to be the function $F\eta_A \colon FA \to FB$.) And the triangle below commutes because $FGF\eta_A = F\eta_{GFA}$ from the equation above and $F\eta_{GFA} = (F\eta G)_{FA}$ (since $\eta_{GFA} = (\eta G)_{FA}$), so the arrows along two sides are simply inverses, and therefore compose to the identity.

The whole diagram therefore commutes. The arrows on longer circuit from top-left to bottom form the composite (*). The arrows on the direct route from top to bottom compose to the identity 1_{FA}. The composites are equal and hence we have established that the first triangle identity holds.

The second triangle identity holds by a similar argument.

Hence $F \dashv G$. And finally we note that if we put $\eta' = \gamma^{-1}$ and $\gamma' = \eta^{-1}$, and put $F' = G$, $G' = F$, the same line of proof shows that $F' \dashv G'$, and so $G \dashv F$. $\square$

34 Adjoints further explored

NB: This chapter, like the previous one, is taken, unrevised, from an earlier set of Notes on Category Theory. It needs a great deal of rewriting, not to mention checking for bad errors! However, if you have got this far then it should still be useful.

We have given a pair of definitions of adjoint functors, mirroring the two alternative definitions of Galois connections. We showed the definitions to be equivalent, and met some initial examples of adjunctions.

In this chapter, after a couple of preliminary sections, we continue to generalize some of the most basic results we found for Galois connections to adjunctions more generally.

34.1 Adjunctions reviewed

Let's gather together what we know about adjunctions so far.

Suppose $F\colon \mathscr{A} \to \mathscr{B}$ and $G\colon \mathscr{B} \to \mathscr{A}$ are functors. Then F is left-adjoint to G (equivalently, G is right-adjoint to F), in symbols $F \dashv G$, or more fully $F \dashv G\colon \mathscr{A} \to \mathscr{B}$, iff the following conditions all hold together:

(1) $\mathscr{B}(FA, B) \cong \mathscr{A}(A, GB)$ naturally in $A \in \mathscr{A}$, $B \in \mathscr{B}$ – the isomorphism in each direction is said to send an arrow f in one hom-set to its transpose $\overline{f}$ in the other.

(2) There are natural transformations $\eta\colon 1_{\mathscr{A}} \Rightarrow GF$ and $\varepsilon\colon FG \Rightarrow 1_{\mathscr{B}}$ such that $\varepsilon_{FA} \circ F\eta_A = 1_{FA}$ for all $A \in \mathscr{A}$, and $G\varepsilon_B \circ \eta_{GB} = 1_{GB}$ for all $B \in \mathscr{B}$. η is said to be the unit, ε the counit of the adjunction.

(3) The component $\eta_A\colon A \to GFA$ of the natural transformation η can be identified as the transpose of $1_{FA}\colon FA \to FA$ under the natural isomorphism between $\mathscr{B}(FA, FA)$ and $\mathscr{A}(A, GFA)$. Likewise, the component ε_B is the transpose of 1_{GB} under the natural isomorphism between $\mathscr{A}(GB, GB)$ and $\mathscr{B}(FGB, B)$.

(4) The inverse isomorphisms from $\mathscr{B}(FA, B)$ to $\mathscr{A}(A, GB)$ and back can be identifed as $G(-)\circ\eta_A\colon \mathscr{B}(FA, B) \xrightarrow{\sim} \mathscr{A}(A, GB)$ and $\varepsilon_B \circ F(-)\colon \mathscr{A}(A, GB) \xrightarrow{\sim} \mathscr{B}(FA, B)$.

(5) For any $f\colon FA \to B$ and $h\colon B \to B'$, $\overline{h \circ f} = Gh \circ \overline{f}$; and for any $g\colon A \to GB$ and $k\colon A' \to A$, $\overline{g \circ k} = \overline{g} \circ Fk$, i.e. $\overline{\overline{g} \circ Fk} = g \circ k$.

These conditions are not independent, however: (1) and (2) are equivalent, and both then imply (3) to (5).

34.2 Two more theorems!

Using (2) and (4) in our conditions on adjunctions, it follows that if $F \dashv G$, then there is a natural transformation $\eta\colon 1_{\mathscr{A}} \Rightarrow GF$ which has the following 'universal mapping property': for any $g\colon A \to G(B)$ there is a unique associated $f\colon F(A) \to B$ such that $g = G(f) \circ \eta_A$.

It is worth noting that we can also prove the converse here, so we get a biconditional:

Theorem 169. *Given functors $F\colon \mathscr{A} \to \mathscr{B}$ and $G\colon \mathscr{B} \to \mathscr{A}$, then $F \dashv G$ iff (i) there is a natural transformation $\eta\colon 1_{\mathscr{A}} \Rightarrow GF$, for which (ii) for any $g\colon A \to G(B)$ in $\mathscr{A}$ there is a unique $f\colon F(A) \to B$ in $\mathscr{B}$ such that $g = G(f) \circ \eta_A$.*

Proof for 'if'. First use clause (i) and define $\varphi_{AB}\colon \mathscr{B}(F(A), B) \to \mathscr{A}(A, G(B))$ by putting $\varphi_{AB}(f) = G(f) \circ \eta_A$.

By same proof as for Theorem 167, when we keep A fixed the various components φ_{AB} assemble into a natural transformation $\varphi_A\colon \mathscr{B}(F(A), -) \Rightarrow \mathscr{A}(A, G(-))$. And when we keep B fixed, the various components φ_{AB} assemble into a natural transformation $\varphi_B\colon \mathscr{B}(F(-), B) \Rightarrow \mathscr{A}(-, G(B))$.

Further, by the uniqueness clause (ii) the components φ_{AB} are bijections, so the natural transformations are indeed natural isomorphisms. Therefore $\mathscr{B}(F(A), B) \cong \mathscr{A}(A, G(B))$ naturally in $A \in \mathscr{A}$, $B \in \mathscr{B}$. ☐

Our theorem has a dual companion of course:

Theorem 170. *Given functors $F\colon \mathscr{A} \to \mathscr{B}$ and $G\colon \mathscr{B} \to \mathscr{A}$, then $F \dashv G$ iff (i) there is a natural transformation $\varepsilon\colon FG \Rightarrow 1_{\mathscr{B}}$, for which (ii) for any $f\colon F(A) \to B$ there is a unique $g\colon A \to G(B)$ such that $f = \varepsilon_B \circ F(g)$.*

Evidently, we could have recruited either of these companion theorems as the basis of two further alternative definitions for $F \dashv G$ – as, for example, in (Awodey 2006, §9.1).

34.3 Adjunctions compose

Recall Theorem 157: in a different notation, if (f, g) is a Galois connection between the posets $\mathscr{P}$ and $\mathscr{Q}$, and (f', g') is a Galois connection between the posets $\mathscr{Q}$ and $\mathscr{R}$, then $(f' \circ f, g \circ g')$ is a Galois connections between $\mathscr{P}$ and $\mathscr{R}$.

Adjunctions similarly compose:

Theorem 171. *Given* $\mathscr{A} \underset{G}{\overset{F}{\underset{\longleftarrow}{\overset{\longrightarrow}{\perp}}}} \mathscr{B}$ *and* $\mathscr{B} \underset{G'}{\overset{F'}{\underset{\longleftarrow}{\overset{\longrightarrow}{\perp}}}} \mathscr{C}$ *, then* $\mathscr{A} \underset{GG'}{\overset{F'F}{\underset{\longleftarrow}{\overset{\longrightarrow}{\perp}}}} \mathscr{C}$*.*

Proof via homsets. Since $F' \dashv G'$, we have $\mathscr{C}(F'FA, C) \cong \mathscr{B}(FA, G'C)$, naturally in A – by Theorem 112(3) – and also naturally in C.

Also, since $F \dashv G$, we have $\mathscr{B}(FA, G'C) \cong \mathscr{A}(A, GG'C)$, naturally in A and in C.

So by Theorem 112(2), $\mathscr{C}(F'FA, C) \cong \mathscr{A}(A, GG'C)$ naturally in A and in C. Hence $F'F \dashv GG'$ $\square$

That was quick and easy. But there is perhaps some additional fun to be had by working through another argument:

Proof by units and counits. Since $F \dashv G$, there are a pair of natural transformations $\eta \colon 1_{\mathscr{A}} \Rightarrow GF$ and $\varepsilon \colon FG \Rightarrow 1_{\mathscr{B}}$, satisfying the usual triangle identities.

Since $F' \dashv G'$, there are natural transformations $\eta' : 1_{\mathscr{B}} \Rightarrow G'F'$ and $\varepsilon' : F'G' \Rightarrow 1_{\mathscr{C}}$, again satisfying the triangle identities.

We now define two more natural transformations by composition,

$$\eta'' : \quad 1_{\mathscr{A}} \xRightarrow{\ \eta\ } GF \xRightarrow{\ G\eta' F\ } GG'F'F$$

$$\varepsilon'' : \quad F'FGG' \xRightarrow{\ F'\varepsilon G'\ } F'G' \xRightarrow{\ \varepsilon'\ } 1_{\mathscr{C}}$$

To show $F'F \dashv GG'$ it suffices to check that η'' and ε'' also satisfy the triangle identities.

Consider, then, the following diagram:

$$\begin{tikzcd}
F'F \arrow[r, "F'F\eta"] \arrow[dr, "1_{F'F}"'] & F'FGF \arrow[r, "F'FG\eta'F"] \arrow[d, "F'\varepsilon F"] & F'FGG'F'F \arrow[d, "F'\varepsilon G'F'F"] \\
 & F'F \arrow[r, "F'\eta'F"] \arrow[dr, "1_{F'F}"'] & F'G'F'F \arrow[d, "\varepsilon'F'F"] \\
 & & F'F
\end{tikzcd}$$

'Whiskering' the triangle identity $\varepsilon F \circ F\eta = 1_F$ by F' shows that the top left triangle commutes. And whiskering the identity $\varepsilon'F' \circ F'\eta' = 1_{F'}$ on the other side by F shows that the bottom triangle commutes.

Further, the square commutes. For by either the naturality of ε or the naturality of η', the following square commutes in the functor category:

$$\begin{tikzcd} FG \arrow[r, "FG\eta'"] \arrow[d, "\varepsilon"] & FGG'F' \arrow[d, "\varepsilon G'F'"] \\ 1 \arrow[r, "\eta'"] & G'F' \end{tikzcd}$$

And whiskering again gives the commuting square in the big diagram. [Exercise: check the claims about whiskering and the naturality square.]

So the whole big diagram commutes, and in particular the outer triangle commutes. But that tells us that $\varepsilon''F'F \circ F'F\eta'' = 1_{F'F}$ – which is one of the desired triangle identities for η'' and ε''.

The other identity follows similarly. □

34.4 The uniqueness of adjoints

Now recall Theorem 158. This tells us that if (f, g) and (f, g') are both Galois connections between the posets $\mathscr{P}$ and $\mathscr{Q}$, then $g = g'$. Likewise, if (f, g) and (f', g) are both Galois connections between the same posets, then $f = f'$.

The corresponding result for adjunctions more generally is this:

Theorem 172. *Adjoints are unique up to natural isomorphism. If $F \dashv G$ and $F \dashv G'$ then $G \cong G'$. If $F \dashv G$ and $F' \dashv G$ then $F \cong F'$.*

Proof. Assume we have $F \dashv G\colon \mathscr{A} \to \mathscr{B}$ and $F \dashv G'\colon \mathscr{A} \to \mathscr{B}$. Then

$$\mathscr{A}(A, GB) \cong \mathscr{B}(FA, B) \cong \mathscr{A}(A, G'B)$$

naturally in $A \in \mathscr{A}$, $B \in \mathscr{B}$. It follows, using Theorem 112, that

$$(*) \quad \mathscr{A}(A, GB) \cong \mathscr{A}(A, G'B)$$

naturally in A and B.

(*)'s naturality in A means that $\mathscr{A}(-, GB) \cong \mathscr{A}(-, G'B)$, i.e. $\mathcal{Y}GB \cong \mathcal{Y}G'B$, where $\mathcal{Y}$ is the Yoneda embedding. And then, by Theorem 134, $GB \cong G'B$.

Moreover, this holds naturally in B – intuitively, because the isomorphism is generated systematically from the isomorphism in (*) which is also natural in B – so $G \cong G'$.

To confirm this, note that $\mathcal{Y}$ sends the diagram on the left in $\mathscr{A}$ to the diagram on the right in **Set** for any $f\colon B \to B'$:

$$\begin{tikzcd} GB \arrow[r, "Gf"] \arrow[d, "\beta_B"] & GB' \arrow[d, "\beta_{B'}"] \\ G'B \arrow[r, "G'f"] & G'B' \end{tikzcd} \qquad \begin{tikzcd} \mathscr{A}(-, GB) \arrow[r, "\mathscr{A}(-, Gf)"] \arrow[d, "\alpha_B"] & \mathscr{A}(-, GB') \arrow[d, "\alpha_{B'}"] \\ \mathscr{A}(-, G'B) \arrow[r, "\mathscr{A}(-, G'f)"] & \mathscr{A}(-, G'B') \end{tikzcd}$$

where the αs are components of the natural transformation required by the naturality of $(*)$ in B, and $\beta_B = \alpha_B(1_{GB})$ by appeal to Theorem 129. But $\mathcal{Y}$ is an embedding, remember, so each diagram commutes if and only if the other does. However, the diagram on the right commutes for all $f\colon B \to B'$ by the naturality in B; hence the diagram on the left does too (embeddings must evidently preserve commutativity relations). So the β assemble into a natural transformation between G and G'.

The proof of the second half of the theorem is dual. $\square$

We should note too an obvious companion theorem:

Theorem 173. *If $F \dashv G$ and $G \cong G'$ then $F \dashv G'$. Likewise, if $F \dashv G$ and $F \cong F'$ then $F' \dashv G$.*

Proof. By definition, given $F \dashv G\colon \mathscr{A} \to \mathscr{B}$, we have $\mathscr{B}(FA, B) \cong \mathscr{A}(A, GB)$ naturally in $A \in \mathscr{A}, B \in \mathscr{B}$.

But given $G \cong G'$, then it is almost immediate that $\mathscr{A}(A, GB) \cong \mathscr{A}(A, G'B)$, again naturally in $A \in \mathscr{A}, B \in \mathscr{B}$.

Hence by Theorem 112 again, $\mathscr{B}(FA, B) \cong \mathscr{A}(A, G'B)$, still naturally in $A \in \mathscr{A}, B \in \mathscr{B}$. Which means that $F \dashv G'$.

The other half of the theorem is dual. $\square$

34.5 How left adjoints can be defined in terms of right adjoints

Theorem 158 states that each component of a Galois connection uniquely fixes the other. So we would hope to be able to explicitly define one such component in terms of the other, and Theorem 159 in fact tells us how to do this. For example, assuming there is a connection (f, g) between the posets $(P, \preccurlyeq)$ and $(Q, \sqsubseteq)$, we can define the left adjoint in terms of the right by setting $f(p)$ to be the minimum of $\{q \in Q \mid p \preccurlyeq g(q)\}$ for every $p \in P$.

We have now shown, more generally, that each component of an adjunction uniquely fixes the other, at least up to isomorphism. We would expect that we can, similarly, characterize one functor in an adjunction in terms of its partner. So let's consider, in particular, how a left adjoint might be defined in terms of its right partner. (There will of course also be a dual story to be told about how right adjoints can be defined in terms of left ones. We can cheerfully leave spelling out the dual constructions and arguments as an exercise.)

Functions in Galois connections between posets correspond to adjoint functors between poset categories (see §33.1). And a minimum for the poset $\{q \in Q \mid p \preccurlyeq g(q)\}$ corresponds to an initial object for the poset-as-category (see §9.1). So this suggests that we might be able to characterize a left adjoint as the initial object of some suitable category.

And this is indeed more or less the case. Suppose $F\colon \mathscr{A} \to \mathscr{B}$ and $G\colon \mathscr{B} \to \mathscr{A}$ are functors such that $F \dashv G$. Now consider the comma category $(A \downarrow G)$, for $A \in \mathscr{A}$ – we met this construction at the end of §24. To recap,

(a) the objects of $(A \downarrow G)$ are pairs $\langle B, f\rangle$ where B is a $\mathscr{B}$-object and $f : A \to GB$ is an arrow in $\mathscr{A}$,

(b) an arrow in $(A \downarrow G)$ from $\langle B, f\rangle$ to $\langle B', f'\rangle$ is a $\mathscr{B}$-arrow $j\colon B \to B'$ making the following commute:

$$\begin{tikzcd} & GB \arrow[d, "Gj"] \\ A \arrow[ur, "f"] \arrow[r, "f'"'] & GB' \end{tikzcd}$$

The definitions for the identity arrows and for composition of arrows in $(A \downarrow G)$ are the obvious ones.

Theorem 174. *Given an adjunction* $\mathscr{A} \underset{G}{\overset{F}{\underset{\perp}{\rightleftarrows}}} \mathscr{B}$, *the pair* $\langle FA, \eta_A\rangle$ *is initial in* $(A \downarrow G)$ *for any* $A \in \mathscr{A}$.

Proof. Let $\langle B, f\rangle$ be any object of $(A \downarrow G)$. We need to show that there is a unique arrow in $(A \downarrow G)$ from $\langle FA, \eta_A\colon A \to GFA\rangle$ to $\langle B, f\rangle$. That is to say, there must be (i) an arrow $j\colon FA \to B$ such that $f = Gj \circ \eta_A$, i.e.

$$\begin{tikzcd} & GFA \arrow[d, "Gj"] \\ A \arrow[ur, "\eta_A"] \arrow[r, "f"'] & GB \end{tikzcd}$$

commutes, and (ii) this arrow must be unique. But we've already proved *that* – see one half of Theorem 169. $\square$

We have a converse result too:

Theorem 175. *Given functors* $\mathscr{A} \underset{G}{\overset{F}{\rightleftarrows}} \mathscr{B}$, *then if (C)* $\eta\colon 1_{\mathscr{A}} \to GF$ *is a natural transformation and the pair* $\langle FA, \eta_A\rangle$ *is initial in* $(A \downarrow G)$ *for every* $A \in \mathscr{A}$, *then* $F \dashv G$.

So that tells us how to characterize a left adjoint for G when it exists, since left adjoints are unique up to isomorphism, i.e. as a functor F satisfying condition (C).

Proof. Suppose η is natural transformation, and that $\langle FA, \eta_A\rangle$ is initial in $(A \downarrow G)$ for every $A \in \mathscr{A}$. Then for every $f\colon A \to GB$ there is a unique $j\colon FA \to B$ such that $f = Gj \circ \eta_A$. Apply the other half of Theorem 169. $\square$

But there was no fun in that instant proof. So, as an instructive and amusing exercise in diagram chasing here is

Another proof, by constructing a counit for η from first principles. We need to find a natural transformation $\varepsilon\colon FG \Rightarrow 1_{\mathscr{B}}$ such that η and ε satisfy the triangle equalities, i.e. such that $\varepsilon_{FA} \circ F\eta_A = 1_{FA}$ for all $A \in \mathscr{A}$, and $G\varepsilon_B \circ \eta_{GB} = 1_{GB}$ for all $B \in \mathscr{B}$.

Taken any $B \in \mathscr{B}$. By hypothesis $\langle FGB, \eta_{GB}\rangle$ is initial in $(GB \downarrow G)$, so there is a unique arrow to the object $\langle B, 1_{GB}\rangle$. Call this unique arrow (hopefully!) ε_B. Then just by its definition, for any B we have (*):

$$\begin{tikzcd} & GFGB \arrow[d, "G\varepsilon_B"] \\ GB \arrow[ur, "\eta_{GB}"] \arrow[r, "1_{GB}"'] & GB \end{tikzcd}$$

which gives us one lot of the triangle identities for free. So it remains to show that (i) we also have the other triangle identities, and (ii) the components ε_B do indeed assemble into a natural transformation. Try before reading on!

For (i), we need to show that the following diagram commutes:

$$\begin{tikzcd} & FGFA \arrow[d, "\varepsilon_{FA}"] \\ FA \arrow[ur, "F\eta_A"] \arrow[r, "1_{FA}"'] & FA \end{tikzcd}$$

Since $\eta\colon 1_{\mathscr{A}} \Rightarrow GF$ is natural, for every $f\colon A \to A'$ with $A, A' \in \mathscr{A}$, there is a commuting naturality square. Take in particular the case where $f = \eta_A(!)$. Then paste on the commuting triangle of the type (*), with $B = FA$, to get the commuting rhombus:

$$\begin{tikzcd} A \arrow[r, "\eta_A"] \arrow[d, "\eta_A"] & GFA \arrow[d, "\eta_{GFA}"] \arrow[dr, "1_{GFA}"] & \\ GFA \arrow[r, "GF\eta_A"] & GFGFA \arrow[r, "G\varepsilon_{FA}"] & GFA \end{tikzcd}$$

Composing arrows, using the functoriality of G, and re-arranging we get the commuting triangle on the left:

$$\begin{tikzcd} & GFA \arrow[d, "G(\varepsilon_{FA} \circ F\eta_A)"] \\ A \arrow[ur, "\eta_A"] \arrow[r, "\eta_A"'] & GFA \end{tikzcd} \qquad \begin{tikzcd} & GFA \arrow[d, "Gj"] \\ A \arrow[ur, "\eta_A"] \arrow[r, "\eta_A"'] & GFA \end{tikzcd}$$

Now, $\langle FA, \eta_A\rangle$ is initial in $(A \downarrow G)$ so there must be a *unique* arrow j from the initial object to itself such the triangle on the right commutes. But evidently $j = 1_{FA}$ makes the triangle commute. But so, as we've just seen, does $j = \varepsilon_{FA} \circ F\eta_A$. Hence $\varepsilon_{FA} \circ F\eta_A = 1_{FA}$. Which establishes (i).

To establish (ii) – the naturality of $\varepsilon\colon FG \Rightarrow 1_{\mathscr{B}}$, when assembled from the components ε_B – we need to show that for any $g\colon B \to B'$, the following commutes (**):

$$\begin{tikzcd}
FGB \arrow[r, "FGg"] \arrow[d, "\varepsilon_B"] & FGB' \arrow[d, "\varepsilon_{B'}"] \\
B \arrow[r, "g"] & B'
\end{tikzcd}$$

We again start by taking a naturality square for η, this time for $f = Gg\colon GB \to GB'$, and then paste on a commuting triangle of type $(*)$, to get the commuting rhombus

$$\begin{tikzcd}
GB \arrow[r, "Gg"] \arrow[d, "\eta_{GB}"] & GB' \arrow[d, "\eta_{GB'}"] \arrow[dr, "1_{GB'}"] & \\
GFGB \arrow[r, "GFGg"] & GFGB' \arrow[r, "G\varepsilon_{B'}"] & GB'
\end{tikzcd}$$

Again composing arrows, using the functoriality of G, and re-arranging we get the commuting triangle on the left:

$$\begin{tikzcd}
& GFGB \arrow[dd, "G(\varepsilon_{B'}\circ FGg)"] \\
GB \arrow[ur, "\eta_{GB}"] \arrow[dr, "Gg"'] & \\
& GB'
\end{tikzcd}
\qquad
\begin{tikzcd}
& GFGB \arrow[d, "G\varepsilon_B"'] \arrow[dd, bend left, "G(g\circ\varepsilon_B)"] \\
GB \arrow[ur, "\eta_{GB}"] \arrow[r, "1_{GB}"] \arrow[dr, "Gg"'] & GB \arrow[d, "Gg"'] \\
& GB'
\end{tikzcd}$$

On the right, we've pasted together (*) with a trivially commuting triangle, and then composed the downwards arrows to give the big triangle. However, by assumption, $\langle FGB, \eta_{GB}\rangle$ is initial in the comma category $(GB \downarrow G)$, so there is a *unique* arrow j to $\langle B', g\rangle$ such that $g = Gj \circ \eta_{GB}$. Whence $\varepsilon_{B'} \circ FGg = j = g \circ \varepsilon_B$, proving (**) commutes and establishing (ii). $\square$

Here's a nice corollary:

Theorem 176. *Suppose $G\colon \mathscr{B} \to \mathscr{A}$ is a functor. If the derived comma category $(A \downarrow G)$ has an initial object for every $A \in \mathscr{A}$, then G has a left adjoint.*

Proof. Choose an initial object for each $(A \downarrow G)$: it is a pair that we will write (hopefully!) as $\langle FA, \eta_A\rangle$, with $FA \in \mathscr{B}$, and $\eta_A\colon A \to GFA$.

So we now define a functor $F\colon \mathscr{A} \to \mathscr{B}$ which sends an object $A \in \mathscr{A}$ to this $FA \in \mathscr{B}$. How should F act on an arrow $f\colon A \to A'$? It must yield an arrow from FA to FA'. But since $\langle FA, \eta_A\rangle$ is initial, we know that there is exactly one arrow in $(A \downarrow G)$ from $\langle FA, \eta_A\rangle$ to $\langle FA', \eta_{A'} \circ f\rangle$. That is to say, there is a unique $g\colon FA \to FA'$ such that $\eta_{A'} \circ f = Gg \circ \eta_A$. Put $Ff = g$, and it is easy enough to check that F is functorial.

So now consider this diagram:

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[d, "\eta_A"] & A' \arrow[d, "\eta_{A'}"] \\ GFA \arrow[r, "GFf"] & GFA' \end{tikzcd}$$

We've defined Ff to make this commute. But this is a naturality square showing that the components η_A assemble into a natural transformation $\eta\colon 1_{\mathscr{A}} \to GF$.

So, in sum, $F\colon \mathscr{A} \to \mathscr{B}$ as defined is such that (C), $\eta\colon 1_{\mathscr{A}} \to GF$ is a natural transformation and the pair $\langle FA, \eta_A\rangle$ is initial in $(A \downarrow G)$ for every $A \in \mathscr{A}$. Hence, by the previous theorem, $F \dashv G$. ☐

34.6 Another way of getting new adjunctions from old

We've already met one way of getting new adjunctions from old, i.e. simple composition. Finally in this chapter, we now introduce another.

Definition 135. Given a functor $F\colon \mathscr{C} \to \mathscr{D}$ and small category $\mathbf{J}$, then the functor $[\mathbf{J}, F]\colon [\mathbf{J}, \mathscr{C}] \to [\mathbf{J}, \mathscr{D}]$ sends a functor $K\colon \mathbf{J} \to \mathscr{C}$ to $F \circ K\colon \mathbf{J} \to \mathscr{D}$.

Strictly speaking that's an incomplete definition. We need to specify not just how $[\mathbf{J}, F]$ acts on objects in $[\mathbf{J}, \mathscr{C}]$ (i.e. acts on functors), but how it acts on arrows (i.e. on natural transformations). But the needed completion, as often in defining functors, writes itself. For what is the obvious way for $[\mathbf{J}, F]$ to act on a natural transformation from K to K' with components $\alpha_J\colon KJ \to K'J$ (for $J \in \mathbf{J}$ and functors $K, K'\colon \mathbf{J} \to \mathscr{C}$)? By sending it, of course, to the natural transformation from $F \circ K$ to $F \circ K'$ with components $F\alpha_J\colon FKJ \to FK'J$. Full functoriality is then immediate.

We can now state our result about how a given adjunction between functors F and G generates a new adjunction between new-style functors $[\mathbf{J}, F]$ and $[\mathbf{J}, G]$:

Theorem 177. *If $F \dashv G\colon \mathscr{C} \to \mathscr{D}$ then $[\mathbf{J}, F] \dashv [\mathbf{J}, G]\colon [\mathbf{J}, \mathscr{C}] \to [\mathbf{J}, \mathscr{D}]$.*

Proof. Take functors $K\colon \mathbf{J} \to \mathscr{C}$, $L\colon \mathbf{J} \to \mathscr{D}$. Then take any natural transformation $\beta\colon FK \Rightarrow L$ living as an arrow in $[\mathbf{J}, \mathscr{D}]$. This has components $\beta_J\colon FKJ \to LJ$ living in $\mathscr{D}(FKJ, LJ)$. By the adjunction $F \dashv G$ these components are in a natural bijection with arrows $\alpha_J\colon KJ \to GLJ$ living in $\mathscr{C}(KJ, GLJ)$, and these assemble into a natural transformation $\alpha\colon K \Rightarrow GL$ which lives in $[\mathbf{J}, \mathscr{C}]$ (the adjunction is easily checked to associate naturality squares with naturality squares). In this way we set up a natural one-to-one correspondence between natural transformations like α and β.

So we have established that there is, naturally, a bijection

$$[\mathbf{J}, \mathscr{D}]([\mathbf{J}, F]K, L) \cong [\mathbf{J}, \mathscr{C}](K, [\mathbf{J}, G]L),$$

which proves $[\mathbf{J}, F] \dashv [\mathbf{J}, G]$. ☐

35 Adjoint functors and limits

NB: This chapter, like the previous two, is taken, unrevised, from an earlier set of Notes on Category Theory. It needs a great deal of rewriting, not to mention checking for bad errors! However, if you have got this far then it should still be useful, and it gets us to a sensible interim stopping point.

We now turn to some key results which tell us how adjoint functors interact with limits. A key bit of news is that right adjoints preserve limits: and dually, exactly as you would now expect, left adjoints preserve co-limits.

35.1 Limit functors as adjoints

(a) Suppose the category $\mathscr{C}$ has all limits of shape $\mathbf{J}$. Three observations:

(1) By Theorem 56, the cones over $D\colon \mathbf{J} \to \mathscr{C}$ with vertex C correspond one-to-one with $\mathscr{C}$-arrows from C to $\underset{\leftarrow \mathbf{J}}{Lim}\, D$.

(2) But by the remark after Theorem 116, the set of cones over $D\colon \mathbf{J} \to \mathscr{C}$ with vertex C is the hom-set $[\mathbf{J}, \mathscr{C}](\Delta(C), D)$. Here $\Delta\colon \mathscr{C} \to [\mathbf{J}, \mathscr{C}]$ is the functor introduced just after that theorem, which sends an object $C \in \mathscr{C}$ to the constant functor $\Delta_C\colon \mathbf{J} \to \mathscr{C}$. (For convenience, understand the cones here austerely).

(3) The set of $\mathscr{C}$-arrows from C to the limit vertex $Lim\, D$ is $\mathscr{C}(C, Lim(D))$, where $Lim\colon [\mathbf{J}, \mathscr{C}] \to \mathscr{C}$ is the functor introduced in §27.6, a functor that exists if $\mathscr{C}$ has all limits of shape $\mathbf{J}$ and that sends a diagram D of shape $\mathbf{J}$ in $\mathscr{C}$ to some limit object in $\mathscr{C}$.

So, in summary, still assuming that $\mathscr{C}$ has all limits of shape $\mathbf{J}$, the situation is this. We have a pair of functors $\mathscr{C} \underset{Lim}{\overset{\Delta}{\rightleftarrows}} [\mathbf{J}, \mathscr{C}]$ such that

$$[\mathbf{J}, \mathscr{C}](\Delta(C), D) \cong \mathscr{C}(C, Lim(D)).$$

Moreover, the isomorphism that is given in our proof of Theorem 56 arises in a natural way, without making any arbitrary choices.[1] So, we can take it that the

[1] Careful: there were arbitrary choices made in determining what *Lim* does. But once *Lim* is fixed, the isomorphism arises naturally.

isomorphism is natural in $C \in \mathscr{C}$ and $D \in [\mathbf{J}, \mathscr{C}]$. Hence Δ has a right adjoint, and one such right adjoint is Lim.

We now argue in the opposite direction starting from the assumption that the diagram Δ has a right adjoint, call it L.

Suppose that D is a diagram $D\colon \mathbf{J} \to \mathscr{C}$. Applying Theorem 170 about a universal mapping property of adjunctions, for any arrow $c\colon \Delta(C) \to D$ in $[\mathbf{J}, \mathscr{C}]$ – in other words for any cone over D with vertex C – there is a unique arrow $u\colon C \to L(D)$ in $\mathscr{C}$, such that $c = \varepsilon_D \circ \Delta(u)$, where ε is the co-unit of the adjunction.

By the definition of Δ, $\Delta(u)$ is the natural transformation from Δ_C to $\Delta_{L(D)}$ with every component equal to u.

And by §34.1 (3), ε_D is the transpose of $1_L(D)$, i.e. is some arrow $\pi\colon \Delta_{L(D)} \to D$ in $[\mathbf{J}, \mathscr{C}]$, i.e. is some particular cone π over D with vertex L_D.

Taken component-wise, the equation $c = \varepsilon_D \circ \Delta(u)$ tells us that for each $J \in \mathbf{J}$, $c_J = \pi_J \circ u$. In other words any cone c factors through our cone π via the unique u. Hence the cone π with vertex $L(D)$ and projection arrows π_J is a limit cone for D. However, D was any diagram $D\colon \mathbf{J} \to \mathscr{C}$. Therefore $\mathscr{C}$ has all limits of shape $\mathbf{J}$.

Summing up, we get the following nice theorem:

Theorem 178. *If category $\mathscr{C}$ has all limits of shape $\mathbf{J}$, then Δ has a right adjoint, and indeed $\Delta \dashv Lim$. Conversely, if Δ has any right adjoint, then $\mathscr{C}$ has all limits of shape $\mathbf{J}$.*

(b) Keeping $\mathbf{J}$ fixed, we can make Δ's dependence on $\mathscr{C}$ explicit by writing $\Delta_{\mathscr{C}} \colon \mathscr{C} \to [\mathbf{J}, \mathscr{C}]$. Similarly we can explicitly write $Lim_{\mathscr{C}} \colon [\mathbf{J}, \mathscr{C}] \to \mathscr{C}$. Then we have the following easy corollary of the last theorem:

Theorem 179. *Suppose the categories $\mathscr{B}$ and $\mathscr{C}$ have all limits of shape $\mathbf{J}$. Then if $G \colon \mathscr{C} \to \mathscr{B}$ is a right adjoint (i.e. has a left adjoint), $G \circ Lim_{\mathscr{C}} \cong Lim_{\mathscr{B}} \circ [\mathbf{J}, G]$.*

Proof. Let $F \colon \mathscr{B} \to \mathscr{C}$ be left adjoint to G, and consider this pair of diagrams:

$$
\begin{tikzcd}
\mathscr{B} \arrow[r, "F"] \arrow[d, "\Delta_{\mathscr{B}}"'] & \mathscr{C} \arrow[d, "\Delta_{\mathscr{C}}"] \\
{[\mathbf{J}, \mathscr{B}]} \arrow[r, "{[\mathbf{J}, F]}"] & {[\mathbf{J}, \mathscr{C}]}
\end{tikzcd}
\qquad
\begin{tikzcd}
\mathscr{B} & \mathscr{C} \arrow[l, "G"] \\
{[\mathbf{J}, \mathscr{B}]} \arrow[u, "Lim_{\mathscr{B}}"] & {[\mathbf{J}, \mathscr{C}]} \arrow[l, "{[\mathbf{J}, G]}"] \arrow[u, "Lim_{\mathscr{C}}"']
\end{tikzcd}
$$

Claim: the left-hand diagram commutes. (i) On the south-west path, an object $B \in \mathscr{B}$ is sent by $\Delta_{\mathscr{B}}$ to the functor $\Delta_B \colon \mathbf{J} \to \mathscr{B}$ which sends every object to B and every arrow to 1_B; and this is sent in turn by $[\mathbf{J}, F]$ to the functor which sends every object to FB and every arrow to 1_{FB}, i.e. the functor Δ_{FB}. (ii) On the north-east path, an object $B \in \mathscr{B}$ is sent by F to FB, and this is sent by $\Delta_{\mathscr{C}}$ to the functor Δ_{FB} again.

Now, given the assumption that $\mathscr{B}$ and $\mathscr{C}$ have all limits of shape $\mathbf{J}$, $\Delta_{\mathscr{B}}$ and $\Delta_{\mathscr{C}}$ have right adjoints $Lim_{\mathscr{B}}$ and $Lim_{\mathscr{C}}$. And since $F \dashv G$, $[\mathbf{J}, F] \dashv [\mathbf{J}, G]$, by Theorem 177.

So our right-hand diagram records the adjoints of the functors in the left-hand diagram. We now know that the composite left-adjoint functors $\Delta_{\mathscr{C}} \circ F$ and $[\mathbf{J}, F] \circ \Delta_{\mathscr{B}}$ are the same. By Theorem 171 about the composition of adjunctions, their right-adjoints are $G \circ Lim_{\mathscr{C}}$ and $Lim_{\mathscr{B}} \circ [\mathbf{J}, G]$. And these composites, being right adjoint to the same functor, must be naturally isomorphic by Theorem 172. □

35.2 Right adjoints preserve limits

We can usefully begin by restating part of a key definition and reminding ourselves of a basic theorem:

Definition 99 A functor $G\colon \mathscr{C} \to \mathscr{B}$ *preserves limits of shape* $\mathbf{J}$ iff, for any diagram $D\colon \mathbf{J} \to \mathscr{C}$, if $[L, \pi_J]$ is a limit cone over D, then $[GL, G\pi_J]$ is a limit cone over $G \circ D\colon \mathbf{J} \to \mathscr{B}$.

Theorem 108 *The covariant hom-functor* $\mathscr{C}(A, –)\colon \mathscr{C} \to \mathbf{Set}$*, for any* A *in the category* $\mathscr{C}$*, preserves all limits that exist in* $\mathscr{C}$*.*

Now, this theorem is easily seen to imply the following:

Theorem 180. *Any set-valued functor* $G\colon \mathscr{C} \to \mathbf{Set}$ *which is a right adjoint (i.e. has a left adjoint) preserves all limits that exist in* $\mathscr{C}$*.*

Proof. Suppose we have a functor F such that $F \dashv G$. Then

$$GA \cong \mathbf{Set}(1, GA) \cong \mathscr{C}(F1, A)$$

with both isomorphisms natural in A (the first relies on the familiar association in **Set** between elements of a set and arrows from a terminal object into that set). Hence G is naturally isomorphic to the hom-functor $\mathscr{C}(F1, –)$. But the latter preserves limits, by Theorem 108. Hence so does G, by Theorem 141. □

We now show that there is in fact nothing special here about set-valued functors. We can prove quite generally:

Theorem 181. *If the functor* $G\colon \mathscr{C} \to \mathscr{B}$ *is a right adjoint (i.e. has a left adjoint), it preserves all limits that exist in* $\mathscr{C}$*.*

Proof from basic principles about limits and adjoints. Suppose that G has the left adjoint $F\colon \mathscr{B} \to \mathscr{C}$; and suppose also that the diagram $D\colon \mathbf{J} \to \mathscr{C}$ has a limit cone $[L, \pi_J]$ in $\mathscr{C}$.

Then $[GL, G\pi_J]$ is certainly a cone over $G \circ D$ in $\mathscr{B}$. We need to show, however, that it is a *limit* cone. That is to say, we need to show that, if we take any cone $[B, b_J]$ over $G \circ D$, there is a unique $u\colon B \to GL$ such that (i) for all $J \in \mathbf{J}$, $b_J = G\pi_J \circ u$.

Well, take such a cone $[B, b_J]$ over $G \circ D$. Then, going back in the other direction, $[FB, \overline{b_J}]$ is a cone over D in $\mathscr{C}$, where $\overline{b_J}\colon FB \to D_J$ is the transpose of $b_J\colon B \to GD_J$ under the adjunction.

Why is $[FB, \overline{b_J}]$ a cone? Suppose we have an arrow $d\colon D_K \to D_K$. Then by assumption, since $[B, b_J]$ is a cone over $G \circ D$, $b_K = Gd \circ b_J$. Hence $\overline{b_K} = \overline{Gd \circ b_J} = d \circ \overline{b_J}$, with the second equation by Theorem 166 (1). Which indeed makes $[FB, \overline{b_J}]$ a cone too.

And now we add that $[FB, \overline{b_J}]$ must factor through $[L, \pi_J]$ via a unique $v\colon FB \to L$ such that (ii) for all $J \in \mathbf{J}$, $\overline{b_J} = \pi_J \circ v$.

So the state of play is: we have found a unique $v\colon FB \to L$; we want to find a suitable $u\colon B \to GL$. The hopeful thought is that one will turn out to be the transpose of the other under the adjunction.

The adjunction means that $\mathscr{C}(FB, C) \cong \mathscr{B}(B, GC)$ naturally in C. Which in turn means that the following square commutes, for any $\pi_J\colon L \to D_J$:

$$\begin{tikzcd}
\mathscr{C}(FB, L) \arrow[r, "\pi_J \circ -"] \arrow[d] & \mathscr{C}(FB, D_J) \arrow[d] \\
\mathscr{B}(B, GL) \arrow[r, "G\pi_J \circ -"] & \mathscr{B}(B, GD_J)
\end{tikzcd}$$

where the vertical arrows are components of the natural transformation which sends an arrow to its transform. Chase the arrow $v\colon FB \to L$ round the diagram in both directions and we get $G\pi_J \circ \overline{v} = \overline{\pi_J \circ v}$. Therefore, using (ii), if we put $u = \overline{v}$, we indeed get as required that (i) for all $J \in \mathbf{J}$, $b_J = G\pi_J \circ u$.

It just remains to confirm u's uniqueness. Suppose that $[B, b_J]$ factors through $[GL, G\pi_J]$ by some $u' = \overline{w}$. Then for all $J \in \mathbf{J}$, $b_J = G\pi_J \circ \overline{w}$. We show as before that $\overline{b_J} = \pi_J \circ w$, whence $[FB, \overline{b_J}]$ factors through $[L, \pi_J]$ via w. By the uniqueness of factorization, $w = v$ again. □

A more compressed proof. Again, suppose that G has the left adjoint $F\colon \mathscr{B} \to \mathscr{C}$; and suppose also that the diagram $D\colon \mathbf{J} \to \mathscr{C}$ has a limit cone $[L, \pi_J]$ in $\mathscr{C}$. Then, using the notation '$\mathscr{C}(X, D)$' as shorthand for the functor $\mathscr{C}(X, -) \circ D$, we have

$$\begin{aligned}
\mathscr{B}(B, GL) &\cong \mathscr{C}(FB, L)\\
&\cong \mathit{Lim}\,\mathscr{C}(FB, D)\\
&\cong \mathit{Lim}\,\mathscr{B}(B, GD)\\
&\cong \mathit{Cone}(B, GD).
\end{aligned}$$

all naturally in B. So the functor $\mathit{Cone}(-, GD)$, being naturally isomorphic to $\mathscr{B}(-, GL)$ is representable, and is represented by GL, and therefore has a universal element of the form $\langle GL, g\rangle$. But such a universal element is a limit cone with vertex GL. Hence G preserves the limit $[L, \pi_J]$. □

But compression doesn't always make for illumination, and our second proof (see Leinster 2014, p. 158; compare Awodey 2006, pp. 225–6) needs some commentary.

The first line of course comes from the adjunction, and the second from the fact that the hom-functor $\mathscr{C}(FB, -)$ preserves limits, by Theorem 108. The move from the third to the fourth line is by Theorem **??** [*The referenced theorem needs* *to be replaced!*]. And the arguments at the end about representability, universal elements and limits appeal to Theorems 148 and 153.

So that leaves the move from the second to the third line, which obviously invokes the adjunction between F and G again. We know that $\mathscr{C}(FB, X) \cong \mathscr{B}(B, GX)$ naturally in X, i.e. $\mathscr{C}(FB, –)$ is naturally isomorphic to $\mathscr{B}(B, G–)$, hence by whiskering, $\mathscr{C}(FB, –) \circ D$ is naturally isomorphic to $\mathscr{B}(B, G–) \circ D$. Now apply Theorem 117 and we can conclude that $Lim\, \mathscr{C}(FB, D) \cong Lim\, \mathscr{B}(B, GD)$.

Which all goes to combine a bunch of earlier results into a neat package: but my own feeling is that the first direct proof from the underlying principles reveals better what is really going on here.

35.3 Some examples

Right adjoints preserve limits. Dually, of course, left adjoints preserve colimits (we surely needn't pause at this stage in the game to state the duals of the theorems in the last couple of sections!). So we now mention just a few elementary examples of (co)limit preservation – and also some examples where we can argue from non-preservation to the non-existence of adjoints.

(1) Back in §22.2, Ex. (4) we noted that the forgetful functor $U\colon \mathbf{Mon} \to \mathbf{Set}$ preserves limits. But we now have another proof: U has a left adjoint (by §33.2, Ex. (3)) i.e. it *is* a right adjoint, so indeed must preserve limits.

There are other examples of this kind, involving a forgetful functor $U\colon \mathbf{Alg} \to \mathbf{Set}$, where $\mathbf{Alg}$ is a category of sets equipped with some algebraic structure for U to ignore. Such a forgetful U standardly has a left adjoint, so must preserve whatever limits exist in the relevant $\mathbf{Alg}$.

Further, a left-adjoint to U must preserve existing colimits in $\mathbf{Set}$. But $\mathbf{Set}$ has *all* colimits; so that this indeed requires the left-adjoints in such cases to be rather lavish constructions (as we saw them to be).

(2) Consider exponentials again.

We noted that if $\mathscr{C}$ is a category with exponentiation, and hence with products, exponentiation is right adjoint to taking products: $(- \times B) \dashv (-)^B$.

Since the functor $(-)^B$ is a right adjoint, it preserves such limits as exist in $\mathscr{C}$. So take in particular the functor $A\colon 2 \to \mathscr{C}$ (where as before 2 is the discrete two object category with objects $0, 1$). Then $A_0 \times A_1$ is the vertex of a limit over A. Hence $(A_0 \times A_1)^B$ is the vertex of a limit over $(-)^B \circ A$. But the canonical limit over that composite functor is $A_0^B \times A_1^B$. Hence $(A_0 \times A_1)^B \cong A_0^B \times A_1^B$

Since the functor $- \times B$ is a left adjoint, it preserves such colimits as exist in $\mathscr{C}$. Assume $\mathscr{C}$ has coproducts. Then, by a similar argument, $(A_0 + A_1) \times B \cong (A_0 \times B) + (A_1 \times B)$.

(3) Take the discussion in §32.3, Ex. (6) where we looked at the Galois connection between two functions between posets of equivalence classes of wffs,

with the left adjoint a trivial 'add a dummy variable' map, and the right adjoint provided by applying a universal quantifier. This carries over to an adjunction of functors between certain poset categories. Since quantification is a right adjoint, it preserves limits, and in particular preserves products, which are conjunctions in this category. Which reflects the familiar logical truth that $\forall x(Px \wedge Qx) \equiv (\forall x Px \wedge \forall x Qx)$.

(4) Claim: the forgetful functor $F\colon \mathbf{Grp} \to \mathbf{Set}$ has no right adjoint. Proof: the trivial one-object group is initial in **Grp**; but a singleton is not initial in **Set**; so there is a colimit which F doesn't preserve and it therefore cannot be a left adjoint.

(5) The proof of Theorem 84 tells us that the forgetful functor $F\colon \mathbf{Mon} \to \mathbf{Set}$ fails to preserve all epimorphisms. By Theorem 102 this implies that F doesn't preserve all pushouts, and hence doesn't preserve all colimits. Hence this forgetful functor too can't be a left adjoint. Compare the arm-waving argument to the same conclusion in §33.2. Ex. (5).

35.4 The Adjoint Functor Theorems

Right adjoints preserve limits. What about the converse? If a functor preserves limits must it be a right adjoint? Well, given some results already to hand, we can easily prove the following:

Theorem 182. *If the category $\mathscr{B}$ has all limits, and the functor $G\colon \mathscr{B} \to \mathscr{A}$ preserves them, then G is a right adjoint.*

Proof. If $\mathscr{B}$ has all limits and G preserves them, then for any $A \in \mathscr{A}$, $(A \downarrow G)$ has all limits (by Theorem 111, and the remark immediately after its proof).

So any $(A \downarrow G)$ in particular has a limit for the big diagram-as-part-of-a-category consisting of the whole of $(A \downarrow G)$ – or in terms of diagrams-as-functors, it has a limit for the identity functor $1_{(A\downarrow G)}$. Hence by Theorem 57, each $(A \downarrow G)$ has an initial object. Hence by Theorem 176, there is a functor $F\colon \mathscr{A} \to \mathscr{B}$ such that $F \dashv G$. □

And now we see the proof, we see that the condition that $\mathscr{B}$ has *all* limits overshoots: the result will go through so long as $\mathscr{B}$ has sufficiently large limits, enough to guarantee that all the functors $1_{(A\downarrow G)}$ have a limit.

This theorem looks neat but is in fact not very useful. Having all sufficiently large limits is a hard condition to fulfil. More precisely, we have

Theorem 183. *If a category $\mathscr{C}$ has limits for diagrams over all categories of size up to the size of the collection of $\mathscr{C}$'s arrows, then $\mathscr{C}$ has at most one arrow between any two objects.*

For example, the condition of having small limits is not satisfied by typical small categories – because, in the terminology of §4.4 preordercatex, a complete small category has to be a *pre-order* category.

Proof. Let $\mathbf{J}$ be a discrete category of the same cardinality as the set of arrows of $\mathscr{C}$. Let $D\colon \mathbf{J} \to \mathscr{C}$ be the diagram which sends every object in $\mathbf{J}$ to B. By hypothesis, D has a limit, namely the product $\prod_{J \in \mathbf{J}} D(J)$ (so this is the product of B with itself, $\mathbf{J}$-many times).

Suppose there are objects $A, B \in \mathscr{C}$ with arrows $f_1, f_2\colon A \to B$ where $f_1 \neq f_2$. Simple cardinality considerations show that this further supposition leads to contradiction. Which proves the theorem.

We start by asking: how many different arrows $A \to \prod_{J \in \mathbf{J}} D(J)$ are there? Theorem **??** showed that if $\mathbf{J}$ is the discrete two object category, then there are four such arrows. Generalizing the proof in the obvious way shows that if $|\mathbf{J}|$ is the cardinality of the objects of $\mathbf{J}$, there are $2^{|\mathbf{J}|}$ different arrows from $A \to \prod_{J \in \mathbf{J}} D(J)$.

Hence our suppositions imply that there is a subset of the arrows in $\mathscr{C}$ whose cardinality is strictly greater than the cardinality of the set of arrows in $\mathscr{C}$. Contradiction. □

So, in sum, Theorem 182 is of very limited application. If we want a more widely useful result of the form 'Given such-and such conditions on the functor $G\colon \mathscr{B} \to \mathscr{A}$ and the categories it relates, then G is a right adjoint', we'll need to consider a new bunch of conditions.

Here are two such theorems of rather wider application (the labels are standard):

Theorem 184 (The General Adjoint Functor Theorem). *If category $\mathscr{B}$ is a locally small category with all small limits, and the functor $G\colon \mathscr{B} \to \mathscr{A}$ is such that for each $A \in \mathscr{A}$, the category $(A \downarrow G)$ has a weakly initial set, then G is a right adjoint iff it preserves all small limits.*

(GAFT: Alternative version) *If category $\mathscr{B}$ is a locally small category with all small limits, and $G\colon \mathscr{B} \to \mathscr{A}$ is a functor, then G is a right adjoint iff it preserves all small limits and satisfies the solution set condition.*

Theorem 185 (The Special Adjoint Functor Theorem). *If the categories $\mathscr{A}$ and $\mathscr{B}$ are locally small, and $\mathscr{B}$ has all small limits, is well powered, and has a coseparating set of objects, then G is a right adjoint iff it preserves all small limits.*

But to investigate these theorems properly would require not just explaining the concepts 'weakly initial set', 'solution set condition', 'well powered' and 'coseparating' and then doing the proofs, but also explaining what might motivate the conditions our new concepts are used to state, and also explaining why the resulting theorems, with just those conditions in play, might be of interest and use. That's a non-trivial expositional task, and one I am going to shirk in this current version of these Notes. If you want to follow up the technical details, which aren't particularly difficult, I can refer you to for example Leinster (2014, pp. 159–164, 171–173) and Awodey (2006, §9.8). But I'm not sure I yet have a sufficiently good grip on the place of these theorems in the scheme of things to give an illuminating account of the motivations here.

Indeed, the Adjoint Functor Theorems arguably sit at the boundary between basic category theory and the beginnings of more serious stuff. So given the intended limited remit of *these* Notes, this is in any case the point at which I should probably stop for the moment.

Bibliography

Adámek, J., Herrlich, H., and Strecker, G., 2009. *Abstract and Concrete Categories: The Joy of Cats*. Mineola, New York: Dover Publications. URL http://www.tac.mta.ca/tac/reprints/articles/17/tr17.pdf. Originally published 1990.

Aluffi, P., 2009. *Algebra: Chapter 0*. Providence, Rhode Island: American Mathematical Society.

Awodey, S., 2006. *Category theory*, vol. 49 of *Oxford Logic Guides*. Oxford: Oxford University Press.

Booth, D. and Ziegler, R. (eds.), 1996. *Finsler Set Theory: Platonism and Circularity*. Basel: Birkhäuser Verlag.

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

Church, A., 1956. *Introduction to Mathematical Logic*. Princeton, NJ: Princeton University Press.

Dummit, D. S. and Foote, R. M., 2004. *Abstract Algebra*. Hoboken, NJ: John Wiley, 3rd edn.

Eilenberg, S. and Mac Lane, S., 1942. Natural isomorphisms in group theory. *Proceedings of the National Academy of Sciences of the United States of America*, 28: 537–543.

Eilenberg, S. and Mac Lane, S., 1945. General theory of natural equivalences. *Transactions of the American Mathematical Society*, 58: 231–294.

Finsler, P., 1926. Über die Grundlagen der Mengenlehre, I. *Mathematische Zeitschrift*, 25: 683–713. Reprinted and translated in Booth and Ziegler 1996: 103–132.

Forster, T., 1995. *Set Theory with a Universal Set*. Oxford: Clarendon Press, 2nd edn.

Freyd, P., 1965. The theories of functors and models. In J. W. Addison, L. Henkin, and A. Tarski (eds.), *The Theory of Models*, pp. 107–120. North-Holland Publishing Co.

Goedecke, J., 2013. Category theory. URL https://www.dpmms.cam.ac.uk/~jg352/pdf/CategoryTheoryNotes.pdf.

Goldblatt, R., 2006. *Topoi: The Categorial Analysis of Logic*. Mineola, New York: Dover Publications, revised edn.

Hamkins, J. D., 2002. How tall is the automorphism tower of a group? In Y. Zhang (ed.), *Logic and Algebra*, vol. 302 of *Contemporary Math.*, pp. 49–57. Providence, RI: AMS. URL http://wp.me/s5M0LV-howtall.

Incurvati, L., 2020. *Conceptions of Set and the Foundations of Mathematics*. Cambridge: Cambridge University Press.

Johnstone, P., 2002. *Sketches of an Elephant: A Topos Theory Compendium, Vol. 1*, vol. 43 of *Oxford Logic Guides*. Clarendon Press.

Lawvere, F. W., 1969. Adjointness in foundations. *Dialectica*, 23: 281–296.

Lawvere, F. W. and Schanuel, S. H., 2009. *Conceptual Mathematics: A first introduction to categories*. Cambridge: Cambridge University Press, 2nd edn.

Leinster, T., 2014. *Basic Category Theory*. Cambridge: Cambridge University Press.

Mac Lane, S., 1997. *Categories for the Working Mathematician*. New York: Springer, 2nd edn.

Maddy, P., 2017. Set-theoretic foundations. *Contemporary Mathematics*, 690: 289–322.

Marquis, J.-P., 2008. *From a Geometrical Point of View: A Study of the History and Philosophy of Category Theory*. New York: Springer.

Mazur, B., 2008. When is one thing equal to some other thing? In B. Gold and R. Simons (eds.), *Proof and Other Dilemmas: Mathematics and Philosophy*. Mathematical Association of America. URL http://www.math.harvard.edu/~mazur/preprints/when_is_one.pdf.

Oliver, A. and Smiley, T., 2016. *Plural Logic*. Oxford: Oxford University Press, 2nd edn.

Quine, W. V. O., 1963. *Set Theory and Its Logic*. Cambridge, MA: The Belknap Press of Harvard University Press.

Schubert, H., 1972. *Categories*. New York, Heidelberg, Berlin: Springer.

Sellars, W., 1963. Philosophy and the scientific image of man. In *Science, Perception and Reality*. Routledge & Kegan Paul.

Simmons, H., 2011. *An Introduction to Category Theory*. Cambridge: Cambridge University Press.

Simpson, S. G., 2010. *Subsystems of Second Order Arithmetic*. Cambridge: Cambridge University Press, 2nd edn. URL https://lead.to/amazon/com/?op=bt&la=en&cu=usd&key=0521150140.

Tao, T., 2016. *Analysis*. Springer, 3rd edn.

Tennant, N., 2009. Natural logicism via the logic of orderly pairing. In S. K. S.-H. V. Lindström S., Palmgren E. (ed.), *Logicism, Intuitionism, and Formalism*, pp. 91–125. Dordrecht: Springer.