Category Theory I

Notes towards a gentle introduction

Peter Smith

LOGIC MATTERS

Published by Logic Matters, Cambridge

Printed copies are inexpensively available:

ISBN 978-1-91690636-5 Paperback (Amazon only, August 2023)

Visit logicmatters.net/categories for further resources related to the topic of this book. Note: external links, as here, are coloured: internal links are live but not marked.

When sending comments and corrections, please note the date of this PDF: August 4, 2023

Contents

Preface to Part I

A little background A few years ago I put together some notes on elementary category theory, initially as an exercise in getting things clearer in my own mind. I later posted versions online. And rather to my surprise, these have been steadily downloaded hundreds of times a month – which has been both embarrassing and encouraging. Embarrassing because those draft efforts were very half-baked. But encouraging enough for me to resolve to do better once other projects were off my desk. Hence this heavily revised new version.

Given the ordering of topics that I have chosen, these Notes divide very naturally into two Parts, as I explain in §1.3. The second Part will have to stay untouched for a while. But this first Part is, I hope, now worth putting into an inexpensive paperback form, if only for ease of reading. However, the text is certainly not yet set in stone! You should think of this as a 'beta version', functional but still rough around the edges and no doubt not bug-free.

So who are these Notes for? I originally got interested in category theory because of its connections with issues in the foundations of mathematics, broadly construed. This angle of approach no doubt influences the shape of these Notes in various ways (someone whose interest in category theory arises from issues in theoretical computer science, say, would organize things with different emphases and a different selection of topics). But there is little overt 'philosophical' discussion here – it is mostly mathematics, served up quite straight. And I expect that the most likely reader is going to be a student of pure mathematics who wants a relatively light-weight first introduction to some category theory, perhaps as a preliminary warm-up before taking on an industrial-strength graduate-level course.

Still, I hope that other readers interested in foundational questions, perhaps with less mathematical background, might also find something useful here. I have tried to give a reasonably accessible exposition of core categorial[1] ideas, enough to give an initial sense of what the fuss is about, and then to provide a launchpad for further explorations, both conceptual and more technical.

One thing will be quite obvious from the outset: I *do* go at quite a leisurely pace. I don't apologize at all for this: after all, if you find the pace *too* slow, there are plenty of faster-track alternative introductions available. However, it

[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.

is not just my own experience which suggests that, for many, getting a secure understanding of category-theoretic ways of thinking by initially taking things gently can make later adventures exploring beyond the basics *very* much more manageable.

But of course, whether my angle of approach and the moderately-relaxed-but-fairly-traditional mode of exposition will satisfy *you* must in the end depend entirely on your particular interests, background, and preferences in matters of mathematical style. I can only suggest dipping in and skimming through to see whether you think these Notes might work for you.

What do you need to bring to the party? One crucial thing which category theory does is give us a story about the ways in which different parts of modern abstract mathematics hang together. Obviously, you can't be in a good position to appreciate this if you really know *nothing* beforehand about modern mathematics! But I do try to presuppose relatively little detail. 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 with the introductory discussions here. 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.

How far do we aim to get? I only explore the beginnings of category theory here. But what count as 'beginnings'?

I'll be guided in part by the coverage of some avowedly introductory books. But I also note, for example, that the famous introduction to topos theory by Mac Lane and Moerdijk (1992) starts with a fourteen page chapter of 'Categorical Preliminaries'. That isn't supposed to be a stand-alone exposition so much as a checklist of assumed basics. And their checklist turns out to correspond pretty closely to the overall coverage of the two Parts of these Notes, which suggests that my menu of topics is sensible enough.

The hope, then, is that these Notes will at least provide a ladder to climb up, so you get to a vantage point from which you can more confidently leap onwards, e.g. to tackle the existing introductions to more advanced topos-theoretic ideas related to logic, or to follow other tracks through category theory.

Theorems as exercises Almost all the proofs of the theorems you meet as you begin category theory are *very* straightforward. Almost always, you just have to 'do the obvious thing': there's little ingenious trickery needed at the outset. So you can think of the statement of a theorem as in fact presenting you with an exercise which you should ideally 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. Sometimes a few theorems are explicitly stated as a series of challenges, with the proofs coming a bit later. I do signal, though, when proofs can skipped without missing too much.

Notation Introduced in early chapters, upright variables such as 'X' are intended to be read *plurally*, as typically denoting many things, to be contrasted with 'X' which is *singular* and might stand for a set or other single thing.

'Iff' is of course short for 'if and only if'. '□' is used as an end-of-proof marker or to conclude the statement of a theorem whose proof needn't be further spelt out. A minor quirk is that I also use '△' 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 fussy about this sort of thing, and let's try to avoid that.

Thanks! Needless to say, all suggestions for improving this beta version, and any other comments and corrections, will continue to be very gratefully received.

I am always struck by the continued kindness of logical strangers. Andrew Bacon, Malcolm F. Lowe and Mariusz Stopa very generously sent long lists of corrections to an early ancestor of these Notes. I had then further corrections to a revised version from Malcolm F. Lowe, David Ozonoff, Jan Thiemann, Zoltán Tóth, and Adrian Yee. Most recently I have had comments and more corrections from Matthew Bjerknes, Sam Butchart, Ruiting Jiang, Phil Nguyen, Rowsety Moid, and Leonardo Pacheco. Very warm thanks to everyone.

1 Introduction

1.1 The categorial imperative

(a) 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-respecting maps between them. Then we can think of this inter-related 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 requirements. But we can also think of a whole family of groups, together with appropriate maps between them – i.e. homomorphisms which respect group structure – as forming a further structure-of-structures.

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

(3) And so it goes. Perhaps what interests you are *some objects equipped with an order*: these constitute another type of mathematical structure – with different kinds of ordering giving us, of course, different kinds of structure. Perhaps it is well-orderings in particular which you are concerned with. There is a whole family of well-ordered structures together with order-respecting maps between them, and 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 pluralities), with a family itself structured by the maps or morphisms between its members.

An obvious point: we see similar relationships recurring within different families. For example, some groups are products of others and some spaces are products of others ('a cylinder is the product of a line and a circle'); likewise, we can form products of e.g. well-ordered collections to get a longer well-ordering. Again, some groups can be seen as the result of quotienting another group by a suitable equivalence relation (in effect, we identify equivalent objects); likewise we can form quotient spaces and can quotient orderings by equivalence relations too. It is entirely natural, then, to want an account – one that applies across different families of structures – about what makes for products in general, what makes for quotients in general, etc. As we will see, entry-level category theory gives us just such an account, because structured families of structures are prime examples of categories.

(b) As a further step, we can go on to consider the interrelations between different families of structures – for they do not exist in glorious isolation from each other. This will involve looking at an additional level of structure-respecting maps, the so-called *functors*, this time linking structures-of-structures – as when, for example, 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 need to explain later).

So here indeed is *one* central imperative of modern mathematics: 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 very 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.

Again, category theory provides exactly what we need as we first set out to explore this territory: its basic ideas and constructions provide a general toolkit for systematically probing not only structures-of-structures but structures-of-structures-of-structures and more. 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 I suppose 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 category theory does much more than give us a helpful way of relating some 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)

So category theory crucially enables us to reveal *new* connections we hadn't made before. What are called 'adjunctions' are a prime example, as we will eventually find.

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 geometry that category theory first came to prominence. But it would be distracting to investigate those roots in these Notes. I will stick to very much more elementary concerns, with an emphasis on unification and conceptual clarification. This will, it has to be said, give us a limited and partial picture: but we will still have more than enough to explore. And this way, I at least can 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 interconnections mean that there certainly isn't a single best route into the theory. Different linear narratives will 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 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 I 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 Part I of these Notes.

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.

In summary, my chosen route here into the basics of category theory steadily ascends through increasing levels of abstraction. This route has considerable logical appeal; but, to be frank, it does mean that we don't meet some of the most important and characteristic categorial ideas until Part II. However, this disadvantage will (I hope) be counterbalanced – at least for enough readers – by the gains in understanding which come from taking our gently sloping path. I will just have to do my best to make the ideas we encounter in Part I already seem pretty interesting and fruitful!

2 Groups, and categories of groups

Category theory gives us a framework for thinking systematically about structured families of mathematical structures. One paradigm case of such a family, I said, comprises some groups organized by homomorphisms between them. I'll run with this example for a little, and by the end of this chapter I will have explained more carefully what it takes to form a category of groups.

But first, let's recall a few really elementary facts about groups. I want to highlight a number of ideas which are already there in familiar mathematics and which will later take on a categorial guise.

Depending on your background, this may *very* well be an unnecessary revision session: if so, you will be able to skim-read most of this chapter at considerable pace – be my guest. However, some of the definitions I give in this chapter are not *quite* the usual ones, so don't skip them entirely. I'll explain the reason for their mildly deviant character in Chapter 3. So bear with me.

2.1 Groups revisited

(a) Take 'G' to stand for one or more objects; and read the likes of '$x \varepsilon$ G' as saying that x is one of the objects G. Then here is my preferred version of the usual definition:

Definition 1. The objects G equipped with a binary operation $*$ form a *group* iff

(i) G are closed under $*$, i.e. for any $x, y \varepsilon$ G, $x * y \varepsilon$ G;
(ii) $*$ is associative, i.e. for any $x, y, z \varepsilon$ G, $(x * y) * z = x * (y * z)$;
(iii) there is a distinguished object $e \varepsilon$ G which acts as a group identity, i.e. for any $x \varepsilon$ G, $x * e = x = e * x$;
(iv) every group object has an inverse, i.e. for any $x \varepsilon$ G, there is at least one object $y \varepsilon$ G such that $x * y = e = y * x$.

A group is *abelian* iff its binary operation is commutative, i.e. for all $x, y \varepsilon$ G, $x * y = y * x$. △

Don't read too much into 'equipped'. It is a standard turn of phrase ('endowed' is an alternative); but it means no more than that we are dealing with some objects G together with an operation defined over them.

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

If y and y' are both inverses of x, then $y = y * e = y * (x * y') = (y * x) * y' = e * y' = y'$; so group inverses are unique too.

(b) Note immediately the huge variety of objects and operations that can form a group. For a start, *any* item e, whatever you like, together with the only possible binary operation $*$ such that $e * e = e$, forms a one-object group. Similarly, any two items e, j, whatever you like, form a group when they are equipped with the binary operation $*$ for which e is the identity and $j * j = e$.

Less trivially, there are for example additive groups of numbers (e.g. the integers equipped with addition, or equipped with addition mod n, either way with 0 as the identity), and there are multiplicative groups of numbers (e.g. the non-zero complex numbers equipped with multiplication, with 1 as the identity).

Likewise, there are groups of functions. For the simplest 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 $n > 2$, this group is non-abelian. Or consider groups of geometrical transformations – for instance, there is the non-abelian group of symmetries of a regular polygon (i.e. the group of rotation and reflection operations which map the given polygon to itself).

Then there are, for example, various groups of real invertible matrices. More intriguingly, perhaps, there are 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. But you knew all that!

(c) Let's agree some notation.

I will use '$(\mathrm{G}, *, e)$' to abbreviate '(the objects) G equipped with the operation $*$ and with the distinguished object e'. Similarly, of course, for e.g. '$(\mathrm{H}, \star, d)$' and so on. Note then that the parentheses here are just helpful punctuation. They are *not* being used to form a term for a new entity such as a set-theoretic triple.

If $(\mathrm{G}, *, e)$ satisfy the conditions for forming a group, then I'll briskly write e.g. 'the group $(\mathrm{G}, *, e)$' rather than 'the group formed by $(\mathrm{G}, *, e)$'. More briefly still, when I want to refer to a group without going into details about its structure, I'll simply use a letter like 'G'.[1]

As we already noted, the group operation can be very different from case to case – all that's required is that the operation satisfies Defn. 1. But it is customary to default to using multiplicative notation and to talk generically of group 'products';[2] we will correspondingly default to denoting the unique inverse of a group object x by x^{-1}.

[1] Looking forward, a standard notation – the one I adopt – uses italic capital letters for objects in categories, objects which might will be structures like groups. Hence for consistency I'll use 'G' (syntactically a singular term, taking a singular verb) for a group, contrasting with 'G' (a plural term) for the objects which form the structure. I'll say a little more about what this distinction might or might not come to in the next chapter.

[2] By tradition, however, additive notation is commonly used when dealing with abelian groups.

2.2 A 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 second-level functions (sending first-level functions to first-level functions), etc.

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

I stress the point in this familiar context because, 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 various natural ways. For a start, there are subgroups, in the entirely predictable sense:

Definition 2. The group S is a subgroup of the group G iff (i) all S's objects belong to G too, and (ii) S's group operation is the restriction of G's operation to S's objects. △

Example: the even integers (still with addition as the group operation, and with zero as the group identity) form a subgroup of the additive group of integers. For another example: the complex numbers on the unit circle form a subgroup of the multiplicative group of non-zero complex numbers.

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

Definition 3. Given the objects X and the objects Y, a scheme for pairing X with Y comprises

(i) some pair-objects O (which can be any suitable objects, and which may or may not be disjoint from X and/or Y);

(ii) a binary pairing function which we can notate '$\langle\ ,\ \rangle$' which sends $x \varepsilon$ X and $y \varepsilon$ Y to a pair-object $\langle x, y\rangle \varepsilon$ O (where every $o \varepsilon$ O is indeed some such $\langle x, y\rangle$);

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

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

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 both X and Y are the natural numbers, then we could perfectly well take suitable pair-objects $\langle m, n\rangle$ to be the numbers $2^m 3^n$, with the obvious pairing and unpairing functions. What matters about the pair-objects in a scheme is not their intrinsic nature but the role they play (a categorially flavoured point I press again in Chapter 9).

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

Definition 4. Suppose G and H are respectively the groups $(\mathrm{G}, *, e)$ and $(\mathrm{H}, \star, d)$. And suppose we have some scheme for pairing the objects G and H using the pair-objects K – so $x \,\varepsilon\, \mathrm{G}$ and $y \,\varepsilon\, \mathrm{H}$ are mapped to a pair-object $\langle x, y\rangle \,\varepsilon\, \mathrm{K}$.

Put $k = \langle e, d\rangle$, and define multiplication of pairs componentwise, so $\langle x, y\rangle \diamond \langle x', y'\rangle = \langle x * x', y \star y'\rangle$. Then $(\mathrm{K}, \diamond, k)$ form a group K, a *product* of the groups G and H, which we can notate $G \times H$. △

It is routine to check that $(\mathrm{K}, \diamond, k)$ really do form a group.

For a very simple example, suppose the group J comprises just the two objects e, j. If a group K_1 is to be a product of 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 objects being the group identity. For brevity's sake, call these four pair-objects $1, a, b, c$ respectively. K_1's group operation $\diamond$ is then defined by the following table (read the table entry as giving the value of row-object $\diamond$ column-object):

$\diamond$	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 K_1 is abelian.

Note that we speak here of 'a' product of the group 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 this 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: any way of forming a product group from a two-object group and itself gives us a group describable by reinterpreting the same table.

The point of course generalizes. Products 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. Given the objects G and an equivalence relation $\sim$ defined over them, a scheme for quotienting G by $\sim$ comprises

(i) some quotient-objects Q (which can be any suitable objects, which may or may not be disjoint from G),
(ii) a unary function which we can notate '[ ]' which sends $x \varepsilon$ G to a quotient-object $[x] \varepsilon$ Q (with every $q \varepsilon$ Q being some such $[x]$), where
(iii) for all $x, y \varepsilon$ 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 do *not* require $[x]$ to be an equivalence class or other set. For example, take the integers and consider the equivalence relation $\equiv_8$, i.e. congruence mod 8. Then in this case we can simply put $[x]$ to be the remainder when x is divided by 8, since $[x] = [y]$ iff $x \equiv_8 y$.

Again, what really matters about quotient-objects is not their 'internal' nature but their 'external' liaisons, the role they serve in a quotient scheme. And indeed, as with the parallel point about pairs, this point about quotients illustrates what will turn out to be a quite central motif of category theory, namely the crucial importance of 'external' relations in pinning down what we care about in various constructions.

We can now define the notion of a quotient group. Suppose we take a group and find some objects which 'behave equivalently' in the group. Then we can, as it were, collapse these objects together, thereby forming a new group.

Less metaphorically, let's say

Definition 6. Given a group $(\mathrm{G}, *, e)$, then $\sim$ is a *congruence* relation for the group iff it is an equivalence relation on the group objects which respects the group structure in the following sense: for any objects $x, y, z \varepsilon$ 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). △

And now we use a quotient scheme to, as it were, collapse congruent objects together:

Definition 7. Suppose that we have a group $(\mathrm{G}, *, e)$, and $\sim$ is a congruence relation for the group. And suppose we also have a quotient scheme for $\sim$, which sends a group object x to $[x]$ (so the function notated '[ ]' in effect ignores the distinction between congruent objects). Let G/$\sim$ be all the objects $[x]$ for $x \varepsilon$ G, and put $[x] \star [y] = [x * y]$.

Then G/$\sim$ equipped with the operation $\star$ and with $[e]$ as the operation's identity also form a group, which we'll denote *G/*$\sim$, a *quotient* of the original group G with respect to $\sim$. △

For this definition to work, $\star$ has to be a genuine function. So 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 $\sim$ is a congruence respecting 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 $(\mathrm{G}/\!\sim, \star, [e])$ do form a group. But that's quite straightforward.

Take a quick example. Let Z be the group $(\mathbb{Z}, +, 0)$ formed by the integers $\mathbb{Z}$ under addition:[3] and consider again the equivalence relation of congruence mod 8. 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, which we will together denote $\overline{8}$. Then $(\overline{8}, +_8, 0)$ – where $+_8$ is addition mod 8 – form a group we can call $Z/\!\equiv_8$, which is a quotient of $(\mathbb{Z}, +, 0)$ by $\equiv_8$.

Note, we again talk of 'a' quotient of a group by a given equivalence relation rather than of 'the' quotient group. There will be many ways of finding quotient schemes for a congruence $\sim$ defined over objects G, hence there can be many alternative candidates $\mathrm{G}/\!\sim$ from which to build a quotient group (though, as with product groups, quotient groups constructed using different quotient schemes will all 'look the same').

2.4 Group homomorphisms

(a) Let's move on and equally briskly recall some basic facts about structure-respecting maps between the groups.

Definition 8. A *group homomorphism* from the group $(\mathrm{G}, *, e)$ as source to the group $(\mathrm{H}, \star, d)$ as target is a function f defined over G with values among H such that for every $x, y \varepsilon \mathrm{G}$, $f(x * y) = fx \star fy$. △

In sum, such a homomorphism sends products of objects in the source group to corresponding products in the target group.

When we want to make explicit the structure of the groups G and H which a homomorphism connects, then we can explicitly write $f\colon (\mathrm{G}, *, e) \to (\mathrm{H}, \star, d)$. But when the structural details are not germane, we will simply write $f\colon G \to H$.

(b) It is immediate that a homomorphism sends a group identity to another group identity, and sends inverses to inverses:

Theorem 1. *If f is a group homomorphism from $(\mathrm{G}, *, e)$ to $(\mathrm{H}, \star, d)$ then*

(i) $fe = d$,

(ii) for any $x \varepsilon \mathrm{G}$, $f(x^{-1}) = (fx)^{-1}$.

[3] I'll recycle familiar set-theoretic notation like '$\mathbb{Z}$' for plural use when convenient.

Proof. For (i), we have $fe = fe \star d = fe \star (fe \star (fe)^{-1}) = (fe \star fe) \star (fe)^{-1} = f(e * e) \star (fe)^{-1} = fe \star (fe)^{-1} = d$.

And for (ii), we note $f(x) \star f(x^{-1}) = f(x * x^{-1}) = f(e) = d$, and similarly $f(x^{-1}) \star f(x) = d$. So $f(x^{-1})$ is the (unique) inverse of $f(x)$. □

(c) Some initial rather trivial examples:

(1) Let $(\mathrm{G}, *, e)$ form a group. Then there is a unique homomorphism f from that group to any given one-object group, which sends every object from G to the sole object of the target group.

(2) Likewise, there is a unique homomorphism g in the opposite direction, from a given one-object group to $(\mathrm{G}, *, e)$. It's the function which sends the sole object of the first group to e, the group identity of the second.

(3) Relatedly, there is always a 'collapse' homomorphism h from a group $(\mathrm{G}, *, e)$ to itself which sends every object from G to the group identity e.

These cases remind us that, although homomorphisms are often described as *preserving* group structure, this does not mean replicating *all* structure. A homomorphism from G to H can compress many or most aspects of the group structure on G simply by mapping distinct G-objects to one and the same H-object. It really is better, then, to talk of homomorphisms as *respecting* group structure.

Three more interesting but still elementary examples:

(4) There is a homomorphism from Z, the additive group of integers $(\mathbb{Z}, +, 0)$, to any two object group J which sends even numbers to J's identity, and sends odd numbers to J's other object. Thought of just as a function from Z, the homomorphism here is surjective but not injective.

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

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

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

Theorem 2. *(1) Any two homomorphisms $f\colon G \to H$, $g\colon H \to J$, with the target of the first being the source of the second, will compose to give a homomorphism $g \circ f\colon G \to J$.*

(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 G, there is an identity homomorphism $1_G : G \to G$ which sends each object to itself. Then for any $f : H \to J$ we have $f \circ 1_H = f = 1_J \circ f$.

Proof sketch. For (1) we simply take $g \circ f$ ('g following f') applied to an object x from the group 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 simply thought of as mapping objects to objects.

(3) is also immediate. □

(e) That was very easy! But note however the important fact that this, only our second theorem, is not (repeat, *not*) a mere logical consequence of our definitions of groups and group homomorphisms. Our proof sketch plainly depends on invoking background assumptions about functions, such as the assumption that functional composition is associative. These assumptions may be entirely uncontentious, but that doesn't mean that they aren't needed.

And so it goes. Contrary to what is sometimes too-casually said, almost *nothing* in group theory follows merely from the definitions alone!

2.5 Group isomorphisms and automorphisms

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

Definition 9. A *group isomorphism* $f : G \xrightarrow{\sim} H$ is a homomorphism where the underlying function is a bijection between the respective objects of the two groups.

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

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

Again, let's have some elementary examples:

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

(2) There are two automorphisms from the additive group $(\mathbb{Z}, +, 0)$ to itself. One is the identity homomorphism; the other is the function which sends an integer j to $-j$.

(3) There are infinitely many automorphisms from the group $(\mathbb{Q}, +, 0)$ 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 K_2 be the group consisting in $1, 3, 5, 7$ equipped with multiplication mod 8 as the group operation. And let 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 $K_2 \cong K_3$.

The easiest way to see this is by constructing an abstract 'multiplication table'. First, take $1, a, b, c$ to be respectively the numbers $1, 3, 5, 7$, and take the group operation $\diamond$ 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 $\diamond$ to be composition. Both times we get the same table: in fact we get the same table again as for 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 $\diamond$ gives us the claimed isomorphism $f\colon K_2 \xrightarrow{\sim} K_3$. By the same reasoning, both groups are isomorphic to 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.

(b) 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 (I said) will 'look the same', and have the same multiplication table. We can now put it like this: suppose J_1 and J_2 are both products of G with H; then $J_1 \cong J_2$.

Why? Just take the map $j\colon J_1 \to J_2$ which sends the pair-object $\langle x, y\rangle_1$ to $\langle x, y\rangle_2$ – where $\langle x, y\rangle_1$ pairs x from G and y from H according to the pairing scheme used in constructing J_1, and $\langle x, y\rangle_2$ pairs the same objects according to the pairing scheme used in constructing J_2. This bijection j is evidently a group isomorphism, so $J_1 \cong J_2$.

Likewise, suppose J_1' and J_2' are now different quotients of a group 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 G. Take the bijection j' 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 this is a group isomorphism, and so we again have $J_1' \cong J_2'$.

(c) Another very easy result, which gives us an alternative characterization of isomorphisms (which we will take over into category theory):

Theorem 3. *A group homomorphism $f\colon G \to H$ is an isomorphism iff it has a two-sided inverse, i.e. there is a homomorphism $g\colon H \to G$ such that $g \circ f = 1_G$ and $f \circ g = 1_H$.*

Proof. Suppose $f\colon (\mathrm{G}, *, e) \to (\mathrm{H}, \star, d)$ is a group isomorphism. Then the underlying function $f\colon \mathrm{G} \to \mathrm{H}$ is a bijection and therefore has a two-sided inverse

$g\colon \mathrm{H} \to \mathrm{G}$. So we only need to confirm that this inverse function g gives rise to a homomorphism $g\colon (\mathrm{H}, \star, d) \to (\mathrm{G}, *, e)$.

But since f is a homomorphism, $(fgx \star fgy) = f(gx * gy)$; and so, since g is a two-sided inverse for f, we have $g(x \star y) = g(fgx \star fgy) = gf(gx * gy) = gx * gy$. So g is indeed a homomorphism.

Conversely, suppose f is a homomorphism with a two-sided inverse. Then just as function between objects it 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 again an isomorphism. Given that isomorphisms are homomorphisms with two-sided inverses which are homomorphisms, it is also immediate that the inverse of an isomorphism is also an isomorphism. Therefore, as we would want,

Theorem 4. *Being isomorphic is an equivalence relation between groups.* □

2.6 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-respecting maps between groups. We now bring the two themes together, foreshadowing again what will be an absolutely key motif of category theory.

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

Theorem 5. *Suppose $f\colon (\mathrm{G}, *, e) \to (\mathrm{H}, \star, d)$ is a group homomorphism, and let $f(\mathrm{G})$ be all the objects which are an f-image of some object from G. Then those objects equipped with the operation $\star$, form a group – the f-image of $(\mathrm{G}, *, e)$ – which is a subgroup of the group $(\mathrm{H}, \star, d)$.*

Proof. (i) Suppose $y_1, y_2 \varepsilon f(\mathrm{G})$. By assumption, these objects are f-images of some objects $x_1, x_2 \varepsilon \mathrm{G}$. So we have $y_1 \star y_2 = fx_1 \star fx_2 = f(x_1 * x_2)$, and hence $y_1 \star y_2 \varepsilon f(\mathrm{G})$ as required.

(ii) Trivally, $d = f(e) \varepsilon f(\mathrm{G})$.

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

Those three points establish that $(f(\mathrm{G}), \star, d)$ form a group, and it is trivially a subgroup of the group $(\mathrm{H}, \star, d)$. □

The reverse theorem is trivial:

Theorem 6. *For any subgroup S of a given group H, there is a homomorphism $f\colon G \to H$ such that S is the f-image of G.* □

Just put $G = S$ and take the injection map which sends a G-object to itself as an H-object.

Combining those results, we can characterize all the subgroups of a given group using homomorphisms with that group as their target. 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 noting another important idea (very familiar if you have done even a little group theory):

Definition 10. The group $(\mathrm{H}, *, e)$ is a *normal subgroup* of $(\mathrm{G}, *, e)$ iff it is a subgroup and, for any $h \varepsilon \mathrm{H}$ and any $g \varepsilon \mathrm{G}$, then $g * h * g^{-1} \varepsilon \mathrm{H}$. △

And now for the desired result linking homorphisms to (normal) subgroups:

Theorem 7. *Suppose* $f\colon (\mathrm{G}, *, e) \to (\mathrm{H}, \star, d)$ *is a group homomorphism, and let* K *be the objects among* G *which* f *maps to the identity element* d*. Then* $(\mathrm{K}, *, e)$ *form a normal subgroup of* $(\mathrm{G}, *, e)$*, the* kernel *of* f*.*

Proof. We need to establish the closure of K under the operation $*$. But suppose $k_1, k_2 \varepsilon \mathrm{K}$. Then $f(k_1 * k_2) = fk_1 \star fk_2 = d \star d = d$. Hence $(k_1 * k_2) \varepsilon \mathrm{K}$.

By the definition of a homomorphism, $f(e) = d$, so $e \varepsilon \mathrm{K}$. Then recall that homomorphisms send inverses to inverses. Therefore if $f(k) = d$, then $f(k^{-1}) = d^{-1} = d$; so if $k \varepsilon \mathrm{K}$, then $k^{-1} \varepsilon \mathrm{K}$. Hence $(\mathrm{K}, *, e)$ form a subgroup of the group $(\mathrm{G}, *, e)$.

For normality, we simply note that for any $k \varepsilon \mathrm{K}$ and $g \varepsilon \mathrm{G}$, $f(g * k * g^{-1}) = f(g) \star f(k) \star f(g^{-1}) = f(g) \star d \star f(g)^{-1} = d$. Therefore $g * k * g^{-1} \varepsilon \mathrm{K}$. □

There is a converse theorem too, that every normal subgroup for a group G is the kernel of some homomorphism with the source 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 8. *Given a group homomorphism* $f\colon G \to H$*, and* x, y *from among* G*'s objects, put* $x \sim y$ *iff* $fx = fy$*. Then* $\sim$ *is a congruence on* G *and the* f*-image of the group* G *is a quotient group* $G/\!\sim$*. Conversely, given a quotient group of* G *with respect to a congruence relation* $\sim$*, we can find a homomorphism* f *with the source* G *such that* $G/\!\sim$ *is the* f*-image of* G*.*

Proof. The relation $\sim$ of being equalized-by-f is trivially an equivalence relation. But we need to check that $\sim$ respects G's group operation $*$ so that $G/\!\sim$ 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) $z * x \sim z * x$.

For (i), if $x \sim y$, then $fx = fy$, hence $f(x * z) = fx \star fz = fy \star fz = f(y * z)$, so $x * z \sim y * z$ (here, $\star$ is H's group operation). Case (ii) is exactly similar.

By the definition of $\sim$, the f-images of G's objects act like quotient-objects with respect to $\sim$; hence G's image under f is indeed a quotient group $G/\!\sim$.

For the converse result, suppose $G/\!\sim$ is a quotient of G with respect to some equivalence relation $\sim$, with $f_\sim \colon x \mapsto [x]$ giving us the relevant quotient scheme. Then $f_\sim \colon G \to G/\!\sim$ is easily checked to be our required homomorphism. □

So again we can trade in certain claims about the internal quotient structure of groups, for corresponding external claims about homomorphisms between groups. And note the revealed duality between quotient groups and subgroups: given a homomorphism $f \colon G \to H$, the f-image of G is a quotient of G and a subgroup of 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. But I'll leave the proof of this for later. For now, I'll just flag up again the key 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 a pivotal theme of category theory.

2.7 'Identical up to isomorphism'

Let's pause over another important point. We have met the groups K_1, K_2, 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 might 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 only on the *functional relations between the objects* induced by the group operation. 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 G and H are isomorphic if there is a bijection between them which preserves the group operations. Intuitively, G and H are the same group except that the elements and the operations may be written differently in G and H. (Dummit and Foote 2004, p. 37)

But that surely isn't a happy way to putting things. We have just reminded ourselves that K_1, K_2 and K_3 are isomorphic groups. But K_2, for example, comprises four *numbers* as its objects, and K_3 comprises four *operations* on a non-equilateral rectangle; and there is no reasonable 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 with different elements and different group operations which, however, can be 'written the same', being represented by the same table under different interpretations.

A seemingly rather happier, and certainly widely used, way of putting things is this: our Klein groups K_1, K_2 and K_3 are *identical up to isomorphism*. And for many purposes, group theory can simply ignore the differences between groups which are identical up to isomorphism. Hence the frequently encountered talk of '*the*' Klein group.

We will eventually have to return to the question of quite what such talk can amount to.[4] For now, we'll simply note that ignoring the differences between isomorphic widgets becomes a lead theme of category theory too.

2.8 Categories of groups

(a) That will do by way of an initial review of some basic facts about groups and the homomorphisms between them.[5] Let's now ask: given some groups and some homomorphisms between them, what does it take for these to form a structured family which counts as a *category* of groups?

We impose just two very natural conditions. First, the homomorphisms in the category should be closed under composition. In other words, if f takes us from G_1 to G_2, and g takes us from G_2 to G_3, then we want $g \circ f$ to be available to take us from G_1 to G_3. Second, for each of the groups in the category, its trivial identity homomorphism needs to be included. And that's *all* it takes: it really is that simple!

Still, let's say the same thing again, but this time in more laborious detail, for clarity's sake:

Definition 11. A *category of groups* comprises

(1) some groups, Grp, and
(2) some homomorphisms, Hom,

where Grp and Hom are governed by the following conditions:

[4] Occasionally, we meet the idea that, as well as 'concrete' Klein groups, i.e. Klein 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. And then it is this group comprising these abstract de-natured elements which is then said to be, properly speaking, *the* Klein group. But does this suggestion really make sense? Evidently, more needs to said!

[5] We could have paused here to take in a similar review of some basic facts e.g. about topological spaces and the continuous maps between *them*, and the parallels would be instructive. But setting the scene would have taken too long, given that you are impatient to hear about categories!

Sources and targets If $f : G \to H$ is one of the homomorphisms Hom – is among Hom, for short – then both its source G and its target H are among Grp.

Composition If $f : G \to H$, $g : H \to J$ are both among Hom, where the target of f is the source of g, then $g \circ f : G \to J$ is also among Hom.

Identity homomorphisms If G is among Grp, the corresponding identity homomorphism $1_G : G \to G$ is among Hom.

Further, we have:

Associativity of composition. For any $f : G \to H$, $g : H \to J$, $h : J \to K$ among Hom, $h \circ (g \circ f) = (h \circ g) \circ f$.

Identity homomorphisms do behave as identities. For any $f : G \to H$ among Hom, $f \circ 1_G = f = 1_H \circ f$. △

Of course, we know the last two conditions will automatically be satisfied because of Theorem 2. But I'm redundantly mentioning those conditions again here so that our definition of categories of groups matches up nicely with our general definition of categories in Chapter 4.

(b) Groups are many and various; so too are categories of groups. For example, a single group G together with its identity homomorphism $1_G : G \to G$ counts as a one-object 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 collects all the finite groups whose objects are some natural numbers together with all the isomorphisms between those groups. 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 built from numbers, but this time combined with *all* the homomorphisms between them (whether isomorphisms or not). And *now* non-isomorphic groups can 'see' each other. So we have enough homomorphisms in play to be able e.g. distinguish the one-object groups in the category by saying that these are the groups which have one and only one homomorphism to and from every other group, as noted 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 §2.6. Developing this sort of story will be a primary item of business in the coming chapters.

(c) And there will be lots more categories of groups – e.g. the category of symmetry groups of finite regular polygons and the homomorphisms between them, the category of infinite abelian groups of natural numbers and their homomorphisms, and so on and so forth.

Now those categories of groups, those relatively small-scale families of structures, all seem rather tamely unproblematic. But we might now wonder about

bigger categories: indeed, will there in fact be an inclusive mega-category of *all* groups and *all* the homomorphisms between them?

A very good question! To get a handle on it, there are some troublesome issues we need to tangle with. Let's make them the business for the next chapter.

3 Where do categories of groups live?

Where do mathematical structures live? In particular, where do groups, and the families of interconnected groups that form categories of groups, live? 'In the universe of sets', comes the speedy conventional reply.

But let's not rush on too fast. After all, the word 'set' didn't even occur in the last chapter (except in saying that the objects in pairing and quotient schemes need *not* be thought of as sets). So we should pause to think a little about what that conventional answer buys us.

Is it compulsory to go set-theoretic?

3.1 Sets, virtual classes, plurals

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

However, if this is the guiding conception, then the very first thing to say is that a great deal of elementary informal talk of sets or classes is really no more than a façon de parler. Yes, it is a useful and now very familiar idiom for talking about many things at once. But in a whole range of elementary contexts informal talk of a set or class doesn't really carry any serious commitment to there being any *additional* object over and above those many things. In other words, singular talk of *the set/class of widgets* can very often be traded in without loss for plural talk of *the widgets*.

Here is Paul Finsler writing a century ago, emphasizing the key distinction we need (and adding a bit of linguistic stipulation):

> It would ... be inconvenient if one always had to speak of many things in the plural; it is much more convenient to use the singular 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. (Finsler 1926, p. 106, quoted in Incurvati 2020, p. 3.)

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 stipulative choice of the particular words 'set' vs 'class' to mark the distinction.[1] What matters is the contrast between uneliminable talk of sets in Cantor's sense of entities in their own right and, on the other hand, non-committal talk, eliminable in favour of plural locutions. And here is Quine making the key point in a later and much more famous passage:

> Much ... of what is commonly said of classes with the help of '$\in$' can be accounted for as a mere manner of speaking, involving no real reference to classes nor any irreducible use of '$\in$'. ... [T]his part of class theory ... I call the virtual theory of classes. (Quine 1963, p. 16)

This same usage plays an important role in set theory itself in some treatments of so-called 'proper classes' as distinguished from sets. For example, in his standard book *Set Theory*, Kenneth Kunen writes

> Formally, proper classes do not exist, and expressions involving them must be thought of as abbreviations for expressions not involving them. (Kunen 1980, p. 24)

But, to complicate matters, other developments of set theory do allow for proper classes (classes which are 'too big to be sets') to count as entities in their own right. So we can't reliably use 'class' and expect to be understood in Finsler's way.

Let me go in, then, for a bit of minor linguistic stipulation of my own. When I talk of a 'set' I will, as I said, understand that in Cantor's sense to be referring to a single object, something over and above its members. I'll avoid talk of 'classes' as dangerously ambiguous. And I'll use 'collection' – a term which carries minimal theoretical baggage – when I want a more non-committal singular term for talking about many things at once.

(b) Finsler perhaps rather exaggerates the supposed inconvenience of plural talk, however. There is nothing at all unusual or forced about the use of plural terms in mathematics. Consider, for example, terms such as 'the complex fifth roots of 1', 'the real numbers between 0 and 1', 'the points where line L intersects curve C', 'the finite groups of order 8', 'the premisses' (of a certain argument), 'Hilbert's axioms for geometry', 'the symmetries of a rectangle', 'the ordinals', etc., etc. Mathematicians habitually use such terms which, taken at face value, refer plurally, to many things; and they use them all the time 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 *the widgets* (or whatever) as disguised singular talk referring to *the set of widgets*. You in fact already know that we can't *always* construe plural talk as being about some corresponding set. We can't,

[1] Bertrand Russell had earlier contrasted a 'class as one' with a 'class as many' to mark a version of the same distinction (Russell 1903, §70).

for example, trade in universally generalizing plural talk about the ordinals for singular talk about the set of ordinals because 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, at least according to standard set theories).

But we don't need to rely on such special cases to rebut the tempting thought. The more fundamental point is that, when we get down to details, it is just impossible to systematically eliminate plural talk in favour of singular talk of sets; and even piecemeal attempts require the most ad hoc and implausible contortions.

It would be far too distracting to pursue the twists and turns of the arguments for that last claim here. But the headline proposition is that *plural talk is in perfectly good logical order as it is*, and does not need to be re-interpreted as referring to sets in anything like Cantor's sense. And that is still true, even if your measure of being in good logical order is formalizability.[2]

(c) Why fuss about this point? Because it matters when we turn to talking about categories. We'll find that a lot of interesting categories are *large* in the sense that they comprise too many structures to form sets. So we can't straightforwardly treat every category as a *set* of structures suitably equipped with maps between them – at least on the usual story about sets.

One option is to get fancier with our set theory, and eventually I will need to say something about this possibility. But a more modest, immediately available, option would be to talk of our large categories as comprising a class of structures, meaning a virtual class, Finsler/Quine-style. However, it is less likely to lead to confusion if, when we first define categories, we instead use frankly plural idioms from the start. As we will see, this is in fact a common tack, even if it isn't always announced as such. And this is exactly the line I took at the end of the last chapter when I adopted plural talk in introducing the idea of a category of groups: I said some groups (one or more) and some homomorphisms (one or more) form a category if they together satisfy certain conditions – no talk yet of sets or classes.

3.2 One 'generous arena' in which to pursue group theory

(a) Back, though, to my initial presentation of some elementary group theory in the last chapter. There too I proceeded using a plural idiom, avoiding explicit talk of groups as sets.

This was the mildly deviant aspect of the presentation. It is of course standard to write something like 'A group is a set G, together with a binary operation $\star$ on G which has the following properties ...' (Beardon 2005, p. 2) or 'A group is an ordered pair $(G, *)$ where G is a set and $*$ is a binary operation on G satisfying

[2]For a lot more on why we shouldn't try to eliminate plurals, and 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 (2016).

the following axioms ...' (Dummit and Foote 2004, p.16). But we should ask: what work is the notion of *set* really doing there?

I'm with Paulo Aluffi, who explicitly acknowledges at the beginning of his fine book *Algebra, Chapter 0* that the informal set idiom which he adopts in the standard way is actually "little more than a system of notation and terminology" (Aluffi 2009, p. 1). That seems surely right. At least at the outset, the story of elementary group theory will unfold in basically the same way in either system of notation and terminology, whether we habitually use singular talk of sets, or alternatively adopt a plural idiom as I did. Part of the point of the presentation in the last chapter was to make this claim begin to seem tenable.

Now, true enough, I did say that a group G (one thing?) is formed by the (perhaps many) objects G suitably equipped with a binary operation, and we can reasonably ask what the role is of the singular expression 'G' here? Is it just giving us a useful-but-eliminable way of talking in the singular about many things at once (compare Finsler/Quine on eliminable talk of classes)? Groups, we might perhaps say, are organized collections of objects in *some* sense of 'collection'. But how weak and attenuated a sense will suffice? We can certainly get a long way without worrying too much about *which* sense of 'collection' might be involved here.

However, some might very well object that this a merely superficial point, because a theory of collections-as-genuine-sets is still essentially required to be there, even if hovering off-stage, in any serious development of group theory. But why so?

(b) Perhaps we can pick out a couple of questions here which might seem to invite a set-theoretic answers.

First, reflect that – as we noted before – even as soon as we reach our trite Theorem 2 in the last chapter, we are in fact going beyond the mere logical consequences of our definitions of groups and group homomorphisms. So what do we in fact need to bring to the table to get group theory going? Answer:

(i) the usual mathematical stock-in-trade of a body of assumptions about *functions*, together with

(ii) a repertoire of available *constructions*.

For example, we assume that functions always do compose when they can (i.e. when the target of the first is the source of the second), and that composition is associative. We assume that a function with a two-sided inverse is a bijection. We assume that it makes determinate sense e.g. to talk about *all* the permutations of some given objects, or *all* the automorphisms on a given group. And so on, and so forth. These, of course, look pretty unproblematic assumptions – but as I emphasized before, they are needed all the same.

Again, we typically assume that we can construct what will serve as ordered pairs ad libitum; and we assume that whenever an equivalence relation partitions some objects we can somehow represent these partitions. More carefully, in our earlier terms, we assume pairing schemes and quotient schemes are available whenever we want them. And going forward, we will assume that we can not

only construct pairs, triples, and finite tuples more generally, but we can form infinite sequences too. We also need to assume that we can freely construct multiple 'copies' of whatever structures we already have. And so on, and so forth again.

I'm leaving the details vague, but quite intentionally so. I am simply gesturing at the way that standard textbook developments of group theory simply help themselves from the outset to a bunch of unproblematic background assumptions as needed as we go along. Which is fair enough. But this does raise an obvious first question. *What if we want to start getting more explicit and methodical about these background assumptions? Suppose we want to regiment these assumptions and organize them into a neat package – what package would suffice?*

(c) Shelve that issue for now, and consider a different issue arising from what we earlier said about groups.

Here's a silly-seeming question: suppose I cut out a cardboard non-equilateral rectangle: have I hereby brought into a being a new Klein four-group, the group of *this* new rectangle's own (approximate!) rotation/reflection symmetries?

Well, we were previously entirely permissive about where we can find our groups: on our Defn. 1, we just need some new objects (in a very broad sense) and a suitable operation on them, and then we get a new 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 I said before, group theory will for many purposes abstract from the differences between groups which are identical up to isomorphism.

OK: suppose that there is a capacious enough fixed abstract mathematical universe in which we can implement isomorphic copies of all the different kinds of groups we will ever want to study. Then we won't care about any additional copies of these groups which are (as it were) roaming outside in the wild, or popping into existence when I cut up a new bit of cardboard (ok, we might well care about physically realized groups when doing applied mathematics; but we won't care for the purposes of pure 'abstract' group theory).

But this last thought prompts a more sensible question. *Where can we find a suitably rich mathematical universe in which we can construct (copies of) all the groups we want?*

(d) We now have two related questions on the table: what package of assumptions about available functions and about structure-building constructions will suffice for group theory? where can we can find (at least copies of) all the groups we want, neatly corralled together?

And there is of course a *very* familiar joint answer! The universe of sets – as described by ZFC or some close relation – provides exactly the sort of generous arena where there is a plenitude of groups along with other mathematical structures, together with all the functions and constructions we want for ordinary mathematical purposes.[3] It provides the desired 'foundation', in one sense of

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

that contested notion, for group theory.

It might reasonably be claimed, therefore, that *that* is why, after all, it is in fact entirely appropriate to take the conventional line and talk about sets right from the very outset in doing group theory (for example). Once the wraps are off, once we make explicit the assumptions there in the background which we need to get our theory up and running, we will find that our theory really is set-theoretic through and through.

3.3 Alternative implementations?

Or so the story goes. A moment's reflection, however, suggests that the argument goes far too fast at the end. Yes, a suitable set theory may provide *one* generous arena in which we can implement all the gadgets we need in developing group theory. *But why suppose that it the only option?*

(a) We are so used to being told that various mathematical widgets and whatnots are to be defined as sets of one kind or another that it can take a bit of effort to loosen the grip of that doctrine. Given our overall project, though, this is worth doing. So let's backtrack for a moment and focus on the simple core case of implementing one-place functions. What's the standard set-theoretic story, and is it compulsory?

Fix on some way of implementing ordered pairs as sets, e.g. as Kuratowski pairs $\langle x, y\rangle_K = \{\{x\}, \{x, y\}\}$. Then here is a familiar and entirely unproblematic definition:

Definition 12. Given a function f with domain X and codomain Y, its *graph* is the set $\hat{f}$ of ordered pairs $\langle x, y\rangle_K$ where $x \varepsilon$ X and $y \varepsilon$ Y, and $fx = y$. △

Then an equally familiar orthodoxy, at least in its baldest and most unqualified form, *identifies* a function f with its graph $\hat{f}$.

Now, there's an immediate problem here. A function's graph doesn't fix its codomain. But, as we saw when talking about group homomorphisms, we in general do care not only about the sources of maps but about their targets too. Hence, for good reasons which also chime with category theory, it would be better – if we are going to identity a function with a set – to identify it with a set-theoretic *triple* whose members are the function's graph, its domain-as-a-set, and its codomain-as-a-set.

But put that point to one side for the present. Our worries about the simple set-theoretic orthodoxy will carry over, mutatis mutandis, to the fancier story, so we need not delay for now over this complication.

(b) Here's why we should resist any outright identification of a function with its graph.

For a start, to play on the set-theorists' own turf for a moment, 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.

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

Well, next note that treating a function as a set of ordered pairs involves arbitrary choices of implementation scheme.

(i) It is arbitrary to fix on Kuratowski's implementation of pairs. Other set-theoretic pairing schemes will work just as well.
(ii) Even relative to a choice of pairing scheme, we could equally well model a function by the set of pairs $\langle y, x\rangle$ where $f(x) = y$, rather than by the set of pairs $\langle x, y\rangle$ – and some textbooks do this. And again other choices are 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 equate a function with its graph as we conventionally just defined it – we have made the uniquely *right* choice, i.e. the choice that correctly identifies which set that function 'really' is. And if there is no fact of the matter about which set a given function is, then we can't flat-out identify the function with some set such as its graph.

(c) But that's still a relatively superficial point. We can dig deeper: *a function and its graph belong to different logical types* – and that is fundamentally why they *can't* be identical. Alonzo Church makes the key observation when he writes that

> 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)

For example, a function such as the factorial defined over the natural numbers is, of its nature, the type of gadget which yields a numerical value for a given 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 e.g. to sets of ordered pairs of numbers (graphs of numerical functions) in particular.

In insisting on a fundamental type-distinction between functions and objects, Church is here following Frege, whose metaphor of 'unsaturatedness' might be helpful. The picture is that functions of their nature are 'unsaturated', have a certain number of empty slots waiting to be filled appropriately when the function is applied to the right number of arguments. By contrast, an object like a set is already 'saturated', with no empty slots waiting to be filled.

In sum, a set of ordered pairs $\hat{f}$ can't *by itself* do the work of a function f, taking arguments and yielding output values. As the mathematician Terence

Tao, who has no philosophical axe to grind, briskly puts it in his introductory book on analysis,

> 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)

Which *of course* isn't 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.

So a function, strictly speaking, isn't a set. But what we can do in a set-theoretic environment is *implement* functions as graphs;[4] and we can then transmute a claim about a function into a corresponding set-theoretic claim about some set of ordered pairs.

(Though, to complicate the story, there is typically another step. Suppose we are considering, say, a one-place function of natural numbers. Then yes, we can implement this as a set of ordered pairs in a suitable universe of sets. But these won't be pairs of numbers – since strictly speaking numbers aren't themselves sets either[5] – but rather they are pairs of whatever-sets-we-choose-for-implementing-numbers. So if $\hat{m}$ is our preferred set-implementation for the natural number m, a numerical claim $f(m) = n$ is then mirrored by a set-theoretic claim of the form $\langle \hat{m}, \hat{n}\rangle \in \hat{f}$.)

Similarly, relations strictly speaking aren't sets either. The only genuine relation to be found in the world of sets is the set-membership relation; but what we can do in a set-theoretic environment is implement other relations via their extensions. So we can then mirror a claim about a relation by a claim about its extension.

(d) What is the point of insisting that the story about functions-as-graphs and relations-as-extensions doesn't tell us what functions and relations 'really' are, but rather reports one way of implementing them in the universe of sets? Am I very boringly splitting hairs?

I hope not! As announced, remember, I'm trying to loosen the grip of the standard identification of functions with their graphs, and thereby make room for the thought that there might in fact be other attractive ways of theorizing about the functions of ordinary mathematics in other foundational frameworks.

[4] No word really seems ideal. Talk of 'proxies', 'surrogates', 'representations' has variously misleading connotations. I'll lean mostly to talk of implementation, as that is common enough and is at least relatively colourless.

[5] The locus classicus for this point is Paul Benacerraf's – very readable! – 'What numbers could not be' (1965).

And now return to thinking about groups, and families of groups. Take e.g. the additive group of integers. Strictly, the objects of this group – the integers – aren't sets: and the binary addition function isn't a set either. The same goes for other groups of ordinary group-theory: their objects typically won't intrinsically be sets and their binary operation can't be. So what we can find in the set-theoretic universe should strictly be speaking be regarded as implementations of, or proxies for, groups.

Which is fine, of course! I am not for a moment wanting to deny that these proxies can serve certain theoretical purposes brilliantly well – I am certainly *not* in the business of scorning the business of modelling or implementing mathematical structures in a set-theoretic framework. I am just emphatically highlighting that we *are* here in the implementation business. And looking at things that way, we can more easily see that we shouldn't too hastily assume that a set-theoretic framework provides the *only* general arena, the only foundational framework, in which we can find a plenitude of surrogates or proxies for implementing the mathematical structures and constructions which we want to regiment and study.

Indeed, we can't rule out that an alternative choice of general framework *might* even do the job rather better in some respects (maybe with different costs and benefits accruing to the different choices). For example, treating all mathematical widgets and whatnots as if they are sets seemingly gives rise to such daft questions as 'is the square root function for complex numbers a member of π?', 'Does any simple group appear as a zero of the Riemann Zeta function?'. We can block such foolish questions by using some type-disciplined framework which more strongly distinguishes types of entities in our mathematical universe in the way that practicing mathematicians habitually do. So a modern type theory could be the way to go. And note, by the way, that types are often said to be 'collections' in some sense, with a type theory thus offering us a different theory of collections to standard set theory.

Or perhaps the general foundational framework we want will not just be broadly type-theoretic but will be essentially category-theoretic in flavour. Here's the logician Dana Scott:

> What we are probably seeking is a 'purer' view of functions: a theory of functions in themselves, not a theory of functions derived from sets. What, then, is a pure theory of functions? Answer: category theory. (Scott 1980, p. 406)

Scott quickly goes on to remark that the general notion of a category won't give us enough. But arguably a suitable *topos* (that's a particular sort of category which we'll eventually meet) does provide another sort of universe in which we can regiment much of our ordinary mathematics. Such a suggestion – which of course we won't be in a position to really understand for quite a while! – would be very puzzling indeed if we have already jumped too quickly to assuming that ordinary mathematics is already quite fixedly conventionally set-theoretic through and through.

3.4 'The' category of groups?

(a) Let's return at last to the question left hanging at the end of §2.8: does it make sense to talk about a mega-category of *all* groups and *all* the homomorphisms between them?

If we think of a group quite unrestrictedly, as just any objects equipped with a suitable binary operation that together satisfy Defn. 1, then it is far from clear that talk of 'all' groups can locate a definite fixed totality. But suppose that we concentrate our attention on *some* generous enough but determinate framework in which we can implement (copies of) of all the groups and group-theoretic gadgets we want. This framework might be the entirely predictable candidate, namely a large enough universe of sets. Or we might go for some alternative generous arena – involving, perhaps, collections of some rather different nature – in which we can still pursue ordinary mathematics (we'll eventually want to elaborate rather more on the options here). Then yes, working now in our chosen arena, it could well be sensible to talk about an inclusive mega-category Grp which comprises all the implementations of groups living in *that* universe and all the homomorphisms between *them*.

So it seems that, if we are going to talk sensibly about inclusive mega-categories like Grp, in the way that category theorists do, then we need to assume that there is *some* appropriately generous foundational framework available there in the background.

(b) Now, the typical approach in introductions to category theory is to suppose that categories *do* live in a world of sets. Hence the mega-category Grp, to stick with that example, is taken to comprise all groups-implemented-as-sets together with the group-homomorphisms-implemented-as-sets.

But we should note that going set-theoretic won't in itself be enough to fully pin down Grp. For a start, different universes of sets will make available implementations of different functions, so can change what group homomorphisms are available. And having different homomorphisms available will, at the margins, affect the answers to some group-theoretic problems.[6] But we really don't want to get involved with such issues here. So, even if we do assume Grp lives in some world of sets, it seems best to remain relatively neutral about the precise character of that universe, at least here at the beginning of our journey into category theory.[7]

[6] For some examples of questions of group theory which get different answers in different set universes, see tinyurl.com/groupqns.

[7] Presentations do differ on this. For example, in his classic *Categories for the Working Mathematician* Mac Lane presents the axioms for category theory as in our §4.1, before saying that a category is "any interpretation of the category axioms within set theory". Then in an early section titled 'Foundations' he does make a first pass at outlining the particular kind of set theory, an extension of ZFC, he has in mind (Mac Lane 1997, p.10, pp. 22–23). By contrast, in his much-used textbook *Categories*, Steve Awodey (2010) is much less committal, and indeed explicitly allows the possibility of working in other background systems than set theory. He defaults, though, to a set theoretic framework, while noting that "we sometimes run into difficulties with set theory as usually practiced" and leaving it somewhat open what

Would it be even better to remain neutral about whether Grp does live in a world of sets at all? Perhaps so. Perhaps we ought from the start to leave the door wide open to the possibility that our categories in the end just shouldn't be thought of in conventionally set-theoretic terms. On balance, however, I think the least distracting policy is to keep marching in step with other introductions to category theory. So pro tempore, when we do talk about a mega-category like Grp, you can take it that we are talking about the category of all the relevant structures implemented in *some* suitably capacious, though not-yet-specified, universe of sets. But I'm flagging up that eventually we might want to loosen even this much initial anchorage in a world of sets as standardly understood. However, there is a great deal of ground to cover first. Let's see how things pan out.

is the best way to handle that fact: "we will not worry about this when it is just a matter of technical foundations" (Awodey 2010, 24–25).

4 Categories in general

We have met categories of only one kind so far, namely categories comprising some groups and enough homomorphisms between them. Here, 'enough' stands in for some pretty minimal requirements – essentially 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

(a) We said that many paradigm examples of categories are – as in our first illustrative case of categories of groups – families of structures with structure-respecting 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-respecting map f from A to B, and another structure-respecting 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-respecting 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

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

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 first 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-respecting 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-

respecting maps between them, and – using less specific terminology – we'll now speak very generally of *objects* and of *arrows* between them. Then we say:

Definition 13. A category C comprises two kinds of things:

(1) C-objects (which we will typically notate by $A, B, C, \ldots$),
(2) C-arrows (which we typically notate by $f, g, h, \ldots$).

These C-objects and 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 For any 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 defined in §2.8 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-respecting maps between them should also satisfy those axioms, and hence should count as forming a category too: I'll give examples in a moment.

(b) Note that – as trailed in §3.1(c) – we haven't defined a category as involving a *set* of objects and *set* of homomorphisms, because some categories have too many objects and homomorphisms to form sets (at least on a standard set-theoretic story). Many would dodge the problem by defining a category, then, as involving a *class* of objects and *class* of homomorphisms: but 'class' in exactly what sense? I prefer not to tangle with such issues at this point, and so use frankly plural locutions instead.[1]

Though it won't matter for now, we'll stretch a point and read the definition as allowing the zero case, of an empty category with no objects and homomorphisms. And we have already allowed the one-object, one-homomorphism case, when we allowed the boring category comprising one group and its identity homomorphism.

[1] I am *not* going out on a limb here! Respected texts which similarly define categories in plural terms – without making a song and dance about it – include those by Awodey (2010, p. 4), Lawvere and Schanuel (2009, p. 21) and McLarty (1992, p. 13).

(c) Here are seven more 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 non-committal 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 a category's first kind of data is quite standard. But note that, 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 structure, a group.

(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-respecting) 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 – other sorts of connections between objects might concern us. Which is why I generally 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. §6.2), which is why I prefer our common alternative terminology.

(v) I have adopted the more usual convention for notating a composite arrow.[2] It is again suggested by the functional paradigm. Suppose the two arrows $A \xrightarrow{f} B \xrightarrow{g} C$ are both functions in the ordinary sense, then $(g \circ f)(x) = g(f(x))$. Occasionally, to reduce clutter, we may write simply 'gf' rather than '$g \circ f$'.

Note the inversion here: g follows f in our mini-diagram, but is written before f in the notation '$g \circ f$'. You'll need to get used to this!

(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.

(vii) Finally for now, a very general point about naming categories. As Emily Riehl nicely puts it:

[2]Though some from computer science writing about categories do things the other way about.

> It is traditional to name a category after its objects; typically, the preferred choice of accompanying structure-preserving morphisms [arrows] is clear. However, this practice is somewhat contrary to the basic philosophy of category theory: that mathematical objects should always be considered in tandem with the morphisms between them. (Riehl 2017, p. 3).

We have in fact already seen that there can be different categories whose objects are the same groups but whose arrows are different selections of the structure-preserving morphisms between them.

4.2 Identity arrows

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

Theorem 9. *Identity arrows on a given object are unique; and the identity arrows on distinct objects are distinct.*[3]

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 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.[4] But I am not unusual in finding this bit of trickery rather unhelpful. 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.

4.3 Monoids and pre-ordered collections

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

(a) We have already met the example of various small-scale categories of groups and the inclusive large category **Grp**. But it is worth thinking now about a case where the algebraic structure is cut nearer to the bone.

Consider, say, the finite strings of symbols from some given alphabet, including the limiting case of the empty string, together with the operation of

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

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

concatenation. This operation is evidently associative, $s_1{}^\frown(s_2{}^\frown s_3) = (s_1{}^\frown s_2){}^\frown s_3$. And concatenating with the empty string leaves us where we were, so the empty string acts like an identity element for concatenation. This structure gives us a paradigm example of a *monoid* – which is, so to speak, a group minus the requirement for inverses. And a monoid homomorphism is then a function which respects monoid structure.

More carefully, we have:

Definition 14. The objects M with a distinguished object e, equipped with a binary operation $*$, form a *monoid* $(\mathrm{M}, *, e)$ – M for short – iff

(i) the binary operation $*$ maps monoid-objects to monoid-objects, i.e. for any $x, y \in \mathrm{M}$, $x * y \in \mathrm{M}$;
(ii) $*$ is associative, i.e. for any $x, y, z \in \mathrm{M}$, $(x * y) * z = x * (y * z)$;
(iii) $e \in \mathrm{M}$, and e acts as a monoid unit or identity, i.e. for any $x \in \mathrm{M}$, $x * e = x = e * x$.[5]

Further, a *monoid homomorphism* from $(\mathrm{M}, *, e)$ as source to $(\mathrm{N}, \star, d)$ as target is a function f defined over M with values among N such that:

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

(b) It is evident that monoid homomorphisms $f\colon M \to N$ and $g\colon N \to O$ compose to give a homomorphism $g \circ f\colon M \to O$. Composition of homomorphisms is associative. And the identity function on M is a homomorphism $f\colon M \to M$ which acts as an identity with respect to composition.

Hence, just as with groups, 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.

Assume now that we are working in some sufficiently capacious universe of sets which contains (implementations for) all the monoids we want together with (implementations for) their homomorphisms. We can then sensibly say:

(C1) Mon is the category whose objects are all the monoids and whose arrows are all the monoid homomorphisms (living in our chosen universe).

Fine print: yes, we should emphasize again that what we have in Mon are strictly speaking implementations of monoids and their homomorphisms. But still, these proxies for monoids and their homomorphisms can count perfectly well as objects and arrows in a category. In particular, note that arrows in a category don't have

[5] A factoid, for future reference in Part II! Any non-empty collection of objects can trivially be equipped with a binary operation to give a monoid. The one-object case is obvious. So suppose we have two or more objects, call them $0, e, a, b, c, \ldots$. Then define the operation $*$ so that $e * o = o * e = o$ for any object o (so e is our identity) while $o * o' = 0$ for any objects o, o' other than e.

By contrast, it isn't at all trivial that every collection of objects can be equipped with a binary operation making it a group. That is in fact an equivalent of the Axiom of Choice.

to be kosher functions. So, in a slogan, Mon will be a genuine category of proxies for monoids, and not a proxy category!

(c) Next, an example involving ordered objects; and again we'll cut structure to the bone by considering the simplest case, pre-orderings. Using 'collection' in a non-committal way (see the end of §3.1(a)) let's say

Definition 15. The objects P equipped with a relation $\preccurlyeq$ form a *pre-ordered collection* $(\mathrm{P}, \preccurlyeq)$ iff, for all $a, b, c \varepsilon \mathrm{P}$,

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

A monotone map $f\colon (\mathrm{P}, \preccurlyeq) \to (\mathrm{Q}, \sqsubseteq)$ between such pre-ordered collections is then defined to be a function f from the objects P into Q which respects order, i.e. such that for any $a, b \varepsilon \mathrm{P}$, if $a \preccurlyeq b$, then $fa \sqsubseteq fb$. △

It is obvious that monotone maps between pre-ordered collections will compose to give monotone maps; and the identity map on some pre-ordered objects gives rise to an identity monotone map.

Evidently, then, we can we have categories with the following kind of data:

(1) objects: various pre-ordered collections $(\mathrm{P}, \preccurlyeq)$,
(2) arrows: enough monotone maps between these various objects,

where (and we won't keep repeating this) 'enough' means the maps are closed under composition and each $(\mathrm{P}, \preccurlyeq)$ gets its own identity map.

OK, now assume again that we are working in some capacious universe where collections are treated as Cantorian sets. So there we can have a set P whose members are (perhaps suitable proxies for) the objects P, and a corresponding set-implementation for the preorder relation $\preccurlyeq$: we'll use $(P, \preccurlyeq)$ to denote such a set-with-a-pre-order. Then we can sensibly say

(C2) Preord is the category whose objects are all the pre-ordered sets $(P, \preccurlyeq)$ and whose arrows are all the monotone set-functions between them (living in our chosen set universe).

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 by itself can also be thought of giving rise to a category. Here's how:

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

(1) M's sole object is some arbitrary entity – choose whatever you like, it *doesn't* have to be among the monoid's objects, and dub it '•';

(2) Then any object $a \varepsilon$ M counts as an M-arrow $a\colon \bullet \to \bullet$ (in other words, 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 then immediate that the category axioms are satisfied (check this!).

Note in this case, since the 'object' in the category 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 happen to be functions, the arrows of the associated category 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'.

Note too that there is a sort of converse to (C3). Any one-object category M gives rise to an associated monoid built from M's arrows, with multiplication in the associated monoid being composition of arrows. Hence 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.

(b) Similarly, while we can put pre-ordered collections and the monotone maps which interrelate them together to form a category, we can also think of a single collection of objects equipped with a pre-order as itself giving us a category. Here's how:

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

(1) P's objects are the objects P again;

(2) there is a (single) P-arrow from A to B if and only if $A \preccurlyeq B$ – this arrow might as well be identified as a pair $\langle A, B\rangle$ (according to *some* pairing scheme), which is assigned the 'source' A and 'target' B. We 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 $\langle A, A\rangle$; there is always such an arrow since $\preccurlyeq$ is reflexive.

It is immediate that, so defined, the arrows for P satisfy the associativity and identity axioms, so we do have another category here (check this!). And again, this isn't a category comprising structures and structure-respecting maps.

Conversely, any category with objects Obj and where there is at most one arrow between two objects can be regarded as a pre-ordered collection $(\mathrm{Obj}, \preccurlyeq)$, where for $A, B \varepsilon$ Obj, $A \preccurlyeq B$ just in case there is an arrow from A to B in the category. It is therefore natural to say

Definition 16. A *pre-order category* is a category such that, for any objects A and B, there is at most one arrow from A to B.

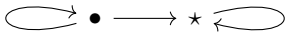
Hence we can think of the unrestricted notion of a category as a generalization of the case of pre-order categories.

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

(C5) For any objects we take, we get the *discrete category* on those by adding as few arrows as possible, i.e. just an identity arrow for each of objects we started with.

As noted before, we can for convenience allow the empty category, with zero objects and zero arrows. Otherwise, the smallest discrete category is 1 which has exactly one object and one arrow (the identity arrow on that object). Let's picture it in all its glory!

(C6) And having mentioned the one-object category 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:;



Call this category 2.

We could think this category as arising from the von Neumann ordinal 2, i.e. the set $\{\emptyset, \{\emptyset\}\}$; 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 1 (or *the* category 2, etc.)? Won't different choices of object make for different one-object categories, etc.? Well, yes and no! We can of course have, e.g. in our chosen set 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 exactly the same spirit as e.g. different Klein four-groups are 'essentially the same' in group theory. We will want to return to this point.

4.5 More categories

Note, I'm already falling into a notational habit which I will try to stick to pretty systematically. I'll use a sans serif font (as in 'Grp', 'Mon', etc.) for names of categories, and will also use the same font for informal variables for categories (as in 'C', 'J' etc.).

Let's continue, then, our list of varieties of category, first generalizing from our basic examples in §4.3, and then adding some geometric and other categories.

And now, both for brevity's sake and also to stay in step with presentations you'll find elsewhere, in most cases we will jump straight to the maximal version living in our favourite default universe of sets (so this will be the category which stands to other instances of the same general sort as e.g. Grp does to other categories of groups).

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

(C7) Ab is the category whose objects are all the (implementations of) abelian groups, and whose arrows are all the (implementations of) group homomorphisms living in our default universe.

(C8) Rng is, the category of rings, whose objects are predictably enough all (implementations of) rings and whose arrows are all (implementations of) ring homomorphisms living in our default universe.

(C9) And Bool is the category of Boolean algebras and structure-respecting maps between them. (It would be very boring to keep on repeating the reference to implementations in our favourite universe – so now take this as read, here and below!).

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 enough order-respecting functions as arrows, and we get another kind of category. For just one example, taking another maximal case,

(C10) Pos is the category whose objects are all the collections-equipped-with-some-partial-order (that's a pre-order which is anti-symmetric), and the arrows are all order-respecting maps again. Implemented in a universe of sets, this will be the category of posets, hence the name.

And so on it goes, for different kinds of orders!

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).

(C11) Top is the category with

(1) objects: all the topological spaces;
(2) arrows: the continuous maps between spaces.

(C12) Met is also a category: this has

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

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

(C13) Vect_k is a category with

(1) objects: vector spaces over the field k (each such space comprising vectors equipped with vector addition and multiplication by scalars in the field k);
(2) arrows: linear maps between the spaces.

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

(C14) Suppose L is a first-order formal language (the details don't matter). Then there is a category of propositions Prop_L with

(1) objects: propositions, closed sentences $X, Y, \ldots$ of the formal language;
(2) 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 Preord whose objects are sets-equipped-with-some-structure and whose arrows are structure-respecting-set-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-ordered-collection-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 it is useful to mention the standard distinction straight away.

(b) Now, the monoids in Mon are sets equipped with not-very-much structure. Likewise for the pre-ordered sets in Preord. Going in one direction, we get concrete categories whose objects are sets equipped with a richer structure and whose arrows are functions constrained to respect this richer structure. Going in the other direction, we get categories of naked sets – i.e. 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 respect structure because there is no structure to respect).

Here's the maximal case:

(C15) Set is the category with

(1) objects: all sets.

(2) 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 we have already reminded ourselves that the standard way of treating functions set-theoretically is simply to define a function f as its *graph* $\hat{f}$, i.e. the set of pairs $\langle x, y\rangle$ such that $f(x) = y$. And as we noted before, 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. Perhaps, as we said, set theorists themselves ought really to define a set-function $f\colon A \to B$ as a triple $\langle A, \hat{f}, B\rangle$. But anyway, that's how category theorists ought officially to regard arrows $f\colon A \to B$ in Set, and in other concrete categories too.

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

(iii) Note that in Set, the empty set is in fact the one and 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 (treat the bullet symbol here as a wildcard). Call your chosen singleton '1'. And consider the possible arrows (i.e. set-functions) from 1 to A.;[6]

We can represent the arrow from 1 to A which sends the element of the singleton 1 to $x \in A$ as $\vec{x}\colon 1 \to A$ (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 §8.3: but we have another glimpse ahead of how we might trade in talk of sets-and-their-elements for categorial talk of sets-and-arrows-between-them.

[6] 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.

(c) So far, so straightforward. But let's pause to remind ourselves that the make-up of the category Set of course is relative to our background universe and the way we implement functions there. We haven't determinately fixed that. But 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 that the sort of size considerations that we hinted at in §3.1(b) and §3.3(a) kick in again. The category of sets has all sets (in your favoured universe) as its objects. Unless you are an NF-iste,[7] however, there is no set of all sets – such a collection is, in a familiar way, 'too big' to be a set. On the standard view, the category of sets has more than a set's-worth of objects – hence the need for our something like our plural characterization.

4.7 Yet more examples

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

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

(C20) FinOrd is the category whose objects are the finite ordinals (you can take these to be the von Neumann ordinals), and whose arrows are the functions between them.

(C21) 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 which is 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.

(C22) $\mathsf{Set}_\star$ is the category (of 'pointed sets') with

(1) objects: all the non-empty sets, with each set A having a distinguished member $\star_A$;
(2) arrows: all the total functions $f\colon A \to B$ which map $\star_A$ to $\star_B$, for any non-empty sets A, B.

Pfn and $\mathsf{Set}_\star$ are in a good sense equivalent categories, as we will eventually be in a position to show (challenge: pause to think why we should expect that).

Let's also mention another kind of category, which arises when we equip sets not just with a single operation picking out a base point but with (so to speak) a whole monoid's worth of operations. So we fix on a monoid $M = (\mathrm{M}, *, e)$; and

[7] 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.

we then equip a set X with a family of operations $\lambda_m : X \to X$, one for each object $m \varepsilon$ M. And the idea is that the λ_m are to combine like the elements of the monoid M: so $\lambda_e = 1_X$ and $\lambda_m \circ \lambda_n = \lambda_{m*n}$. Denote a set X thus equipped with the operations λ_m by simply (X, λ). Then

(C23) For a given monoid M, the category M-Set is the category with

(1) objects: (X, λ) for all sets X and M-like families of operations λ;
(2) arrows: an arrow $f : (X, \lambda) \to (Y, \mu)$ is a function $f : X \to Y$ which respects the action of the corresponding operations on X and Y. In other words, if the operation λ_m on X sends x to x', then the corresponding operation μ_m on Y will send $f(x)$ to $f(x')$.

you might like to check that, with composition of arrows defined as ordinary function composition, and the obvious identity arrow, this really is a category.

Let's now have yet another example where the arrows in a category are *not* functions:

(C24) 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 whose extension is $\{\langle a, a \rangle \mid a \in A\}$. And $S \circ R : A \to C$, the composition of arrows $R : A \to B$ and $S : B \to C$, 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.

(b) There are various kinds of graph, depending on whether we allow more than one edge between nodes, whether edges are directed, and whether we allow edges looping round from a node back to itself. But here take a graph to comprises zero or more nodes together with zero or more directed edges between them, loops allowed. So the objects/arrows of a category can be thought of as constituting the nodes/edges of a special sort of graph in this sense, where every node has its own identity loop, and where edges compose (i.e. if there is an edge from A to B and an edge from B to C, then there is an edge from A to C).[8]

Now, in Part II of these Notes, we'll have to ask whether it makes sense to talk about a category of all categories. But it certainly makes sense to talk about a maximal category of graphs living in our default universe of sets. Thus

(C25) The category Graph has the following data:

(1) objects: all the graphs G living in our favoured universe (so nodes and edges are identified as sets).

[8] 'So is category theory just a department of graph theory?' Well, a glance at e.g. Béla Bollobás's classic 1998 text shows how very different the concerns of graph theory and category theory are. One key reason is that, for a graph in general, where edges don't compose, we get all kinds of highly non-trivial extremal problems about the length of paths between nodes. And as Bollobás remarks, "Extremal problems are at the very heart of graph theory."

(2) arrows: all the graph homomorphisms – these are functions f between graphs G_1 and G_2, which respect graph structure.

In more detail, a graph homomorphism f from G_1 and G_2 will have two components, f_N acting on nodes, and f_E acting on edges, where these components fit together in the obvious way. So if f_N maps the nodes a and b to $f_N(a)$ and $f_N(b)$, then f_E maps an edge from a to b to an edge from $f_N(a)$ to $f_N(b)$.

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

In fact, by this stage, you might very reasonably be wondering whether it isn't far *too* easy to be a category. If such very different sorts of structures as (i) a particular very small monoid, and (ii) a whole universe of topological spaces and their continuous maps, equally count as categories, then how much mileage can there possibly 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 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. These points are important enough to deserve a brief chapter to themselves.

5.1 Diagrams, in two senses

Talk of diagrams is in fact commonly used in three related ways. Later, in Part II, 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. Firstly:

Definition 17. A *representational diagram* is a directed graph with nodes representing objects from a given category C, and directed edges between nodes (drawn as arrows!) which represent arrows of C. Nodes and edges will normally be appropriately labelled, to make it clear what is being represented.

Two different nodes in a diagram can be joined by zero, one, or more directed edges. There can also be edges 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.

A directed edge (drawn arrow) labelled 'f' going from the node labelled 'A' to the node 'B' of course represents the arrow $f : A \to B$ of C. △

And then, relatedly:

Definition 18. A *diagram in a category* C is what is represented by a representational diagram – in other words, it will be some C-objects and some C-arrows between them. △

Note, diagrams (in either sense) needn't be *full*. That is to say, a diagram-as-a-picture need only represent *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 directed edges. So a path in a representational diagram from a node labelled 'X' to a node labelled 'Y' (for example) might look like this:

$$X \xrightarrow{f} Z_1 \xrightarrow{g} Z_2 \xrightarrow{h} Z_3 \xrightarrow{j} Y$$

This path-as-picture represents a connected chain of arrows (with the target of one arrow being the source of the next). The axiom about composition tells us that, in the represented category, there will also be an arrow $j \circ (h \circ (g \circ f))$ from the object X to the object Y: we will say that this arrow is obtained by composing along the path.

Two points. First, because of the associativity of composition we needn't actually worry about bracketing here, and can simply describe that composite as $j \circ h \circ g \circ f$ (or even plain $jhgf$). From now on, then, we freely insert or omit brackets in writing composites, doing whatever promotes local clarity.

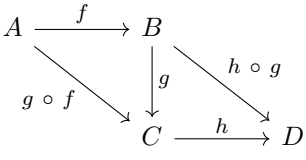
Second, in §4.1(b) we explained the rationale for our notational choice for the order in which we write the composite of two arrows. But note again that this does mean that the components in the notation '$j \circ h \circ g \circ f$' (or $jhgf$) occur in the opposite order to the path diagram.

(b) We now say – as our initial shot –

Definition 19. A representational diagram *commutes* iff, for any nodes X and Y and any two directed paths from X to Y, the arrow you obtain by composing along the first path is equal to the arrow you obtain composing along the second path.

Relatedly, a diagram in a category commutes iff it can be represented by a commutative diagram – i.e., iff composites taken along two chains of arrows between a source and final target are always equal.[1] △

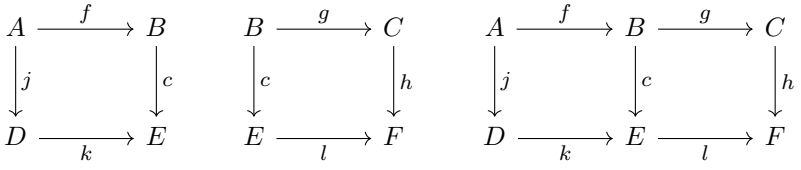
Hence, for example, the associativity axiom $h \circ (g \circ f) = (h \circ g) \circ f$ can be presented by saying that diagrams like the following always commute:



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

[1] Arbib and Manes (1975, p. 2) put it nicely: "*commutare* is the Latin for *exchange*, and we say that a diagram commutes if we can exchange paths, between two given points, with impunity."

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:



To check this, note that

$$h(gf) = (hg)f = (lc)f = l(cf) = l(kj)$$

with the equations holding alternately by associativity and by the assumed commutativity of the squares as we chase around the right-hand diagram.

These two cases illustrate a general claim: because of associativity, a diagram commutes if each minimal polygon in the diagram commutes. We can prove that by induction on the number of polygons. But since we won't need to invoke the general claim, we need not pause to prove it.

(c) We will meet many more examples of commutative diagrams in the coming chapters, so I won't give more illustrations yet. For the moment, two quick points worth emphasizing.

First, I've been a bit fussy in explicitly distinguishing the two ideas, a diagram-as-representation, and a diagram-as-what-is-represented. But having made the distinction, we will rarely need to bother about it, and can in future let context determine a sensible reading of informal claims about diagrams.

Second, do note – this is obvious but important! – that merely drawing a diagram with different routes from X to Y in the relevant category emphatically doesn't always mean that we have a *commutative* diagram: the identity of the composites along the paths in each case needs to be argued for (e.g. by showing the each component subdiagram commutes).

(d) OK, I have given a basic definition of a commutative diagram. But it turns out to be helpful to tweak it very slightly. Why so?

Well, later we will quite often be encountering e.g. 'fork' diagrams like this:

$$E \xrightarrow{e} A \underset{g}{\overset{f}{\rightrightarrows}} B$$

And it is really quite convenient to also count such a diagram as commuting so long as $f \circ e = g \circ e$, *without* also requiring the 'parallel' arrows from A to B to be equal. So, on our tweaked definition, we won't require arrows along *every* path between *any* X and Y to be equal: instead, we'll now officially say this:

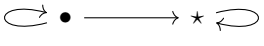
Definition 19*. A representational diagram *commutes* iff, for any nodes X and Y and any two directed paths from X to Y (*so long as at least one path contains more than one edge*), the arrow you obtain by composing along the first path is equal to the arrow you obtain composing along the second path.

Relatedly, a diagram in a category commutes iff it can be represented by a commutative diagram. △

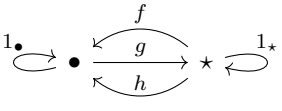
Of course nothing important hangs on explicitly tweaking the definition this way: some authors do, some don't.[2]

5.3 A reality check

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



Now consider the next diagram: is there a small category that would be fully pictured by a commutative diagram like the following, where $f \neq h$?



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

$$(hg)f = 1_\bullet f = f \neq h = h1_\star = h(gf).$$

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

[2]For some that do, see Arbib and Manes (1975, p. 4) and Roman (2017, p. 17).

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 additional constructions later, but these first examples should suffice to be going on with!

6.1 Subcategories, products, quotients

Three familiar ways of getting new widgets from old involve pruning to get sub-widgets, forming products of widgets, and quotienting by a suitable equivalence relation. We met these sorts of constructions on groups in §2.3. And we can do the same constructions with categories.

(a) We get a new subgroup from an old group by slimming down the group while retaining enough group structure. Similarly, the simplest way of getting a new category is by slimming down an old one while retaining enough categorial structure:

Definition 20. Given a category C, if S consists of the data

(1) objects: some or all of the C-objects,
(2) arrows: some or all of the C-arrows,

subject to the conditions

(i) for each S-object A, the C-arrow 1_A is also an S-arrow,
(ii) for any S-arrows $f\colon A \to B$, $g\colon B \to C$, the C-arrow $g \circ f\colon A \to C$ is also an S-arrow,

then, with composition of arrows in S defined as in the original category C, S is a *subcategory* of 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 S is still a category.

Many cases where we prune an existing category will leave us with constructions of no particular concern. Other cases can be more interesting:

(1) Lots of categories of groups-implemented-as-sets will be subcategories of Grp.

Note that, as we've set things up, not every category of groups will be a subcategory of Grp – for we don't rule out that suitably structured families of groups and their homomorphisms live outside the particular universe of sets in which the objects of Grp are to be found. Though by design Grp should contain *copies* of all the groups we want.

By contrast, we can say outright

(2) Ab is a subcategory of Grp.

For having fixed on our background universe of sets, all the abelian groups implemented in that universe will of course be among the groups living in that universe. Similarly to that case,

(3) FinOrd is a subcategory of FinSet;

(4) FinSet is a subcategory of Set;

(5) Set is a subcategory of Pfn.

And trivially,

(6) The discrete category on the objects of C is a subcategory of C for any category.

So, we can shed objects and/or arrows in moving down from a category to a subcategory. In examples (5) and (6) we keep all the objects but shed some or all of the non-identity arrows. While in cases (2), (3) and (4) we drop some objects while keeping all the existing arrows between the remaining objects, and there is a standard label for such cases:

Definition 21. If S is a subcategory of C where, for all S-objects A and B, the S-arrows from A to B are *all* the C-arrows from A to B, then S is said to be a *full subcategory* of C. △

(b) Next, the definition of products of categories is entirely predictable (I add this for the record, but the idea won't be of much interest to us):

Definition 22. If C and D are categories, then a product category $\mathsf{C} \times \mathsf{D}$ is such that

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

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

(c) Thirdly, and more interestingly, consider quotients. Following closely what we said about quotients for groups in Defn. 7, we can say:

Definition 23. (i) If 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, a congruence $f \sim g$ is an equivalence such that (i) if $f \sim g$, then f and g share the same source and target (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 C is a category, and suppose $\sim$ is a congruence on its arrows. And suppose we have a quotient scheme for $\sim$, which sends an arrow f (and its $\sim$-equivalents) to $[f]$. Then $\mathsf{C}/\sim$ is the *quotient category* whose objects are the same as those of C and whose arrows are all the $[f]$ for f in C, with $[f]$ assigned the same source and target as an arrow in $\mathsf{C}/\sim$ as f has in C. △

We've defined the notion of congruence so that it becomes trivial to check that $\mathsf{C}/\sim$ 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.

(d) Recall from §4.4 that a single monoid M gives rise a corresponding category M – this has some arbitrarily chosen thing as the sole object, while the category's arrows are the monoid-objects, with the monoid-operation as composition.

Now suppose $\sim$ is a congruence relation for M, i.e. an equivalence relation on the monoid objects which respects the monoid structure. This will then also be congruence relation between arrows in the associated category M.

Starting from M, then, we can now quotient to get a category in two ways. We can form the category M corresponding to the monoid M, and then construct the quotient category $\mathsf{M}/\sim$ in the way just described. Or we can first form a quotient monoid $M/\sim$ (in exactly the same way as we form a quotient group, see §2.3(c)), and then form the corresponding category to *that*.

Challenge: convince yourself that we end up with same category – on some sensible understanding of 'same category' – either way.

6.2 Duality

(a) Now for a centrally important new idea. A crucial way of getting a new category from old is by simply *reversing all the arrows*. More carefully, let's say:

Definition 24. Given a category C, then its *opposite* or *dual* C^{op} is the category such that

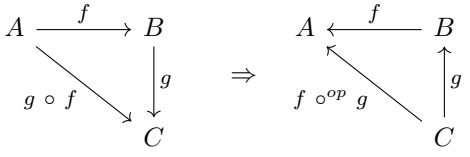
(1) The objects of C^{op} are the same as the objects of C.

(2) If f is an arrow of C with source A and target B, then f is also an arrow of C^{op} but there it is now assigned source B and target A.

(3) Identity arrows remain the same, i.e. $1_A^{op} = 1_A$.

(4) Composition-in-C^{op} is defined in terms of composition-in-C: put $f \circ^{op} g = g \circ f$. △

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



in C in C^{op}

It is easy to see that our definition is in good order and that C^{op} really is a category.

It is also trivial to check that $(\mathsf{C}^{op})^{op}$ is 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 a very different sort of category to Set (it is in fact equivalent to the category of complete atomic boolean algebras – but you don't need to know that!). And in general, taking the opposite category gives us something essentially new. Though not always. Consider the category Rel^{op}, for example, and remember that every relation in our background universe of sets comes as one of a couple with its converse.

(c) However, what will matter more for us is not the construction of particular opposite categories, but the following *duality principle* which arises from the fact that every category is the opposite of another category.

Let's get a bit formal for a moment. Take 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 '… ∘ …' which expresses the function which takes two composable arrows to another arrow. We can therefore regiment general propositions in the theory of categories in the language L.

[1] This is one of those cases where talking of 'domains' and 'codomains' instead of 'sources' and 'targets' could 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.

The following is then a natural definition:

Definition 25. Suppose φ is a wff of L. 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. △

And note, the duals of the axioms for a category are also instances of the axioms, as is quickly checked – which is why C^{op} is a category.

That last observation immediately gives us the duality principle we want:

Theorem 10. *Suppose φ is an 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 of category theory, so we have a proof of φ^{op} from the axioms. □

Semantic proof. If φ always holds, i.e. holds in every category C, then φ^{op} will hold in every C^{op} – but the 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 might be very simple but it is a hugely labour-saving result; we'll see this time and time again, starting in the next chapter.

6.3 Slice categories

(a) We will now look at another way of constructing new categories from old – or rather, we'll define a dual pair of constructions.

Suppose, then, that C is a category, and X is a particular C-object. We are first going to define a new category C/X whose *objects* are the pairs (A, f) for any C-object A and any C-arrow $f\colon A \to X$ (we will define the corresponding *arrows* of C/X in a moment).

And then there will be a dual construction, a new category X/C whose objects are again pairs (A, f), where A is any C-object but this time the arrow goes in the opposite direction, i.e. is an arrow $f\colon X \to A$ in the original category C.

But why should we be interested in such constructions? Let's have a couple of very simple examples:

(1) 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 a pair $(S, f\colon S \to I_n)$ gives us a set S whose members are coloured by f from that palette of n colours.

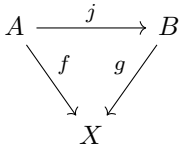
Hence with the arrows of the category appropriately defined – as I said, we'll come to that in a moment! – we can think of FinSet/I_n as the category of n-coloured finite sets, exactly the sort of structure that combinatorialists will be interested in.

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

Now think about a category $1/\mathsf{Set}$ whose objects are all the pairs of the form $(S, \vec{x}\colon 1 \to S)$. Each such object of $1/\mathsf{Set}$ provides us with a set and then a distinguished element of that set; in other words, the object works as a pointed set. Therefore, $1/\mathsf{Set}$ will be (or at least, in some strong sense to be explained later, comes to the same as) the category Set_* of pointed sets.

True, pointed sets aren't very exciting. But pointed topological spaces are. And, with 1 now some one-point topological space, $1/\mathsf{Top}$ similarly gives us the category of pointed topological spaces.

(b) OK, we have a modest amount of motivation for being interested in slice categories (explaining fancier examples would take us too far astray). So let's explore further.[2]



The obvious next question is: given C/X's objects $(A, f\colon A \to X)$ and $(B, g\colon B \to X)$ what's a sensible candidate for an *arrow* between them? If we want to construct C/X's data from ingredients available in C, what can we use to construct an arrow from (A, f) to (B, g)? The obvious candidate is a C-arrow j from A to B which interacts appropriately with the arrows f and g, so we get a commuting diagram in C.

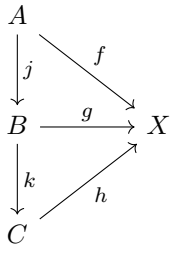
This thought prompts the following attempted definition:

Definition 26 (?)**.** Let C be a category, and X be a C-object. Then the category C/X, the *slice category over* X, has the following data:

(1) The C/X-objects are pairs (A, f) for any C-object A and C-arrow $f\colon A \to X$.
(2) The C/X-arrows from (A, f) to (B, g) are C-arrows $j\colon A \to B$ where $f = g \circ j$ in C.
(3) The identity arrow in C/X on the object (A, f) is the C-arrow 1_A.
(4) Given C/X-arrows j and k, where the target of j is the source of k, their composition is $k \circ j$ (where $k \circ j$ is the composite arrow in C). △

Of course, we need to check that these data do satisfy the axioms for constituting a category. Let's make a start on doing that. In particular, we need to confirm that our definition of composition for C/X-arrows makes sense.

[2] Fine print. Since on our definition of categories any arrow has a unique source, we could without loss of information take the object-data of C/X to be not *pairs* of objects and arrows from C such as $(A, f\colon A \to X)$ but simply the C-arrows $f\colon A \to X$ *by themselves*. Many officially opt for this more economical definition for C/X-objects. Obviously nothing hangs on this. And I'll talk in the economical idiom too when it makes for neatness.



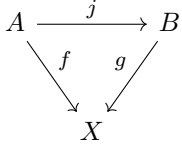
If $j\colon f \to g$ is a C/X-arrow, then $g \circ j = f$ and the top triangle commutes in C. If $k\colon g \to h$ is a C/X-arrow, then $h \circ k = g$ and the bottom triangle commutes in C. So pasting the triangles together, the whole resulting diagram commutes in C. Or in equations, we have $(h \circ k) \circ j = g \circ j = f$ in C, and therefore $h \circ (k \circ j) = f$. Hence $k \circ j$ really will count as an arrow in C/X from f to h on our definition, as we require.

So far, so good!

(c) Unfortunately, however, there is a very annoying snag. Our definition of a slice category C/X in fact doesn't quite work as it stands. Why not?

Suppose $f\colon A \to X$ and $f'\colon A \to X$ are distinct arrows in C. Then (A, f) and (A, f') are distinct objects in C/X. But according to (3) these objects will both have 1_A as their identity arrows. However, by Theorem 9, we can't have distinct C/X-objects with the same identity arrow on them.

What to do? We could fiddle with the definition of a category so Theorem 9 no longer holds, but that seems excessive. It is simpler to revise the definition of a C/X-arrow. Instead of saying that an C/X-arrow from (A, f) to (B, g) is a C/X-arrow which makes the diagram commute, say



(2′) The C/X-arrows from (A, f) to (B, g) are the complete commutative triangles in C formed by f, g together with an arrow $j\colon A \to B$ where $f = g \circ j$.

Equivalently, a C/X-arrow is a *triple* of arrows (j, f, g) such that $f = g \circ j$.[3] The definition of composition of arrows is then adjusted to match.

But it is irritating to have to fuss about this. From now on, we'll in fact cheat a tiny bit (as seems to be the standard way). When we talk of slice categories and the like, we'll continue to talk of an arrow from f to g as if it is simply a suitable j making the triangle commute. Since we can read off the triple when given an arrow specified as $j\colon f \to g$, no information at all is lost by just giving that single arrow. So that's what we'll do.

(d) Now for the dual notion, namely the idea of a *co-slice category* X/C (or the slice category *under* X). As we said, the objects of this category are C-objects paired with C-arrows going in the opposite direction, i.e. they are of the form $(A, f\colon X \to A)$.[4] Then the rest of the definition is exactly as you would predict given our explanation of duality: just go through the definition a slice category reversing arrows and the order of composition. It is a useful exercise to check that this works!

[3] "The arrows of C/X from f to g are arrows j of C such that $f = g \circ j$, i.e. they are the same thing as commutative triangles from f to g." So say Lawvere and Rosebrugh (2003, p. 25), with lettering changed. But an arrow j and a trio of arrows including j and forming a commuting triangle plainly *can't* be the same thing.

[4] Or, if you prefer, you could take the X/C-objects just to be arrows, in this case arrows of the form $f\colon X \to A$.

6.4 Just for the record: arrow categories

(a) We have seen that an arrow in the slice category C/X is, strictly speaking, a complete commutative *triangle* in C. Here's another type of derived category where the arrows are commutative *squares* in C.

Definition 27. If C is a category, then the corresponding *arrow category* $\mathsf{C}^{\to}$ has the following data:

(1) The $\mathsf{C}^{\to}$-objects are all the C-arrows.
(2) The $\mathsf{C}^{\to}$-arrows from $f\colon X \to Y$ to $g\colon W \to Z$ are the commutative squares in C formed by arrows $j\colon X \to W$ and $k\colon Y \to Z$ such that $k \circ f = g \circ j$.

Composition is defined by amalgamating commuting squares in C to get another commuting square (how?), and the identity arrow on a $\mathsf{C}^{\to}$-object is also defined in the obvious way (how?). △

I've mentioned arrow categories because you may well come across references to them: but at this stage, they are frankly not of much interest to us – it is difficult to come up with naturally arising elementary examples. So let's move on.

7 Kinds of arrows

So we have defined the general notion of a category. And we have met a lot of initial examples, and then seen how to construct yet more categories from old ones in various ways. Note again that our net is now cast very widely, going well beyond the initial motivating idea of a family of structures equipped with enough structure-respecting maps between them.

We are eventually going to want to impose some order on this proliferating universe of categories. And just as we organize groups by looking at the maps between them that respect group structure, we will want to introduce the key notion of *functors*, maps between categories which respect categorial structure. But not yet: functors will be the fundamental organizing idea of Part II of these Notes. However, here in Part I, we are going to be continuing to look *inside* categories, before we proceed to look at relations *between* categories.

And in this chapter, 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.

7.1 Left-cancellable, right-cancellable arrows

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

Definition 28. An arrow f in the category C is *left-cancellable* iff, whenever g and h are such that $f \circ g = f \circ h$, then $g = h$. △

Note by the way that if the composites $f \circ g$ and $f \circ h$ are to exist and be equal, then g and h must be parallel arrows sharing the same source and target (why?). So we can also put it this way: $f\colon Y \to Z$ is left-cancellable if whenever a fork of the form $X \underset{h}{\overset{g}{\rightrightarrows}} Y \xrightarrow{f} Z$ commutes, then $g = h$.[1]

Why is this notion of being left-cancellable interesting? Well, first note that we have the following general result for categories where arrows are structure-respecting-functions (or set-proxies for functions):

[1] Annoyingly, in the diagrammatic version, f is 'cancelled' on the right. Bother! Recall that remark about notational inversion in §4.1(c)(v). And recall Defn. 19* on what it takes for a fork to commute.

Theorem 11. *In a category where the arrows are indeed functions, such as* Set, Pos *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 $g(x) = h(x)$. Hence in arrow-speak, if $f \circ g = f \circ h$ then $g = h$, and so f is left-cancellable. □

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

Theorem 12. *In* Set, Pos 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 in Set 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. □

Proof for Pos. The same argument works: just note that a singleton 1 can be equipped with a partial order to make a poset, and functions from 1 are trivially monotone, so also live in Pos. □

Proof for Grp. This takes a bit more work. Suppose that $f\colon C \to D$ is a group homomorphism between the groups $(C, *, c)$ and $(D, \star, d)$ but is not injective. So for some particular objects x, y we have $f(x) = f(y)$ but not $x = y$.

Now, note that for these objects,

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

Let K (for 'kernel') be the objects among C that f sends to D's group identity d – compare §2.6(b). Then $x^{-1} * y$ belongs to K. And c is another *distinct* object that f sends to d (for if $x^{-1} * y = c$, then $x = x * c = x * x^{-1} * y = y$ contrary to hypothesis). Hence K includes more than one object.

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

So, putting our last two theorems together, we have now proved that in Set, Pos 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.[2]

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

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

Definition 29. An arrow f in the category C is *right-cancellable* iff, whenever g and h are such that $g \circ f = h \circ f$, then $g = h$. △

Equivalently, $f\colon X \to Y$ is right-cancellable if whenever a fork of the form $X \xrightarrow{f} Y \underset{h}{\overset{g}{\rightrightarrows}} Z$ commutes, then $g = h$.

Left and right cancellability are evidently dual properties – i.e. f is right-cancellable in C if and only if it is left-cancellable in C^{op}. And we easily get a companion result to Theorem 11:

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.*

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 y from among D, $g(y) \neq h(y)$. But by the surjectivity of f, we know that $y = f(x)$ for some x in f's source domain, and therefore $g(f(x)) \neq h(f(x))$. 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. □

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 $g \neq h$, though $g \circ f$ and $h \circ f$ will agree everywhere on C; so f is not right-cancellable. Contraposing, if f is right-cancellable, it is surjective. □

We can also show e.g. that in Grp, the right-cancellable functions are surjective; but this is not so obvious.[3] And later in this chapter, §7.5, we'll meet an easy case where we have a right-cancellable arrow which *is* a function but which is *not* surjective.

7.2 Notation and terminology

(a) There is a notational convention that we use special styles of drawn arrows to represent cancellable arrows, and we will follow this convention occasionally:

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

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

[3] 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*.

That convention is easy enough to remember: 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.

(b) But now we need to introduce some distinctly less memorable but absolutely standard terminology that you certainly need to know, and which we'll immediately start using:

Definition 30. An arrow is a *monomorphism* (or is *monic*) iff it is left-cancellable. And an arrow is an *epimorphism* (or is *epic*) iff it is right-cancellable. △

How are you supposed to remember which way round the labels 'monomorphism' and 'epimorphism' go? Well, you *could* try recalling that 'mono' means one, and the 'monomorphisms' are (we've seen) rather often the injective, one-to-one functions. While 'epi' is Greek for 'on' or 'over', and the 'epimorphisms' are (we've seen) fairly often surjective, onto, functions. But to be honest, what actually works for me is going by the brute alphabetic proximity of *ML* and of *PR*: for a *M*onomorphism is *L*eft cancellable, while an e*P*imorphism is *R*ight cancellable.

(c) As the very gentlest of exercises, putting our crisper terminology to work, let's have an easy mini-theorem:

Theorem 15. *(1) Identity arrows are always monic. Dually, they are always epic too.*

(2) If f, g are monic, so is $f \circ g$ (assuming f and g compose). 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 immediate.

For (2), we need to show that if $(fg)j = (fg)k$, then $j = k$. So suppose the antecedent. By associativity, $f(gj) = f(gk)$. Whence, assuming f is monic, $gj = gk$. 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 §6.2, it follows that if f and g are epic, so is $(f \circ g)$.[4]

For (3) assume $f \circ g$ is monic. Suppose $gj = gk$. We need to show $j = k$. But $f(gj) = f(gk)$, hence $(fg)j = (fg)k$, hence since $f \circ g$ is monic we have $j = k$. The corresponding result for epics holds by duality. □

7.3 Inverses

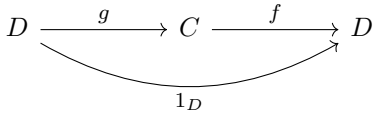
(a) We define some more types of arrow:

Definition 31. Given an arrow $f : C \to D$ in the category C,

[4] Check this, as it our first mini-application of the duality principle.

(1) $g\colon D \to C$ is a *right inverse* of f iff $f \circ g = 1_D$.
(2) $g\colon D \to C$ is a *left inverse* of f iff $g \circ f = 1_C$.
(3) $g\colon D \to C$ 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' again. Note that if we represent the situation in (1) with a commuting diagram like this



then f's right inverse g appears on the left. As with left/right cancellability, it is just a matter of convention that we standardly describe the handedness of inverses by reference to the representation '$f \circ g = 1_D$' rather than by reference to our representing diagram.

Second, note that $g \circ f = 1_C$ in C iff $f \circ^{op} g = 1_C$ in C^{op}. So a left inverse in C is a right inverse in 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 notion 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 where arrows are functions. Here's a *very* easy result:

Theorem 16. *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 simply 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 11 and 13, the following hold for concrete categories (with '⇒' for 'implies', of course!).

f has a left inverse ⇒ f is injective ⇒ f is left-cancellable (monic).
f has a right inverse ⇒ f is surjective ⇒ f is right-cancellable (epic).

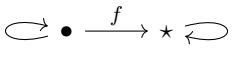
(c) What about categories where the arrows aren't functions (so the question of being injective or surjective doesn't arise)?

Well, the first item on each of those lines *still* implies the last: having a left (right) inverse implies being left (right) cancellable. Or to ring the changes on the terminology, since you really need to get used to this, we have the first part of the following theorem:

Theorem 17. *(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 of (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. □

Proof of (2). We can use a toy example. Take the two-object category 2:



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. □

Very slightly more interesting proof of (2). Take the category Grp, and consider Z and $2Z$, respectively the additive groups $(\mathbb{Z}, +, 0)$, and $(2\mathbb{Z}, +, 0)$, where of course $2\mathbb{Z}$ is the set of the even integers.[5] There is an obvious injection homomorphism $i\colon 2Z \to Z$, and i is monic in Grp (why?).

But i is not a right inverse. That is to say, there is no $f\colon Z \to 2Z$ such that $f \circ i = 1_{2Z}$. For suppose otherwise. Then

$$f(1) +_{2Z} f(1) = f(1 +_Z 1) = f(2) = f \circ i(2) = 1_{2Z}(2) = 2$$

But that's impossible since f maps only to even numbers! □

(d) 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 18. *(1) In* Set*, every monomorphism is a right inverse apart from arrows of the form $\emptyset \to D$.*
(2) Also in Set*, the proposition that every epimorphism is a left inverse is (a version of) the Axiom of Choice.*

Proof of (1). 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 chosen object in C. Such a g is always possible to find in Set unless C is the empty set.

So by construction, $g \circ f = 1_C$, and f is a right inverse. □

Proof of (2). 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 $f(c) = d$. Note that, in the general case, we *do* have to make an infinite number of choices, picking

[5] Well, not really – it's the set of set-implementations of the evens! But are we going to keep fussing about this sort of thing when it is not really relevant to the local point?

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 in Set is equivalent to the Axiom of Choice. □

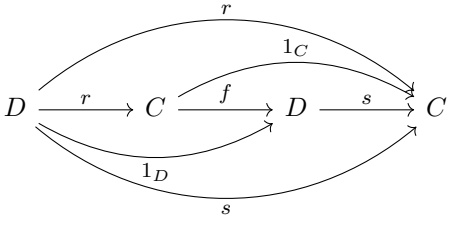
(e) An arrow can have zero or one left inverse. It can also have more than one. For a mini-example in Set, consider $f\colon \{0,1\} \to \{0,1,2\}$ where $f(0) = 0$, $f(1) = 1$. Then $g\colon \{0,1,2\} \to \{0,1\}$ is a left inverse so long as $g(0) = 0$, $g(1) = 1$; we have two choices for $g(2)$, and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

Theorem 19. *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

$$r = 1_C r = (sf)r = s(fr) = s1_D = s.$$

Or, to put it diagrammatically, the following commutes:



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. □

(f) Our little example at the beginning of (e) shows us that we can, of course, have arrows $f\colon A \to B$ and $g\colon B \to A$ where $g \circ f = 1_A$ but $f \circ g \neq 1_B$.

Is there anything interesting that *can* always be said about $f \circ g$ when $g \circ f = 1$? Well, note we will then have

$$(f \circ g) \circ (f \circ g) = f \circ (g \circ f) \circ g = f \circ 1 \circ g = f \circ g$$

Suppose we say that an arrow e is *idempotent* when $e \circ e = e$ (for that to make sense, an idempotent arrow must have the same source and target). Then, when $g \circ f = 1$, the corresponding composite $f \circ g$ is an idempotent arrow.

7.4 Some more – less memorable? – terminology

There is a rather annoying oversupply of alternative and unfriendly jargon hereabouts. Unlike the quite essential 'monomorphism' and 'epimorphism', I won't be making much further use of these more opaque bits of terminology in these Notes. But you'll certainly come across them elsewhere, so I need to explain them.

Definition 32. Assume we have a pair of arrows $s\colon C \to D$, and $r\colon D \to C$ such that $r \circ s = 1_C$. Then r, which is a left inverse of s, is said to be a *retraction* of s. And s is a right inverse of r; but s is also called a *section* of r. △

In this usage, then, s *is* a section iff it *has* a retraction, etc.

For a hint of the origin of this jargon, consider the following geometric example. Take P to be the plane minus a point as origin, and let S be the unit circle round the origin. Imagine P parameterized by polar co-ordinates r, θ centred at the origin, and S parameterized by θ (in each case, $0 \leqslant \theta < 2\pi$). Then consider the map $r\colon P \to S$ which sends a point (r, θ) on P to θ on S– this 'retracts' the whole plane onto the unit circle. While the map $s\colon S \to P$ which sends the point θ on the circle to $(1, \theta)$ in the plane locates, as it were, a 'section' of the plane. And trivially, $r \circ s$ is an identity map. But you can now forget all that!

Definition 33. If f has a left inverse/is a right inverse, then f is also 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 17 tells us that right inverses are monic, so a split monomorphism is indeed properly called a monomorphism. Dually, a split epimorphism is an epimorphism. But why 'split'? I haven't anything short and helpful to offer!

7.5 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 notion of isomorphism in arrow-theoretic, categorial, terms?

(a) First, 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 (if A and B have all the same relevant structure, then a map preserving all that structure should reverse). 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 17, is both monic and epic, but lacks an inverse. Or here's a generalized version of the same idea:

(1) Take the category P corresponding to some pre-ordered objects $(\mathrm{P}, \preccurlyeq)$, as in §4.4 (C4). Then there is at most one arrow f between any given objects of P. But for any f, 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 case where the arrows *are* functions but where being monic and epic *still* doesn't imply having an inverse:

(2) Consider this artificial little example in Pos, the category of posets and monotone functions.

Suppose the posets A and B each involve just two objects x, y; and let $\leqslant_A$ be the empty relation, while $x \leqslant_B y$. Let $f\colon A \to B$ be the identity map, which is trivially monotone. f is monic and epic, but plainly doesn't have a monotone inverse.

And for a much more interesting case:

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

It is worth pausing to prove that last claim:

Proof. To prove i is monic, assume $i \circ g = i \circ h$. We need to show $g = h$. If those assumed composites are to exist and be equal, g and h must be parallel arrows from some monoid M to N. Suppose $g \neq h$. Then there is some 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 $i \circ g \neq i \circ h$. Contradiction. So $g = h$ as required.

Second, to prove i is epic, again take a monoid M and this time consider any two monoid homomorphisms $g, h\colon Z \to M$ 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 in this example is an epic arrow which is a function but isn't surjective.

(b) The moral of our various examples? If we want our isomorphisms in general to be invertible, as we surely do, then it looks as though we'll have to build in that feature by definition!

So, at last, here's the official story:

Definition 34. An *isomorphism* in category 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 20. *(1) Identity arrows are isomorphisms.*

(2) An isomorphism $f : C \xrightarrow{\sim} D$ has a unique inverse which we can call $f^{-1} : D \xrightarrow{\sim} C$, 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 quickly 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) But as we noted, in a pre-order category, i.e. a category P corresponding to some pre-ordered objects $(P, \preccurlyeq)$, the only isomorphisms are the identity arrows.

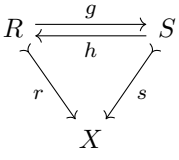
(c) Isomorphisms are monic and epic by Theorem 17. But as we have noted, arrows which are monic and epic need not be have inverses so need not be isomorphisms, e.g. in Pos and Mon. However, we do have this result:

Theorem 21. *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 monic, $g \circ f = 1$. 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. □

Here's another easy result in the vicinity:

Theorem 22. *Suppose the following diagram commutes:*



In other words, suppose r and s are both monic arrows with the same target, and there are g, h such that $r = s \circ g$ and $s = r \circ h$. Then g and h are isomorphisms and inverse to each other.

Proof. We have $r \circ 1_X = r = s \circ g = r \circ h \circ g$. Since r is monic, $h \circ g = 1_R$. Similarly, $g \circ h = 1_S$. So g and h are each other's two-sided inverse, and both are isomorphisms. □

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

Definition 35. A category 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 Pos and Mon are not. Top is another example of an unbalanced category.

7.6 Isomorphic objects

(a) We can now introduce another key notion:

Definition 36. If there is an isomorphism $f : C \xrightarrow{\sim} D$ in C then the objects C, D are said to be *isomorphic* in C, and we write $C \cong D$. △

From the ingredients of Theorem 20, we immediately get

Theorem 23. *Isomorphism between objects in a category C is an equivalence relation.* □

Then, roughly speaking, just as group theory typically doesn't care about the distinction between isomorphic groups, category theory typically doesn't care about the distinction between isomorphic objects. For example, we'll see that categorially we only care about pinning down the product of objects 'up to isomorphism'.

(b) But is it right that we needn't care about distinguishing isomorphic objects? Here's an example I mentioned before: instances of a Klein four-group are

group-theoretically indiscernible by virtue of being isomorphic (indeed, between any two instances, there is a unique group isomorphism). And yes, we then do cheerfully talk about *the* Klein four-group. There is a real question, however, about 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 yes, as I said, category theory typically doesn't care about distinguishing isomorphic objects. However, in FinSet or again in Set any two singletons count as isomorphic. Yet it would strike us as rather odd – wouldn't it? – to say that we can happily talk about *the* singleton in the way we talk about *the* Klein group. To be sure, there are contexts where any singleton will do: recall how in §4.6 we associated elements x of a set X one-to-one with arrows $\vec{x}\colon 1 \to X$ (where any singleton 1 you care to choose will serve to make the point). But in other contexts, the pairwise distinctness of singletons would seem be important, e.g. when we treat $\{\emptyset\}, \{\{\emptyset\}\}, \{\{\{\emptyset\}\}\}, \{\{\{\{\emptyset\}\}\}\}, \dots$ as a sequence of *distinct* singletons in one possible construction (Zermelo's) for the natural numbers.

We can't delay to explore this issue any further just at the moment: I am just flagging up that there are questions we would eventually want to discuss around and about the idea of isomorphism-as-sameness. But these are best tackled *after* getting a lot more category theory under our belt!

(c) Back to technicalities, then. Let's have a little theorem: an isomorphism between objects in a category induces a bijection between the arrows to (or from) those objects:

Theorem 24. *If $C \cong D$ in C, then there is a one-one correspondence between arrows $X \to C$ and $X \to D$ for all objects X in 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 $j^{-1}\overline{f} = j^{-1}\overline{g}$ and hence $f = g$). It is also surjective (for any $g\colon X \to D$, put $f = j^{-1}g$ then $\overline{f} = g$). That gives us a one-one correspondence between arrows $X \to C$ and $X \to D$.

The dual claim is proved similarly. □

7.7 Epi-mono factorization

That's the main business of this chapter done. But I'll finish with a couple of additional sections noting some further points worth knowing.

(a) Start with a definition:

Definition 37. An arrow $f\colon B \to D$ has an *epi-mono factorization* iff there is an epic arrow $e\colon B \twoheadrightarrow C$ and a monic arrow $m\colon C \rightarrowtail D$ such that $f = m \circ e$. △

Then we have an easy result:

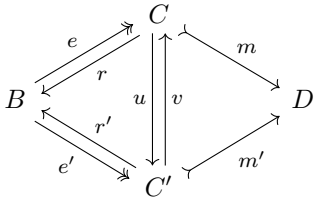
Theorem 25. *(1) In* Set *every arrow has an epi-mono factorization.*
(2) Still in Set*, if* $f\colon B \to D$ *factors both as* $B \xrightarrow{e} C \xrightarrow{m} D$ *and as* $B \xrightarrow{e'} C' \xrightarrow{m'} D$ *then* $C \cong C'$.

Proof of (1). In Set an arrow $f\colon B \to D$ is a function, and let $f(B)$ be the f-image of B. Then let the function $e\colon B \to f(B)$ agree everywhere with f, and let $m\colon f(B) \to D$ be the inclusion function which sends an element of $f(B)$ to itself as an element of D. Trivially e is surjective (so epic), m is injective (so monic), and $f = m \circ e$; so we are done. □

Another proof of (1). Again take $f\colon B \to D$ in Set, and consider the equivalence relation E_f on B where xE_fy if and only if $fx = fy$. Take a quotient of B by E_f (as in Defn. 5). Then let $e'\colon B \to B/E_f$ send every element x of B to $[x]$. And let $m'\colon B/E_f \to D$ send $[x]$ to fx. e' is epic and m' monic (why?). And $f = m' \circ e'$. □

Note that there is a bijection between $f(B)$ and B/E_f, the intermediary objects of these two epi-mono factorizations. And (2) generalizes the point:

Proof of (2). We'll help ourselves to Choice so we can assume that both the epic arrows e and e' are left inverses, i.e. there are arrows r and r' such that $e \circ r = 1_C$ and $e' \circ r' = 1_{C'}$. Now put $u = e' \circ r$ and $v = e \circ r'$. Then I claim the following diagram commutes:



The composites along the two outer paths from B to D are equal by assumption. The two left triangles commute by definition of u and v. And $mv = mer' = m'e'r' = m'$ and likewise $m'u = m$, making the right triangles commute.

And now we note $mvu = mer'e'r = m'e'r'e'r = m'e'r = mer = m = m1_C$; hence, since m is monic and left-cancellable, $vu = 1_C$. Similarly, $uv = 1_{C'}$. So u is our desired isomorphism from C to C'. □

(b) We should note, however, that epi-mono factorization is not always available in a given category. For a toy example, consider the mini-category which has a single object o, its identity arrow 1_o, and a further non-identity arrow $f\colon o \to o$ such that $ff = f$. Then evidently f is neither epic nor monic (or else we could cancel from that equation to get $f = 1_o$). And so f doesn't factor into epic and monic.

Again, when more than one epi-mono factorization is available, they needn't go via isomorphic objects. Recall from §7.5 the situation in Mon where $i\colon N \to Z$

– i.e. $i\colon (\mathbb{N}, +, 0) \to (\mathbb{Z}, +, 0)$ – is both epic and monic. Then trivially we can have two epi-mono factorizations for i, as it equals both

$$N \xrightarrow{1_N} N \xrightarrow{i} Z \quad \text{and} \quad N \xrightarrow{i} Z \xrightarrow{1_Z} Z$$

and of course we don't have $N \cong Z$.

So the kind of epi-mono factorization which we get in Set (universal for all arrows, and unique-up-to-isomorphism) is only found in a sub-family of categories.

7.8 Groups as categories

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

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

Now, since every element in the group has an inverse, it follows immediately that every arrow in the corresponding category 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, i.e., is an isomorphism.[6]

[6] 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 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 Steve Awodey, offering some similarly arm-waving remarks about what he calls "category-theoretical definitions":

> These are characterizations of properties of objects and arrows in a category solely in terms of other objects and arrows, that is, just in the language of category theory. Such definitions may be said to be abstract, structural, operational, relational, or perhaps 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 2010, p. 29)

We proceed 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.

8.1 Initial and terminal defined

(a) As we noted in §4.6, in Set,

(i) For any set X, there is one and only one set-function from the empty set $\emptyset$ 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, then: 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.[1]

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

Definition 38. The object I is an *initial* object of the category C iff, for every C-object X, there is a unique arrow $!\colon I \to X$.

Dually, the object T is a *terminal* object of C iff, for every C-object X, there is a unique arrow $!\colon X \to T$.[2] △

Evidently, an object is initial in C if and only if it is terminal in C^{op}. The use of '!' to signal the unique arrows from an initial object (or to a terminal object) is quite common. If we want explicitly to indicate the target (or source) of such a unique arrow, we can write e.g. '$!_X$'.

Then, in summary, we've noted that

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

Let's immediately have some more examples;

(2) In Pos too, the empty poset is initial, while any singleton equipped with the partial order that relates the singleton's member to itself is terminal.

(3) In the pre-ordered natural numbers $(\mathbb{N}, \leqslant)$ thought of as a category, zero is the unique initial object and there is no terminal object. By contrast the pre-ordered integers $(\mathbb{Z}, \leqslant)$ form a category which lacks both initial and terminal objects.

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

(4) $\mathsf{Set}_\star$, 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.

[1] Fine print. The official story is that we are working in a suitably capacious, though not-yet-fully-specified, universe of sets. So we haven't determinately pinned down Set. But familiar variants on the usual stories about sets do of course agree that there is an empty set and that there are singletons. I suppose I should, however, note for the record that there can be competent set theories *without* an empty set (Cantor's own, perhaps!) and/or *without* singletons distinguished from their members. For provocation on this topic, you might be diverted by Oliver and Smiley (2006).

[2] Some call terminal objects *final*; and then that frees up 'terminal' to mean *initial or final*. So when reading other treatments, you do need to check how 'terminal' is being used.

(5) In Rel, the category of sets and relations, the empty set is both the sole initial and sole 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) The one-element ring is terminal in Rng too. But the initial object is more interesting – it's the ring of integers.

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

(9) In the category Prop_L of propositions in the first-order language L, $\bot$ is initial and $\top$ is terminal.

(10) In the category Bool, the 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.

(11) In the category Graph the empty graph is initial; the graph with one node and one edge looping from that node to itself is terminal (why?).

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

Consider the C/X object which is the C-arrow 1_X. A C/X arrow from $f\colon A \to X$ to $1_X\colon X \to X$ is a 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 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 39. An object O in the category C is a *null object* of the category C iff it is both initial and terminal. △

8.2 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, if a category C has any initial objects, they may be one or many. However, we have the following key pair of theorems:

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

Proof. Suppose I and J are both initial objects in C. By definition there must be unique C-arrows $f\colon I \to J$, and $g\colon J \to I$. Then $g \circ f$ is an arrow from I to itself. But we know that one arrow from I to itself is the identity arrow 1_I. And since I is initial, there can only be one arrow from I to itself. Therefore $g \circ f = 1_I$. Exactly similarly, we can show $f \circ g = 1_J$.

Hence the unique arrow f has a two-sided inverse and is an isomorphism. □

Theorem 27. *If I is initial in C and $I \cong J$, then J is also initial. Dually for terminal objects.*

Proof. Suppose (i) I is initial and (ii) $I \cong J$. By (i), for any X, there is a unique arrow $f\colon I \to X$. By (i) and (ii) the unique arrow $i\colon I \to J$ is an isomorphism.

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

The dual of this line of argument delivers, of course, the dual result. □

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

Definition 40. We use '0' to denote an initial object of C (assuming one exists), and likewise '1' to denote a terminal object.[3] △

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

Theorem 28. *In a category with a terminal object, any arrow $f\colon 1 \to X$ is monic.*

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

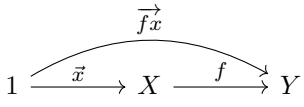
8.3 Elements

(a) A category can have a terminal object (so every object has a unique arrow *to* it), without having any arrows *from* it (except to itself or to other terminal objects). For example, take a pre-ordered-set-as-a-category where the order has a maximum element.

By contrast, consider the category Set again. As we have remarked before, in this case 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 monic arrows $\vec{x}\colon 1 \to X$ as the categorial version of talking of elements of X.

[3] Null objects which are both initial and terminal are often alternatively called 'zero' objects. But that perhaps doesn't sit happily with the pretty standard practice 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.

Now imagine using $\vec{x}\colon 1 \to X$ to pick out an element x from X, and then applying the function $f\colon X \to Y$ to this element, to land at the element fx in Y. This element corresponds, of course, to the arrow $\overrightarrow{fx}\colon 1 \to Y$, and we get this commuting diagram:



So: $\overrightarrow{fx} = f \circ \vec{x}$.[4]

(b) I said that in Set we can treat talk of arrows $\vec{x}\colon 1 \to X$ as the categorial version of talking of elements of the set X. I suppose we had better check, though, that it can't matter for us *which* terminal object 1 we use here.

So suppose 1 and $1'$ are two terminal objects in Set. There is a unique isomorphism $j\colon 1' \xrightarrow{\sim} 1$. And if $\vec{x}\colon 1 \to X$ picks out a member x of X, then $\vec{x} \circ j\colon 1' \to X$ picks out the very same object. And of course conversely, if $\vec{x}'\colon 1' \to X$ picks out a member x' of X, $\vec{x}' \circ j^{-1}\colon 1 \to X$ does the same job. Therefore – at least as far as general claims about elements are concerned – it can't matter whether elements are defined as arrows from 1 or as arrows $1'$.

(c) We now generalize and carry the idea over to other categories:

Definition 41. In a category C with a terminal object 1, an *element* or *point* of the C-object X is a (monic) arrow $f\colon 1 \to X$.[5] △

We can immediately see, however, that in categories C other than Set, these so-called 'elements' $1 \to X$ often won't 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 items there are forming the group X.

Put it this way. An arrow $1 \to X$ shines a very narrow beam into X. Still, varying through all the possible arrows $1 \to X$ (for a given fixed X) in Set will turn the spotlight through all the different elements (in the ordinary sense) of X. By contrast, an arrow $1 \to X$ in Grp can only spotlight the identity element of X.

8.4 Separators and well-pointed categories

(a) Continuing from the last point, let's introduce a couple of useful bits of terminology. First,

[4] If we drop the over-arrow notation to mark elements-as-arrows, and drop the optional sign for composition, this would become, not very helpfully, $fx = fx$: so it really is best to stick to our more explicit notation here!

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

Definition 42. The object S is a *separator* in category C iff for every pair of parallel C-arrows $f, g\colon X \to Y$, where $f \neq g$, there is an arrow $s\colon S \to X$ such that $f \circ s \neq g \circ s$. △

In other words, S is a separator if given two parallel arrows f and g (so we can't separate them merely by looking at their sources and/or targets), we can still always tell them apart 'looking from S' – we can probe their shared source using some arrow s from S, and find that f and g combine differently with that probe.

Then there is the special case where a terminal object 1 is a separator:

Definition 43. Suppose the category C has a terminal object 1 which is a separator. Then C is said to be *well-pointed*. △

In other words, in a well-pointed category, whenever parallel arrows $f, g\colon X \to Y$ agree on all elements – i.e. $f \circ \vec{x} = g \circ \vec{x}$ for all $\vec{x}\colon 1 \to X$ – then $f = g$. That's trivially how things are in Set. But compare again the situation in Grp. 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.

Well-pointedness, note, is defined by generalizing over elements-as-arrows – so being well-pointed doesn't depend on which terminal object we pick to fix these elements.

And so to summarize, for the record:

Theorem 29. Set *is well-pointed. But* Grp*, for example, isn't.* □

(b) It is worth remarking that, although the terminal object is not a separator for Grp, the category does have one (in fact many, but one will do!):

Theorem 30. *Z, the additive group of integers, is a separator for* Grp.

Proof. Suppose in Grp, we have parallel group homomorphisms $f, g\colon X \to Y$ where $f \neq g$. Then choose some x such that $fx \neq gx$. Now consider the map $s\colon Z \to X$ that sends the integer j to x^j.[6] Then s is easily seen to be a group homomorphism, and by construction $f(s(1)) \neq g(s(1))$ so $f \circ s \neq g \circ s$. □

8.5 'Generalized elements'

(a) We have seen that, even when arrows in a category are functions, acting the same way on all point 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* always imply being the same arrow, whatever the category?

Well, suppose we say, as some do, that

[6] The notation should be obvious. For positive j, $x^j = x * x * x * \ldots * x$, for j multiplicands, with $*$ the group operation in X. For negative j, $x^j = x^{-1} * x^{-1} * x^{-1} * \ldots * x^{-1}$, for $-j$ multiplicands. And $x^0 = e$, the group identity of X.

Definition 44. A *generalized element* (of shape S) of the object X in C is an arrow $s\colon S \to X$. △

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

Theorem 31. *Parallel arrows in a category* 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 of X, they in particular act identically on the generalized element $1_X\colon X \to X$: so $f \circ 1_X = g \circ 1_X$, and $f = g$.

And of course if $f, g\colon X \to Y$ are identical, they act identically on any generalized element. □

But that's too trivial to be very unexciting. More interesting will be the cases where there is some special class of generalized elements such that acting identically on them is enough to ensure arrows are equal. We saw, for example, that acting identically on 'generalized elements of shape Z' is enough to ensure equal arrows in Grp.

(b) I do find, though, that there is something rather odd about calling *any* arrow $S \to X$ a sort of 'element' of X. For example, suppose we are in the category of topological spaces, and S^1 is a circle. Then a 'generalized element of shape S^1' in X is an arrow from $S^1 \to X$. In other words, it is a continuous map which yields a loop in X. But do we really want to think of such a loop as in any sense an 'element' of X? Doesn't this 'generalized element' correspond, rather, to a subspace of X?

I prefer, then, largely to avoid the 'generalized element' jargon. And it is noticeable that quite a few writers on category theory do the same.[7]

8.6 'And what about arrows to 0?'

A terminal object 1 in C is defined as having a unique arrow from any C-object *to* it. And arrows *from* 1, when they exist, give us a notion of element which – at least in some categories like Set – corresponds to the intuitive pre-categorial notion.

What about the dual case (we will repeatedly ask this sort of question in category theory)? An initial object 0 in C is defined as having a unique arrow any C-object *from* it to any C-object. And what about arrows *to* 0?

In some categories, there are no such arrows (other than the identity arrow). For a trivial example, take any pre-ordered-set-as-a-category where the order

[7] To take three relatively recent examples of introductory books, none of Simmons (2011), Roman (2017) and Fong and Spivak (2019) have occasion to use the terminology. I believe it was introduced by Lawvere, so unsurprisingly we find it used in e.g. Lawvere and Rosebrugh (2003, pp. 15–17); but the remarks there about why talk of 'generalized elements' is apt do seem philosophically pretty confused.

has a bottom element. More importantly, there are no arrows $X \to 0$ in Set other than from 0 itself. (What happens in Set^{op}? – you don't need to have a conception of the data of this category, but can just invoke duality.) In Grp where an initial object is a one-object group, arrows $X \to 0$ are the trivial collapse homomorphisms which send every group-member to the single object of 0.

And perhaps that will do for now. Though we'll return to look again at arrows to 0 in the special context of so-called Cartesian closed categories in §17.2.

9 Pairs and products, pre-categorially

The discussion in the last chapter illustrates an absolutely central categorial theme. We defined initial objects and terminal objects not 'internally' but 'externally' in terms of the arrows for which they are source or target, and then we showed that the objects defined this way are themselves 'unique up to unique isomorphism'. This is a pattern which will keep on recurring in rather more exciting contexts, starting in the next chapter when we give a categorial definition of products.

Now, we are very familiar in pre-categorial maths with constructing products for all kinds of widgets. The paradigm case, of course, is where we take sets S_1 and S_2 and form their Cartesian product $S_1 \times S_2$, the set of ordered pairs of their elements. But we now want to pause to ask rather more carefully than usual: what does it take for some candidates to count as the required pair-objects?

In this chapter, we'll tackle that question in pre-categorial terms. Our answer will then point forward in a pretty natural way to the standard categorial treatment of products given at the beginning of the next chapter.[1]

9.1 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 without taking on any new commitments, by the simple 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 $\langle m, n \rangle_B =_{\text{def}} \{(m+n)^2 + m + 3n\}/2$. Or, if we don't insist on every number coding a pair, we could – as we noted in §2.3 – instead adopt the

[1] I do think it is illuminating to take things slowly and to work up to the categorial story this way. Some authors introduce products much more briskly, later in the game and in a more sophisticated setting, after developing category theory much further than we have so far done: see for example Leinster (2014, p. 107), Riehl (2017, p. 77) and – with even less by way of intuitive motivation – Roman (2017, p. 98).

That approach, which is technically fine of course, does however miss the opportunity to make the categorial treatment of products seem as uncontrived as it really is.

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 even 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 quite 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 $3^m 5^n$ as a code for the pair m, n).

And that is all true, of course. But 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 $\{\{\{m\}, \emptyset\}, \{\{n\}\}\}$ etc., etc., would have done equally as well as alternative implementations.[2]

On these counts, at any rate, 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 ironic 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 particular 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

[2]The second of these is based on the original set-theoretic definition of an ordered pair, due to Norbert Weiner in 1914.

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 a general-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 a hammer to crack open 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. Likewise, 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 2009) – then pair-numbers are *exactly* the kind of thing we need.

9.2 Pairing schemes more generally

(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 to have such a workable scheme for pairing numbers with numbers? Or more generally, to have a scheme for pairing objects X with objects Y?

We've been here before of course, in §2.3. In essence, we need some *pair-objects* O to code up pairs; we need a binary *pairing function* that sends a given $x \varepsilon$ X and a given $y \varepsilon$ Y to a particular pair-coding object $o \varepsilon$ O; and (of course!) we need a couple of *projection functions* which allow us to recover x and y from o. And the point illustrated by the case of rival pairing schemes for numbers is that we shouldn't care too much about the 'internal' nature of the pair-objects, so long as we can associate them 'externally' with suitable pairing and unpairing functions which fit together in the required way.

(b) Our earlier Defn. 3 which characterized pairing schemes was somewhat informally phrased. Let's now tidy things up – and for local typographical neatness, I'll now use '*pr*' generically for a pairing function (rather than '$\langle\ ,\ \rangle$').

Assume, as before, that X are some objects, as are Y, and as are O (these may or may not be all distinct), and assume that $x \varepsilon$ X, $y \varepsilon$ Y and $o \varepsilon$ O. Then:

Definition 3*. Let $pr\colon \mathrm{X}, \mathrm{Y} \to \mathrm{O}$ be a two-place function, while $\pi_1\colon \mathrm{O} \to \mathrm{X}$, and $\pi_2\colon \mathrm{O} \to \mathrm{Y}$, are one-place functions. Then $(\mathrm{O}, pr, \pi_1, \pi_2)$ *form a scheme for pairing* X *with* Y iff for all x, y and o, the following conditions hold:

(I) $\pi_1(pr(x, y)) = x$ and $\pi_2(pr(x, y)) = y$;
(II) $pr(\pi_1 o, \pi_2 o) = o$. △

Of course, (I) just records the desideratum that if we pair up objects using pr, and then unpair using the projection functions π_1 and π_2, we get back to where

we started. While (II) records the complementary desideratum that if we take a pair-object o, extract the two paired objects by using π_1 and π_2, and then pair up the results again using pr, we also get back where we started.

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 $\pi_1 o$ and $\pi_2 o$ returning respectively the exponent of 2 and 3 in the factorization of o, then $(\mathrm{O}, pr, \pi_1, \pi_2)$ officially form a scheme for pairing naturals with naturals. And similarly for pairing based on Kuratowski pairs.

(c) Two points on notation. First, I could have used 'X × Y' rather than the blankly unhelpful 'O' to denote the pair-objects in our scheme for pairing X with Y. But avoiding that notation for a while will keep us honest and help loosen the hold of the idea that products must 'internally' be anything like Cartesian products.

Second, there is no need to over-interpret the brackets in '$(\mathrm{O}, pr, \pi_1, \pi_2)$': they are no more than punctuation, so you can read this as 'O together with the functions pr, π_1 and π_2'

(d) We need to check some (very!) elementary facts about pairing schemes as defined. First,

Theorem 32. *If $(\mathrm{O}, pr, \pi_1, \pi_2)$ form a scheme for pairing* X *and* Y*, then (1) different pairs of objects are sent by pr to different pair-objects, i.e. $pr(x, y) = pr(x', y')$ iff $x = x'$ and $y = y'$. Also (2) pr, π_1 and π_2 are all surjective.*

Proof. For (1) suppose $pr(x, y) = pr(x', y')$. Then by condition (I) on pairing schemes, $x = \pi_1(pr(x, y)) = \pi_1(pr(x', y')) = x'$, and likewise $y = y'$.

We want (2) to be true so that O includes no more than we need, a condition which we built into our original Defn. 3. But it is immediate that pr is surjective by condition (II): any o is the value of pr for the corresponding inputs $\pi_1 o, \pi_2 o$.[3]

We also want every x among X to be the first projection of a pair object o, etc. And it is again immediate that the projection function π_1 is surjective because, given x, we can take any y and put $o = pr(x, y)$, and then by (I), x is the value of π_1 for input o. Similarly for π_2. □

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

Theorem 33. *(1) If $(\mathrm{O}, pr, \pi_1, \pi_2)$ and $(\mathrm{O}, pr, \pi_1', \pi_2')$ are both schemes for pairing* X *with* Y*, then $\pi_1 = \pi_1'$ and $\pi_2 = \pi_2'$.*

(2) If $(\mathrm{O}, pr, \pi_1, \pi_2)$ and $(\mathrm{O}, pr', \pi_1, \pi_2)$ are both schemes for pairing X *with* Y*, then $pr = pr'$.*

[3] Careful! We only need the binary function pr to be surjective *on the pair-objects*. The function $pr(m, n) = \langle m, n\rangle_P = 2^m3^n$ in the pairing scheme for numbers which we considered a moment ago is of course not surjective *over all numbers*. Likewise a Kuratowski-style pairing function is not surjective over all sets.

Proof. For (1), take any o, and suppose $o = pr(x, y)$ (there must be some such x and y since pr is surjective). Hence, applying (I) 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 and y and let $pr(x, y) = o$. Then $\pi_1 o = x$ and $\pi_2 o = y$. Applying (II) to the second scheme, we have $pr'(x, y) = pr'(\pi_1 o, \pi_2 o) = o$. Whence $pr(x, y) = pr'(x, y)$. □

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

Theorem 34. *If* $(\mathrm{O}, pr, \pi_1, \pi_2)$ *and* $(\mathrm{O}', pr', \pi'_1, \pi'_2)$ *are both schemes for pairing* X *with* Y, *then there is a unique bijection* $f\colon \mathrm{O} \to \mathrm{O}'$ *which respects pairing, 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 by putting $f(o) = pr'(\pi_1 o, \pi_2 o)$. Then it is immediate that $f(pr(x, y)) = pr'(\pi_1(pr(x, y)), \pi_2(pr(x, y))) = pr'(x, y)$.

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 condition (I) in Defn. 3*, 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 (II), $o = o^*$.

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 (I) and (II) confirms that $f(o) = o'$.

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

(f) We've confirmed that pairing schemes as (re)defined in Defn. 3* do work exactly as we should want. Theorem 32 tells us that in a pairing scheme, different pairs get coded by different pair-objects, and also there are no redundancies, i.e. there are no more pair-objects in the scheme than we need. Theorem 33 tells us that, as we'd expect, pairing and unpairing functions fit together tightly – fix the first and that determines the second, and vice versa. Theorem 34 tells us that variant pairing schemes for pairing up X with Y will in a good sense all 'look the same'.

So far, then, so obvious. Should I have left those theorems as really elementary exercises? Very possibly. But these things are rarely properly spelt out. Anyway, let's continue:

Theorem 35. *Given various pluralities of objects* $\mathrm{O}, \mathrm{X}, \mathrm{Y}$ *as before, and functions* $\pi_1\colon \mathrm{O} \to \mathrm{X}$, $\pi_2\colon \mathrm{O} \to \mathrm{Y}$, *suppose that there is a* unique *binary function* $pr\colon \mathrm{X}, \mathrm{Y} \to \mathrm{O}$ *such that*

(I) $\pi_1(pr(x, y)) = x$ *and* $\pi_2(pr(x, y)) = y$.

Then $(\mathrm{O}, pr, \pi_1, \pi_2)$ *form a scheme for pairing* X *with* Y.

Proof. We show (1) that uniqueness for pr implies it is surjective – or equivalently, that being non-surjective implies being non-unique. And then (2) the surjectivity of pr implies condition (II) for a pairing scheme, given (I).

(1) Suppose pr satisfies (I) but is *not* surjective. Then there will be at least one escapee o^*, such that there is no x and y such that $pr(x, y) = o^*$. In particular, $pr(\pi_1 o^*, \pi_2 o^*) \neq o^*$.

Consider then the function pr^* which agrees with pr on all inputs except that we stipulate $pr^*(\pi_1 o^*, \pi_2 o^*) = o^*$.

For all values of x and y other than $x = \pi_1 o^*, y = \pi_2 o^*$, (I) still holds. And by stipulation, for the remaining case $\pi_1(pr^*(\pi_1 o^*, \pi_2 o*)) = \pi_1 o^*$ and $\pi_2(pr^*(\pi_1 o^*, \pi_2 o^*)) = \pi_2 o^*$.

Hence condition (I) always holds for pr^*, although $pr^* \neq pr$, thereby showing pr isn't unique.

(2) Since pr is surjective, for every o, there is some x and y such that $o = pr(x, y)$, and hence by (I) $\pi_1 o = x$ and $\pi_2 o = y$. But then $pr(\pi_1 o, \pi_2 o) = pr(x, y) = o$. Which gives us condition (II) for being a pairing scheme. □

9.3 Defining products, pre-categorically

OK. With those general facts about *pairing schemes* to hand, let's turn to thinking about what we want to say about *products*.

This chapter started by recalling the set-theoretic idea of a Cartesian product as a set of pair-objects – and such a product is typically implemented by selecting Kuratowski pairs to function as pair-objects, which come with their obvious projection functions for unpairing. But NB, while the choice of projection functions might be obvious, a choice is most certainly needed. After all, we need to know, given the unordered set $\{\{m\}, \{m, n\}\}$, whether m is the first or second component of the represented pair.

We have now introduced the more general idea of a pairing scheme. And by this stage we can see that we really don't want to say that the relevant collection of pair objects O *just by itself* forms a product of the relevant X with Y – because it again depends crucially on the rest of the pairing scheme whether the candidate pair-objects can play the right role, and in particular on how they are equipped with projection functions. Our last theorem, however, makes the following an appropriate definition (with notation still as before):

Definition 45. $(\mathrm{O}, \pi_1, \pi_2)$ *forms a product of* X *with* Y, where O are some objects, and $\pi_1 \colon \mathrm{O} \to \mathrm{X}, \pi_2 \colon \mathrm{O} \to \mathrm{Y}$ are functions, so long as

(C) There is a *unique* two-place function $pr \colon \mathrm{X}, \mathrm{Y} \to \mathrm{O}$ satisfying condition (I) $\pi_1(pr(x, y)) = x$ and $\pi_2(pr(x, y)) = y$. △

And look! – from our very elementary pre-categorial results out has popped a rather categorially flavoured definition, which characterizes a product not in terms of the 'internal' make up of the pair-objects O, whatever they are, but 'externally' in terms of there being a unique map doing a certain job. Terrific! We can work with this idea ...

But note, our definition doesn't yet quite have the shape of a kosher arrow-theoretic definition. Arrows in categories have single objects as sources – so when arrows are functions, they are *monadic* functions.[4] However, the function pr in Defn 45 is essentially *binary*. What to do? Well, we can rephrase condition (C) to avoid overt reference to pr like this:

Definition 45*. $(\mathrm{O}, \pi_1, \pi_2)$ *form a product of* X *with* Y, where O are some objects, and $\pi_1 \colon \mathrm{O} \to \mathrm{X}, \pi_2 \colon \mathrm{O} \to \mathrm{Y}$ are functions, so long as

(C′) For any $x \varepsilon \mathrm{X}$ and $y \varepsilon \mathrm{Y}$, there is a *unique* corresponding $o \varepsilon \mathrm{O}$ such that
(I′) $\pi_1 o = x$ and $\pi_2 o = y$. △

Why is (C) equivalent to (C′)? Because if there is an x and y for which there are after all multiple choices for the value of o satisfying (I′), then there will be different candidates for pr which satisfy (I) but whose values peel apart at that x and y. While conversely, if there are different candidates for pr satisfying (I) whose values peel apart at some particular x and y, then at those x and y there will be different values of $o = pr(x, y)$ satisfying (I′).

[4] Does it have to be this way? Definitely so, on the standard conception of a category. There is a contrasting notion of multicategory where the source of an arrow/morphism can be a list of objects. But this and related notions are *far* beyond our scope here.

10 Categorial products and coproducts

And now, as we will see, our Defn. 45* from the end of the previous chapter gives us a really rather natural lead into a categorial treatment of products.

10.1 Products defined categorially

(a) Let's take things in stages.

First, suppose we are initially working in Set. So, instead of talking in the plural about some objects X and objects Y objects plural we can now talk in the singular about the set X and set Y of those objects.

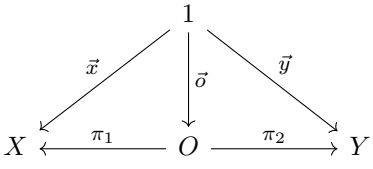
As we saw in §8.3, in the case of Set, the 'elements' in the sense of arrows from a terminal object to X behave as elements intuitively should behave. Then

1. In this setting, instead of talking of an object x (from X) and object y (from Y), we can talk instead of two corresponding arrows $\vec{x}\colon 1 \to X$ and $\vec{y}\colon 1 \to Y$. Again, instead of talking of some object o (which belongs to a set O), we can talk of an arrow $\vec{o}\colon 1 \to O$.

2. Hence, instead of saying as in condition (C′) in Defn. 45* that $\pi_1 o = x$ and $\pi_2 o = y$, we could equivalently say $\pi_1 \circ \vec{o} = \vec{x}$ and $\pi_2 \circ \vec{o} = \vec{y}$.

3. And *that* is of course equivalent to saying that the following diagram commutes:

(P)



Hence we can transmute our Defn. 45* into a first-shot categorial definition applying to Set as follows:

> O equipped with the projection arrows $\pi_1\colon O \to X$, $\pi_2\colon O \to Y$ forms a product for X and Y in Set iff 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 (P) commute.

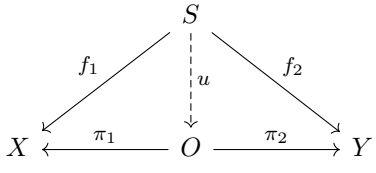
(b) So far, so good. But a moment's reflection tells us this certainly won't give us what we want across *all* categories. For example, in Grp, our diagram (P) will always commute if O is a one-object group and π_1 and π_2 are the only possible arrows from it to X and Y (why?). But, trivial cases apart, such an O won't in any sense constitute a product of the groups X and Y.

So: in categories other than Set, concentrating only on what happens with *point elements* $1 \to X$ and $1 \to Y$ usually isn't enough to give us a sensible notion for products of X with Y. What to do?

Following on from the discussion of §8.5, the obvious thing to try is moving from considering just point elements to thinking about interactions with *'generalized elements'* too. In other words, instead of thinking only about what happens when we probe X and Y with narrow-beam spotlights with source 1, we should also consider using wider-beam probes from other sources, i.e. using arrows $S \to X$ and $S \to Y$ more generally.

Which motivates the following official cross-category definition:[1]

Definition 46. In any category C, *a (binary) product* (O, π_1, π_2) *for* X *with* Y is an object O together with projection arrows $\pi_1 : O \to X, \pi_2 : O \to Y$, such that for any object S and arrows $f_1 : S \to X$ and $f_2 : S \to Y$ in C, there is always a *unique* 'mediating' arrow $u : S \to O$ such that the following diagram commutes:



△

Here, by the way, we now adopt a very common convention: in a commutative diagram, we use a dashed arrow ⇢ to indicate an arrow which is to be uniquely fixed by the requirement that the diagram commutes.

(c) This new definition is often served up 'neat', without much preceding ceremony. And I suppose it is true that you can just stare hard at the diagram, and 'see' that it gives 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 thought might go something along these lines. First, the fact that for any S (and f_1, f_2) our diagram always commutes for *some* arrow u tells us that, whatever our vantage point S, O packages up and preserves *enough* information about X and Y as seen from S (seen via f_1, f_2) for us to be able retrieve the information again by going to O via u and then using the projection arrows to recover f_1, f_2. While second, the fact that the mediating arrow u is unique tells that O packages the information without

[1] NB carefully: 'motivates' doesn't mean 'forces'! (In Set, because of special features of the category, the first-shot definition of a product in terms of point elements is in fact equivalent to the official definition in terms of generalized elements: but that's a very special case. We don't in general get equivalence, even in well-pointed categories.)

redundancy, there is no slack for u to vary over, so O (so to speak) preserves *no more than enough*.

But that really is not wonderfully transparent, is it? So I do think it has been well worth taking the longer route to our destination. We can already see that Defn. 45, tweaked to become Defn. 45*, defines a product pre-categorially in an entirely natural way, given what we want from a pairing scheme. The pre-categorial Defn. 45* then immediately gives us a categorial story about products in Set. And the route from that to the cross-category Defn. 46 then does involve a pretty natural generalization.

10.2 Examples

(a) Let's now check that our official definition behaves well in various categories. So, first,

(1) In Set, as you would most 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 $\langle x, y\rangle \overset{\pi_1}{\longmapsto} x$ and $\langle x, y\rangle \overset{\pi_2}{\longmapsto} y$, form a binary product.

For 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, 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.

Inspired by this paradigm case, we will now often default to the notation $X \times Y$ for the object forming a binary product of X with Y, thus $(X \times Y, \pi_1, \pi_2)$. By this stage, you should really be inoculated against the temptation to over-read this notation as always indicating something Cartesian-like.

(2) We can similarly construct products in the category Pos which has posets as objects and order-preserving maps as arrows.

Suppose we take two posets $(X, \leqslant_X)$ and $(Y, \leqslant_Y)$ and form the usual Cartesian product $X \times Y$ of their underlying sets, and equip *this* with the component-wise product order: i.e., in the obvious notation, we define $\langle x, y\rangle \leqslant_{X\times Y} \langle x', y'\rangle$ to hold if and only if $x \leqslant_X x'$ and $y \leqslant_Y y'$. This gives us a poset $(X \times Y, \leqslant_{X\times Y})$.

Now, note that the obvious projection map from $(X \times Y, \leqslant_{X\times Y})$ to $(X, \leqslant_X)$ will be order preserving, given our definition of the product order: so this projection map along with the companion projection map from $(X \times Y, \leqslant_{X\times Y})$ to $(Y, \leqslant_Y)$ will count as arrows in Pos. It is then easily confirmed that $(X \times Y, \leqslant_{X\times Y})$ equipped with these two projection maps forms a categorical product of our original two posets in Pos.

(b) At the end of §2.6, I promised that it would turn out that various claims about product groups can be recast as claims about the existence of appropriate homomorphisms between groups. And we now know how to define a categorial product of groups using arrows. We should check that this accords with the pre-categorial definition when applied to groups living in Grp

(3) For implementations of groups in the category Grp (so living in some universe of sets), we can use the same Kuratowksi construction for pairs. And then, relative to that pairing scheme, the standard direct product of the groups $(G, *, e)$ and $(G', *', e')$ will be the group $(G \times G', \star, d)$, where $\star$ is 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 then easily checked to be a group homomorphism $\pi_1 \colon G \times G' \to G$. And we define π_2 similarly, of course.

We now need to check that, thus defined, the group $G \times G'$ (for short) equipped with π_1 and π_2 is indeed a categorial product.[2] That's easy, following the same line of argument as in example (1).

Continuing our examples:

(4) 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. So this category too has a binary product for any of its objects.

(5) Now revisit the category Prop_L introduced in §4.5, (C14). Its objects are propositions, closed wffs of a given first-order language 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 Prop_L.

Why? Take any arrows in Prop_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 beginning to line up nicely with products intuitively understood.

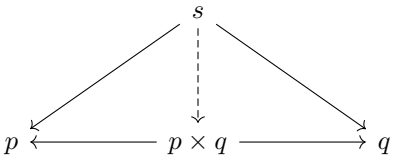
(c) Let's have one more example to be going on with:

[2] To be extra clear: the group $G \times G'$ is an object living in Grp, and π_1 and π_2 are arrows also living in Grp. And that's all it takes for the product $(G \times G', \pi_1, \pi_2)$ to count as existing in Grp. For the product is not an extra item over and above the product-object and the projection arrows.

As already stressed, we shouldn't overinterpret our notation – the parentheses in '$(G \times G', \pi_1, \pi_2)$' are just punctuation, unlike say the curly brackets in '$\{G \times G', \pi_1, \pi_2\}$' which would serve to introduce a new sort of set that doesn't itself live in Grp.

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

What is a product of p and q in 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 from s to q, there is a unique arrow from s to $p \times q$ making this diagram commute:



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 $s \preccurlyeq p \times q$. So the object $p \times q$ must be the 'meet' or greatest lower bound of p and q in $(\mathrm{P}, \preccurlyeq)$.

A simple but very important general moral is neatly emphasized by 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).

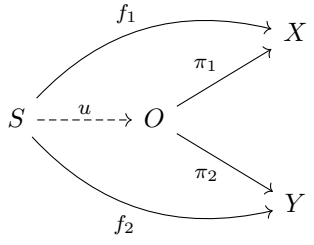
10.3 Products as terminal objects

(a) Defn. 46 defines the notion of *a* product of a pair 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 obviously equivalent, way of defining products.

We need an auxiliary notion. Let's say

Definition 47. A *wedge to X and Y* (in category C) is an object S and a pair of arrows $f_1 \colon S \to X$, $f_2 \colon S \to Y$. Call S the vertex of the wedge. △

Then a wedge $\begin{array}{ccc} & \overset{\pi_1}{\nearrow} & X \\ O & & \\ & \underset{\pi_2}{\searrow} & Y \end{array}$ is a product of X with Y iff, for any wedge $\begin{array}{ccc} & \overset{f_1}{\nearrow} & X \\ S & & \\ & \underset{f_2}{\searrow} & Y \end{array}$ to X and Y, there exists a unique arrow u making the following commute:



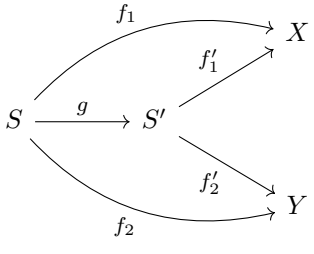
That's just our previous definition put in different terms, with the diagram rotated! No mystery here.

In such a case where f_1 factors as $\pi_1 \circ u$ and f_2 as $\pi_2 \circ u$, we will say that the whole wedge $X \xleftarrow{f_1} S \xrightarrow{f_2} Y$ *(uniquely) factors through* the product wedge via the mediating arrow u.

(b) A quick terminological aside. It is a common categorial idiom to use the informal 'factors through' in a pretty relaxed spirit. Wikipedia's List of Mathematical Jargon puts it this way: "If for three objects A, B, and C a map $f\colon A \to C$ can be written as a composition $f = h \circ g$ with $g\colon A \to B$ and $h\colon B \to C$, then f is said to factor through any (and all) of B, g, and h." Talk of a wedge, i.e. a pair of arrows, factoring through another wedge, another pair of arrows, is a natural extension.

(c) Now for another definition involving wedges.

Recall, the category C/X, the slice category of C over X, has as it objects pairings of C-objects and C-arrows of the form $(S, f\colon S \to X)$.We are now going to introduce a new category C/XY, the *wedge category* of C over X and Y. Its objects are going to be triples of a C-object and *two* C-arrows of the form $(S, f\colon S \to X, g\colon S \to Y)$. In other words – what a surprise! – the objects of the wedge category C/XY are C-wedges to X and Y. And looking at the definition of slice categories, the corresponding definition for arrows in wedge categories should be predictable – just meditate on the following diagram:



Thus, we will say:

Definition 48. Given a category C and C-objects X, Y, then the *wedge category* C/XY has the following data.

(1) Its objects are all the wedges (S, f_1, f_2) from any S to X, Y.
(2) And an arrow from (S, f_1, f_2) to (S', f_1', f_2') is a C-arrow $g\colon S \to S'$ such that the two resulting triangles commute: i.e. $f_1 = f_1' \circ g$, $f_2 = f_2' \circ g$.

Composition of two arrows in C/XY is defined as being the same as their composition as arrows of C.[3] △

[3]OK – we are cheating a bit again! For recall the irritating complication we mentioned in §6.3 when defining slice categories. We get the same irritating complication here, and really should define C/XY-arrows as whole commuting diagrams in C, not just as single C-arrows. I can perhaps leave it to pernickety readers to fuss about the details, and about why they don't matter for our purposes!

(d) Finally, our new notion of the derived category C/XY to hand, we can revisit our previous definition of a product. A moment's more reflection shows that it is straightforwardly equivalent to

Definition 49. A product of X with Y in C is a terminal object of the wedge category C/XY. △

Think about it! – this really is rather cute.

10.4 Uniqueness up to unique isomorphism

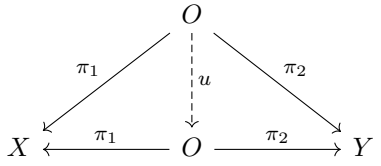
(a) As noted, products need not exist for arbitrary objects X and Y in a given category C; and when they exist, they need not be strictly unique. However, when they do exist, then – as announced – they *are* 'unique up to unique isomorphism' (compare Theorems 26 and 34). That is to say,

Theorem 36. *If both (O, π_1, π_2) and (O', π'_1, π'_2) are products for X with Y in the category C, then there is a unique isomorphism $f : O \xrightarrow{\sim} O'$ 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 of objects both taken from X. There are evidently two isomorphisms between $X \times X$ 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 O of any two products (O, π_1, π_2) for X with Y *which commutes with the products' respective projection arrows.*

We are now going to prove our theorem twice over – rather ploddingly from first principles, and then more zippily using our redefinition of products as terminal objects.

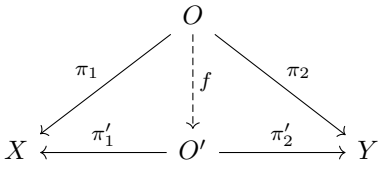
Plodding proof. Since (O, π_1, π_2) is a product for X with Y in C, every wedge to X and Y factors uniquely through it, including itself. In other words, there is a unique u such that this diagram commutes:



But evidently putting 1_O for the central arrow makes the diagram commute. So by the uniqueness requirement we know that

(i) Given a product (O, π_1, π_2) and an arrow $u : O \to O$, if $\pi_1 \circ u = \pi_1$ and $\pi_2 \circ u = \pi_2$ (so the product factors through itself via u), 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:



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 versa versa, since (O, π_1, π_2) is 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 $\pi_2 \circ g \circ f = \pi_2$.

But $g \circ f$ is an arrow from O to O. So it follows by (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. □

You'll recognize the key proof idea here is closely akin to the one we used in proving Theorem 26, showing that initial objects are unique up to unique isomorphism. And we indeed can simply appeal to that earlier result:

Succinct proof using the alternative definition of products. Both (O, π_1, π_2) and (O', π_1', π_2') are terminal objects in the category C/XY. Therefore by our earlier theorem there is a unique C/XY-isomorphism f between them. But, by definition, this has to be a C-arrow $f\colon O \to O'$ commuting with the projection arrows. It is immediate that an isomorphism in C/XY is also an isomorphism in C. □

(b) Here's a simple corollary of our last theorem.

Theorem 37. *In a category where the relevant products exist,* $X \times Y \cong Y \times X$.

Proof. Suppose $(X \times Y, \pi_1\colon X \times Y \to X, \pi_2\colon X \times Y \to Y)$ is a product of X with Y; then – applying the definition – $(X \times Y, \pi_2\colon X \times Y \to Y, \pi_1\colon X \times Y \to X)$ will count as a product of Y with X. Hence, by Theorem 36, there is an isomorphism between this particular product-object $X \times Y$ and the object $Y \times X$ of any other product of Y with X. □

(c) When discussing terminal objects, we not only showed that they are unique up to unique isomorphism (Theorem 26) but that any objects isomorphic to them are also terminal (Theorem 27). Similarly for products: we've just shown that they are unique up to unique isomorphism. We now prove that wedges that factor through a product via an isomorphism are themselves products.

Theorem 38. *If the wedge* $X \xleftarrow{\pi_1} O \xrightarrow{\pi_2} Y$ *is a product of* X *with* Y *and there is an isomorphism* $o\colon O' \xrightarrow{\sim} O$ *such that* $\pi'_1 = \pi_1 \circ o$ *and* $\pi'_2 = \pi_2 \circ o$*, then the wedge* $X \xleftarrow{\pi'_1} O' \xrightarrow{\pi'_2} Y$ *is itself a product of* X *with* Y*.*

Proof. Take any wedge $X \xleftarrow{f_1} S \xrightarrow{f_2} Y$. We need to show (i) there is an arrow $v\colon S \to O'$ such that $f_j = \pi'_j \circ v$ (for $j = 1, 2$), and (ii) v is unique.

But we are given that $X \xleftarrow{\pi_1} O \xrightarrow{\pi_2} Y$ is a product so we know that there is a unique arrow $u\colon S \to O$ such that $f_j = \pi_j \circ u$. And we know $\pi'_j = \pi_j \circ o$, hence $f_j = \pi'_j \circ o^{-1} \circ u$. Therefore put $v = o^{-1} \circ u$, and that satisfies (i).

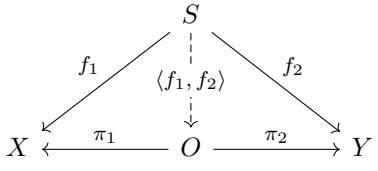
Now suppose there is another arrow $v'\colon S \to O'$ such that $f_j = \pi'_j \circ v'$. Then we have an arrow $o \circ v'\colon S \to O$, and also $f_j = \pi_j \circ o \circ v'$. Which makes the wedge with the apex S factor through the original product wedge via $o \circ v'$. But by uniqueness of mediating arrows, that means $o \circ v' = u$. Hence $v' = o^{-1} \circ u = v$. Which proves (ii). □

To put this another way, if $X \xleftarrow{\pi_1} O \xrightarrow{\pi_2} Y$ is a product of X with Y and there is an isomorphism $o\colon O' \xrightarrow{\sim} O$, then $X \xleftarrow{\pi_1 \circ o} O' \xrightarrow{\pi_2 \circ o} Y$ is also a product of X with Y.

10.5 Notations for mediating arrows

(a) We should note some fairly predictable notation:

Definition 50. 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 $\langle f_1, f_2 \rangle\colon S \to O$, as here:



△

We should of course check that this product-style notation $\langle f_1, f_2 \rangle$ for mediating arrows doesn't mislead. But we have:

Theorem 39. *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 $f_2 = g_2$. □

We will put our shiny new notation for mediating arrows to work in the (optional) Chapter 12, but you should know it in any case.

(b) A special case is worth noting:

Definition 51. 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 $\langle 1_X, 1_X \rangle \colon X \to X \times X$.

That unique mediating arrow can also be notated δ_X, and 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 $\langle x, x \rangle$. We can 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'.

10.6 'Universal properties'

Let's pause for a quick general comment.

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 also say that products are defined by a universal mapping property.

We've already met other examples of such universal mapping properties: terminal and initial objects are also defined by how any other object has a unique map/arrow to or from them. We will meet lots more examples over coming chapters.

It is too soon to attempt a formal definition of what it is to be defined by a universal (mapping) property. For the moment, then, I am just making a first informal gesture towards a common pattern of categorial definition which you will start to recognize again when you repeatedly come across it.

10.7 Coproducts

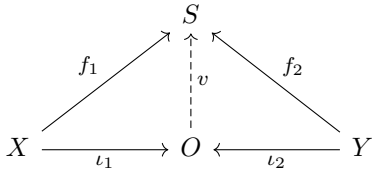
(a) We are going now to discuss the duals of products. But first, we should note a common terminological trope:

Definition 52. Duals of categorially defined widgets are very often called *co-widgets*. Thus a *co-widget* of the category C is a widget of C^{op}. △

For 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 53. In any category C, *a 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:



The object O in a coproduct for X with Y is very often notated '$X \oplus Y$' or '$X \amalg Y$'. △

Note, however, that the 'injections' in this sense need not be injective or even monic.

(c) It is useful to introduce another an auxiliary notion. Let's say

Definition 54. A *corner from* X *and* Y (in category C) is an object S and a pair of arrows $f_1 \colon X \to S$, $f_2 \colon Y \to S$. Call S the vertex of the corner. △

Draw this situation, to see why corners are sensibly called corners! 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 define a coproduct of X with Y as an initial object of this category. It is a useful check on understanding to work through the easy details: just reverse arrows!

(d) 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 Prop_L (which we met in §10.2) 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. $c \preccurlyeq d$.

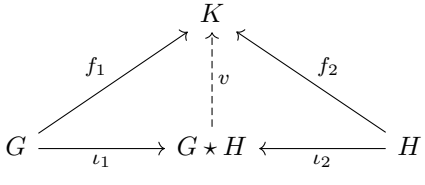
Which means that the coproduct of p and q, if it exists, must be their least upper bound (equipped with the obvious two arrows, of course).

(e) In some cases, however, the story about coproducts gets markedly more complicated. I'll finish this chapter by mentioning one more example. However, the details really aren't going to matter later, so this is an afterword for enthusiasts. By all means skip this result:

(4) In the category Grp, coproducts are (isomorphic to) the so-called 'free products' of groups.

Take the groups $(G, \cdot, e)$ and $(H, \odot, d)$. 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 $G \star H = (G \star H, \diamond, e)$ is a group – the so-called free product of the two groups G and H – and there are obvious 'injection' group homomorphisms $\iota_1 \colon G \to G \star H$, $\iota_2 \colon H \to G \star H$ (these send an object g or h respectively to itself as a member of $G \star H$).

Claim: $(G \star H, \iota_1, \iota_2)$ is a coproduct for the groups G and H. That is to say, for any group $K = (K, *, k)$ and group homomorphisms $f_1 \colon G \to K$, $f_2 \colon H \to K$, there is a unique v such that this commutes:



Proof. Put $v \colon G \star H \to K$ 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, $v \circ \iota_1 = f_1$, $v \circ \iota_2 = f_2$. 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 $v'(gh) = v'(g) * v'(h) = v'(\iota_1(g)) * v'(\iota_2(h)) = f_1(g) * f_2(g) = v(\iota_1(g)) * v(\iota_1(g)) = v(\iota_1(g) * \iota_1(g)) = v(gh)$. 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. □

11 Products more generally

So we have arrived at a categorial definition of *binary* products. What, though, about products of three or more objects? What about products of an infinite collection of objects?

In fact the story extending beyond the binary case more or less writes itself. But this chapter spells things out.

11.1 Ternary products

(a) Here's the obvious generalization of our previous definition, moving from two-ply to three-ply products:

Definition 55. In any category 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 three 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 $u \colon S \to O$ such that $f_i = \pi_i \circ u$. △

And then, exactly as we would expect, using the same proof ideas as in the binary case, we can prove that ternary products are unique up to a unique isomorphism, i.e.

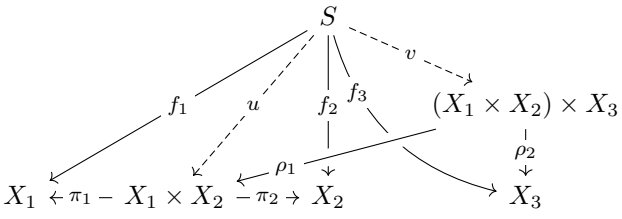
Theorem 40. *If the ternary products* (O, π_1, π_2, π_3) *and* $(O', \pi'_1, \pi'_2, \pi'_3)$ *for* X_1, X_2, X_3 *both exist in* C*, then there is a unique isomorphism* $f \colon O \xrightarrow{\sim} O'$ *commuting with the projection arrows.* □

I can safely leave filling in the details as an exercise.

(b) We now note that if C has binary products for all pairs of objects, then it automatically has ternary products too, for

Theorem 41. $(X_1 \times X_2) \times X_3$ *together with the obvious projection arrows forms a ternary product of* X_1, X_2, X_3*.*

Proof. We just hack through the details (sorry!). So 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. And now think about the following diagram.



So take any object S and the three 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 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. And we've just seen that 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. So it only remains to confirm v's uniqueness in this role.

Suppose that triple wedge also factors through $(X_1 \times X_2) \times X_3$ via the arrow w. In other words, suppose 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 since we also have $f_3 = \rho_2 \circ w$, it follows by (b) that $w = v$. □

(c) 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. So we immediately get the following corollary:

Theorem 42. *Assuming the products exist,* $X \times (Y \times Z) \cong (X \times Y) \times Z$.

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 40 entails that $X_1 \times (X_2 \times X_3) \cong (X_1 \times X_2) \times X_3$. □

11.2 More finite products

Defn. 55 defines ternary, i.e. three-way, products: we can give exactly similar definitions for four-way, five-way, n-way products for any finite $n \geq 2$. And just as we can build a three-way product of X_1, X_2 and X_3 from two binary products, as in $((X_1 \times X_2) \times X_3)$, we can build a four-way product of X_1, X_2, X_3 and X_4 from three binary products as in $((X_1 \times X_2) \times X_3) \times X_4$. More generally, if we can freely construct any binary products we like, we can also construct n-ary products for any finite $n \geq 2$.

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 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 tells us that a nullary product is a terminal object of the category.

And a *unary* product of X would be an object O and an 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 56. A category C *has all binary products* iff for all C-objects X and Y, there exists a binary product of X with Y in C. C *has all finite products* iff it has n-ary products for any n C-objects, for all $n \geq 0$. △

Then our preceding remarks establish

Theorem 43. *A category* C *has all finite products iff* C *has a terminal object and has all binary products.* □

Need we add that these theorems of course all dualize to coproducts? How?

11.3 Infinite products

We can generalize still further in an obvious way, going beyond finite products to infinite cases.

Definition 57. Suppose that we are dealing with C-objects X_j indexed by items j in some suite of indices J (not assumed finite). Then the product of the X_j, if it exists in 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 is not *too* ludicrously large, so that it can certainly be represented as a *set* in standard set theory. We then say:

Definition 58. A category C *has all small products* iff for any C-objects X_j, for $j \in J$ where J is some index set, these objects have a product. △

Here, 'small' is the category theorist's idea of a joke. It doesn't mean small by any normal standards – it only 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 Part II.

12 Binary products explored

The conceptual basics about products are all in place. This chapter now reads into the record a variety of theorems, either showing that categorial products have properties we naturally want them to have, or giving results which will be useful later. It is up to you how much you want to nail down the technicalities here. If you like, you could just glance at the first section and then skip on to the next chapter. Then later you could return to delve into some of this chapter's details, on a need-to-know basis.

To make things a bit more fun (if that's the right word!), I will state the chapter's theorems in the next section as a series of challenges for enthusiasts to prove, before I go on to give answers to the challenges in the remaining sections. These answers do quite nicely illustrate some typical elementary proof strategies.

12.1 Challenges!

Start by showing this easiest of results:

Theorem 44. *Given a product* $(X \times Y, \pi_1, \pi_2)$ *and arrows* $S \overset{u}{\underset{v}{\rightrightarrows}} X \times Y$, *then, if* $\pi_1 \circ u = \pi_1 \circ v$ *and* $\pi_2 \circ u = \pi_2 \circ v$, *it follows that* $u = v$.

Next check a result which looks as though it ought to hold (why?)

Theorem 45. *In a category which has a terminal object 1, products* $1 \times X$ *and* $X \times 1$ *exist, and* $1 \times X \cong X \cong X \times 1$.

Here, as always, when we mention just the object of a product, take this to be equipped with the obvious projection arrows. Also show

Theorem 46. *There are categories with initial objects where the product* $0 \times X$ *(or* $X \times 0$*) always exists but is not generally isomorphic to* 0.

(Hint: you needn't look beyond the very earliest examples of categories you met.)

Now a lemma which is of later use:

Theorem 47. *If* $1 \xleftarrow{!} 1 \times X \xrightarrow{\pi} X$ *is a product, then* π *is an isomorphism. Similarly for the mirror image result.*

More interestingly, show:

Theorem 48. *Assuming $\langle f, g\rangle$ and e compose, $\langle f, g\rangle \circ e = \langle f \circ e, g \circ e\rangle$.*

Theorem 49. *Given an arrow $q\colon S \to X$, $\delta_X \circ q = \langle q, q\rangle$.*

Next, provide a definition:

Definition? Suppose we have two arrows $f\colon X \to X'$, $g\colon Y \to Y'$. Then we will 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 $g(y)$. *What is an appropriate categorial definition for $f \times g$?*

Check your definition with the one given at the beginning of §12.4. Now ask yourself: why *ought* the following theorems hold? Then prove them!

Theorem 50. *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 $o \circ (f \times g) = (g \times f) \circ o$.*

Theorem 51. *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 $(f \times g) \circ \delta_X$.*

Theorem 52. *Assume that there are arrows*

$$\begin{array}{ccccc} X & \xrightarrow{\quad f \quad} & X' & \xrightarrow{\quad j \quad} & X'' \\ Y & \xrightarrow{\quad g \quad} & Y' & \xrightarrow{\quad k \quad} & Y'' \end{array}$$

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 $(j \times k) \circ (f \times g) = (j \circ f) \times (k \circ g)$.

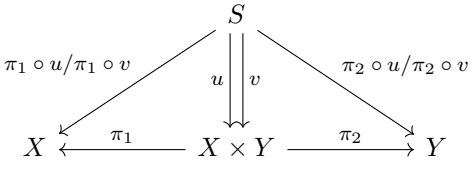
Of course, everything in this chapter will dualize: but let's leave it as a further exercise to supply all the corresponding theorems about coproducts.

12.2 Four simple theorems, and a non-theorem

(a) Our first challenge was the gentlest of warm-up exercises. Prove:

Theorem 44. *Given a product $(X \times Y, \pi_1, \pi_2)$ and arrows $S \overset{u}{\underset{v}{\rightrightarrows}} X \times Y$, 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:



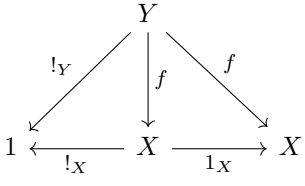
Hence $u = v$ by uniqueness of mediating arrows. □

(b) Now, as noted, we might reasonably hope the following is true: –

Theorem 45. *In a category which has a terminal object 1, products* $1 \times X$ *and* $X \times 1$ *exist, and* $1 \times X \cong X \cong X \times 1$.

Proof. Following the notational convention of Defn. 38, we will use $!_X$ for the unique arrow from X to the terminal object 1, and 1_X is of course the identity arrow on X.

Consider then the wedge $1 \xleftarrow{!_X} X \xrightarrow{1_X} X$, and take any other wedge to 1 and X, namely $1 \xleftarrow{!_Y} Y \xrightarrow{f} X$. The following diagram then commutes:



(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 36, given any product $(1 \times X, \pi_1, \pi_2)$, we have $1 \times X \cong X$. Exactly similarly, $X \times 1 \cong X$. □

(c) Question: do we similarly have $0 \times X \cong 0$ in categories with an initial object and the relevant product? Answer: Not always.

Theorem 46. *There are categories where the product* $0 \times X$ *(or* $X \times 0$*) always exists but is not generally isomorphic to* 0.

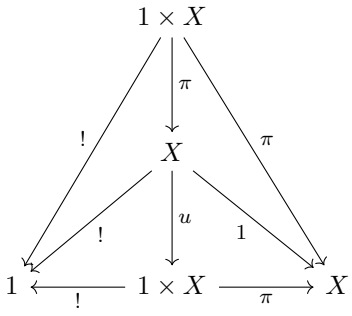
Proof. Take a category like Grp which has a null object, i.e. where $0 = 1$. Then $X \cong (1 \times X) = (0 \times X)$. But in general, it won't be the case that $X \cong 0$. Hence it won't be the case in general that $(0 \times X) \cong 0$. □

(d) To reduce clutter, let's drop subscripts from unique arrows to terminal objects (so we e.g. write simply '!' rather than '$!_{1\times X}$' for the unique arrow from $1 \times X$ to 1), and also let's drop subscripts from identity arrows. It then becomes a nice reality check to mentally replace the subscripts in the following diagram and in the proof of our next little theorem:

Theorem 47. *If (W)* $1 \xleftarrow{!} 1 \times X \xrightarrow{\pi} X$ *is a product, then* π *is an isomorphism. Similarly for the mirror image result.*

We know from Theorem 45 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 like (W), π has to be an isomorphism.

Proof. Consider, then, the following diagram:



This commutes. Why?

First, there is a (unique) mediating arrow u making the bottom two triangles commute. In other words – and see §10.3 again for 'factors through' – the middle wedge (V) $1 \xleftarrow{!} X \xrightarrow{1} X$ factors through the bottom product (W) via a unique u, giving $\pi \circ u = 1$.

Similarly the top wedge, a copy of (W) again, factors through (V) as shown. (The top left triangle commutes, i.e. $!_X \circ \pi = !_{1\times X}$ because arrows to the same terminal object are unique.)

But putting the upper and lower triangles together means that (W) factors through (W) via the mediating arrow $u \circ \pi$. But since (W) also factors through itself via 1, and such mediating arrows are unique by the definition of a product, it follows that $u \circ \pi = 1$.

Having inverses on both sides, π is therefore an isomorphism. □

(e) And now, again for future reference, we should remark in passing on a non-theorem.

Suppose we have a pair of parallel composite arrows built up using the same projection arrow like this: $X \times Y \xrightarrow{\pi_1} X \underset{g}{\overset{f}{\rightrightarrows}} X'$. In Set, the projection arrow here '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. 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:

$$X' \underset{g}{\overset{f}{\leftleftarrows}} X \xleftarrow{\pi_1} V \xrightarrow{\pi_2} Y$$

If that is all the data we have to go on, we can consistently stipulate that in this mini-category $f \neq g$ but $f \circ \pi_1 = g \circ \pi_1$.

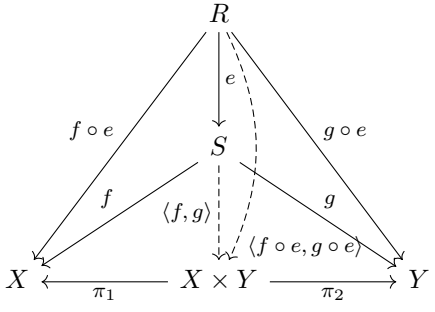
Now, there is only one wedge of the form $X \longleftarrow ? \longrightarrow Y$, so trivially all wedges of that shape uniquely factor through it. In other words, the wedge $X \xleftarrow{\pi_1} V \xrightarrow{\pi_2} Y$ is a product and π_1 is indeed a projection arrow. But by construction it isn't epic.

12.3 Diagonal arrows

Theorem 48. *Assuming $\langle f, g\rangle$ and e compose, $\langle f, g\rangle \circ e = \langle f \circ e, g \circ e\rangle$.*

Proof. $\langle f, g\rangle$ is a mediating arrow from some S to a product of (say) X and Y. And since we are assuming $\langle f, g\rangle \circ e$ is defined, the target of e must be S.

So this diagram commutes because each triangle commutes:

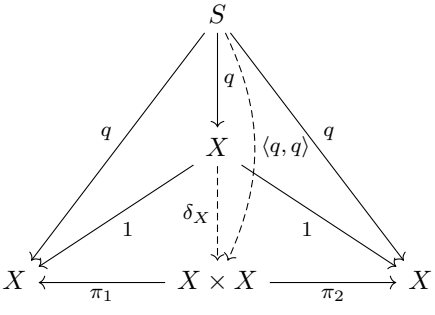


Hence in particular $\langle f, g\rangle \circ e$ is a mediating arrow factoring the wedge with apex R through the product $X \times Y$.

But by definition, the unique mediating arrow is $\langle f \circ e, g \circ e\rangle$. □

Theorem 49. *Given an arrow $q\colon S \to X$, $\delta_X \circ q = \langle q, q\rangle$.*

Proof. Consider the following similar diagram:



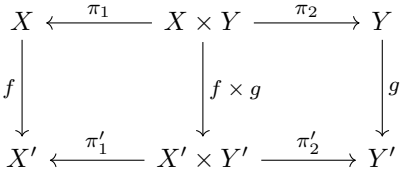
The inner triangles commute, hence $\delta_X \circ q$ is a mediating arrow factoring the wedge with apex S through the product $X \times X$.

But looking at the outer triangle, the unique mediating arrow which does that is by definition $\langle q, q\rangle$. □

12.4 Arrows between two products

(a) Suppose we have two arrows $f\colon X \to X'$, $g\colon Y \to Y'$. How can we characterize an arrow between products, $f \times g\colon X \times Y \to X' \times Y'$, which works component-wise?

In categorial terms, we require $f \times g$ to be such that the following diagram commutes:



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 fixed uniquely by the requirement that that diagram commutes, and hence must equal $\langle f \circ \pi_1, g \circ \pi_2 \rangle$.

(b) This shows that the following definition is in good order:

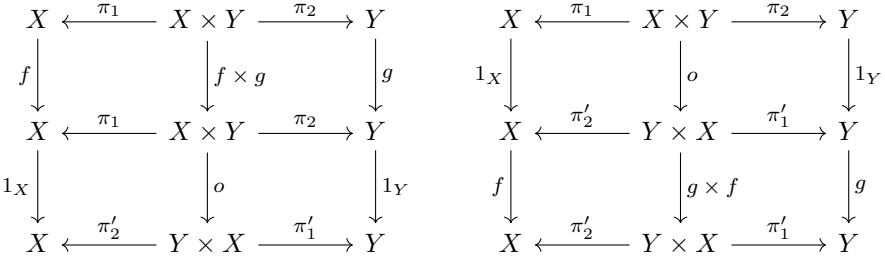
Definition 59. Given the arrows $f\colon X \to X'$, $g\colon Y \to Y'$, and the products $(X \times Y, \pi_1, \pi_2)$ and $(X' \times Y', \pi'_1, \pi'_2)$, then put $f \times g = \langle f \circ \pi_1, g \circ \pi_2 \rangle\colon X \times Y \to X' \times Y'$. $f \times g$ then acts component-wise on the product $X \times Y$, like f on X and g on Y. △

And to check everything works as it should, let's prove our pair of results which should look reasonably intuitive:

Theorem 50. *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 $o \circ (f \times g) = (g \times f) \circ o$.*

This ought to hold because it shouldn't matter whether we first apply f and g component-wise to a product, and then swap the order of terms in the product, or alternatively first swap the order of terms, and then apply g and f component-wise.

Proof. Suppose we have products $(X \times Y, \pi_1, \pi_2)$ and $(Y \times X, \pi'_1, \pi'_2)$, and an order-swapping isomorphism $o\colon X \times Y \to Y \times X$. And now consider the following pair of diagrams (being very careful with the directions of the projection arrows!):

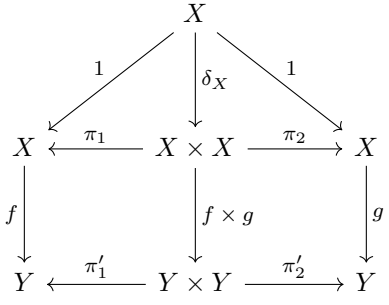


Both diagrams commute. Hence the wedge $X \xleftarrow{f \circ \pi_1} X \times Y \xrightarrow{g \circ \pi_2} Y$ factors through the bottom product via both $o \circ (f \times g)$ and $(g \times f) \circ o$. Those arrows must therefore be equal by the uniqueness of mediating arrows. □

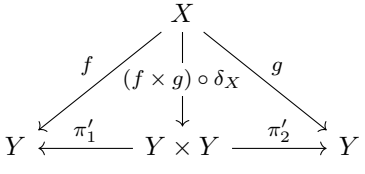
Theorem 51. *Suppose we have parallel arrows $f, g\colon X \to Y$ in a category with binary products. Then the arrow $\langle f, g\rangle$ is equal to the composite $(f \times g) \circ \delta_X$.*

Think of the situation in Set, for example. The idea is that it should not matter whether we apply the functions f and g separately to some member x of X and then take the product of the results, or alternatively form the product of x with itself and then apply f and g to the resulting pair componentwise.

Proof. Take the diagram



This commutes by the definitions of δ_X and $f \times g$. Hence the following also commutes:



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 §12.2, whose only arrows are as diagrammed

$$X' \underset{g}{\overset{f}{\leftleftarrows}} X \xleftarrow{\pi_1} V \xrightarrow{\pi_2} Y$$

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 $\langle f \circ \pi_1, 1_Y \circ \pi_2\rangle = \langle g \circ \pi_1, 1_Y \circ \pi_2\rangle$, and hence $f \times 1_Y = g \times 1_Y$.

(d) Finally in this chapter, here is another result we will need later:

Theorem 52. *Assume that there are arrows*

$$\begin{array}{ccccc} X & \xrightarrow{\quad f \quad} & X' & \xrightarrow{\quad j \quad} & X'' \\ Y & \xrightarrow{\quad g \quad} & Y' & \xrightarrow{\quad k \quad} & Y'' \end{array}$$

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. 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 warm-up lemma Theorem 44. □

Well, that was all quite fun as promised, if you enjoy that sort of diagram-wrangling!

13 Groups in categories

I'm going to pause at this point, before developing any more categorial apparatus. Because I want to show that we have already said enough to characterize so-called *internal groups* living in categories. I needn't take the discussion very far: the aim is simply to illustrate how we can begin to talk in category-theoretic terms about one familiar type of mathematical structure.

13.1 Instead of binary functions

A preliminary point. In category theory, arrows have single sources. So, as already noted in §9.3, in categories where arrows are functions they are always monadic functions. But then how can we accommodate binary functions (or polyadic functions more generally)? In particular, how can we accommodate the binary operations that characterize groups?

Well, recall how things are handled on the orthodox set-theoretic approach: we in fact model e.g. a two-place total function from numbers to numbers (addition, say) as a function $f\colon \mathbb{N}^2 \to \mathbb{N}$. And 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, a pair-object, to a value: such an f is not a real binary function.

Of course, in set-theory, we can arrange it that for any two things there is a pair-object that codes for them – we usually choose a Kuratowski pair. Hence we can indeed unproblematically trade in a function from two objects for a related function from the corresponding pair-object. And standard notational choices can make the move invisible. Suppose we adopt the modern convention of using '(m, n)', with common-or-garden parentheses, as our notation for the ordered pair of m with n. Then the notation '$f(m, n)$' invites being parsed either way, as representing a two-place function taking the two arguments m and n, or as a corresponding one-place function taking a single argument, the pair (m, n). But note: the fact that the trade between the two-place and the one-place function is now notationally disguised doesn't mean that it isn't being made!

In sum, the usual set-theoretic procedure is to trade in an underlying binary function $\underline{f}\colon A, B \to C$ for a related unary function $f\colon A \times B \to C$. And this same procedure is of course now available to us in any category with products.

13.2 Groups in Set

So to our main theme. How can we characterize groups (ok, if you insist on being pernickety, implementations of groups) living e.g. in the category Set?

We need an object G in our category which collects together the elements of the relevant group, and we need three arrows (which are functions in this category):

(i) $m\colon G \times G \to G$ which represents the group operation (so, as just announced, we have traded the informal two-place group operation 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 to be the identity – we'll also informally call this distinguished member of the group 'e', allowing context to disambiguate);

(iii) $i\colon G \to G$ (this will be 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)
$$\begin{array}{ccccc} (G\times G)\times G & & \xrightarrow{\quad\cong\quad} & & G\times(G\times G) \\ \downarrow{\scriptstyle m\times 1_G} & & & & \downarrow{\scriptstyle 1_G\times m} \\ G\times G & \xrightarrow{\quad m\quad} & G & \xleftarrow{\quad m\quad} & G\times G \end{array}$$

Here the arrow at the top represents the naturally arising isomorphism between the two triple products that is established by Theorem 40.

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 our distinguished object e to act like an identity for the group operation.

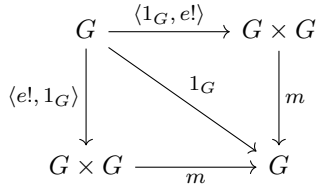
To characterize this condition categorically, start by defining the map $e!\colon G \to G$ by composing the unique map $!\colon G \to 1$ with $e\colon 1 \to G$. In Set, $e!$ is then the function which sends anything in G to the identity element e. We then have the following product diagram:

$$\begin{array}{ccccc} & & G & & \\ & {\scriptstyle 1_G}\swarrow & \downarrow{\scriptstyle \langle 1_G, e!\rangle} & \searrow{\scriptstyle e!} & \\ G & \xleftarrow{\quad\pi_1\quad} & G\times G & \xrightarrow{\quad\pi_2\quad} & G \end{array}$$

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:

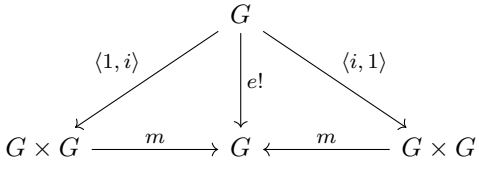
(G2)



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)



For take an element $g \in G$. Going left, the arrow $\langle 1, i \rangle$ maps g to $(g, i(g))$ which is then sent by m to $m(g, i(g))$. The central vertical arrow meanwhile simply sends g to e. Therefore, the requirement that the left triangle commutes tells us, as we want, that $m(g, i(g)) = e$. Similarly the requirement that the right triangle 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 in Set in fact makes sense whenever we are working in a category with binary products and a terminal object. So it is natural to generalize, as follows:

Definition 60. Suppose C is a category which has binary products and a terminal object. Let G be a C-object, and $m \colon G \times G \to G$, $e \colon 1 \to G$ and $i \colon G \to G$ be C-arrows. Then (G, m, e, i) is an *internal group* in C iff the three diagrams (G1), (G2), (G3) commute, where $e!$ in the latter two diagrams is the composite map $G \xrightarrow{!} 1 \xrightarrow{e} G$. △

An internal group is, alternatively, also called a 'group object'.

Then, if we don't fuss about the type-difference between an arrow $e \colon 1 \to G$ (in a internal group) and a designated element e (in a group), we have established the summary result

Theorem 53. *In the category* Set*, an internal group constitutes a group.* □

And conversely, every group living in Grp (our universe of groups-implemented-as-sets) can be regarded as constituting an internal group in Set.

13.3 Groups in other categories

(a) Here are a couple more examples of internal groups in other categories:

Theorem 54. *(1) In the category* Top*, which comprises topological spaces with continuous maps between them, an internal group is a topological group in the standard sense.*

(2) In the category Man*, which comprises smooth manifolds with smooth maps between them, an internal group is a Lie group.*

The proofs of these two claims are pretty straightforward, at least if you know the usual definitions of topological groups and Lie groups. But I won't pause over the details here. I just note that the categorial story here will nicely bring out what is common between the various cases.

(b) But here's another, much less predictable result:

Theorem 55. *In the category* Grp*, an internal group must itself be a abelian.*

How strange! Yet the proof is relatively straightforward, really quite cute, and a rather useful reality-check. So – at least for enthusiasts – here it is:

Proof. Suppose G, equipped with m, e, i is an internal group in Grp.

Then, since we are in the category Grp, the object G is *already* a group – let's use a dotty notation for *this* group G, so it will be a set of objects $\dot{G}$ equipped with a group operation we can notate as in '$x \cdot y$', and an identity element we'll dub '$\dot{1}$'.

Now note that the arrow $e\colon 1 \to G$ of the internal group must also pick out a distinguished element of $\dot{G}$, call it '$\tilde{1}$', an identity for m.

Now, by assumption 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 the operation $\times$ is defined component-wise 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 $\tilde{1}$ 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

$$\dot{1} = \dot{1} \cdot \dot{1} = (\tilde{1} \star \dot{1}) \cdot (\dot{1} \star \tilde{1}) = (\tilde{1} \cdot \dot{1}) \star (\dot{1} \cdot \tilde{1}) = \tilde{1} \star \tilde{1} = \tilde{1}$$

We can therefore just write 1 for the shared unit, and now show secondly that

$$\begin{aligned} x \cdot y &= (x \star 1) \cdot (1 \star y) = (x \cdot 1) \star (1 \cdot y) = x \star y \\ &= (1 \cdot x) \star (y \cdot 1) = (1 \star y) \cdot (x \star 1) = y \cdot x. \end{aligned}$$

By the end of the first line we have shown that $x \cdot y = x \star y$; so the internal group's arrow m thought of as a binary function is the same as G's own group operation. And by the end of the second line we have shown that $x \cdot y = y \cdot x$, so G's own group operation commutes so G is abelian. Hence the internal group's operation m commutes, and so the internal group is indeed also abelian in the obvious sense. □

Not every group in the category Grp is abelian. Every internal group of that category *is* abelian. So not every group in Grp counts as (or is equivalent to) an internal group of that category. Does that still seem strange? It shouldn't when we recall that the group operations of members of Grp are any suitable binary operations satisfying certain conditions, while the group operations of internal groups are, specifically, group homomorphisms, which (as we have now seen) imposes extra constraints.

13.4 The story continues . . .

(a) We can continue the story, defining further group-theoretic notions in categorial terms. For a start, we can categorially define the idea of a homomorphism between internal groups in a category.

Suppose (G, m, e, i) and (G', m', e', i') are internal groups in Set. Then a homomorphism between them is a 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 if and only if the following three diagrams commute:

$$\begin{array}{ccc} G \times G & \xrightarrow{h \times h} & G' \times G' \\ {\scriptstyle m}\downarrow & & \downarrow{\scriptstyle m'} \\ G & \xrightarrow{h} & G' \end{array} \qquad \begin{array}{ccc} & 1 & \\ {\scriptstyle e}\swarrow & & \searrow{\scriptstyle e'} \\ G & \xrightarrow{h} & G' \end{array} \qquad \begin{array}{ccc} G & \xrightarrow{h} & G' \\ {\scriptstyle i}\downarrow & & \downarrow{\scriptstyle i'} \\ G & \xrightarrow{h} & G' \end{array}$$

So we can in this way begin to recast core group-theoretic ideas into a categorial framework. And the richer the category we work in, the more group theory we can do: for example, if our category also has the resources for constructing quotients (see the next chapter), then we can get quotient groups.

(b) The explorations we have gestured towards here can be continued in various directions. We can similarly define other kinds of algebraic objects and their morphisms within categories. And noting that we can now define group-objects and group-homomorphisms inside a given category, we could go on to categorially define whole *categories* of groups living in other categories. However, things do begin to get pretty abstract (and not in a way that is particularly helpful for us at this stage in the proceedings). So let's move on.

14 Quotients, pre-categorially

Forming product widgets is a ubiquitous procedure in pre-categorial maths. So too is forming new widgets by quotienting old widgets by suitable equivalence relations. The general idea, roughly put, is that we start with a given widget together with an equivalence relation on its objects, where equivalent objects behave the same way in the widget. We then, as it were, 'collapse together' equivalent objects into a single object, and we arrive at a new widget formed from these 'collapsed' objects. In §2.3(c) we saw how this general idea begins to play out in a particular but typical case, when we form a new group by quotienting an old one by a suitable congruence relation.

In this chapter, we will think a little more about how equivalence relations can be generated and about what is involved in quotienting by an equivalence relation. Then, as with our story about products, the pre-categorial reflections will then give a natural shape to our categorial account in the next chapter.

14.1 Equivalence relations

(a) Let's start by defining some terminology and notation:[1]

Definition 61. The function $k\colon \mathrm{Y} \to \mathrm{Z}$ *respects the relation* R defined over the objects Y iff, whenever yRy', $k(y) = k(y')$.

Definition 62. Given any relation R, then $\overline{\overline{R}}$ is the reflexive, symmetric, transitive closure of R, i.e. it is the smallest equivalence relation containing R.

And here is an immediate lemma for future use:

Theorem 56. *If* $k\colon \mathrm{Y} \to \mathrm{Z}$ *respects the relation* R*, it also respects its reflexive, symmetric, transitive closure* $\overline{\overline{R}}$*.*

Proof. First, if we extend a relation R by making it reflexive, we make true all instances of the form yRy; but correspondingly we of course always have $k(y) = k(y)$, so k still respects R.

[1] I continue to use non-italic letters like 'X' and 'Y' as plural schematic variables, standing in for some objects (usually more than one!). That's for consistency with what's gone before, and to re-emphasize that that the idea of taking quotients is *not* essentially set-theoretic.

Second, if we extend a relation to make it symmetric, we add instances of the form $y'Ry$ whenever we previously had yRy'. But if k respected the original R by making $k(y) = k(y')$, we of course also have $k(y') = k(y)$, so k still respects the extended R.

Third, if yRy' and $y'Ry''$ and k respects R, then we have $k(y) = k(y')$ and $k(y') = k(y'')$, and hence of course $k(y) = k(y'')$. So if we make extend R by making it transitive, i.e. by adding yRy'' whenever yRy' and $y'Ry''$, k will still respect R.

Repeated operations of extending R in these ways will give us $\overline{\overline{R}}$. So if k respects R, it respects $\overline{\overline{R}}$. □

(b) We want ideas which we can carry over to the categorial context where we emphasize the role of arrows/functions: so let's think further about the way that equivalence relations on some objects Y can relate to functions to and from Y.

Let's have some more terminology:

Definition 63. Take any function $k\colon \mathrm{Y} \to \mathrm{Z}$. Let E_k be the equivalence relation on the objects Y such that yE_ky' if and only if $k(y) = k(y')$. E_k is then said to be the *equivalence kernel* of k.

Definition 64. Take any pair of functions $f, g\colon \mathrm{X} \to \mathrm{Y}$. Let P_{fg} be the relation on the objects Y such that $yP_{fg}y'$ iff there is some $x \varepsilon X$ such that $f(x) = y$ and $g(x) = y'$. Then $\overline{\overline{P_{fg}}}$ is the reflexive, symmetric, transitive closure of P_{fg}: I'll call this the *equivalence projection* of the pair f and g.

We can note, again for future use, that every equivalence relation $\sim$ on Y is the equivalence projection of some parallel functions into Y. Just take some pairing scheme for Y with Y, and let X comprise the pairs $\langle y, y'\rangle$ where $y \sim y'$. Then let f send a pair $\langle y, y'\rangle$ to y and g send $\langle y, y'\rangle$ to y'. Then of course $\sim$ is the equivalence projection of f, g.

(c) Suppose next that we have both a parallel pair of functions f, g into the objects Y, and also another function k onwards from Y. In other words, suppose we have a situation which we can depict like this, using a sort of 'fork':

$$\mathrm{X} \underset{g}{\overset{f}{\rightrightarrows}} \mathrm{Y} \xrightarrow{\ k\ } \mathrm{Z}$$

We can ask: what does it take for equivalence projection of f and g to be in fact none other than the equivalence kernel of k?

Certainly, this much is required: in the notation of our last definition, when $yP_{fg}y'$, then yE_ky'. Hence, for any $x \varepsilon$ X, $k(f(x)) = k(g(x))$. So we need $k \circ f = k \circ g$; in other words, we need our fork diagram to commute.[2] Or putting it another way, k needs to respect the relation P_{fg}.

That's a necessary condition. It ensures that the equivalence relation E_k contains P_{fg} and hence contains $\overline{\overline{P_{fg}}}$, the smallest equivalence relation containing

[2] Commute, that is to say, in the sense of our tweaked Defn. 19*.

P_{fg}. But plainly the condition isn't sufficient. Suppose, for example, k sends each and every Y-object to the same target. Then entirely trivially, $k \circ f = k \circ g$. But in this case, the corresponding E_k is the *largest* equivalence relation on Y, relating any two Y-objects: and in the general case this won't be the same as the projected equivalence of f and g.

So the obvious next question to ask is: what happens when we keep the prongs f and g of our fork fixed, but vary the handle k while still ensuring we get a commuting diagram? The resulting equivalence kernel E_k will then vary. And what we are after is the limiting case where the equivalence kernel is the *smallest* it can be and so is in fact $\overline{\overline{P_{fg}}}$. When does this happen?

We'll put this question on hold for the moment (though it is a nice challenge to pause to think about), and first return to consider ...

14.2 Quotient schemes again

(a) Suppose we have an equivalence relation on some objects Y, and want to 'collapse' equivalent objects. We need some objects Q (which may or may not be some of Y) to play the role of the 'collapsed' objects, and a function $q\colon \mathrm{Y} \to \mathrm{Q}$ which sends equivalent objects among Y to the same target object. But we don't want q to be too indiscriminate and to collapse non-equivalent objects. And we also want to avoid redundant complications, so Q should only include the necessary target objects.

Wrapping these desiderata into a definition, let's say

Definition 5* Given some objects Y and an equivalence relation $\sim$ defined over them, then the objects Q and the function $q\colon \mathrm{Y} \to \mathrm{Q}$ provide *a scheme* (Q, q) *for quotienting* Y *by* $\sim$ just when:

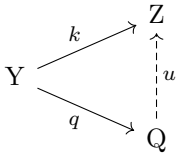
(i) if $y \sim y'$ then $q(y) = q(y')$;
(ii) if $q(y) = q(y')$ then $y \sim y'$; and
(iii) q is surjective, so for any $o \,\varepsilon\, \mathrm{Q}$, there is some $y \,\varepsilon\, \mathrm{Y}$ such that $o = q(y)$. $\triangle$

Notation apart, this is of course the very same idea of a quotient scheme as introduced in Defn. 5 right back in §2.3.

As we noted before, the canonical example of a scheme for quotienting Y by $\sim$ is provided by taking Q to be $\sim$-equivalence classes formed from Y, and q to be the function that sends an object among Y to the equivalence class it belongs to. But as we also emphasized, this is just one way of forming a quotient scheme. Exactly as with a pairing scheme, what actually matters about a quotient scheme is that it provides *some* (Q, q) which – viewed 'externally' – work together as described in Defn. 5*: the particular 'internal' nature of the quotient-objects Q is not of the essence. There is, in particular, no requirement that quotient-objects really *are* classes.

(b) Clause (i) of our definition tells us that q respects $\sim$. Clauses (ii) and (iii) together then tell us that q is a limiting case among the functions from Y which respects $\sim$. This thought can be captured by the following theorem:

Theorem 57. $(\mathrm{Q}, q\colon \mathrm{Y} \to \mathrm{Q})$ *is a scheme for quotienting* Y *by the equivalence relation* $\sim$ *if and only if (1)* q *respects* $\sim$ *and (2) for any function* $k\colon \mathrm{Y} \to \mathrm{Z}$ *which respects* $\sim$*, there exists a unique* $u\colon \mathrm{Q} \to \mathrm{Z}$ *such that* $k = u \circ q$*, i.e. such that this commutes:*[3]



Proof ('only if'). Assume that (Q, q) form a scheme for quotienting Y by $\sim$. By condition (iii) in Defn. 5*, every object among Q is $q(y)$, for some y from Y. So we can define a function $u\colon \mathrm{Q} \to \mathrm{Z}$ by saying that, for each y, it sends $q(y)$ to $k(y)$.

We need to check that this does well-define a function. But by conditions (i) and (ii), and the assumption that k respects $\sim$, we can't have $q(y) = q(y')$ without having $k(y) = k(y')$.

And this is evidently the unique u such that $k = u \circ q$. □

Proof ('if'). We need to show that if (2) holds, so too do conditions (ii) and (iii) in Defn. 5*.

For (ii), suppose that q were to send objects in two different $\sim$-partitions of Y to the same q-value, while k (as could be the case) always sends objects in different $\sim$-partitions to different k-values. Then no u will make $k = u \circ q$. So the existence condition in (2) means q *can't* send objects in two different $\sim$-partitions to the same q-value. So (ii) is satisfied.

For (iii), suppose that, as well as all the requisite q-images of objects from Y, there are also one or more junk objects among Q. Then $k = u \circ q$ for any u which sends the q-values of objects in Y to their k-values but sends the junk objects wherever you like; so u wouldn't then in general be unique. Contraposing, if the uniqueness condition in (2) holds, so does (iii). □

(c) As with pairing schemes, we can similarly show that different schemes for quotienting Y by the equivalence relation $\sim$ will all 'look the same'. More carefully, we have

Theorem 58. *If* (Q, q) *and* (Q', q') *are both schemes for quotienting* Y *by* $\sim$*, then there is a unique bijection* $f\colon \mathrm{Q} \to \mathrm{Q}'$ *which preserves the way objects from* Y *are 'collapsed together' by the schemes, i.e. such that* $q' = f \circ q$*. And* $\sim$ *will be the equivalence kernel of both* q *and* q'*.*

Proof. Since q is surjective onto Q, every object among Q is some $q(y)$ for y among Y. Likewise, every object among Q' is some $q'(y)$. So define $f\colon \mathrm{Q} \to \mathrm{Q}'$ as sending $q(y)$ to $q'(y)$ for each y among Y. It is straightforward to check that

[3] Yes, we are for the moment still working pre-categorially; but we can of course still helpfully use diagrams in this chapter!

this is our needed bijection. And the concluding claim is simply a consequence of our definitions. □

(d) A minor point. We introduced *pairing schemes* as combining four items $(\mathrm{O}, pr, \pi_1, \pi_2)$. And we suggested that don't want to identify the *products* delivered by a pairing scheme simply with the pair-objects O; because it is only when combined with their projection functions that some pair-objects will play the role we need. But we can usefully treat products as just combinations $(\mathrm{O}, \pi_1, \pi_2)$ satisfying a certain condition.

Now we have *quotient schemes* defined as combining two items (Q, q). Again, we don't want to officially identify the *quotients* delivered by a quotient scheme as just the objects Q, because without a function q to match items to their quotient-objects, those objects won't play the role we need. But then there is no pruning to be done – so we might as well treat quotients as simply the combinations (Q, q) satisfying the right conditions.

14.3 A key result about quotients to carry forward

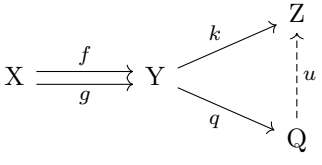
We can return now to the question we left hanging at the end of §14.1. Given a commuting fork of the shape

$$\mathrm{X} \underset{g}{\overset{f}{\rightrightarrows}} \mathrm{Y} \xrightarrow{k} \mathrm{Z}$$

when is the equivalence projection of f and g the same as the equivalence kernel of k?

And here is the now-probably-predictable answer:

Theorem 59. *Suppose that* $\mathrm{X} \underset{g}{\overset{f}{\rightrightarrows}} \mathrm{Y} \xrightarrow{q} \mathrm{Q}$ *commutes, and for each commuting fork* $\mathrm{X} \underset{g}{\overset{f}{\rightrightarrows}} \mathrm{Y} \xrightarrow{k} \mathrm{Z}$ *there is a unique* $u\colon \mathrm{Q} \to \mathrm{Z}$ *such that the following whole diagram commutes:*



Then (Q, q) *is a scheme for quotienting* Y *by the equivalence projection of* f *and* g*, and this equivalence projection is also the equivalence kernel of* q*.*

Proof. Since the forks commute, we know that q and k respect the relation P_{fg} and hence by Theorem 56 respect the equivalence relation $\overline{\overline{P_{fg}}}$. Hence by Theorem 57, (Q, q) is a scheme for quotienting Y by $\overline{\overline{P_{fg}}}$. But then the equivalence kernel of q is indeed $\overline{\overline{P_{fg}}}$ by Theorem 58. So we are quickly done! □

We can therefore relate claims about quotients with claims about limiting cases of commuting forks. And this looks to be *exactly* the kind of thing we can directly carry over into a categorial setting. Let's do that in the next chapter.

15 Equalizers and co-equalizers

The informal story that we told in the last chapter suggests that we can treat quotients in a categorial setting by invoking some particular commuting forks with unique arrows *from* them. In this way, the construction will be analogous to that for initial objects and coproducts.

However, starting out now on our official story, it is conventional to begin with the dual case. So we will in fact first look at commuting fork diagrams with arrows in the opposite direction, and then pick out special commuting forks with unique arrows going *to* them.

15.1 Forks and equalizers defined

For convenience, we'll start calling commuting fork diagrams simply 'forks' for short. With the direction of arrows reversed, then,

Definition 65. A *fork* (from W through X to Y) consists of an arrow $k\colon W \to X$ together with parallel arrows $f\colon X \to Y$ and $g\colon X \to Y$, such that $f \circ k = g \circ k$. In other words, to count as a fork, the resulting diagram must commute:[1]

$$W \xrightarrow{\ k\ } X \underset{g}{\overset{f}{\rightrightarrows}} Y \qquad \triangle$$

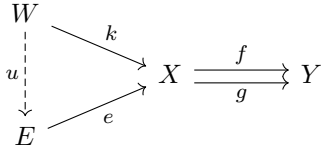
Now, when we were talking about products, we defined a product wedge from O to X and Y as a limiting case. It's a wedge such that any other wedge from W to X and Y uniquely 'factors through' it (in the sense of §10.3). Our next move is exactly analogous: we will define an equalizing fork starting from E and with the prongs $f, g\colon X \to Y$ as another limiting case. It's a fork such that any other fork sharing the same prongs f, g again uniquely 'factors through' it (in a closely related sense).

To spell that out:

Definition 66. Let $f, g\colon X \to Y$ be a pair of parallel arrows in the category C. Then the object E and arrow $e\colon E \to X$ form an *equalizer* (E, e) in C for those arrows if and only if (1) $f \circ e = g \circ e$ (making $E \xrightarrow{\ e\ } X \underset{g}{\overset{f}{\rightrightarrows}} Y$ a fork),

[1] Must commute, need I say again, in the sense of our tweaked Defn. 19*.

and (2) for any fork $W \xrightarrow{k} X \underset{g}{\overset{f}{\rightrightarrows}} Y$ there is a unique mediating arrow $u\colon W \to E$ making the following diagram commute:

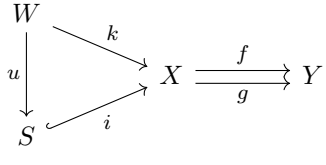


△

15.2 Examples of equalizers

(1) Suppose in Set we have parallel arrows $f, g\colon X \to Y$ (in this case, the arrows are straightforwardly functions). Now consider the subset $S \subseteq X$ which is the set of $x \in X$ such that $fx = gx$. Let $i\colon S \hookrightarrow X$ be the simple inclusion map which sends an element of S to the very same element of X.[2] By construction, $f \circ i = g \circ i$. So $S \xhookrightarrow{i} X \underset{g}{\overset{f}{\rightrightarrows}} Y$ is indeed a fork.

Claim: (S, i) so defined is an equalizer for f and g. Why? Well, let's suppose $W \xrightarrow{k} X \underset{g}{\overset{f}{\rightrightarrows}} Y$ is another fork in Set. Can we make this diagram commute?



By the assumption that the top fork *is* a commuting fork, we know that for each $w \in W$, $f(k(w)) = g(k(w))$. Hence the k images of objects in W must live in $S \subseteq X$. Hence if we define the arrow $u\colon W \to S$ to agree with $k\colon W \to X$ for all $w \in W$, this will make the whole diagram commute. Moreover this is the unique possibility: in order for the diagram to commute, we need $k = i \circ u$, and since the inclusion i doesn't alter the value in X we reach, k and u must agree on all inputs (the functions just have different co-domains).

Note that, since the described construction is always available, Set has an equalizer for any pair of parallel arrows.

(2) As you would probably predict, equalizers in concrete categories whose objects are sets-with-structure behave similarly.

Consider the category Mon. Given a pair of monoid homomorphisms $(X, *, e_X) \underset{g}{\overset{f}{\rightrightarrows}} (Y, \star, e_Y)$, take the subset E of X on which the functions

[2] A hook at the start of an arrow is conventionally used to indicate an inclusion function.

agree. Evidently E must contain the identity element of X (since f and g agree on this element: being homomorphisms, both have to send e_X to the element e_Y). And suppose $a, b \in E$: then $f(a * b) = f(a) \star f(b) = g(a) \star g(b) = g(a * b)$, which means that E is closed under products of members.

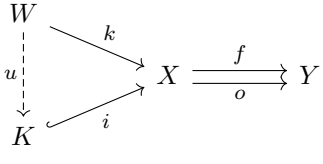
So take E together with the monoid operation from $(X, *, e_X)$ restricted to members of E. Then $(E, *, e_X)$ is a monoid – for the shared identity element still behaves as an identity, E is closed under the operation, and the operation is still associative. And if we take $(E, *, e_X)$ and equip it with the injection homomorphism into $(X, *e, 1_X)$, this will evidently give us an equalizer for f and g.

(3) Next, take Top. What is the equalizer for a pair of continuous maps $X \underset{g}{\overset{f}{\rightrightarrows}} Y$? 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.

(4) A more interesting case. Suppose we are in Grp and have a group homomorphism, $f\colon X \to Y$. There is automatically also a homomorphism $o\colon X \to Y$ which sends any element of the group X to the identity element in Y (this can be defined as the composite $X \to 1 \to Y$ of the only possible homomorphisms, where 1 is the one-object group which is both initial and terminal in the category of groups). 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 $i\colon K \hookrightarrow X$ be the inclusion map (trivially a homomorphism). Then $K \overset{i}{\hookrightarrow} X \underset{o}{\overset{f}{\rightrightarrows}} Y$ is a fork since $f \circ i = o \circ i$.

Let $W \overset{k}{\longrightarrow} X \underset{o}{\overset{f}{\rightrightarrows}} Y$ be another fork. Now, $o \circ k$ sends every element of W to the unit of Y. By assumption, $f \circ k = o \circ k$, so $f \circ k$ also sends every element of W to the unit of Y; hence $k\colon W \to X$ must send any element of W to some element which lives in f's kernel K. Let $u\colon W \to K$ agree with $k\colon W \to X$ on all arguments. Then the following commutes:



And evidently u is the only possible homomorphism which will make the diagram commute.

In sum, then, the equalizer of f and o is (up to isomorphism) 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. Which is rather nice.

15.3 Uniqueness up to unique isomorphism

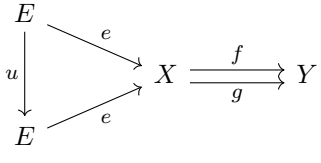
(a) A quick terminological aside before proceeding further. I've defined an equalizer as an object E equipped with an arrow whose source is E, satisfying certain conditions. Since fixing the arrow fixes its source, we could without loss of information officially define an equalizer to be just the relevant arrow. Many do this. Nothing hangs on the choice.

(b) To continue. Just as products are unique up to unique isomorphism, equalizers are too:

Theorem 60. *If (E, e) and (E', e') are both equalizers for $X \overset{f}{\underset{g}{\rightrightarrows}} Y$, then there is a unique isomorphism $v\colon E \xrightarrow{\sim} E'$ commuting with the equalizing arrows, i.e. such that $e' \circ v = e$.*

Plodding proof from first principles. We can use an argument that goes along very similar lines to the plodding proof we used to prove the uniqueness of products. This is of course no accident, given the similarity of the definitions of products and equalizers via a universal mapping property. (Challenge: pause to give this similar line of argument!)

OK, if you want the details, assume (E, e) equalizes f and g. Then a fork from (E, e) on through f and g factors uniquely through *itself*, via some the mediating arrow u, meaning that this commutes:

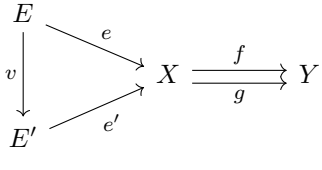


And obviously, this u (being unique) must in fact be equal to 1_E.

But now note that for any u such that $e \circ u = e$ this diagram also commutes. Which gives us a first result

(i) If (E, e) is an equalizer, then $e \circ u = e$ implies $u = 1_E$.

Now suppose (E', e') is also an equalizer for f and g. Then the fork starting (E, e) must factor uniquely through this new equalizer via a (unique) mediating $v\colon E \to E'$ such that $e' \circ v = e$:



Similarly, swapping (E, e) and (E', e'), there is a unique w such that $e \circ w = e'$. Therefore $e \circ w \circ v = e$, and hence by (i) it follows that (with v and w as defined)

(ii) $w \circ v = 1_E$.

Since everything is symmetric in (E, e) and (E', e'), an exactly similar argument shows that

(iii) $v \circ w = 1'_E$.

Which gives the unique v a two-sided inverse – i.e. as we wanted to show to complete the proof

(iv) v is an isomorphism. □

(c) We now quickly note that, as with products (see Defn. 48), we can give an alternative definition which defines an equalizer as a terminal object in a suitable category.

First we say

Definition 67. Given a category C and parallel arrows $f, g\colon X \to Y$, then the derived category of forks $\mathsf{C}_{f\|g}$ has as objects all forks $W \xrightarrow{k} X \underset{g}{\overset{f}{\rightrightarrows}} Y$.

And an arrow from $W \xrightarrow{k} \cdots$ to $W' \xrightarrow{k'} \cdots$ in $\mathsf{C}_{f\|g}$ is a C-arrow $v\colon W \to W'$ such that the resulting triangle commutes: i.e. such that $k = k' \circ v$.[3]

The identity arrow in $\mathsf{C}_{f\|g}$ on the fork $W \xrightarrow{k} \cdots$ is the identity arrow 1_W in C; and the composition of arrows in $\mathsf{C}_{f\|g}$ is defined as the composition of the arrows as they feature in C. △

It is easily checked that this does define a category,[4] and that our definition of an equalizer then comes to the following:

Definition 66*. An equalizer of $f, g\colon X \to Y$ in C is some (E, e), where E is a C-object, and e is a C-arrow $E \to X$, such that the fork $E \xrightarrow{e} X \underset{g}{\overset{f}{\rightrightarrows}} Y$ is terminal in $\mathsf{C}_{f\|g}$. △

But this redefinition immediately gives us

A slicker proof of Theorem 60. (E, e) and (E', e') are both terminal objects in the fork category $\mathsf{C}_{f\|g}$. So by Theorem 26 there is a unique $\mathsf{C}_{f\|g}$-isomorphism j between them. But, by definition, this has to be a C-arrow $j\colon E \xrightarrow{\sim} E'$ such that there is an arrow $j'\colon E' \xrightarrow{\sim} E$, where $j' \circ j = 1_E$ and $j \circ j' = 1_{E'}$. So j has to be an C-isomorphism too. □

[3] Or rather, strictly speaking, we should take the arrow to be the whole commuting triangle, as we did for slice categories and for the same reason: see §6.3(c). But we won't fuss about this.

[4] Modulo that last footnoted tweak.

15.4 A few easy challenges about equalizers

Let's note the following simple results (we'll need one of them later).

Theorem 61. *A pre-order category has an equalizer for any parallel arrows.*

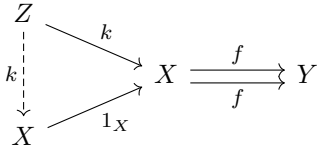
Theorem 62. *A group-considered-as-a-category has no equalizers for any distinct parallel arrows.*

Theorem 63. *If (E, e) is an equalizer, then e is a monomorphism. And if e is also epic, then it is an isomorphism.*

Theorem 64. *In* Set, *any subset S of a set X together with the inclusion map from S to X is an equalizer for a certain pair of parallel arrows from X.*

Apart perhaps from the last one, it is in fact rather a stretch to present these theorems as 'challenges' to prove! But do pause to derive the results, before reading on.

Proof: A pre-order category has all equalizers. Recall Defn. 16: a pre-order category has at most one arrow between any two objects. So we only have arrows $f, g\colon X \to Y$ when $f = g$. But then it is easy to see that $(X, 1_X)$ equalizes f with itself, by thinking about the commuting diagram



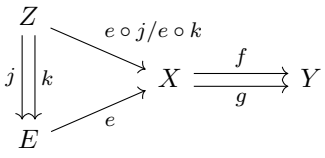
Since 1_X comes for free with any category containing X, the equalizer $(X, 1_X)$ always exists, as claimed. □

Proof: A category without any equalizers for distinct arrows. Recall §7.8: given a group $(\mathrm{G}, *, e)$, we can define 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 G is defined as group-multiplication $*$.

Now, take distinct objects $g, h\ \varepsilon$ G. Then there will be no x such that $g * x = h * x$, or else we would have $g * x * x^{-1} = h * x * x^{-1}$ and hence $g = h$ after all. So correspondingly, in G, given distinct parallel arrows $g, h\colon \bullet \to \bullet$, there is no x such that $g \circ x = h \circ x$, and hence those arrows can't have an equalizer. □

Proof: Equalizing arrows are monic. Suppose (E, e) equalizes $X \overset{f}{\underset{g}{\rightrightarrows}} Y$, and also suppose $e \circ j = e \circ k$.

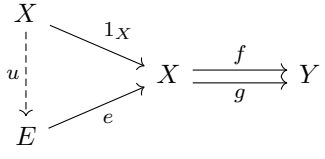
For the second supposition to make sense, j and k must be parallel arrows from some Z to E. And then the following diagram commutes,



Therefore $Z \xrightarrow{e \circ j/e \circ k} X \underset{g}{\overset{f}{\rightrightarrows}} Y$ 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. □

Proof: An epic equalizer is an isomorphism. We know that epic monos need not in general be isomorphisms: but they are in the special case when the mono is an equalizing arrow.

Assume again that (E, e) equalizes $X \underset{g}{\overset{f}{\rightrightarrows}} Y$, so that $f \circ e = g \circ e$. So if e is epic, it follows that $f = g$. Then consider the following diagram



We know that the top fork commutes and uniquely factorizes through the equalizer, i.e. 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 a two-sided inverse. Therefore e is an isomorphism. □

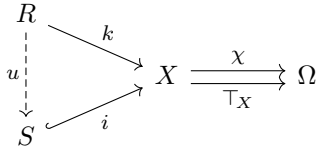
Our first example in §15.2 showed that in Set the equalizer of parallel maps $f, g\colon X \to Y$ will be provided by a suitable subset of X together with the inclusion map from that subset to X. Our next result, Theorem 64, shows the converse:

Proof: Subsets as equalizers. We are working in Set, and we'll use a familiar device for thinking about subsets. So take a suitable two-object set we'll call Ω (whose members we might suggestively dub *true* and *false*). Then, a very familiar idea, a subset $S \subseteq X$ has an associated characteristic function $\chi\colon X \to \Omega$, where $\chi(x) = \mathit{true}$ if and only if $x \in S$.

Now compare this with the boring function we'll dub $\top_X\colon X \to \Omega$ which indiscriminately sends *everything* in X to *true*. If $i\colon S \hookrightarrow X$ is the simple inclusion function, then

$$S \xhookrightarrow{i} X \underset{\top_X}{\overset{\chi}{\rightrightarrows}} \Omega$$

is a commuting fork, and (S, i) is an equalizer for $X \underset{\top_X}{\overset{\chi}{\rightrightarrows}} \Omega$. For consider the following diagram:



If we have a commuting fork at the top, the k-image of R must be contained in S (why?). In which case the unique way of making the whole diagram commute is to make u agree with k for all inputs.

As claimed, then, any subset S of a set X together with the inclusion map from S to X is an equalizer for a pair of parallel arrows from X. □

15.5 Co-forks and co-equalizers defined

Now we dualize to get the notion of a co-equalizer. We simply reverse the arrows on forks as defined Defn. 65 in §15.1 (and for convenience, swap back the labels 'X' and 'Y'). So we return back again to the sort of forks informally introduced in the last chapter.

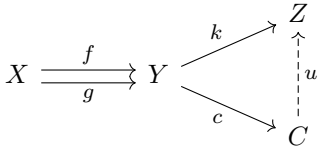
However, for clarity, we will now officially call commuting forks of that earlier kind 'co-forks', as in:

Definition 68. A *co-fork* (from X through Y to Z) consists of parallel arrows $f\colon X \to Y$, $g\colon X \to Y$ together with an arrow $k\colon Y \to Z$, such that $k \circ f = k \circ g$. In other words, to form a co-fork, this corresponding diagram must commute:

$$X \underset{g}{\overset{f}{\rightrightarrows}} Y \xrightarrow{k} Z \qquad \triangle$$

Dualizing Defn. 66, our definition of equalizers, we then get

Definition 69. Let $f, g\colon X \to Y$ be a pair of parallel arrows in the category C. Then the object C and arrow $c\colon Y \to C$ form a *co-equalizer* (C, c) in C for those arrows iff $c \circ f = c \circ g$ (making $X \underset{g}{\overset{f}{\rightrightarrows}} Y \xrightarrow{c} C$ a co-fork), and for any co-fork $X \underset{g}{\overset{f}{\rightrightarrows}} Y \xrightarrow{k} Z$ there is a unique mediating arrow $u\colon C \to Z$ making the following diagram commute:



△

To stress the duality, in an equalizer (E, e) the object E is the *source* of the arrow e, in a co-equalizer (C, c) the object C is the *target* of the arrow c.

We need not pause to spell out the dual argument that, like equalizers, co-equalizers are unique up to a unique isomorphism. And we can also leave it as an exercise to show other dual results, such as that the arrows of co-equalizers are epic.

15.6 Examples of co-equalizers

(a) Let's immediately turn to consider the key example.

What, then, do co-equalizers do in a concrete category like Set? Important though the question is, we can be very brief, since we did all the necessary ground-work in the last chapter. Simply trade in plural talk about some items Q for singular talk about some appropriate 'object' Q in a category which collects them together.

So, take a pair of parallel arrows $f, g\colon X \to Y$ in Set. These set up a relation P_{fg} on Y, where $yP_{fg}y'$ if and only if there is some $x \in X$ such that $y = f(x)$ and $y' = g(x)$. That in turns gives us an equivalence relation $\overline{\overline{P_{fg}}}$ on Y (the closure of P_{fg}). Then, Theorem 59 tells us that if we have

$$X \underset{g}{\overset{f}{\rightrightarrows}} Y \xrightarrow{\ q\ } Q$$

with (Q, q) a co-equalizer for f and g, then (Q, q) provides a scheme for quotienting Y by $\overline{\overline{P_{fg}}}$. It is also a scheme for quotienting Y by the equivalence kernel of q (the relation that holds between y and y' if $q(y) = q(y')$ – because that is the *same* relation as $\overline{\overline{P_{fg}}}$).

Conversely. if we form the set Q of equivalence classes for the relation $\overline{\overline{P_{fg}}}$ in the conventional way, then Q equipped with the map that sends an object in Y to its equivalence class in Q will provide a categorial co-equalizer (Q, q). And since we can form equivalence classes ad libitum in a standard universe of sets, that tells us that the corresponding category Set will have a co-equalizer for any pair of parallel arrows.

(b) Predictably enough, we get parallel results in other categories whose objects can be thought of as sets-equipped-with-structure. Take Grp, for example, and suppose (Q, q) is a co-equalizer for the parallel group homomorphisms $f, g\colon X \to Y$. Then $q\colon Y \to Q$ is a group homomorphism, and if we put $y_1 \sim y_2$ iff $q(y_1) = q(y_2)$, then by Theorem 8, $\sim$ is a congruence on the group Y, and Q will be a quotient of Y with respect to that congruence relation.

We will leave thinking about e.g. co-equalizers in e.g. Top until Part II, where we have more apparatus available to smooth the discussion.

16 Exponentials

We have been thinking about how we form products, quotients, and a few other constructs, in 'ordinary mathematics'. And we've seen that what matters about product-widgets, quotient-widgets, and the like, is not their 'internal' make up, but how they 'externally' map to and from other widgets. This is the key insight which gets picked up in the categorial treatment of products, quotients, etc.

Now we move on to consider another kind of construction, namely exponentials. Again I'll start with some informal remarks: then we'll see how things play out in a categorial setting.

16.1 Instead of binary functions, again

(a) I have stressed more than once that in categories where arrows are functions, they are always monadic functions. So as we asked before, how can we accommodate binary functions?

In fact, we have a couple of already-familiar frameworks which manage to do without genuine multi-place functions by providing workable substitutes. The first we've met before:

(1) The default set-theoretic procedure is to trade in an underlying binary function $\underline{f}\colon A, B \to C$ for a related official unary function $f\colon A \times B \to C$.

But now let's note that varieties of type theory usually deal with two-place functions in a quite different way.

To illustrate: addition – naively a binary function – 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 $N \to N$). So we now get from two numbers as input to a numerical output in two steps. We feed the first number to a function of type $N \to (N \to N)$, which delivers another function of type $N \to N$ as output; and then we can feed the second number to this second function.

This so-called 'currying' manoeuvre from type theory[1] is of course also perfectly adequate for certain formal purposes, and we can borrow the same device

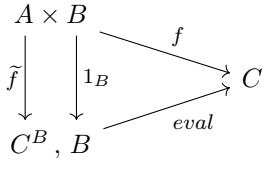
[1]The trick of replacing the evaluation of a function that takes multiple arguments by the evaluation of a sequence of unary functions was developed by Haskell Curry: hence 'currying'. Moses Schönfinkel had the idea first, but somehow 'Schönfinkeling' never caught on!

to use in a set-theoretic framework. We can do the work of a binary function $\underline{f}\colon A, B \to C$ by a unary function which sends a member of A to a particular function from B to C. And where do functions from B to C live? In the 'exponential' C^B. Hence, in set-theoretic terms,

(2) Currying is essentially a matter of trading in a binary function $\underline{f}\colon A, B \to C$ for a related unary function $\widetilde{f}\colon A \to C^B$, i.e. the function which sends a to the function f_a (where f_a is the unary function whose value for input b is $\underline{f}(a, b)$).[2]

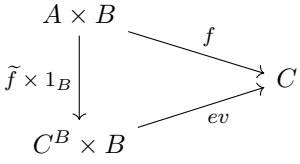
(b) The obvious next question, is how do these two substitutes f and $\widetilde{f}$ for the underlying binary function $\underline{f}$ fit together?

At a first shot, we want something like the following informal diagram to commute, where *eval* is a binary function that takes a function living in C^B (i.e. a function from B to C) and evaluates it for a given argument in B.



In other words, taking a pair $\langle a, b\rangle$ from $A \times B$, we can (1) use that pair as input to f. Or (2) we can use $\widetilde{f}$ to send a to a function $f_a\colon B \to C$ while carrying along b unchanged: and then *eval* takes f_a and b as its two inputs and outputs $f_a(b)$. By either route, we get the same result.

Now, that first shot gives us the core idea, except that it leaves us with a binary function *eval* still in play. So let's slightly revise. Let *ev* now be a unary function which takes an ordered pair of a function living in C^B and an argument from B, and still evaluates that function for that argument. Then, as a second shot, we'll say we need the following to commute:



where $\widetilde{f} \times 1_B$ acts component-wise on $A \times B$, sending a pair $\langle a, b\rangle$ to $\langle f_a, b\rangle$, and *ev* takes the pair $\langle f_a, b\rangle$ and returns the value $f_a(b)$. Note: given f and given *ev* with its intended meaning, $\widetilde{f}$ will be the *unique* function from A to C^B which makes the diagram commute.

[2] You might find an alternative notation helpful. Suppose we use '$f(\cdot, \cdot)$' to explicitly mark how the function is waiting to be applied to two terms. Then similarly, instead of 'f_a' we could write '$f(a, \cdot)$', now marking how this function is to be applied to a single term. So $\widetilde{f}$ sends a to $f(a, \cdot)$.

16.2 Exponentials in categories

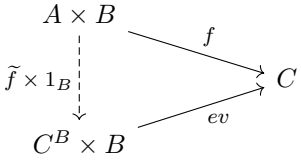
And now everything is nicely set up to carry over smoothly to our categorial framework.

We don't have native binary morphisms in category theory; we don't have binary arrows with two sources, $\underline{f}\colon A, B \to C$. But, as we've already seen, once we are working in a category which has products, we can use a version of the first set-theoretic trick and deploy corresponding arrows like $f\colon A \times B \to C$.

And we can also deploy an analogue of the currying trick, where we trade in our binary $\underline{f}\colon A, B \to C$ for the unary $\widetilde{f}\colon A \to C^B$. Or at least, we can do this if we have suitable exponential objects C^B and corresponding evaluation arrows ev available in our category. But which objects and arrows would these be? Given the discussion in the last section (when we were in effect looking inside the category Set), this is evidently the general story we want:

Definition 70. Assume C is a category with binary products. Then (C^B, ev), where C^B is an object and ev is an arrow $C^B \times B \to C$, forms an *exponential of C by B* iff the following holds: for every object A and arrow $f\colon A \times B \to C$, there is a *unique* arrow $\widetilde{f}\colon A \to C^B$ making the following commute:

(Exp)



△

Here, all the objects and arrows are of course living in C. The product arrow $\widetilde{f} \times 1_B$, which acts componentwise on pairs in $A \times B$, is defined categorially in §12.4. And $\widetilde{f}$ – some write $curry(f)$ – is said to be f's *exponential transpose*.

Three quick comments. First, note that, just as f fixes its exponential transpose, the converse is also true. If $\widetilde{f} = \widetilde{g}$ then $f = ev \circ \widetilde{f} \times 1_B = ev \circ \widetilde{g} \times 1_B = g$.

Second, note too 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. Hence it might occasionally help to think of the notation 'ev' as really being lazy shorthand for something like '$ev_{B,C}$'.

Third, we can add an obvious supplementary bit of terminology:

Definition 71. A category C *has all exponentials* iff for all C-objects B, C, there is a corresponding exponential (C^B, ev). △

16.3 Some categories with exponentials

(a) A category may have *no* exponentials: for example, take a preorder category with no products. Or it may only have trivial exponentials – for we'll see that if a category has all products and hence a terminal object 1, then it will automatically have at least the exponentials X^1 and 1^X; but it may not have others.

But here are two initial examples of categories which *do* have all exponentials:

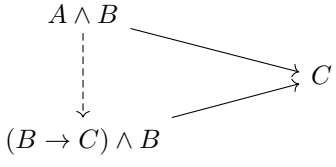
(1) Defns. 70 and 71 were of course purpose-built to ensure that Set counts as having all categorial exponentials – such an exponential of C by B is provided by the set C^B (in the standard set-theoretic sense, a set of functions-as-sets) equipped with the appropriate function-as-set ev.

Or at least, this is will be the case on most understandings of Set. Recall, however, we have so far left it open exactly how we are to conceive of our preferred universe of sets: and at this point, details begin to matter. In particular, sets-according-to-NF (or NFU, the nicer version of Quine's theory which allows for urelements) does not provide a well-behaved ev function.[3] So from now on, then, we need to be at least a little more specific about character of Set, and we will assume henceforth that it is indeed sufficiently non-deviant to have all exponentials.

(2) We can note now that the construction of exponentials in Set as standardly understood 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.

(3) We last met the category Prop_L in §10.2. This is the category whose objects are wffs of a given first-order language L, and where there is a unique arrow from A to B iff $A \vDash B$. Assuming 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 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 $B \vDash B$ – we have $A \wedge B \vDash (B \to C) \wedge B$. We therefore get the required commuting diagram of this shape,



where the down arrow is the product of the implication arrow from A to $B \to C$ and the identity arrow from B to B.

It can also be the case that a category has *some* non-trivial exponentials, though not *all* exponentials.

(4) Consider 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

[3] Assuming you know just a little about NF, here is a very brief but clear explanation by Randall Holmes: tinyurl.com/holmesev.

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 do.

(5) We'll see at the end of this chapter that **Grp** can't have all exponentials.

(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.

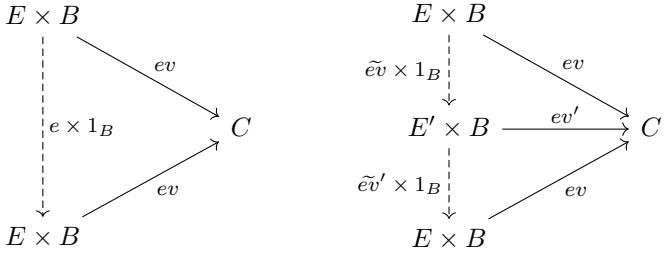
It would, however, take us far too far afield to explain and justify this example to non-topologists.

16.4 Uniqueness up to unique isomorphism

(a) Defn. 70 talks of 'an' exponential of C with B. But – as you might expect by now, given that the definition is by a universal mapping property – exponentials are in fact unique, at least up to unique isomorphism:

Theorem 65. *Suppose the category* C *has two ways of forming an exponential of* C *by* B*, namely* (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 36:



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 the mediating arrow. Hence $e = 1_E$.

The diagram on the right then reminds us that (E, ev) and (E', ev') factor through each other, and putting the two commuting triangles together we get

$$ev \circ (\widetilde{ev}' \times 1_B) \circ (\widetilde{ev} \times 1_B) = ev.$$

Applying Theorem 52, we know that $(\widetilde{ev}' \times 1_B) \circ (\widetilde{ev} \times 1_B) = (\widetilde{ev}' \circ \widetilde{ev}) \times 1_B$, and hence

$$ev \circ (\widetilde{ev}' \circ \widetilde{ev}) \times 1_B = ev,$$

and now applying the uniqueness result from the first diagram

$$\widetilde{ev}' \circ \widetilde{ev} = 1_E.$$

Similarly, by interchanging E and E' in the second diagram, we get

$$\widetilde{ev} \circ \widetilde{ev}' = 1_{E'}.$$

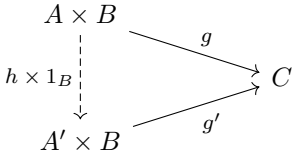
Whence $\widetilde{ev}\colon E \to E'$ has a two-sided inverse and is an isomorphism. □

(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 type of proof, we noted that products are terminal objects in a category of wedges, and 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? We will expect so. And for the record, let's confirm this. Start with

Definition 72. Given objects B and C in the category C, then the category $\mathsf{C}_{E(B,C)}$ of parametized maps from B to C has the following data:

1. Objects (A, g) comprising a C-object A, and a C-arrow $g\colon A \times B \to C$,
2. An arrow from (A, g) to (A', g') is any arrow C-arrow $h\colon A \to A'$ which makes the following diagram commute:



The identity arrows and composition are as in C. △

It is easily checked that this does define a category, and evidently we have

Theorem 66. *An exponential (C^B, ev) in C is a terminal object in $\mathsf{C}_{E(B,C)}$.*

Since exponentials are terminal in a suitable category, that yields the second type of proof of their uniqueness.

16.5 Another example of a category with exponentials

Theorem 67. *The category* Pos *has all exponentials.*

This result is worth knowing. To be frank, though, I guess you can skip the proof! I'm including it as an illustration of what it takes to check such a claim.

Proof. We are looking for a general recipe for constructing an exponential of C by B, where those objects are posets (posets living in a non-deviant world of sets with all exponentials). Represents C's ordering by $\preccurlyeq_C$.

OK: take the set of monotone functions $f\colon B \to C$ (remember the arrows in Pos are order-respecting functions). And now equip this set with the order that puts $f \preccurlyeq_{C^B} f'$ iff for all $x \in B$, $f(x) \preccurlyeq_C f'(x)$. That gives us a poset C^B.

We can define products in Pos (as in §10.2), so we can in particular form $C^B \times B$. And now to get our categorical exponential, we need a suitable evaluation function $ev\colon C^B \times B \to C$. The obvious candidate to chose is the function which takes $\langle f, b\rangle$ as input, applies the monotone function f from C^B to the element b from B, and outputs $f(b)$.

Remember, however, that ev is supposed to be an arrow living in Pos, so it needs to be an order-respecting map too. So we *do* have to check that ev as just defined is indeed monotone. But actually that's easy. If $\langle f, b\rangle \preccurlyeq \langle f', b'\rangle$ in the product order for Pos, then by definition $f \preccurlyeq_{C^B} f'$ and $b \preccurlyeq_B b'$ (see §10.2 again). But then, as wanted, $ev\langle f, b\rangle = f(b) \preccurlyeq_C f'(b) \preccurlyeq_C f'(b') = ev\langle f', b'\rangle$.

So now the claim is that (C^B, ev) is our desired exponential in Pos. Well, is there always an exponential transpose for a monotone map $f\colon A \times B \to C$ which will get the required diagram to commute? As in Set, this transpose will need to be the function $\widetilde{f}\colon A \to C^B$ which maps $a \in A$ to the monotone function f_a which sends any $b \in B$ to $f\langle a, b\rangle \in C$.

But if this is to work, we need $\widetilde{f}$ to be available in Pos, i.e. it needs itself to be monotone. Again, we need to check! Suppose $a \preccurlyeq_A a'$. Then by definition of the order on products in Pos, $\langle a, b\rangle \preccurlyeq \langle a', b\rangle$ for any b from B. Hence, since f is monotone, $f\langle a, b\rangle \preccurlyeq_C f\langle a', b\rangle$ for all b, i.e. $f_a(b) \preccurlyeq_C f_{a'}(b)$ for all b. Hence by definition, $f_a \preccurlyeq_{C^B} f_{a'}$. So $\widetilde{f}$ is monotone as required. □

16.6 Further general results about exponentials

(a) Back to more general theorems. First, we note

Theorem 68. *If there exists an exponential of C by B in the category* C*, then, for any object A in the category, there is a one-one correlation between arrows $A \times B \to C$ and arrows $A \to C^B$.*

Proof. By definition of the exponential (C^B, ev), an arrow $g\colon A \times B \to C$ is associated with a unique 'transpose' $\widetilde{g}\colon A \to C^B$ making the diagram (Exp) commute.

The function $g \mapsto \widetilde{g}$ is injective. For suppose $\widetilde{g} = \widetilde{h}$. Then $g = ev \circ (\widetilde{g} \times 1_B) = ev \circ (\widetilde{h} \times 1_B) = h$.

The function $g \mapsto \widetilde{g}$ is also surjective. Take any $k\colon A \to C^B$; then if we put $g = ev \circ (k \times 1_B)$, $\widetilde{g}$ is the unique map such that $ev \circ (\widetilde{g} \times 1_B) = g$, so $k = \widetilde{g}$.

Hence $g \mapsto \widetilde{g}$ is the required bijection between arrows $A \times B \to C$ and arrows $A \to C^B$. □

This gives us a categorial analogue of the idea of we met at the outset, where a two-place function of type $A, B \to C$ can get traded in for either a function of the type $A \times B \to C$ or alternatively for one of the type $A \to C^B$.

We also have:

Theorem 69. *Assuming the exponentials exists, there is also a one-one correlation between arrows $A \to C^B$ and arrows $B \to C^A$.*

Proof. We simply 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 24 and 37). We then apply the last theorem. □

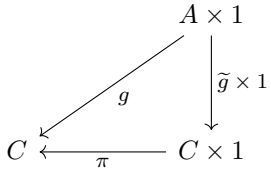
(b) We now show, as promised, that any category with all products – or at least, binary products and a terminal object – has trivial exponentials as follows:

Theorem 70. *If the category* C *has binary products and a terminal object* 1*, then for any* 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).

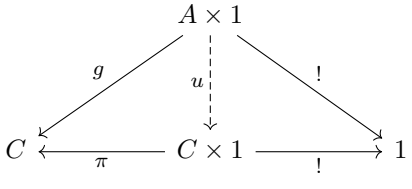
Proof for (1). By Theorem 68 again, for each A, there is a one-one correlation between arrows $A \to 1^B$ and arrows $A \times B \to 1$. But since 1 is terminal, there is exactly one arrow $A \times B \to 1$; hence, for each A, there is exactly one arrow $A \to 1^B$. Therefore 1^B is terminal, and hence $1^B \cong 1$. □

Proof for (2). Suppose we are given an arrow $g\colon A \times 1 \to C$. We want to show that there is always a unique $\widetilde{g}$ making this diagram commute:

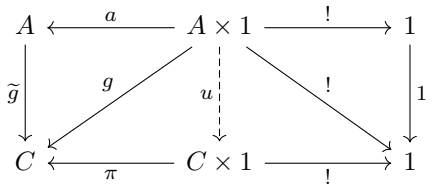


where π is the first projection from the product. Then (C, π) will serve as an exponential of C by 1 and hence, by the uniqueness theorem, any $C^1 \cong C$.

So how do we construct $\widetilde{g}$? Try brute force! 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 this next diagram commute:



Now complete the diagram with the product wedge $A \xleftarrow{a} A \times 1 \xrightarrow{!} 1$:



Both a and π must be isomorphisms by Theorem 47. So put $\widetilde{g} = g \circ a^{-1}$ where a^{-1} is the inverse of a. Then the whole diagram commutes.

But this means that $u = \widetilde{g} \times 1$ by definition of the operation $\times$ on arrows in §12.4. Hence for each $g\colon A \times 1 \to C$ there is indeed a corresponding $\widetilde{g}$ making the first of our three diagrams commute.

Moreover $\widetilde{g}$ is unique. If $k \times 1\colon A \times 1 \to C \times 1$ makes the third diagram commute then (i) it must equal u, and so $g = \pi \circ k \times 1$. But also, applying Defn. 59, we have $\pi \circ k \times 1 = k \circ a$. Hence $g = k \circ a$, so $k = g \circ a^{-1} = \widetilde{g}$. □

16.7 'And what is the dual construction?'

A good question to ask! After all, when introducing terminal objects, products, and equalizers, we more or less immediately went on to give the dual constructions of initial objects, coproducts, and co-equalizers, just by turning arrows around. So, what happens if we turn around the arrows in our definition of exponentials?

Well, doing that won't by itself give us what we need, if we are to adhere to the mantra 'co-widgets in C are widgets in C^{op}'. To make things work properly, as well as reversing the arrows in Defn. 70, we need to replace the products with coproducts. Then we do get a coherent definition of co-exponentials in C which will be exponentials in C^{op}.

But is the concept actually of much immediate interest, at our level of enquiry? Not as far as I know! So I'll say no more about co-exponentials here.

17 Cartesian closed categories

Categories like Set, Prop and Pos which have all exponentials and also have binary products and terminal objects (and hence all finite products) form an interesting class. This chapter briefly investigates.

17.1 A definition and some initial results

Definition 73. A category C is a *Cartesian closed category* iff it has all finite products and all exponentials.[1] △

Such categories indeed have some nice properties. In particular, exponentials in such categories behave as exponentials morally *ought* to behave. So we get:

Theorem 71. *If* C *is a Cartesian closed category, then for all* $A, B, C \in \mathsf{C}$

(1) If $B \cong C$*, then* $A^B \cong A^C$,
(2) $(A^B)^C \cong A^{B \times C}$,
(3) $(A \times B)^C \cong A^C \times B^C$.

I will here give a proof of (1). But I will only gesture at how to prove (2) and will omit the proof of (3) altogether, for a reason I'll explain.

Proof that if $B \cong C$*, then* $A^B \cong A^C$. Here's the 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, $\widetilde{g}\colon A^B \to A^C$. Similarly, there is an arrow $\widetilde{h}\colon A^C \to A^B$. It remains to confirm that these arrows are (as you'd expect) inverses of each other, whence $A^B \cong A^C$.

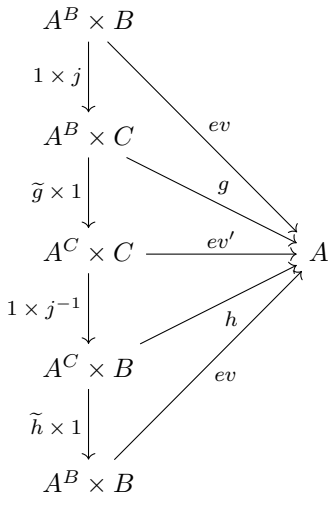
To spell that out, consider the following diagram (where $j\colon B \to C$ is an isomorphism witnessing that $B \cong C$):

[1] A terminological complexity: there is a notable variation which you need to be aware of between what different authors count as a Cartesian closed category.

Awodey (2010, p. 123), like many, follows the classic Mac Lane (1997, p. 97) in requiring only finite products and exponentials.

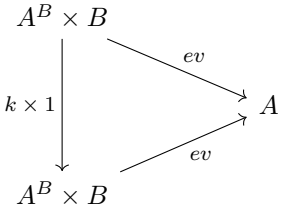
But e.g. Borceux (1994, p. 335) and Goldblatt (2006, p. 72) require all finite limits in the sense of Chapter 20's Defn. 84, rather than merely all finite products.

While Johnstone (2002, p. 46) notes that the weaker definition is the more embedded but rather deprecates that, and he calls categories satisfying the stronger condition 'properly Cartesian closed'.



Here I'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 $1 \times j$, when therefore has an inverse. And if we put $g = ev \circ (1 \times j)^{-1}$, then the top triangle commutes. The next triangle commutes by definition of the transpose $\widetilde{g}$; 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 $\widetilde{h}$.

Products of arrows compose componentwise, as shown in Theorem 52. Hence the composite vertical arrow reduces to $(\widetilde{h} \circ \widetilde{g}) \times 1$. However, by the definition of the exponential (A^B, ev) we know that there is a unique mediating arrow, k such that this commutes:



We now have two candidates for k which make the diagram commute, the identity arrow and $\widetilde{h} \circ \widetilde{g}$. Hence by uniqueness, $\widetilde{h} \circ \widetilde{g} = 1$.

A similar argument shows that $\widetilde{g} \circ \widetilde{h} = 1$. We are therefore done. □

Proof of $(A^B)^C \cong A^{B \times C}$. We can given a similarly direct proof 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 real enthusiasts (masochists?) to work out details here. That's because we will eventually be able to bring to

bear some heavier-duty general apparatus which will yield a fast-track proof for $(A^B)^C \cong A^{B\times C}$, and for the other parts of Theorem 71 too. □

17.2 Challenges

A Cartesian closed category doesn't need to have an initial object. But when it does, we get a number of further results, some of which we will eventually need. Since they can be fairly smoothly established using what you already know, I'll group them together as a sextet of challenges to prove:

Theorem 72. *If* C *is a Cartesian closed category which also has an initial object* 0*, then*

(1) $A \times 0 \cong 0 \cong 0 \times A$,

(2) $A^0 \cong 1$,

(3) any arrow $f\colon A \to 0$ *is an isomorphism,*

(4) every arrow $f\colon 0 \to A$ *is a monomorphism,*

(5) there exists an arrow $1 \to 0$ *iff all* C*'s objects are isomorphic to each other.*

And as a simple corollary,

(6) the categories Grp *and* Set_* *are not Cartesian closed.*

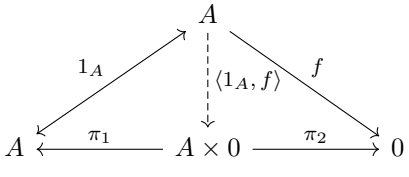
So: pause to derive those results!

Proof that $A \times 0 \cong 0 \cong 0 \times A$. Since $A \times 0$ and $0 \times A$ exist by hypothesis, and are isomorphic by Theorem 37 we need only prove $0 \times A \cong 0$.

By Theorem 68, 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 that $A^0 \cong 1$. By Theorem 68 again, for all C, there is a bijection between arrows $C \to A^0$ and arrows $C \times 0 \to A$. And by the previous result and Theorem 24 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 that any arrow $f\colon A \to 0$ *is an isomorphism.* If there's an arrow $f\colon A \to 0$ then the wedge $A \xleftarrow{1_A} A \xrightarrow{f} 0$ exists and factors uniquely through $A\times 0$:



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) $\langle 1_A, f\rangle \circ 1_A \circ \pi_1 = 1_{A\times 0}$. Therefore $\langle 1_A, f\rangle \circ \pi_1 = 1_{A\times 0}$, and $\langle 1_A, f\rangle$ has a two-sided inverse. Whence $A \cong A \times 0 \cong 0$

But then f is an arrow between two initial objects (since objects isomorphic to 0 are also initial by Theorem 27). And there can can only one such arrow and it will be an isomorphism (by Theorem 26). □

Proof that every arrow $f\colon 0 \to A$ is a monomorphism. Since 0 is initial, there is an arrow $f\colon 0 \to A$ for any target A.

Suppose we have arrows g, h such that $f \circ g = f \circ h$. Then for the composites to exist and be equal, g and h must be parallel arrows $g, h\colon X \to 0$ for some X. And hence $g = h$ by the final remark in the previous proof. □

Proof that there's an arrow $1 \to 0$ iff all C *'s objects are isomorphic.* The 'if' direction is trivial. For 'only if', suppose there is an arrow $f\colon 1 \to 0$. Then, for any A there must be a composite arrow $A \longrightarrow 1 \xrightarrow{f} 0$, and hence by our result a moment ago $f \circ 1_A\colon A \to 0$ is an isomorphism and $A \cong 0$. So every object in the category is isomorphic to 0 and hence to each other. □

Proof that Grp *and* Set_* *are not Cartesian closed.* Recall, 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 not all groups are isomorphic! Therefore the category Grp cannot be Cartesian closed. (Since Grp has all finite products, it follows that this category must lack at least some exponentials.)

The same argument shows that Set_* is not Cartesian closed, since the one-element set is both initial and terminal. □

17.3 Degeneracy!

(a) Our initial examples of Cartesian closed categories, Set, Prop and Pos, are generously endowed with multiple non-isomorphic objects. At the other end of the scale, there is the example of the one-object one-identity-arrow instance 1 which – quite trivially – has a terminal object, all binary products, and exponentials (check that!).

And we can generalize from the one-object case:

Definition 74. A category where any object has just one isomorphism to itself and one isomorphism to any other object, and there are no other arrows, is a *degenerate* Cartesian closed category.

That definition is in order because adding isomorphic copies of objects to the one-object one-identity-arrow case won't change a category in significant ways.

(b) Let's have a theorem:

Theorem 73. *A Cartesian closed category with an initial object such that* $0 \cong 1$ *is degenerate.*

Proof. If $0 \cong 1$, then there is an arrow, in fact an isomorphism, $f\colon 1 \to 0$. Hence by part (5) of the last theorem, all the objects of our category are isomorphic, and hence all the objects are in particular isomorphic to 1, i.e. are terminal. So there is a *unique* isomorphism to it from itself and from any other object. In other words, our category is degenerate. □

18 Limits and colimits defined

A terminal object is defined in terms of how *all* the other objects in the category relate to it (by each sending a unique arrow to it). A product wedge is defined in terms of how it relates to *all* the other wedges in a certain family (each factoring through it via a unique arrow to it). 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 to it). As noted before, then, terminal objects, products, and equalizers are *limiting cases*, defined in closely analogous ways using universal properties. Likewise, needless to say, for their duals.

Exponentials too are defined by a universal mapping property – recall, '(C^B, ev) is an exponential iff for all f there is a unique $\widetilde{f}$ such that ...'. But intuitively, exponentials are not limiting cases of the same general sort as before. This chapter will confirm the intuition. We will capture what's common to terminal objects, products and equalizers by defining an official general class of *limits*. We also define a dual class of *colimits*, which has initial objects, coproducts and co-equalizers as examples. We'll see that exponentials are neither limits nor colimits.

Now, in giving a general categorial definition of products, we are already abstracting from various notions of products for different kinds of widgets, bringing out what is common between them. So in this chapter we abstracting further, bringing out what is common between products, terminal objects, equalizers and more. As I promised at the outset, category theory does indeed stack layers of abstraction on layers of abstraction, but in hopefully revealing ways.

Having introduced the general idea of limits and colimits, in the next chapter we go on explore a further pair of examples, so-called pullbacks and their duals. We meet some familiar constructions of 'ordinary' mathematics in this new guise. Then in Chapter 20 we prove an important general result of the following shape: if a category has certain basic limits then it will have *all* finite limits. (And this result will both dualize and extend to the infinite case in obvious ways.)

18.1 Cones over diagrams

We need to 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*.

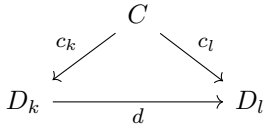
Way back in Defn. 18, we loosely characterized a diagram D in a category C as

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'll now assume that the objects in D can handily be labelled by terms like 'D_j' where 'j' is an index from some suitable suite of indices J. We will also allow the limiting cases of diagrams where there are no arrows, and even the empty case where there are no objects either. So, recasting our earlier definition:

Definition 18* A *diagram in a category* C is some (or no) objects D_j for indices j from the suite of indices J, together with some (or no) C-arrows between these objects. △

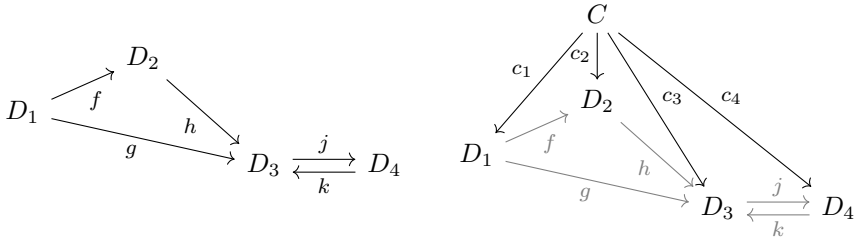
(We will eventually, in Part II, give a more abstract definition of diagrams; but this current one will certainly do to be getting on with.)

Definition 75. Let D be a diagram in category C. A *cone over* D comprises a C-object C, the *vertex* or *apex* the cone, together with C-arrows $c_j : C \to D_j$ (often called the *legs* of the cone), one for each object D_j in D, such that *whenever* there is an arrow $d : D_k \to D_l$ in D, then $c_l = d \circ c_k$ – in other words the triangles from the vertex C always commute (for each k, l):



We use '(C, c_j)' as our notation for such a cone.[1] △

Think of it diagrammatically(!) like this. Arrange the objects in the diagram D in a plane, along with whatever arrows D contains between those objects (as on the left). Now sit the object C above the plane, with a quiverful of arrows from C zinging down, one targeted at each object D_j in the plane (as on the right).



Those arrows $c_j : C \to D_j$ form the 'legs' of a skeletal cone. And the key requirement is that any new triangles formed, with C at the apex and some D_k, D_l at the base, must commute. So in our little example, we require $c_2 = f \circ c_1$, $c_3 = g \circ c_1$, $c_3 = k \circ c_4$, $c_4 = j \circ c_3$.

[1] I 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. I'll take the more explicit line.

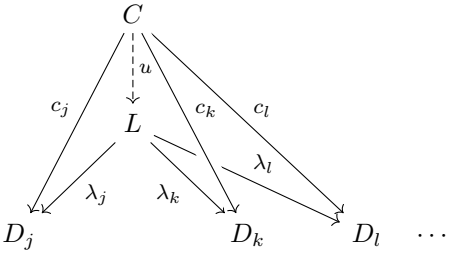
18.2 Limits

(a) There of course can be many cones, with different vertices C, over a given diagram D. But, as with e.g. our earlier definition of products, we can define a limiting case, by means of a universal property:

Definition 76. A cone (L, λ_j) over a diagram D in C is a *limit (i.e. a limit cone) over* D iff *any* cone (C, c_j) over D uniquely factors through it, meaning that there is a unique mediating arrow $u\colon C \to L$ such that for each index j, $\lambda_j \circ u = c_j$.

And we will say (simply) that D *has a limit*, if there exists a limit cone over it. △

A picture will again help to make it clear what's going on. For (L, λ_j) to be a limit cone over D, there must for each cone (C, c_j) over D be a corresponding unique $u\colon C \to L$ which makes each pictured triangles commute:



Here, for clarity's sake, I've left out of the picture any arrows there are between the D-objects.

In sum: you can think of a limit cone as one of the *shallowest* cones over D, which other cones will factor through via a unique mediating arrow.

(b) Hopefully that last diagram makes the general concept clear. But to fix ideas, let's immediately confirm that our three announced examples of 'limiting cases' so far – namely terminal objects, products, equalizers – are (or are tantamount to) limits in the sense defined.

(1) We start with the easy null case. Take the empty diagram in C – *zero* objects and so, necessarily, no arrows.

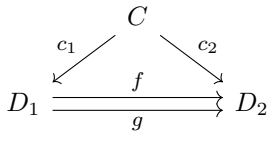
Then a cone over the empty diagram is simply an object C, a lonely vertex (there is no further condition to fulfil), and an arrow between such minimal cones C, L is simply an arrow $C \to L$. Hence L is a limit cone over the empty diagram if and only if there is a unique arrow to it from any other object – i.e. just if L is a terminal object in C!

(2) Consider now a diagram which is just *two* objects D_1 and D_2, still with no arrow between them.

Then a cone over such a diagram is simply a wedge with vertex C and arrow to D_1, D_2; and hence a limit cone is simply a product of D_1 with D_2.

(3) Next consider a diagram which again has two objects objects D_1 and D_2, but now with two parallel arrows f and g between them.

A cone over this is a commuting diagram of this shape:



If there is such a cone, then we must have $f \circ c_1 = c_2 = g \circ c_1$. Which means that $C \xrightarrow{c_1} D_1 \underset{g}{\overset{f}{\rightrightarrows}} D_2$ is a fork. Conversely, of course, given such a fork, we can turn it into a cone by adding the arrow $c_2 = f \circ c_1 : C \to D_2$. Since c_1 fixes what c_2 has to be to complete the cone, we can without loss focus on the part of the cone consisting of just (C, c_1).

What is the corresponding part of a limit cone over $D_1 \underset{g}{\overset{f}{\rightrightarrows}} D_2$? Changing notation, 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! So our recent new friends equalizers are (parts of) limits.

By contrast, however, exponentials are evidently *not* limits in the sense of limit cones.

18.3 Uniqueness up to unique isomorphism

(a) You know what comes next! We have the predictable result:

Theorem 74. *The limit over a given diagram D, if one exists, is unique up to a unique isomorphism commuting with the cones' arrows. That is to say, if (L, λ_j) and (L', λ'_j) are both limit cones over D, then there is a unique isomorphism v such that $\lambda_j \circ v = \lambda'_j$ for all j.*

And as with comparable uniqueness theorems like Theorems 36, 60 and 65, we can prove this in two ways (you shouldn't really need me to fill in the details).

Firstly, then, we can use brute force:

Plodding proof from first principles. Suppose (L, λ_j) is a limit cone over D. Note that if $\lambda_j \circ u = \lambda_j$ for all indices j, then (L, λ_j) would factor though itself via u. But trivially, the limit cone factors through itself via 1_L. Hence, since mediating arrow are unique,

(i) if $\lambda_j \circ u = \lambda_j$ for all indices j, then $u = 1_L$.

Now suppose (L', λ'_j) is another limit cone over D. Then (L', λ'_j) uniquely factors through (L, λ_j), via some v, so

(ii) $\lambda_j \circ v = \lambda'_j$ for all j.

And likewise (L, λ_j) uniquely factors through (L', λ'_j) via some w, so

(iii) $\lambda'_j \circ w = \lambda_j$ for all j.

Whence

(iv) $\lambda_j \circ v \circ w = \lambda_j$ for all j.

Therefore by (i),

(v) $v \circ w = 1_L$.

And symmetrically we can show

(vi) $w \circ v = 1_{L'}$.

Whence v is not just unique (by hypothesis, the only way of completing the relevant diagrams to get the arrows to commute) but an isomorphism. □

(b) We have already seen that

(1) A terminal object in C is ... wait for it! ... terminal in the category C.

(2) A product of X with Y in C is a terminal object in the derived category C/XY of wedges to X and Y.

(3) An equalizer of parallel arrows $f, g\colon X \to Y$ in C is (part of) a terminal object in the derived category $\mathsf{C}_{f\|g}$ of forks through X to Y.

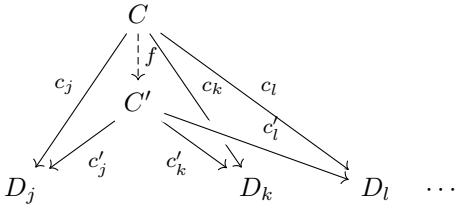
Predictably, limit cones more generally are terminal objects in appropriate categories.

To spell this out, we first note that the cones (C, c_j) over a given diagram D in C form a category in a very natural way:

Definition 77. Given a diagram D in category C, the derived category $\mathsf{C}_{Cone(D)}$ – *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 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:



The identity arrow on a cone (C, c_j) is the C-arrow 1_C. And composition for arrows in $\mathsf{C}_{Cone(D)}$ is composition of the corresponding C-arrows. △

It is entirely routine to confirm that $\mathsf{C}_{Cone(D)}$ is a category. Our earlier definition of a limit cone is then immediately equivalent to this:

Definition 78. A *limit for* D in C is a terminal object in $\mathsf{C}_{Cone(D)}$. △

And we now have an alternative proof of our desired uniqueness result:

Succint proof of Theorem 74. Since a limit cone over D is terminal in $\mathsf{C}_{Cone(D)}$, it is unique in $\mathsf{C}_{Cone(D)}$ up to a unique isomorphism. But such an isomorphism in $\mathsf{C}_{Cone(D)}$ must be an isomorphism in C commuting with the cones's arrows. □

18.4 Challenges!

We now want to prove some further general results about limits, some of which we'll need later. And to make things a bit more interesting, let's present the three theorems as a series of challenges to prove.

But first, some quick questions. We have seen that limit cones in C over the null diagram are terminal objects of C, those over arrow-less two-object diagrams are products, and those over two-object two-arrow diagrams are in effect equalizers. So it is natural to ask:

Queries *What is a limit over a two-object diagram which is arrow-less except that each object comes with its identity arrow? What is a limit over an arrowless* one-*object diagrams? How about limits over arrowless* three-*object diagrams? What are limits over diagrams with two objects and a single arrow between them? What about limits over wedges?*

Now for our needed theorems:

Theorem 75. *Suppose* (L, λ_j) *is a limit cone over the diagram* D *in* C*, and* (L', λ'_j) *is another cone over* D *which factors through* (L, λ_j) *via an isomorphism* $o\colon L' \xrightarrow{\sim} L$*. Then* (L', λ'_j) *is also a limit cone.*

Theorem 76. *Again let* (L, λ_j) *be a limit cone over a diagram* D *in* C*. Then the cones over* D *with vertex* C *correspond one-to-one with* C*-arrows from* C *to* L*.*

Theorem 77. *If* (C, c_j) *is a cone over a diagram* D *in* C *it is also a cone over the smallest sub-category of* C *which contains* D*.*

Hint: the smallest sub-category which contains D has the same objects, with all the necessary identity arrows, and all compositions of D's arrows (and then compositions of *these* arrows, etc.).

18.5 Responses

(a) Let's start with our five queries. A limit over an arrow-less diagram with two objects is a product of those objects. But what if we decorate those initial objects with their identity arrows? Nothing changes!

The underlying point is this. To make a cone over a diagram D, if there is any arrow $d\colon D_k \to D_l$, then the legs of the cone $c_k\colon C \to D_k$ and $c_l\colon C \to D_l$ must form a commuting triangle making $c_l = d \circ c_k$. Whatever the shape of D, if we add to the diagram an identity arrow $1\colon D_k \to D_k$, then trivially we will automatically have $c_k = 1 \circ c_k$, and so any cone over D is still a cone over the augmented diagram.

(b) Next, what is a limit cone over a diagram in C which consists in a single object D? A cone (C, c) over D is any object C together with an arrow $c\colon C \to D$. So – mindlessly applying the definition! – a limit cone will comprise an object L and arrow $\lambda\colon L \to D$ such that for any $c\colon C \to D$ there is a unique $u\colon C \to L$ such that $\lambda \circ u = c$.

One candidate such limit is evidently given by $L = D$ and $\lambda = 1_D$. But you know that limits are usually only unique up to isomorphism. You should expect the same here. So you'd expect any L which is isomorphic to D, when equipped with that isomorphism, gives us another limit over D.

And that's right. Here's a slow-motion argument for the same conclusion. All cones over D need to factor uniquely through a candidate limit cone (L, λ). In particular, this applies (i) to the cone $(D, 1_D)$ and also (ii) to the cone (L, λ) itself.

From case (i) we know that there is a unique u such that $\lambda \circ u = 1_D$. From case (ii) we know there is a unique v such that $\lambda \circ v = \lambda$, and evidently $v = 1_L$. But $\lambda \circ u \circ \lambda = \lambda$, so we now know that $u \circ \lambda = 1_L$. Which means that λ has a two-sided inverse, i.e. has to be an isomorphism between L and D.

(c) To answer our next query, we note that a limit cone over a diagram which consists of three isolated objects is a limiting case of a three-way wedge over those objects, i.e. is a ternary product. See Defn. 55.

(d) What is the shallowest cone over $D_1 \xrightarrow{f} D_2$? Evidently, the cone with vertex D_1 and legs $1\colon D_1 \to D_1$ and $f\colon D_1 \to D_2$. But remembering the point in (a), a cone whose vertex L is isomorphic with D_1 and with legs $\lambda\colon L \xrightarrow{\sim} D_1$ and $f \circ \lambda\colon L \to D_2$ will do equally well.

Similarly, the shallowest cone over the wedge $D_1 \xleftarrow{f} D_2 \xrightarrow{g} D_3$ is, up to isomorphism, the cone with vertex D_2 and legs $f, 1_{D_2}, g$, so is the wedge we started off with decorated with the identity arrow on D_2.

(e) Next, Theorem 75 should be very reminiscent of Theorem 38. It is proved in exactly the same way (simply generalize from the case where we have two 'legs' π_j to the case where there are possibly many legs λ_j).

So let's move on to

Theorem 76. *Suppose (L, λ_j) is a limit cone over a diagram D in C. Then the cones over D with vertex C correspond one-to-one with 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, λ_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 since (L, λ_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 described correspondence is a natural one, involving no arbitrary choices. □

(f) Let's now introduce another natural idea:

Definition 79. The (reflexive, transitive) *closure* of a diagram D in a category C is the smallest diagram which includes all the objects and arrows of D, but which also (i) has an identity arrow on each object, and (ii) for any two of its composable arrows, it also contains their composition. △

In other words, the closure of a diagram D in 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 C, it is immediate that the closure of a diagram D in C is itself a category, the smallest subcategory of C which contains D.

And now we can prove

Theorem 77. *If (C, c_j) is a cone over a diagram D in C it is also a cone over the smallest sub-category of C which contains 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. As noted before, it is trivial that, given an identity arrow $1_k\colon D_k \to D_k$, we have $c_k = 1_k \circ c_k$. Therefore we just need to show a cone over composable arrows in D is still a cone when their composite is added to D.

Suppose we have a cone over a diagram including the arrows $d\colon D_k \to D_l$ and $d'\colon D_l \to D_m$. By the definition of a cone, that means $c_l = d \circ c_k$ and $c_m = d' \circ c_l$. Hence $c_m = (d' \circ d) \circ c_k$. In other words, the new triangle with apex C and base $d \circ d'$ also commutes.

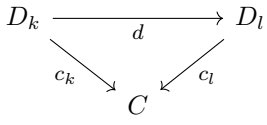
Hence the (commuting-where-required) cone remains a (commuting-where-required) cone if we add the composite arrow $d' \circ d\colon D_k \to D_m$. Iterating gives us our theorem. □

This result will prove rather significant in Part II.

18.6 Cocones and colimits

(a) Now let's dualize! Reverse the relevant arrows and you get definitions of cocones and colimits. So, dualizing Defns. 75 and 76 we get:

Definition 80. Let D be a diagram in category C. Then a *cocone under D* is a C-object C, together with an arrow $c_j\colon D_j \to C$ for each object D_j in D, such that whenever there is an arrow $d\colon D_k \to D_l$ in D, the following triangle commutes:

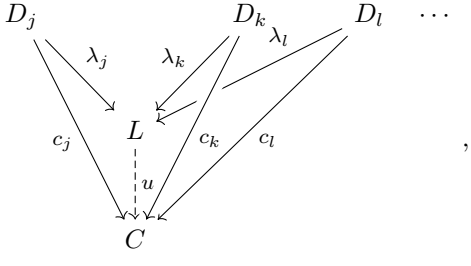


We again use '(C, c_j)' as our notation for such a cocone (letting context settle whether we are talking of cocones rather than cones). △

Definition 81. A cocone (L, λ_j) under a diagram D in C is a *colimit* (i.e. a limit case of a cocone) under D iff it factors uniquely through *any* cocone (C, c_j) under D, meaning that there is a unique mediating arrow $u\colon L \to C$ such that for each index j, $u \circ \lambda_j = c_j$.

And we will say (simply) that D *has a colimit*, if there exists a limit cocone under it. △

Here's another picture, to fix ideas. For (L, λ_j) to be colimit under D, there must for each cone (C, c_j) under D be a corresponding unique $u\colon L \to C$ which makes each pictured triangles commute:



,

Again, for clarity's sake, I've left out of the picture any arrows there are between the D-objects.

(b) 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 C-arrow $f\colon C \to C'$ such that $c'_j = f \circ c_j$ for all indexes j. So here's an equivalent definition; a colimit for D is an initial object in the category of cocones under D. (Check that!)

(c) 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 C gives rise to a cocone which is simply an object in C. So the category of cocones over the empty diagram is the category C we started with, and a limit cocone is just an initial object in C.

(2) Consider now a diagram with only *two* objects D_1 and 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 §10.7); and a limit cocone in the category of such cocones is simply a coproduct.

(3) And if we start with the diagram $D_1 \underset{g}{\overset{f}{\rightrightarrows}} D_2$ then a limit cocone over this diagram gives rise to a co-equalizer.

Evidently, exponentials are no more colimits than they are limits. (Check that!)

(d) Finally in this chapter, I suppose I ought to mention for the record some notation which you need to recognize when you see it, but which I will avoid. It is standard to denote the object at the vertex of a colimit cocone for the diagram D with objects D_j by '$\underset{\rightarrow j}{Lim}\, D_j$'. And correspondingly, the object at the vertex of a limit cone for the diagram D by '$\underset{\leftarrow j}{Lim}\, D_j$'. The directions of the arrows in this conventional notation is a bit of mystery to me, but there you have it!

19 Pullbacks and pushouts

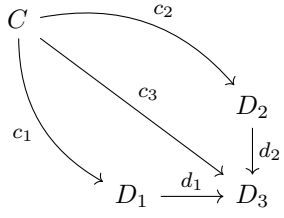
In this chapter, we explore one more kind of limit, so-called pullbacks. Predictably enough, we'll also (though much more briefly) touch on their duals, so-called pushouts.

With products and quotients, we first thought about these pre-categorially, and then gave a treatment in categorial terms. This time we'll go the other way about. We will first define the notion of a pullback in abstract categorial terms, and then think about how this relates to various already-familiar concrete constructions. For example, we'll make the link with forming intersections of sets, inverse images of functions, and kernels of group homomorphisms. Later, in §22.2, we'll meet pullbacks again in another crucial role when discussing subobjects.

19.1 Pullbacks defined

As we saw in the last chapter, taking a limit over a wedge boringly returns the same wedge. But what about taking a limit over a co-wedge or *corner*, i.e. a diagram D like $D_1 \xrightarrow{d_1} D_3 \xleftarrow{d_2} D_2$?

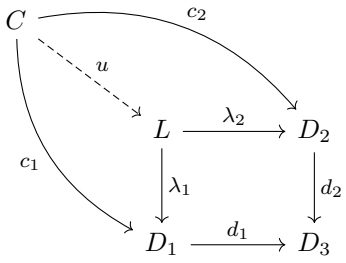
Things now get much more interesting! Let's draw this corner *as* a corner; then a cone over it an be pictured as a commutative diagram like this:



But note, the diagonal arrow c_3 is fixed by the requirement that $d_1 \circ c_1 = c_3 = d_2 \circ c_2$. So we don't really need to make it explicit. And from now on, to keep things uncluttered, I'll follow convention and leave out such arrows when drawing cone diagrams over corners.

So, what is the *limit* case for this type of cone? It will be a cone with vertex L and three projections $\lambda_j \colon L \to D_j$ such that any cone (C, c_j) over D factors

uniquely through (L, λ_j) – in other words, for any (C, c_j) there is a unique $u\colon C \to L$ such that this diagram commutes:



To repeat, I've not drawn in the diagonal arrow $c_3\colon C \to D_3$ or the diagonal arrow $\lambda_3\colon L \to D_3$; these are fixed by the requirement that the rest of the diagram commutes.

There is standard terminology for such a limit:

Definition 82. A limit for a corner diagram is its *pullback*.

The commuting square formed by a corner and its limit, with or without its diagonal, is a *pullback square*.

Further, in the illustrated square, the arrow λ_1 is said to arise by *pulling back* d_2 along d_1 – and symmetrically, of course, λ_2 is said to arise by pulling back d_1 along d_2. △

To put that last bit of terminology to work,

Theorem 78. *If both $\lambda_a\colon L_a \to D_1$ and $\lambda_b\colon L_b \to D_1$ arise from pulling back d_1 along d_2, then $L_a \cong L_b$ and λ_a and λ_b factor through each other via an isomorphism.*

Proof. As with any limit, pullbacks are unique up to unique isomorphism. So if the corner formed by d_1 and d_2 has two pullbacks with vertices L_a and L_b, there must be an isomorphism $u\colon L_1 \xrightarrow{\sim} L_2$ such that the one pullback wedge factors through the other via u. Whence $L_a \cong L_b$, and $\lambda_a = \lambda_b \circ u$ (and of course $\lambda_b = \lambda_a \circ u^{-1}$, with u^{-1} also an isomorphism). □

19.2 Examples

(a) Let's immediately look at some examples. And we get a clue where to look first from the following observation.

A pullback for a corner is a limit wedge (W), $D_1 \xleftarrow{\lambda_1} L \xrightarrow{\lambda_2} D_2$ (forgetting about the diagonal arrow we've left undrawn). And, in a sense, the notion of a pullback is a generalization of the notion of a product which is another kind of limit wedge. Compare:

(i) If (W) is to be an ordinary product, *every* wedge $D_1 \xleftarrow{c_1} C \xrightarrow{c_2} D_2$ needs to factor uniquely through that limit wedge (W).

(ii) But, for a pullback, only those wedges $D_1 \xleftarrow{c_1} C \xrightarrow{c_2} D_2$ *which form a commuting square with the opposite corner* $D_1 \xrightarrow{d_1} D_3 \xleftarrow{d_2} D_2$ need to factor uniquely through the limit wedge (W).

Evidently, then, not all pullbacks need be ordinary products.

(b) Consider then what happens in the category Set. So, changing the labelling, consider a corner formed by two set functions f, g as in $X \xrightarrow{f} Z \xleftarrow{g} Y$.

We know from the last remarks that the limit object of a pullback for this corner in Set won't in general be the whole product $X \times Y$. But it will be isomorphic to the part of that product which makes the square commute, i.e. it will be isomorphic to $\{\langle x, y\rangle \in X \times Y \mid f(x) = g(y)\}$, equipped with the obvious projection maps to X and Y.

Things work similarly in other categories where arrows are functions. For example in Top the pullback of two continuous maps $f\colon X \to Z$ and $g\colon Y \to Z$ will give us a certain subspace of the product space $X \times Y$ equipped with the product topology.

(c) Now suppose, again working in Set, that in fact both X and Y are subsets of Z, and the arrows into Z are both inclusion functions. We then get a pullback square (and now note the conventional device of indicating that a diagrammed square is a pullback by marking the vertex of the limit over the opposite corner with a little corner-symbol),[1]

$$\begin{array}{ccc} L & \longrightarrow & Y \\ \downarrow & \lrcorner & \downarrow{\scriptstyle i_2} \\ X & \xhookrightarrow{i_1} & Z \end{array}$$

with $L \cong \{\langle x, y\rangle \in X \times Y \mid x = y\} = \{\langle z, z\rangle \mid z \in X \cap Y\} \cong X \cap Y$, and with the arrows from L the obvious inclusions. Hence, we can regard the intersection of a pair of sets as tantamount to a pullback object.

As a special case, then, when $Z = X \cup Y$, the least set including both X and Y, the following will be a pullback square in Set where the arrows are inclusion functions again:

$$\begin{array}{ccc} X \cap Y & \hookrightarrow & Y \\ \downarrow & \lrcorner & \downarrow \\ X & \hookrightarrow & X \cup Y \end{array}$$

[1] I guess I should also mention a frequently-used notation for the limit vertex in a pullback diagram for such a corner, as in $X \times_Z Y$. But this seems rather unhelpful, as the limit object essentially depends on the *arrows* from X and Y and not just on the object Z occupying the corner vertex.

(d) Again in a category where arrows are functions, suppose we start from a corner of this kind:

$$\begin{array}{ccc} & & Y \\ & & \downarrow{\scriptstyle 1_Y} \\ X & \xrightarrow{f} & Y \end{array}$$

Then the pullback object will be

$$L \cong \{\langle x, z\rangle \in X \times Y \mid f(x) = z\} \cong \{x \mid \exists z f(x) = z\} = f^{-1}[Y],$$

So L is isomorphic to the inverse image of Y, and we have the following pullback square:

$$\begin{array}{ccc} f^{-1}[Y] & \longrightarrow & Y \\ \downarrow & \lrcorner & \downarrow{\scriptstyle 1_Y} \\ X & \xrightarrow{f} & Y \end{array}$$

Hence we can also think of forming the inverse image as a pullback construction, with the arrow $f^{-1}[Y] \to X$ formed by pulling back the arrow 1_Y along f.

Relatedly, starting from the corner

$$\begin{array}{ccc} & & 1 \\ & & \downarrow{\scriptstyle \vec{y}} \\ X & \xrightarrow{f} & Y \end{array}$$

the resulting pullback will give the inverse image of a particular element under a function.

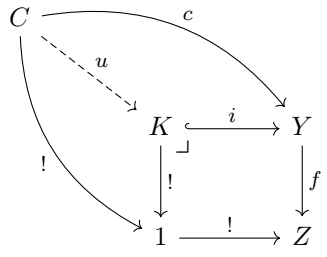
(e) Let's turn now to look at a nice example in Grp.

Remember, the one-object group 1 is both initial and terminal in Grp. So now consider the corner $1 \xrightarrow{!} Z \xleftarrow{f} Y$. Does it have a pullback? Well, a cone over this corner will look like this (omitting the diagonal as usual):

$$\begin{array}{ccc} C & \xrightarrow{c} & Y \\ \downarrow{\scriptstyle !} & \lrcorner & \downarrow{\scriptstyle f} \\ 1 & \xrightarrow{!} & Z \end{array}$$

And if this diagram is to commute, then $f \circ c$ has to send every object in the group C to the group identity of Z. So c has to send every object of C somewhere in the kernel of Y.

We'll expect, then, that we get a limiting case – a pullback cone – when its vertex is the kernel K of Y, and the non-trivial 'leg' is simply the inclusion map $i\colon K \hookrightarrow Y$. Let's confirm that. Consider the diagram



For the outer wedge to make a commuting square with the corner, as we said, c needs to map C into the kernel of Y. But if $u\colon C \to K$ agrees everywhere with $c\colon C \to Y$, we'll get a commuting upper triangle (and in the only way possible). So K with the appropriate 'legs' gives us our limit cone.

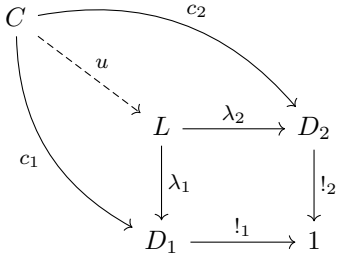
In short, kernels of group homorphisms can be thought of as pullback objects.

19.3 More on pullbacks and products

We noted at the beginning of the last section that not all wedges forming a pullback square need be products. However, we do have the following easy result in the converse direction:

Theorem 79. *In a category with a terminal object, all binary products are pullbacks.*

Proof. Consider the diagram



By definition, if the wedge with vertex L is in fact a product for D_1 with D_2 then, for any wedge $D_1 \xleftarrow{c_1} C \xrightarrow{c_2} D_2$, there will be a unique u making the outer triangles commute. But of course, adding the only possible arrows from D_1 and D_2 to the vertex with the terminal object 1 will make the resulting square commute. Therefore that unique u will make the whole diagram commute. So the product wedge for D_1 with D_2 with vertex L is a pullback limit for the corner $D_1 \xrightarrow{!} 1 \xleftarrow{!} D_2$. □

Our proof of the last theorem generates an easy corollary:

Theorem 80. *A category which has a terminal object and a pullback for every corner also has all binary products.*

Proof. Given objects D_1 with D_2 in the category, then by the definition of the terminal object 1 there will be a corner $D_1 \xrightarrow{!} 1 \xleftarrow{!} D_2$. This corner, by assumption, will have a pullback. But, as the last proof shows, the wedge that forms the pullback is a product of D_1 with D_2. □

19.4 Pullbacks, monos, (co)equalizers

(a) Let's next note two simple theorems relating pullbacks and monomorphisms:

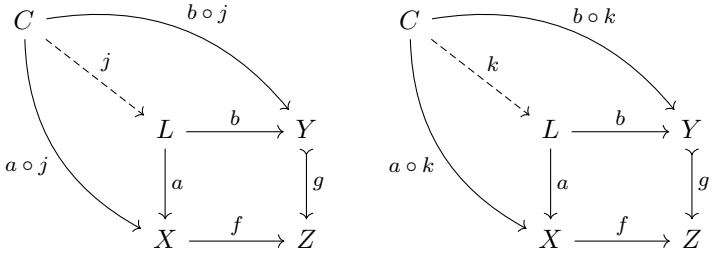
Theorem 81. *Pulling back a monomorphism yields a monomorphism.*

In other words, start with a corner $X \xrightarrow{f} Z \xleftarrow{g} Y$ with g monic. Then if we *can* pull back g along f to give a pullback square

$$\begin{array}{ccc} L & \xrightarrow{b} & Y \\ {\scriptstyle a}\downarrow & \lrcorner & \downarrow{\scriptstyle g} \\ X & \xrightarrow{f} & Z \end{array}$$

then the resulting arrow a is monic. (NB, this doesn't 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, $b \circ j = b \circ k$. So now consider the two diagrams



Since $f \circ a \circ j = g \circ b \circ j$, the wedge $X \xleftarrow{a \circ j} C \xrightarrow{b \circ j} Y$ is a cone over the original corner, so uniquely factors through the limit via j. Likewise the wedge $X \xleftarrow{a \circ k} C \xrightarrow{b \circ k} Y$ is also a cone over the original corner, factoring uniquely through the limit via k. But we've just shown that those two cones are in fact the same. Hence $j = k$.

So, in short, we've proved that if $a \circ j = a \circ k$, then $j = k$, and hence a is monic. □

Theorem 82. *The arrow $f : X \to Y$ is a monomorphism in C if and only if the following is a pullback square:*

$$\begin{array}{ccc} X & \xrightarrow{1_X} & X \\ \downarrow{\scriptstyle 1_X} & \lrcorner & \downarrow{\scriptstyle f} \\ X & \xrightarrow{f} & Y \end{array}$$

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. In other words, 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.

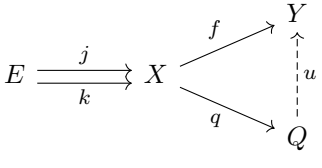
Conversely, suppose f is monic. Then given any cone $X \xleftarrow{a} C \xrightarrow{b} X$ over the original corner, $f \circ a = f \circ b$, whence $a = b$. But that means the cone factors through the cone $X \xleftarrow{1_X} X \xrightarrow{1_X} X$ via the unique a, making that cone a limit and the square a pullback square. □

(b) Following on from the last theorem, what about the case where we pull back an arrow f along itself, but f isn't assumed to be monic? Suppose, then, we have a pullback square of this kind in Set:

$$\begin{array}{ccc} E & \xrightarrow{k} & X \\ \downarrow{\scriptstyle j} & \lrcorner & \downarrow{\scriptstyle f} \\ X & \xrightarrow{f} & Y \end{array}$$

Then, up to isomorphism, E will be the set $\{\langle x, x' \rangle \in X \times X \mid f(x) = f(x')\}$, which is the extension of E_f, the equivalence kernel of f (see Defn. 63).

Now, the routes round the pullback square give us a co-fork starting from the parallel arrows $j, k \colon E \to X$. So a nice question to ask is: what is the co-equalizer of these arrows?



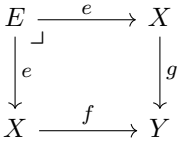
A moment's reflection shows that, following the line of argument in §15.6, Q will be (isomorphic to) the quotient X/E_f (why)? So we've shown that $f \colon X \to Y$ factors through X/E_f as $u \circ q$. Now compare the discussion of §7.7, where we proved that in Set an arrow $f \colon X \to Y$ factors through the quotient X/E_f. But we haven't yet quite recovered the earlier pre-categorial result that this can provide an epi-mono factorization. We know that q is epic, by the remark following Defn 69. But it needs to be shown that u is monic in Set (it is!).

Generalizing beyond Set, in any category with pullbacks and co-equalizers, any arrow $f \colon X \to Y$ will factor through the object Q in the co-equalizer of the pullback of f along itself – will factor as $u \circ q$, with $q \colon X \to Q$ epic. If

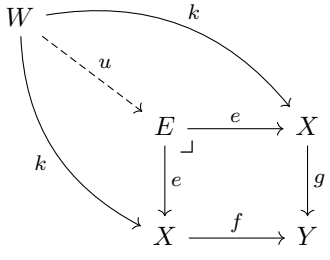
the category is otherwise nice enough, u will be monic as in Set and we get an epi-mono factorization. But we won't pursue the details here.

(c) Now a theorem about pullbacks and equalizers:

Theorem 83. *(E, e) is an equalizer for the parallel arrows $f, g\colon X \to Y$ if the following is a pullback square:*



Proof. Suppose that is a pullback square. Then whatever object W and arrow $k\colon W \to X$ we take, there will be a unique u making the following diagram commute:



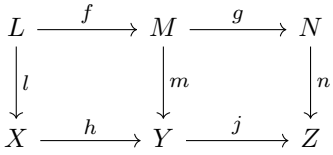
But that is of course just a re-drawn version of our Defn. 66 of (E, e) as an equalizer. □

19.5 Some challenges about pullbacks

(a) Do at least note the statements of the following theorems. And why not challenge yourself to find the proofs?

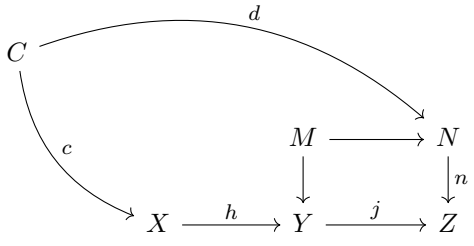
First, a very useful result, often called simply *the pullback lemma*:

Theorem 84. *Suppose we have two joined commuting squares like this:*

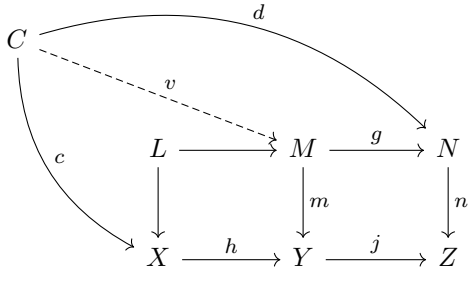


Then (1) if the two inner squares are pullback squares, the outer rectangle is a pullback square(!) too. And (2) if the right-hand square and the outer rectangle are both pullback squares, so is the left-hand square.

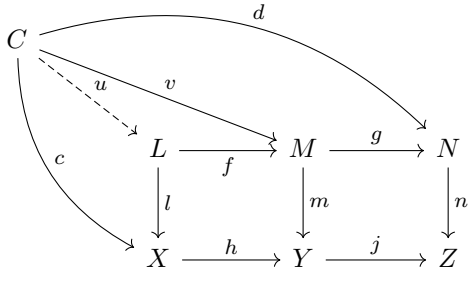
Proof of (1). If the outer rectangle is to be a pullback square, we need to show that any wedge $X \xleftarrow{c} C \xrightarrow{d} N$ over the corner $X \xrightarrow{j \circ h} Z \xleftarrow{n} N$ factors uniquely through L. Suppose then that we assume that this next diagram commutes:



Since the right-hand square is a pullback, there must be a unique $v\colon C \to M$ from the vertex of the wedge $Y \xleftarrow{h \circ c} C \xrightarrow{m} M$ which makes this next diagram commute:



Since the left-hand square is also a pullback, the wedge $X \xleftarrow{c} C \xrightarrow{v} M$ must factor through L via a unique arrow $u\colon C \to L$, making all this commute:



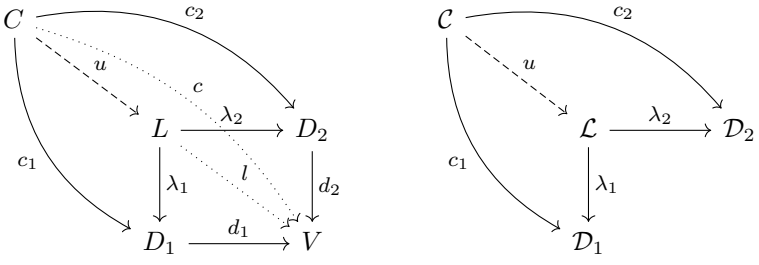
So the wedge $X \xleftarrow{c} C \xrightarrow{d} N$ factors through L via a unique map $u\colon C \to L$. Hence the whole rectangle *does* form a pullback. □

I can perhaps leave you to similarly prove part (2) of the pullback lemma.

(b) We saw that there is a sense in which pullbacks are like modified products. Here is another less obvious way of looking at how pullbacks and products they are related.

Theorem 85. *A pullback square for a corner $D_1 \longrightarrow V \longleftarrow D_2$ in the category C is a product of $D_1 \to V$ and $D_2 \to V$ as objects of the slice category C/V.*

Proof. Let's consider these two diagrams, while keeping our wits about us!



So, we have a pullback in C over the corner $D_1 \longrightarrow V \longleftarrow D_2$. Being a pullback square, for any wedge with vertex C that forms a commuting square with that corner, there will be a unique u making the whole diagram commute. And this time I have indicated the two diagonals $l\colon L \to V$ and $c\colon C \to V$.

Next, recall from §6.3 that objects and arrows in C/V are both arrows in C. Using a different font to label C/V-objects for clarity, let $\mathcal{D}_j$ be the C-arrows $d_j\colon D_j \to V$, and let $\mathcal{L}$ be $l\colon L \to V$ and $\mathcal{C}$ be $c\colon C \to V$.

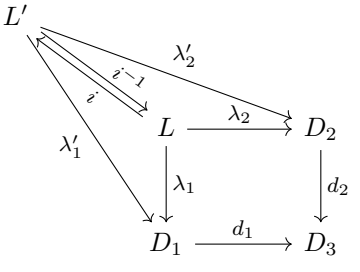
Now, since $c = d_j \circ c_j$ in C, we have (by definition) C/V-arrows $c_j\colon \mathcal{C} \to \mathcal{D}_j$. And since $l = d_j \circ \lambda_j$, we have C/V-arrows $\lambda_j\colon L \to D_j$ such that $l = d_j \circ \lambda_j$. There they are, diagrammed on the right.

Finally note that, as we vary in C over any wedge $D_1 \xleftarrow{c_1} C \xleftrightarrow{c_2} D_2$ which makes a commuting square with the opposite corner, we vary in C/V over every wedge $\mathcal{D}_1 \xleftarrow{c_1} \mathcal{C} \xrightarrow{c_2} \mathcal{D}_2$. But in C, however we vary the wedge, there is a unique arrow u in C such that $\lambda_j \circ u = c_2$ (by the definition of the pullback); which means that for every corresponding wedge in C/V, there is again the same unique arrow such that $\lambda_j \circ u = c_2$.

Therefore, lo and behold, the wedge $\mathcal{D}_1 \xleftarrow{\lambda_1} \mathcal{L} \xrightarrow{\lambda_2} \mathcal{D}_2$ is a product of $\mathcal{D}_1$ and $\mathcal{D}_2$. But the wedge is just the whole pullback square in C (with the diagonal drawn in), and the $\mathcal{D}_j$ are the arrows $D_j \to V$, so we are done. □

(c) Finally, I should mention a lemma which will prove useful.

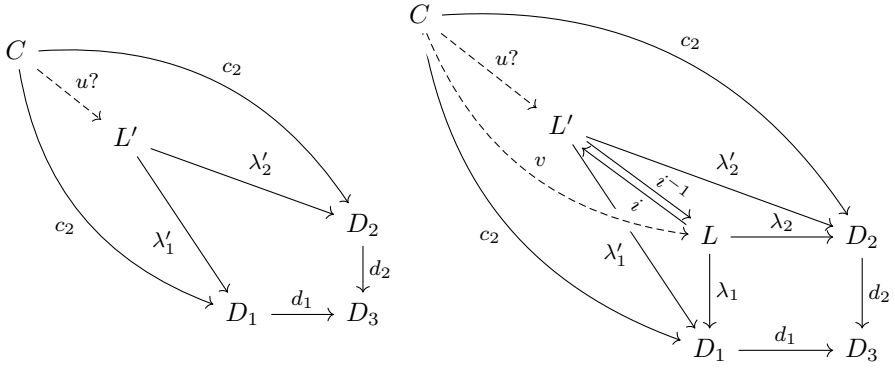
Theorem 86. *Take a pullback square. Suppose there is an isomorphism i from its limit object L to L' with inverse i^{-1}, and all (four!) displayed triangles commute.*



Then the outer square with corner L' is also a pullback.

Proof of Theorem 86. This looks very messy, and we'll only see the point of the result later. But we just have to chase arrows round some diagrams. So here goes.

We want to know whether, for any wedge $D_1 \xleftarrow{c_1} C \xrightarrow{c_2} D_2$ such that $d_1 \circ c_1 = d_2 \circ c_2$ there is a unique $u\colon C \to L'$ making the left hand diagram commute.



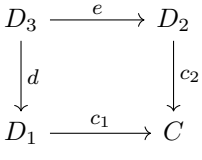
Since the original square is a pullback, the wedge with vertex C (the C-wedge for short) must factor through the L-wedge via a unique $v\colon C \to L$ as shown in the right hand diagram. But we can then read off the commuting diagram that the C-wedge factors through L'-wedge via the arrow $i \circ v$.

And that's the unique possibility, for suppose the C-wedge factors through the L'-wedge by some u. Then again we can read off the commuting diagram that the C-wedge factors through L-wedge via the arrow $i^{-1} \circ u$. However, we know that v is the unique arrow that can play this role, so $i^{-1} \circ u = v$, whence $u = i \circ v$. □

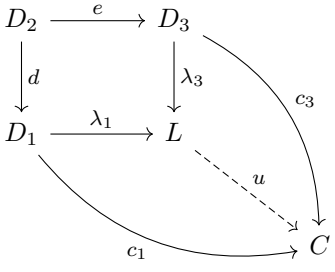
19.6 Pushouts

(a) A pullback is the *limit* of a *corner*. What is a colimit for a corner? Check the relevant diagram and it is obviously the corner itself. No, the real dualization of the notion of a pullback is provided when we take the *colimit* of a *'co-corner'*, i.e. of a wedge.

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).



And a limit cocone of this type will be a cocone with apex L and projections $\lambda_j : D_j \to L$ such that for any cocone (C, c_j) under D, there is a unique $u : L \to C$ such that the obvious dual of the whole pullback diagram in §19.1 commutes:

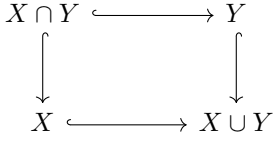


So we say:

Definition 83. A colimit for a wedge diagram is a *pushout*. △

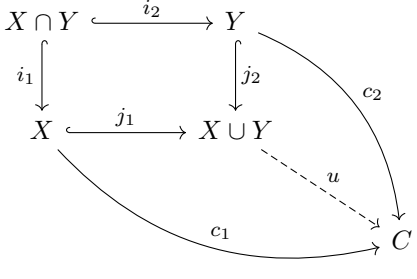
As you would expect, we then have dual theorems about pushouts – e.g. they are unique up to unique isomorphism, pushing out an epimorphism yields an epimorphism, etc.

(b) What about examples of pushouts? Let's start with a nice example in Set. This diagram which we met before as a pullback is also a pushout:



Why so?

We need to show that for any cocone (C, c_j) under the wedge $X \leftarrow X \cap Y \to Y$ there is a unique arrow $u : X \cup Y \to C$ making this diagram commute (I've now labelled the arrows for convenience):



Define u to agree with c_1 on members of X and c_2 on members of Y. That does well-define a function because by the assumption that (C, c_j) is a cocone for the wedge, $c_1 \circ i_1 = c_2 \circ i_2$, and by the assumption that i_1 and i_2 are inclusions (so don't affect values), that means c_1 and c_2 do agree on the overlap of X and Y. This definition of u makes the added triangles commute because j_1 and j_2 are inclusions too, and is evidently the unique possibility.

(c) What about pushouts more generally in Set?

In this category, as we've seen, the object of the limit cone over a corner diagram $X \xrightarrow{f} Z \xleftarrow{g} Y$ is not in general the whole product $X \times Y$ but a subset of that, namely the subset of the product consisting of pairs $\langle x, y \rangle$ where $f(x) = g(y)$. Dually, we'll expect the object of the colimit cocone under a wedge diagram $X \xleftarrow{f} Z \xrightarrow{g} Y$ to be not the whole co-product (disjoint union) $X \oplus Y$ but ...?

Well, how *do* we need to tinker with the co-product (thought of as constructed in the usual way by tagging members of X with 0 and members of Y with 1, and combining the results)? It turns out that we need to quotient by the smallest equivalence relation generated by the relation which holds between each $(f(z), 0)$ and $(g(z), 1)$.

For our purposes, however, I probably don't need to pause to spell out the details why (though it is a nice challenge to think things through). Nor will I pause to explain why in Grp, pushouts produce so-called free products with amalgamation. I just note that pushouts do tend to generate rather less familiar constructions than pullbacks.

20 The existence of limits

We have now 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, and taking inverse images.

Not *every* familiar kind of construction in a category C involves taking a limit cone or cocone in C. We have already met one important exception, namely exponentials: we meet another exception in the next chapter. But plainly we are mining a very rich seam here – and we are already making good on the 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 very tedious to explore case by case what it takes for a category to have limits for lots of further kinds of diagram. 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). Similarly, needless to say, for the dual case.

20.1 The key theorems stated

(a) Let's start with a natural definition:

Definition 84. The category C *has all finite limits* iff for any finite diagram D in C – i.e. for any diagram whose objects are D_j for indices j from a finite suite of indices J – C has a limit over D.

A category with all finite limits is also said to be *finitely complete* △

So now the obvious question is: what does it take for a category to be finitely complete?

We'll work up to to an answer by initially establishing (in §20.2)

Theorem 87. *If a category has all binary products and has equalizers for every pair of parallel arrows, then it has a pullback for every corner.*

Then (in §20.3) we'll adapt the proof-strategy for that initial result to establish our first main result:

Theorem 88. *If a category has a terminal object, and has all binary products and equalizers, it is finitely complete.*

And we could leave it at that. But there is some interest in also proving a variant completeness result. Recall, we have already shown the easy

Theorem 80. *A category which has a terminal object and a pullback for every corner also has all binary products.*

But we can also show (in §20.4)

Theorem 89. *If a category has a terminal object and has a pullback for every corner, then it will have an equalizer for every pair of parallel arrows.*

And from those two theorems and our first main result we can obviously derive this variant completeness theorem:

Theorem 90. *If a category has a terminal object and has a pullback for every corner, it is finitely complete.*

Given ingredients from our previous discussions, since the categories in question have terminal objects, binary products and equalizers, we can immediately conclude

Theorem 91. Set *and* FinSet *are finitely complete, as are categories of algebraically 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.

(b) Theorem 88 and its companion Theorem 90 are perhaps the first unobvious Big Results we have met. Their proofs are not exactly difficult, but get a bit intricate. You certainly need to know the Results; however, nothing later depends on your knowing the proof-details explained over the next three sections. So by all means – at least on a first reading – skip forward to §20.5 where I briefly outline the move from the finite to the infinite case, and to §20.6 where everything gets snappily dualized.

20.2 Products plus equalizers imply pullbacks

So, as announced, we are going to begin by proving

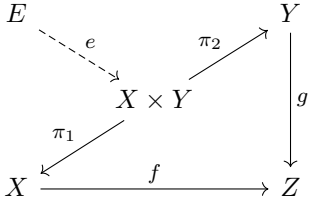
Theorem 87. *If a category has all binary products and has equalizers for every pair of parallel arrows, then it has a pullback for every corner.*

Proof. Start with an arbitrary corner $X \xrightarrow{f} Z \xleftarrow{g} Y$.

There is nothing to equalize yet. Our only option, then, for constructing a pullback is to begin by constructing some product. So: what else can we do

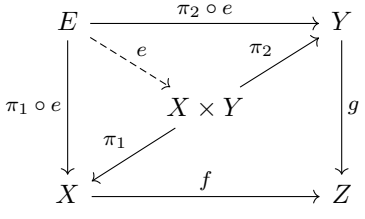
but take the product $X \times Y$ with the usual projections $\pi_1 \colon X \times Y \to X$ and $\pi_2 \colon X \times Y \to Y$. This immediately gives us parallel arrows $X \times Y \overset{f \circ \pi_1}{\underset{g \circ \pi_2}{\rightrightarrows}} Z$.

We are assuming that we can always equalize parallel arrows, so there is some (E, e) for which $f \circ \pi_1 \circ e = g \circ \pi_2 \circ e$. We can picture the situation like this:



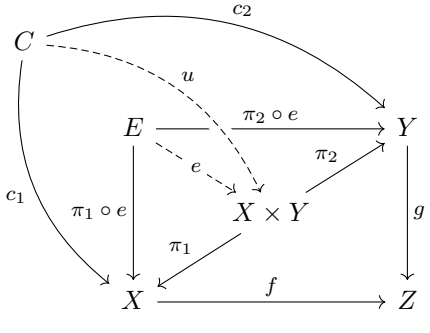
But careful! This is *not* a fully commuting diagram. We are *not* assuming that the two composite arrows from $X \times Y$ round to Z are equal – after all, we are in the business of equalizing those arrows!

Now add the arrows which give us two new commuting triangles like this:



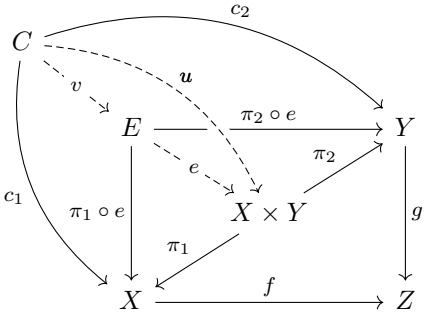
And note that the wedge formed by E with the arrows $\pi_1 \circ e$, $\pi_2 \circ e$ looks as if it should be a sort of limiting case among wedges completing a commuting square with the original corner (after all, E is part of a limit). So hopefully, that wedge is a pullback for the corner.

Some fairly routine checking confirms that conjecture. For consider any other cone over the original corner. In other words, leaving the diagonals to take care of themselves, consider any wedge $X \xleftarrow{c_1} C \xrightarrow{c_2} Y$ with $f \circ c_1 = g \circ c_2$: we need to show that this factors uniquely through E. So stare at this diagram!



Our new wedge will also uniquely factor through the product $X \times Y$, which is why we have added to the diagram 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 $C \xrightarrow{u} X \times Y \overset{f \circ \pi_1}{\underset{g \circ \pi_2}{\rightrightarrows}} Z$ is itself a fork, which must factor uniquely through the equalizer E via some v, like this:



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 our wedge with vertex C factors through the wedge with vertex E, as we need.

To finish, we have to establish that v is the only way that the wedge with vertex C can factor through E. 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 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 63, so $v' = v$, and we are done. □

20.3 Deriving the finite completeness theorem

(a) Our project now is prove the announced main result:

Theorem 88. *If a category has a terminal object, and has all binary products and equalizers, it is finitely complete.*

And we are going to generalize the strategy pursued in proving the cut-down version in the previous section, where we showed that having binary products and equalizers implies at least having pullbacks. So, here's the outline plan:

> Given a finite diagram D, we start by forming the product P of the objects from D (which we can do since C has all finite products if it has terminal objects and binary products – see §11.2).
>
> We then find some appropriate parallel arrows out of this product P. But what will be their target? In fact, another product Q. Then we can take an equalizer (E, e) of those parallel arrows (which we can do since C has all equalizers).
>
> We show that we can use E as the vertex of the desired limit cone over the diagram D on the model of the proof of Theorem 87.

The devil, of course, is in the details! And to be frank, you won't lose anything if you skip right past them.

(b) Still with me? Consider again the proof of our last theorem. We started with a mini-diagram D, i.e. a corner with two arrows sharing a target, $f\colon X \to Z$, $g\colon Y \to Z$. We then took the product of the diagram's objects X and Y; and we can think of Z as already a 'unary product' (as in §11.2) of those arrows' shared target. And we next formed parallel arrows between these two products, namely $f \circ \pi_1, g \circ \pi_2\colon X \times Y \to Z$. *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 from the diagram a pair of parallel arrows with the same source and same target so that we can then take an equalizer which will give us a limit cone over D. To construct the needed single source and single target we use products – products again formed from the diagram's objects and from the targets of the diagram's arrows, as follows:

(i) First, we simply form the multi-product $(P, p_j\colon P \to D_j)$ formed by the object P equipped with all the arrows p_j targeting diagram objects D_j.
(ii) Second, we look at all the arrows d in D, and form the multi-product $(Q, q_d\colon Q \to D_d)$ of all the corresponding target objects D_d which are the target of some $d\colon D_k \to D_d$ (note that a particular object D_d is to feature in different arrows $q_d\colon Q \to D_d$ as many times as it is the target of an arrow d).

The name of the game is now to define a pair of parallel arrows $v, w\colon P \to Q$ which we are going to equalize by some (E, e).

There are two arrows from P to Q which arise rather naturally:

(1) First, take the vertex P together with the arrows $p_d\colon P \to D_d$, for each D_d occurring in the product Q. These arrows will form a 'multi-wedge' over the same objects as the product (Q, q_j) is over. So this multi-wedge must factor through the product (Q, q_d) by a unique mediating arrow v, so that $p_d = q_d \circ v$ for each l.
(2) Second, take the vertex P again, but this time with arrows $d \circ p_k\colon P \to D_d$ zinging down, one for each arrow $d\colon D_k \to D_q$ in D. This 'multi-wedge' must also factor through the product (Q, q_d) by a unique mediating arrow w, so that $d \circ p_k = q_d \circ w$ for each arrow $d\colon D_k \to D_l$.

Since we are assuming that all parallel arrows have equalizers in 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 87: $(E, p_j \circ e)$ will be a limit cone over the given diagram D.

(c) Let's state this as a more specific Theorem 88* which immediately implies the less specific Theorem 88:

Theorem 88*. *Let D be a finite diagram in a category 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_d) be the product of the objects D_d (one occurrence for each arrow of the kind $d\colon D_k \to D_d$). Then there are arrows*

$$P \underset{w}{\overset{v}{\rightrightarrows}} Q$$

such that the following diagrams commute for each $d\colon D_k \to D_d$:

$$\begin{array}{ccc} P & \xrightarrow{v} & Q \\ & \searrow^{p_d} & \downarrow{\scriptstyle q_d} \\ & & D_d \end{array} \qquad \begin{array}{ccc} P & \xrightarrow{w} & Q \\ {\scriptstyle p_k}\downarrow & & \downarrow{\scriptstyle q_d} \\ D_k & \xrightarrow{d} & D_d \end{array}$$

Let the equalizer of v and w be (E, e). Then $(E, p_j \circ e)$ will be a limit cone over D in C.

Proof. (P, p_j) and (Q, q_d) exist because a category with terminal objects and binary products has all finite products. And we have already shown that v and w exist such that the given diagrams commute. By assumption an equalizer (E, e) for them exists.

So next we confirm $(E, p_j \circ e)$ is at least a cone of the diagram D. Suppose then that there is an arrow $d\colon D_k \to D_d$. For a cone, we require $d \circ p_k \circ e = p_d \circ e$.

But we have $d \circ p_k \circ e = q_d \circ w \circ e = q_d \circ v \circ e = p_d \circ e$, 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 cone over D and we have $c_j = p_j \circ u$.

Since (C, c_j) is a cone, for any $d\colon D_k \to D_d$ in D we have by assumption that $d \circ c_k = c_d$. Hence $d \circ p_k \circ u = p_d \circ u$, and hence for each q_d from the product (Q, q_j), $q_d \circ w \circ u = q_d \circ v \circ u$. But then we can apply the obvious generalized version of Theorem 44 about products, and conclude that $w \circ u = v \circ u$. Which means that

$$C \xrightarrow{u} P \underset{w}{\overset{v}{\rightrightarrows}} Q$$

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, $c_j = p_j \circ u = p_j \circ e \circ s$.

But 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. □

Phew! But that's our first completeness result in then bag.

20.4 Deriving the variant completeness theorem

Our next target is to prove

Theorem 89. *If a category has a terminal object and has a pullback for every corner, then it will have an equalizer for every pair of parallel arrows.*

Then, as we noted at the beginning of the chapter, this – together with Theorems 80 and 88 – immediately implies our variant completeness theorem

Theorem 90. *If a category has a terminal object, and has a pullback for every corner, it is finitely complete.*

So we just need

Proof of Thm. 89. Take the parallel arrows we want to equalize, say $f, g\colon X \to Y$, but now think of these as forming a wedge $Y \xleftarrow{f} X \xrightarrow{g} Y$.

Like any wedge, this wedge will factor uniquely through the appropriate product of the outside objects of the wedge – in this case $Y \times Y$. And by Theorem 79, like any other product this particular one is available in our category with all pulbacks and a terminal object. Following our earlier convention in §10.5 we can notate this unique mediating arrow $\langle f, g\rangle\colon X \to Y \times Y$.

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. 51). This is nice to think about since (to arm-wave a bit!) the first arrow packages up the parallel arrows we want to equalize. While the second sort of arrow is always available in a category with products, and in effect does some sort-of-equalizing – e.g. in Set it sends something from Y to an ordered pair of two equal objects.

Now take this corner's pullback (the only thing to do with it!):

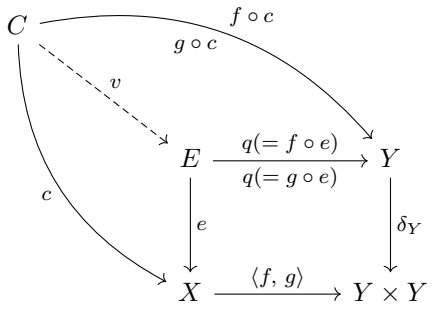
$$\begin{array}{ccc} E & \xrightarrow{\quad q \quad} & Y \\ \downarrow{\scriptstyle e} & \lrcorner & \downarrow{\scriptstyle \delta_Y} \\ X & \xrightarrow{\quad \langle f,g\rangle \quad} & Y \times Y \end{array}$$

Intuitively, $E \xrightarrow{e} X \xrightarrow{\langle f,g\rangle} Y \times Y$ sends something from E to what is, according to the other route round the commuting square, a pair of equals. So, morally, (E, e) ought to be an equalizer for $X \underset{g}{\overset{f}{\rightrightarrows}} Y$.

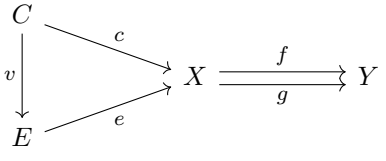
And, from this point on, it is a routine proof to check that (E, e) is the equalizer we want. Here goes ...

By the commutativity of the pullback square, $\delta_Y \circ q = \langle f, g\rangle \circ e$. Appealing to Theorems 39, 49 and 48, 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 $E \xrightarrow{e} X \underset{g}{\overset{f}{\rightrightarrows}} Y$ is a fork. It remains to show that it is a limit fork.

Take any other fork $C \xrightarrow{c} X \underset{g}{\overset{f}{\rightrightarrows}} Y$, and form the corresponding wedge $X \xleftarrow{c} C \xrightarrow[g\circ c]{f\circ c} Y$. . This wedge must factor through E (because E is the vertex of the pullback) via a unique mediating arrow v. That is to say, more perspicuously putting all that in a nice diagram, we have:



It follows that v makes this next diagram commute:



And any $v' \colon C \to E$ 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. □

20.5 Infinite limits

We can extend our key Theorem 88 to reach beyond the finite case. First, we need:

Definition 85. The category 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 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 in §11.3, 'small' is something of a joke. Small limits can be outrageously huge constructions – they just must be taken over diagrams that are no-bigger-than-set-sized.

An easy inspection of the proofs of Theorem 88 shows that the argument will continue to go through as before even if the suite of indices J is non-finite, so long as we can sensibly handle that many objects. Assume then we keep everything set-sized, so we are still dealing with a category like Set which has products for all collections of objects O indexed by set-sized suites of indices. Then, without further ado, we can state:

Theorem 92. *If* C *has all small products and has equalizers, then it has all small limits, i.e. is complete.*

We can similarly extend Theorem 91 to show that

Theorem 93. 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.

20.6 Dualizing again

Our results in this chapter dualize in obvious ways (of course!). Thus we need not delay over the further explanations and proofs of

Definition 86. The category C *has all finite colimits (is finitely co-complete)* iff for any finite diagram D in C, the category has a colimit over D.

C *has all small colimits* if for any diagram D whose objects are D_j for indices $j \in I$, for some set I, then C has a colimit over D. A category with all small colimits is also said to be *co-complete*. △

Theorem 94. *If a category has initial objects, binary coproducts and co-equalizers, then it has all finite colimits, i.e. is finitely co-complete.*

If a category has all small coproducts and has co-equalizers, then it is co-complete.

Theorem 95. Set *is co-complete – as are the categories of structured sets* Mon*,* Grp*,* Ab*,* Rng*.* Top *too is co-complete.*

But note that a category can be (finitely) complete without being (finitely) co-complete 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 co-equalizers (see Theorem 61). 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.

21 Subobjects

We have seen how many familiar mathematical constructions, when treated categorially, involve taking limits or colimits. However, Chapter 16 on exponentials has already discussed one notable exception. And in this chapter, we consider another much more basic exception, namely the simple business of forming subobjects (as in subsets, subgroups, subspaces).[1]

Initially, the treatment of subobjects might seem surprising. By the end of the chapter, however, I hope the account should appear in a reasonably attractive light!

21.1 Subsets revisited

(a) Let's start with a reminder, picking up an idea from §15.4.

In Set, we can choose some two-membered set Ω to be a *truth-value object*, treating its members as coding for *true* and *false*. Then a subset $S \subseteq X$ has a unique associated *characteristic function* $\chi\colon X \to \Omega$ which sends $x \in X$ to *true* if $x \in S$ and sends x to *false* otherwise.

Let $\top\colon 1 \to \Omega$ be the function which sends the sole object in our chosen singleton 1 to *true*. $\top \circ !_X$ is then the composite map $X \xrightarrow{!_X} 1 \xrightarrow{\top} \Omega$ which sends every element of X to *true*. We can usefully abbreviate '$\top \circ !_X$' to '$\top_X$'.

The proof of Theorem 64 then showed that (S, i) – where $i\colon S \hookrightarrow X$ is the inclusion function – is a sort of limit; it's an equalizer for the parallel arrows $\chi, \top_X\colon X \to \Omega$.

(b) We might reasonably wonder whether there is a cross-category generalization of this story. Can we always treat subobjects as in effect limits like this? Immediately, however, we face two questions:

[1] There are good introductory treatments of category theory which actually don't discuss this topic at all – see for example Pierce (1991), Simmons (2011), Roman (2017). You might in fact prefer to follow suit and – for the moment, at least – jump over the rest of Part I and make a start on the topics of Part II. But my initial focus in these Notes has been on the way that category theory throws light on the various constructions of ordinary mathematics; and so it would be rather odd to downplay here what seems the very simplest of such constructions, i.e. taking subobjects. Also, those interested in foundational questions will want to make at least their first glancing acquaintance with the notion of a topos sooner rather than later, and will need the categorial notions of subobjects and subobject classifiers for that.

(Q1) First, regarding subobjects as limits means we can only pin them down up to isomorphism: is this acceptable?

(Q2) Second, can we find a truth-value object like Ω in other categories, in a way that enables us then to go on to define subobjects in terms of limits involving the local Ω?

Answers:

(A1) Yes, we can live with identifying subobjects only up to isomorphism.

(A2) No, things really have to go the other way about. In the general case, we need to get a prior notion of subobject into play first. Only *then* do truth-value objects Ω together with characteristic functions as arrows to Ω get defined by their interplay with subobjects as already understood.

This chapter elaborates on (A1); the next chapter explains (A2).

21.2 Subobjects and monic arrows

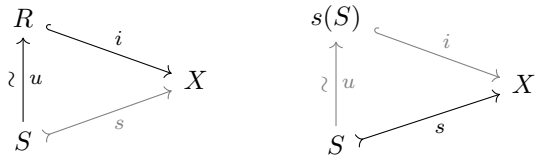
(a) By now you are getting very used to the idea that, at least using the resources of category theory, we don't get a direct handle on the constituents or 'innards' of an object in a category. We only get to see how an object is related 'externally' to others by the arrows of the category. Categorially we come to know an object not by digging inside (so to speak) but by finding the company it keeps: as the old adage has it, "by their friends ye shall know them".

Now ordinarily, when we ask whether S is a *part* of X, we are asking whether the innards of X in some good sense include S. Categorially, however, we can't ask straight out whether one object S in a category is a part of another object X in *that* sense. Rather, for S to count as a subobject of X, we will need instead for there to be the right sort of arrow $S \to X$ (what else?).

"Yes, yes, of course: we want an inclusion arrow!" But hold on! How can we specifically define an *inclusion* arrow in categorial terms? The non-categorial definition, looking e.g. at objects in Set, is that an inclusion arrow $i\colon S \hookrightarrow X$ sends every constituent element inside S to the very same element inside X. However, the best that category theory has to offer by way of an account of 'elements of S' and 'elements of X' are, respectively, arrows $1 \to S$ and $1 \to X$, and *they* can't be the same. What to do?[2]

(b) Let's meditate a bit on these next two diagrams, still in Set, and still (temporarily) assuming that we have a grip on the 'internal' idea of an inclusion map i:

[2]To be clear. At some earlier points – as in §15.4 which we have just referred back to – we have perfectly reasonably invoked our pre-categorial understanding of the idea of an inclusion function in order to illustrate some categorial idea; *now* we are noting a problem about defining such an inclusion function in purely categorical, arrow-theoretic, terms.



Suppose, in the left hand diagram, that we are given a subset R included (in the ordinary sense) in X, with i the associated inclusion function. And suppose we are also given an isomorphism u between S and R. Categorially, i will be monic (being injective), and u will be monic (as any isomorphism is); hence their composition $s = i \circ u$ will be monic by Theorem 15.

Now alternatively suppose, in the right hand diagram, that we are given a monic arrow $s\colon S \rightarrowtail X$. Being injective in Set, this sets up a derived isomorphism u from S to its s-image $s(S)$ which then can be naturally sent by the obvious inclusion map into X, so that again $s = i \circ u$.

Hence, *at least up to isomorphism*, S equipped with a monic arrow $s\colon S \rightarrowtail X$ will be tantamount to a subobject of X defined by inclusion.

(c) OK, now we throw away the ladder we've just climbed up.

Let's accommodate ourselves to the idea that (as with other categorical notions) we really only want to define the notion of a subobject up to isomorphism. In this spirit, and dropping any reliance on the 'internal' notion of inclusion, we can propose the following definition:

Definition 87. (S, s) is a *subobject* of an object X in the category C when S is some object and $s\colon S \rightarrowtail X$ is a monomorphism. △

For example, in the category Grp, we take a subobject of a group G living in the category to be any group S which has a monic homomorphism s into G. We don't care whether the constituent elements of S are 'really' to be found inside G – because, in the spirit of group theory itself, we mostly only care about distinguishing groups up to isomorphism. So we will only be interested in whether S has, so to speak, the right shape to map injectively into G via a monic. Or so goes the story.

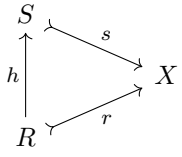
You can see, however, why I didn't introduce this categorial treatment of subobjects much earlier. For you do need to be softened up by enough exposure to attractive up-to-isomorphism categorial treatments of other informal mathematical ideas to be primed to find the initially surprising Defn. 87 half-way sensible!

21.3 Ordering subobjects

I've said that subobjects of X are objects-equipped-with-monics-targetting-X, as it might be $(S, s\colon S \rightarrowtail X)$. Though in fact, since the monic determines its source, we could without loss of information treat subobjects as just being the monic arrows themselves, a point I'll return to.

But whichever way we treat a subobject of X, we can't immediate re-apply Defn. 87 to give us subobjects of subobjects. There is, however, a very natural definition of *inclusion* between subobjects of X:

Definition 88. If (R, r) and (S, s) are both subobjects of X, then we say (R, r) *is included in* (S, s) – or in symbols $(R, r) \leqslant (S, s)$ – if and only if r factors through s, i.e. there is an arrow $h\colon R \to S$ such that $r = s \circ h$. △



Question: Wouldn't it be even more natural to also require the mediating arrow h in our inclusion diagram to be monic too? Answer: We don't need to write that into the definition because h *is* monic by Theorem 15(3).

And note that if $(R, r) \leqslant (S, s)$ because there is an arrow $h\colon R \to S$ such that $r = s \circ h$, then h is unique. For suppose $r = s \circ h = s \circ h'$: then $h = h'$ because s is monic.

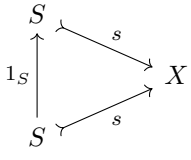
It is then almost trivial to check

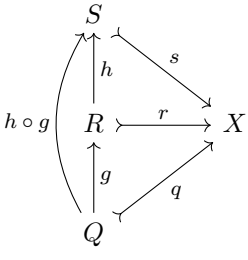
Theorem 96. *(1) The subobjects of X are partially ordered by inclusion (so the notation '$\leqslant$' is apt).*

(2) $(X, 1_X)$ is a maximum in that order.

(3) In a Cartesian closed category with an initial object, $(0, 0_X)$ is a minimum, where 0_X is the unique arrow $0 \to X$.[3]

Proof. (1) Inclusion is easily seen to be reflexive and transitive, as the following diagrams reveal:





(2) Since $1_X\colon X \to X$ is monic (by Theorem 15), $(X, 1_X)$ is a subobject of X. If (R, r) is any subobject of X, then r factors through 1_X since there is an arrow h such that $r = 1_X \circ h$ (trivially, put $h = r$). Hence $(R, r) \leqslant (X, 1_X)$.

(3) Since $0_X\colon 0 \to X$ is monic (by Theorem 17.2(4)), $(0, 0_X)$ is a subobject of X. If (R, r) is any subobject of X, then 0_X factors through r since there is an arrow h such that $0_X = r \circ h$ (put $h = 0_R$, and rely on the fact that there is only one arrow from 0 to X). Hence $(0, 0_X) \leqslant (R, r)$. □

[3]Note that '0_X', so defined, names an arrow from 0 to X. However, '1_X' of course doesn't names an arrow from 1 to X, but the identity arrow from X to itself. So, to be honest, the neatness of the parallel between results (1) and (2) is exaggerated by a slightly sneaky choice of notation here!

21.4 Equivalent subobjects

(a) A natural further definition:

Definition 89. If (R, r) and (S, s) are both subobjects of some object X, then we put $(R, r) \equiv (S, s)$ if and only if $(R, r) \leqslant (S, s)$ and $(S, r) \leqslant (R, r)$. △

The introduced symbol is again appropriate, because it is immediate that $\equiv$ really is an equivalence relation. Then from the definition (for one direction) and by a simple application of Theorem 22 (for the other direction) we get the biconditional

Theorem 97. *$(R, r) \equiv (S, s)$ if and only if r and s factor through each other by an isomorphism and $R \cong S$.* □

(b) Another quick result, again one we certainly want to be true:

Theorem 98. *In* Set *(and other categories where arrows are functions), $(R, r) \equiv (S, s)$ if and only if r and s have the same codomain, i.e. $r(R) = s(S)$.*

Proof. First, suppose $(R, r) \equiv (S, s)$ so there is an isomorphism $i\colon R \to S$ such that $r = s \circ i$. Therefore if $y \in r(R)$, then there is an $x \in R$ such that $y = r(x) = s(i(x))$ where $i(x) \in S$, so $y \in s(S)$. Hence $r(R) \subseteq s(S)$. Likewise $s(S) \subseteq r(R)$. So $r(R) = s(S)$.

Conversely, suppose the monic arrows $r\colon R \rightarrowtail X$, $s\colon S \rightarrowtail X$ have the same image. In Set a monic is an injection, so one-to-one between its source and image. Hence we can define a map $i\colon R \to S$ which matches x in R with the unique y in S such that $r(x) = s(y)$, and then $r = s \circ i$, and $(R, r) \leqslant (S, s)$. Similarly $(S, s) \leqslant (R, r)$. So $(R, r) \equiv (S, s)$. □

(c) Now, in Set, a singleton set $\{\bullet\}$ will have *two subsets* in the ordinary sense; however it will *infinitely many subobjects* in the categorial sense. For any singleton $\{\star\}$ equipped with (the unique possible) arrow $\{\star\} \rightarrowtail \{\bullet\}$ will count as a subobject of $\{\bullet\}$. Indeed our singleton will have more than set-many such subobjects! – since in standard set theory there are too many singletons to form a set.

Is that desperately awkward? No! For note that all the subobjects of that form will evidently be equivalent (check that!), leaving one outlier subobject, the empty set equipped with the empty function. So our singleton set can at least be said to have just two subobjects up to isomorphic equivalence.

And the result generalizes.

Theorem 99. *In* Set*, the subsets of X correspond one-to-one with equivalence classes of subobjects of X.*

Proof. Given a subset S of X (in the ordinary sense), there is a monic inclusion function $s\colon S \hookrightarrow X$, and hence a corresponding subobject (S, s). So consider the map M that sends each subset S to the equivalence class of subobjects containing the corresponding (S, s).

Evidently, M is onto (why?). So it remains to confirm M is one-to-one. So suppose S_1 and S_2 are different subsets. Then the inclusions $s_1 \colon S_1 \hookrightarrow X$ and $s_1 \colon S_1 \hookrightarrow X$ will have different images, so by the previous theorem $(S_1, s_1) \not\equiv (S_2, s_2)$. So M sends S_1 and S_2 to different equivalence classes. □

We get parallel results in other categories too. For example, equivalence classes of subobjects in the category Grp correspond one-to-one to subgroups, and similarly equivalence classes of subobjects in the category Vect_k correspond to vector subspaces, and so on.

But topologists might like to work out why in Top the equivalence classes of subobjects don't straightforwardly correspond to subspaces.

21.5 Defining subobjects, again

(a) I have defined a subobject of X as an object S equipped with a monic arrow $s \colon S \rightarrowtail X$. But fixing the monic fixes its source, so (as I remarked before) we could without loss of information more tersely identify a subobject as simply a monic arrow. Adámek et al. (2009, p. 117), for example, also go my way; but ours is in fact the minority choice. Absolutely nothing hangs on this, of course.

In fact, for aesthetic reasons, I'll sometimes later fall into the snappier idiom. And then, instead of writing e.g. $(R, r) \leqslant (S, s)$ or $(R, r) \equiv (S, s)$ we can write simply $r \leqslant s$ or $r \equiv s$.

(b) But there is a much more significant divergence in definitions that you will also come across, which can now be explained. Having noted that – e.g. in Set – subobjects of an object X typically don't line up one-to-one with subobjects of X as ordinarily understood, it is common for authors to want to get things back in sync by proving our last theorem and then officially (re-)defining subobjects as equivalence classes of subobjects in our original sense. For example Goldblatt (2006, p. 77) and Leinster (2014, Ex. 5.1.40) do this. Some others stick to our definition of subobjects as monics, with or without sources made explicit: e.g. Awodey (2010, §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'.

In these Notes, I will stick to our first, simpler, definition.

22 Subobject classifiers

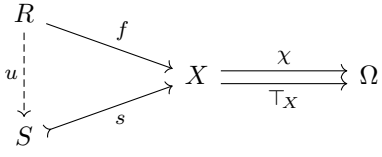
In the last chapter, we took an already-familiar old idea, the idea of a monic arrow, and showed that it gives us all we need to define a quite sensible categorial notion of subobject. In this chapter, by contrast, we introduce two linked new ideas, which give us categorial versions of (sets of) truth-values and of characteristic functions that map to truth-values.

22.1 Subobjects and limits again

(a) Go back to §15.4. A moment's thought will confirm that what we showed there about *inclusion* functions and equalizers in Set applies to *monic* arrows there more generally. Let me spell that out.

Still working in Set, suppose that $s\colon S \rightarrowtail X$ is *any* monic arrow into X, and suppose again that Ω is a set of truth values $\{true, false\}$. Then let $\chi\colon X \to \Omega$ (think 'characteristic function') be the unique map that sends $x \in X$ to *true* iff $x \in s(S)$, and let $\top_X\colon X \to \Omega$ by contrast send every member of X to *true*.

As before, $S \rightarrowtail^{s} X \underset{\top_X}{\overset{\chi}{\rightrightarrows}} \Omega$ is a commuting fork. And moreover it is a limit fork such that any other fork through $\chi, \top_X$ factors uniquely through it. Take again the diagram

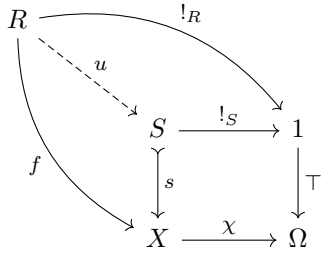


For (R, f) to produce a fork, members of $f(R)$ must be sent to *true* by χ. Hence $f(R) \subseteq s(S) \subseteq X$. Hence, if we define u to send an object $x \in R$ to the pre-image of $f(x)$ under s (which is unique since i is monic), then the diagram commutes. And this u is evidently the only arrow to give us a commuting diagram. Therefore any subobject (S, s) is an equalizer in Set.

(b) That's a very minor tweak on what we had before. But now let's note an equivalent (and not-so-obvious) way of describing this situation, still in Set. Start with the observation that the map $\top_X \circ s\colon S \to \Omega$, which sends everything in S to the value *true*, is trivially equal to the composite map $S \xrightarrow{!_S} 1 \xrightarrow{\top} \Omega$ where

1 is a terminal object in the category. Similarly for the map $\top_X \circ f : R \to \Omega$: this is equal to the composite map $R \xrightarrow{!_R} 1 \xrightarrow{\top} \Omega$.

Hence, re-arranging the previous diagram, the claim that (S, s) equalizes $\chi, \top_X$ in **Set** is equivalent to the following proposition. For any $f : R \to X$ such that $\chi \circ f = \top_X \circ 1_R$ there is a unique $u : R \to S$ which makes the whole diagram commute:



And after our work in §19.1, we now know a snappy way of putting that: *the lower square will be a pullback square.* Any subobject arrow s is a pullback from $\top$ along a suitable χ.

(c) Now, we should be able to carry this linkage between subobjects and limits across to other categories, so long as they have a terminal object, a suitable truth-value object Ω and a *true*-selecting map $\top : 1 \to \Omega$. Excellent!

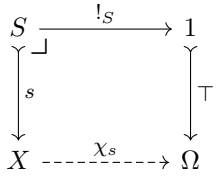
However, to pick up the thought I trailed at the end of §21.1, we *can't* parlay this linkage into an alternative definition of a subobject in terms of a pullback limit – since that would presuppose we *already* have a handle on a general notion of truth-value object, and we don't.

Rather, we need to look at things the other way about. In fact, what we can extract here is a general characterization of a truth-value object Ω and an associated *true*-selecting map $\top : 1 \to \Omega$ (which, recall, must be monic by Theorem 28). We pin down such things across categories by requiring that they interact in the illustrated way with subobjects as already defined.

22.2 Subobject classifiers

All that motivates introducing a new, distinctively categorial, idea:

Definition 90. In a category C with a terminal object 1 and pullbacks, an object Ω and arrow $\top : 1 \rightarrowtail \Omega$ provide a *subobject classifier* $(\Omega, \top)$ if and only if for any $(S, s : S \rightarrowtail X)$ there is a unique *characteristic* arrow $\chi_s : X \to \Omega$ making this a pullback square:



△

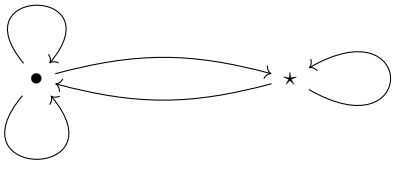
To emphasize: it is not enough that χ_s makes the square commute (after all, the undiscriminating composite of the arrows $!\colon X \to 1$ and $\top\colon 1 \to \Omega$ does that).[1] The characteristic arrow χ_s needs to be, so to speak, the *most discriminating* arrow making the square commute.

This last point means that, in some categories which have subobject classifiers, the *truth-value objects* Ω are significantly more complex than two-element sets. So let's immediately have ...

22.3 An instructive example

Consider the category of graphs.[2] Recall, as noted in §8.1(11), that Graph has a terminal object 1, which is a graph with a single node and one loop. Then

Theorem 100. *The category* Graph *has a subobject classifier. Its truth-value object* Ω *is a graph of this form*



And the required arrow $\top\colon 1 \rightarrowtail \Omega$ *needed to complete the subobject classifier is the graph homomorphism which sends* 1*'s single node to* Ω*'s node* $\bullet$ *and* 1*'s loop to one of* $\bullet$*'s loops.*

Proof sketch. This initially appears entirely mystifying! But suppose we start with a graph X and take a subgraph $(S, s\colon S \rightarrowtail X)$, where s is a monic graph homomorphism of course (so s acts on both nodes and edges). And for brevity, we will slightly abuse language and say that a node n from X is 'in' S when there is a node m in S such that $s(m) = n$; similarly an edge e from X is 'in' S when there is an edge d in S such that $s(d) = e$. In other words, we speak as if s is always inclusion (though, in the general case, it needn't be).

Now think how S looks from the point of view of X:

(i) As far as nodes are concerned, a node n in X can be *in* S or *out*.

(ii) As far as edges are concerned, an edge e in X can be in S, along with its nodes of course, i.e. *everything* about e is also in S.

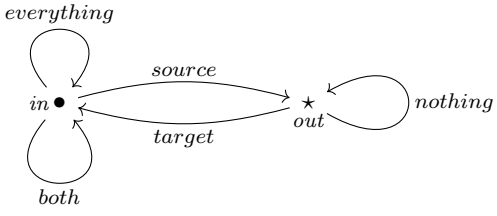
(iii) Otherwise the edge e from X is not in S: but that leaves four possibilities as far as e's nodes are concerned – either *both* its nodes are in S, or just e's *source* is in S, or just e's *target*, or else *nothing* of e's is in S.

[1] Why 'undiscriminating'? Well, in Set for example, that arrow $\top \circ !\colon X \to \Omega$ just sends everything in X to *true*.

[2] Inspired by the brisk presentation in Barr and Wells (1995, p. 319).

Ahah! Compare the situation in Set: looking at one of the subsets S of X from the point of X, we just ask 'is this element of X still in S?'. That's a simple yes/no, two possibility, question – which is why we only need a two-member truth-value set to encode the possibilities. Here in Graph there are more possibilities: when we look at our subgraph S from the point of view of X, there are two possibilities for nodes, and five for edges. Which demystifies why our truth-value object is more complex, a graph with two nodes and five edges, reflecting the different modes of 'inclusion'.

Suppose then that we label the nodes and arrows of Ω like this, reflecting those different possibilities for nodes and arrows in subgraphs:



Our arrow $\top\colon 1 \rightarrowtail \Omega$ now more specifically sends 1's node to Ω's node *in* and 1's loop to Ω's edge *everything*.

OK: so given our subgraph $(S, s\colon S \rightarrowtail X)$, now consider the graph homomorphism $\chi\colon X \to \Omega$ which sends each node n of the graph X to *in* or *out* depending on whether n is in S, and sends each edge e from X to the appropriate arrow in Ω which represents how much of e is in S. Then χ will make the required square commute – and moreover it is the most discriminating graph homomorphism which will do this. Hence, arm-waving a bit, this gives us a pullback square and we are done. □

I won't pause to tighten up the details of that proof sketch, as I have said enough to get across the basic idea, and this is one of those cases where playing with a few more diagrams will convince. I'm more interested in the general moral here: as signalled at the end of the previous section, truth-value objects (in those categories which can have them) can be significantly more complex than simple two-value, yes/no, affairs.

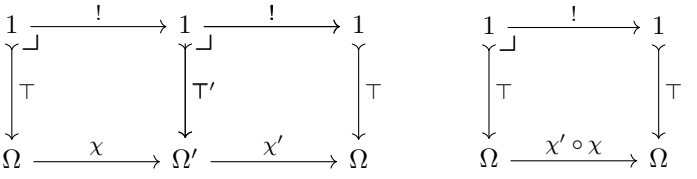
22.4 Four general theorems about subobject classifiers

(a) It is familiar enough from elementary logic that even when we have picked some things to function as truth-values – as it might be, 0 and 1 – it is arbitrary which of them we care to identify as *true*. Similarly here: even once we've fixed on a 'truth-value object' Ω, it can in many cases be arbitrary which arrow $1 \rightarrowtail \Omega$ to choose as $\top$ to form a subobject classifier. This is evidently so in Set; and it is the case in Graph too – given a graph of the right shape to be the truth-value object, it is up for grabs which of the loops on the two-loop node we decide to be the value of $\top$ when applied to 1's loop.

However, if a category has a subobject classifier at all, the truth-value object Ω itself will be determinate up to isomorphism:

Theorem 101. *If a category has subobject classifiers* $(\Omega, \top)$, $(\Omega', \top')$, *then* $\Omega \cong \Omega'$.

Proof. Theorem 28, to repeat, tells us arrows with source 1 are monic, so $(1, \top)$ counts as a subobject of Ω. Given $(\Omega', \top')$ is a subobject classifier, then there is a unique χ such the left square in diagram (i) below is a pullback:



Further, since $(1, \top')$ counts as a subobject of Ω' and $(\Omega, \top)$ is a subobject classifier, there is also a unique χ' such the right square in (i) above is a pullback. Hence by Theorem 84 the whole rectangle, equivalently the square (ii), is a pullback.

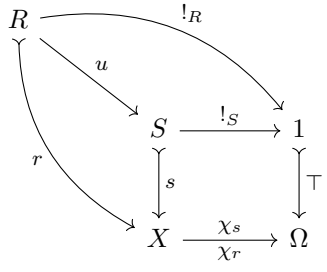
But by the very definition of the subobject classifier $(\Omega, \top)$, given the particular subobject $(1, \top)$ there must be a unique arrow $\Omega \to \Omega$ that will make a pullback square like (ii), and obviously 1_Ω will do the trick. Hence we must have $\chi' \circ \chi = 1_\Omega$. Exactly similarly, $\chi \circ \chi' = 1_{\Omega'}$.

Therefore χ and χ' are mutually inverse isomorphisms. And we are done. □

(b) To check everything is working as intended, let's verify the following claim which we indeed want to be true (why?):

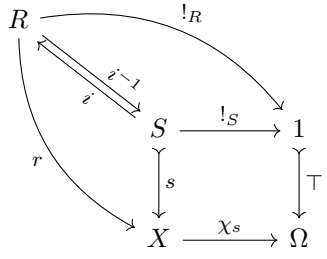
Theorem 102. *Assuming we are in a category with a subobject classifier,* (R, r) *and* (S, s) *are equivalent subobjects of* X *if and only if* $\chi_r = \chi_s$.

Proof of 'if'. Suppose $\chi_r = \chi_s$ and consider this diagram:



The outer 'square' commutes by definition of χ_r. The inner square is a pullback, so the wedge with vertex R has to factor through the wedge with vertex S by a some u (in fact, uniquely so). So we have $r = s \circ u$. But that means $(R, r) \leqslant (S, s)$. Similarly we can prove $(S, s) \leqslant (R, r)$. Therefore $(R, r) \equiv (S, s)$. □

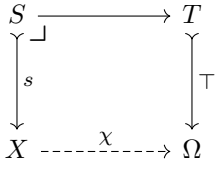
Proof of 'only if'. Suppose $(R, r) \equiv (S, s)$ and now look at this variant diagram:



The inner square is given as a pullback. By the equivalence hypothesis, there is an isomorphism i between R and S such that the bottom triangles commute, and the top triangles trivially commute. So now we find a use for one of our results about pullbacks, the otherwise unexciting lemma Theorem 86 – it shows that the outer square with vertex R is a pullback. But then by definition of the subobject classifier $(\Omega, \top)$, the bottom arrow from X to Ω must be equal to χ_R. Therefore $\chi_R = \chi_S$. □

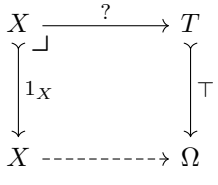
(c) Next, let's show that it was actually unnecessary to explicitly require that our subobject classifier involves an arrow from a *terminal* object to the truth-value object Ω. For we have following theorem:

Theorem 103. *Suppose in* C *that there is an object* Ω *and monic* $\top\colon T \rightarrowtail \Omega$ *such that for any* $(S, s\colon S \rightarrowtail X)$ *there is a unique arrow* $\chi\colon X \to \Omega$ *making a pullback square of this form:*



Then T *is terminal in* C.

Proof. By assumption, for any X in C there must be at least one arrow $X \to T$ making this next diagram a pullback.



Suppose there are two candidates for that mystery arrow, f and g. Then the bottom arrow of the squares would be respectively equal to $\top \circ f$ and $\top \circ g$. But by assumption, there is there is a unique arrow $X \to \Omega$ making a pullback square. So $T \circ f = T \circ f$, and therefore $g = f$ since $\top$ is monic. Hence there is one and only one arrow from any X to T and T is terminal. □

(d) Finally, recalling Defn. 35, a balanced category is one where every arrow which is both monic and epic is an isomorphism. We immediately have

Theorem 104. *If a category has a subobject classifier, it will be balanced.*

Proof. Assume C has a subobject classifier. As we saw, a monic arrow $s\colon S \to X$ in C equalizes the parallel arrows $\top_X, \chi_s\colon X \to \Omega$. So an epic monic will be an epic equalizer, and hence an isomorphism by Theorem 63. So C is balanced. □

22.5 Which categories have subobject classifiers?

Back to basics. We know Set has a subobject classifier. The same two-object classifier works in Finset, and even in 2set, the category of sets which have no more than two members. Note, though, that the impoverished category 2set doesn't have all binary products for the tritest of reasons – the product of two two-member sets needs to have four members! So a category can (radically!) fail to be Cartesian closed yet still have a subobject classifier.

Things can also go the other way around. A category can be rich enough to be Cartesian closed yet lack a subobject classifier: Pos is an example. Why? Because Pos is a not balanced category as we proved in §7.5, and hence it can't have a subobject classifier by Theorem 104.

The category Grp by contrast is balanced: but that too doesn't have a subobject classifier: I'm not going to pause to prove that. But some other cases are easier to see. The category of rings doesn't have any ring homomorphisms from its terminal object, the trivial one-object ring, to any other non-trivial ring: so we can't get a non-trivial Ω and arrow $\top\colon 1 \to \Omega$. So the possibility of having a subobject classifier in Rng is immediately squashed. Similarly some other categories simply lack the needed arrows from their terminal objects.

It would, however, again take us too far off course to pursue further tests for having or lacking subobject classifiers. But the take-away message is that, while subobjects are easy to come by, subobject classifiers are not at all so common. And after all, why in general, should a category have a specialized object, a truth-value object, such that arrows into it line up one to one with equivalence classes of subobjects? Having a subobject classifier is in fact a very strong property that can combine with other features of a category in unexpected ways.

22.6 Truth vs falsehood (a question raised)

I hope the earlier discussion prompted a very natural question in your mind! A subobject classifier involves a map $\top\colon 1 \to \Omega$ which we can think of as picking out the value *true*. Can we define a companion map $\bot\colon 1 \to \Omega$ which similarly picks out a value we can think of as *false*?

Consider the situation in Set, which has an initial object 0 (the empty set). There must be a unique arrow $0 \to 1$ and this will be a monomorphism (by

Theorem 72). Like any monic arrow, then, this arrow gives us a subobject, i.e. subset, of 1. Which subset? The empty set, of course.

And what's the characteristic arrow $1 \to \Omega$ associated with this subobject? Intuitively, it has to be the function $\perp$ which maps the sole member of the singleton 1 to *false*.

So now the obvious move is to try to generalize that idea. But we will need to be still working in a category which is sufficiently like Set for similar constructions to go through – we'll want a subobject classifier to give us a characteristic function, and also want an initial object to play with. We will also want Theorem 72 still to be available, so it will be good if our category is Cartesian closed category. So, putting those thoughts together, we will wanting to be working in what's called an *(elementary) topos*, or at least in a near miss. Let me explain!

23 Toposes

We have arrived at a significant juncture: we can now motivate the key idea of a *topos*, which is a category rich enough to provide a framework in which a wide range of constructions is available. So I'll give a definition and some simple examples. I'll then return to pick up the question we left hanging, of how to define falsehood in a suitable category.

Topos theory famously becomes fairly tough going: but don't panic – there will be nothing scary *here*!

23.1 Defining an elementary topos

(a) Looking back, we have seen that if a category (1) is finitely complete – has all finite limits – then it will have a terminal object and we can e.g. construct products of any objects in the category and e.g. construct inverse images too and other pullbacks. If our category (2) is also finitely co-complete – has all finite colimits – then we'll get an initial object and we can in addition e.g. form all quotients. Add (3) all exponentials and our category will, for any objects A to B, also supply a corresponding object B^A which behaves like a function space collecting the arrows from A to B.

And we have now seen that if a category (4) has a subobject classifier, it has arrows that work like analogues of characteristic functions. This in fact enables a range of further constructions. Let me briefly mention one. Once we have a truth-value object Ω in play, together with exponentiation, then for any object X there will be a corresponding exponential Ω^X. Now, in Set, where Ω can be thought of as a simple two-object set, Ω^X is a familiar object tantamount to the powerset of X (every subset of X corresponds to a function from X to $\{\mathit{true}, \mathit{false}\}$). The point generalizes to any category where we have (1) to (4): Ω^X is available to behave like a 'power object' for X, and – for example – subobjects of X, i.e. monic arrows targetting X, can be shown to nicely correspond to elements of the power object, i.e. to arrows $1 \to \Omega^X$. We needn't pause over the details for current purposes.

(b) Putting everything together, then, a category which has all of (1) to (4) will provide an arena in which we can implement a lot of 'ordinary' mathematical constructions. Not surprisingly, then, there is a standard term for such a well-endowed category:

Definition 91. A category is an *(elementary) topos*[1] if and only if it

(1) is finitely complete,
(2) is finitely co-complete,
(3) has all exponentials,
(4) has a subobject classifier. △

We will take this as our official definition even though it in fact involves a redundancy. It turns out that condition (2) is in fact automatically satisfied if the other three conditions are satisfied. However, this is *not* easy to prove using the apparatus we currently have available: even proving the existence of an initial object given (1), (3) and (4) isn't entirely simple. So we'll keep (2) explicitly built in.

But the fact that (2) *can* be be deduced from the other conditions explains why you'll often find an elementary topos defined, briskly, as being a (properly) Cartesian closed category with a subobject classifier.[2]

(c) Why that qualifier 'elementary'? It marks a contrast with the original stronger notion of a *Grothendieck* topos which has its roots in hard-core algebraic geometry. I judge it would take us too far afield, however, even to begin to explain Grothendieck's notion.

So for now, let's simply record the fact that the idea of a topos does come in both a stronger (and scarier) and a weaker (and notably more friendly) version. And there really should be nothing at all puzzling about the more elementary notion which we are concerned with here in these Notes: to repeat, it is just the notion of a category which usefully combines a number of easily-explained and now-familiar features.

23.2 Examples

(a) Note that – as with so many definitions of types of structure – Defn. 91 admits a boring minimal case. It is easily checked that the category $\mathbf{1}$ comprising a single object and its identity arrow forms a topos. The same goes for a category formed by lots of isomorphic copies of $\mathbf{1}$. Echoing Defn. 74, then, let's say:

Definition 92. A *degenerate* topos is a category where any object has just one isomorphism to itself and one isomorphism to any other object, and there are no other arrows.

Picking up from Theorem 73 it is immediate that

Theorem 105. *A topos where* $0 \cong 1$ *is degenerate.*

[1] A vexed question: is it one topos and many toposes? or many topoi? I'm with team Johnstone (see his 1997, p. xx) in preferring the first.

[2] For the idea of being *properly* Cartesian closed see Defn. 73 and its footnote.

(b) For non-trivial examples, start with the paradigm case of the category Set which has all limits and colimits, all exponentials, and a subobject classifier.

Further, if we just look at the *finite* sets, all the needed constructions are still available: so FinSet is a topos. But any finite set is isomorphic to a finite ordinal: so all the 'unique-up-to-isomorphism' categorial constructions we can do with finite sets can be done with finite ordinals. Hence we also have

Theorem 106. FinOrd *is a topos.* □

And there is a sense in which FinOrd is the minimal interesting case of a topos, for it can be shown that every non-degenerate topos has a copy of FinOrd sitting inside it. So putting aside the degenerate cases, every topos has an infinite number of objects.

(c) Other examples? We know that e.g. Set_* is not even Cartesian closed. And while Pos is Cartesian closed, it isn't a topos because it doesn't have a subobject classifier (see §22.5). But Graph *is* a topos. Limits are relatively easy to establish and Theorem 100 tells us that the category has a subobject classifier: we haven't proved that the category has all exponentials, but that isn't hard once we have some of the apparatus we introduce in Part II.

The slice category Set/I is also a topos for any set I. It is relatively easy to show that such slice categories have terminal objects and all pullbacks (and hence are finitely complete). It is not too hard either to show that they have exponentials and subobject classifiers: but I won't pursue the details here. Now, recall from §6.3 that an object of Set/I will be a set S equipped with a function $f\colon S \to I$. But such a function f partitions S into disjoint sets S_i, where $x \in S_i$ iff $f(x) = i \in I$. So we can equally think of an object of Set/I as a bundle of sets (perhaps some empty) indexed by I. And we can get other toposes by taking indexed bundles of more structured sets such as topological spaces. But we needn't go into details here.

Next, recall from §4.7(a) the idea of an M-Set category, whose objects are sets equipped a family of operations which behave like the monoid M. For any monoid M, the corresponding category M-Set is again a topos. This is a useful result, because ringing the changes on M will give us toposes with e.g. differently structured truth-value objects. Again, I just report this and won't sketch proofs. For perhaps all we need for our limited purposes here is the thought that categories which are rich enough to be toposes are indeed many and various.[3]

23.3 Truth vs falsehood (a question answered)

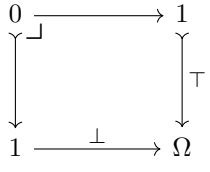
(a) Let's return to the question we left hanging at the end of the last chapter: given a suitable setting – supposing we are in a topos, let's say – can we define

[3] You'll find more details about slice categories, M-Sets and bundles as toposes in Goldblatt (2006, §4.5). And if you want further examples, see §4 of the nLab page ncatlab.org/nlab/show/topos.

a map $\bot\colon 1 \to \Omega$ which picks out a value we can think of as *false*, a companion to $\top\colon 1 \to \Omega$ which arrows in on *true*?

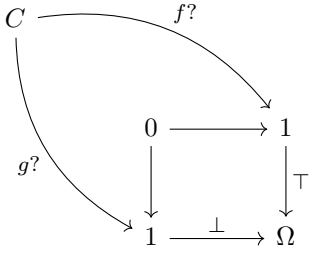
In §22.6, we saw that in Set, we can introduce a map $\bot\colon 1 \to \Omega$ targeting *false* as the characteristic arrow associated with the subobject $\emptyset \rightarrowtail 1$. This motivates the following natural generalization:

Definition 93. Suppose E is an (elementary) topos. Then $\bot\colon 1 \to \Omega$ in E is the unique arrow which makes this a pullback diagram:



△

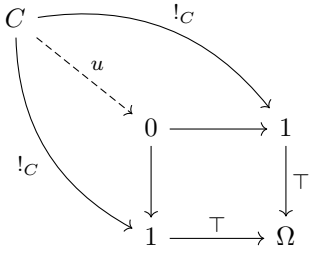
You might wonder: how can that be a pullback square if, as we want (see the next theorem) $\top$ and $\bot$ are distinct? Well, consider the situation in Set for example:



If $C \neq 0$, there can be *no* arrows f, g in Set for which $\top \circ f = \bot \circ g$ (for any member of C gets sent to different values by the two composites). Hence if $C \neq 0$ it is vacuously true that whenever $\top \circ f = \bot \circ g$ (which is never!) there is an arrow from C to 0 giving the required commuting triangles. So our square *will* count as a pullback, albeit vacuously.

Theorem 107. *So long as the topos* E *is not degenerate,* $\bot \neq \top$.

Proof. We suppose $\bot = \top$ and derive degeneracy. Our supposition makes the lower square here a pullback:



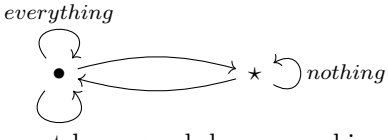
Hence, for any C, since the composite arrows from C to Ω along the two outer routes are trivially identical, there must be an arrow $u\colon C \to 0$ completing

the diagram since this is a pullback. But C was arbitrary, and so there will in particular be an arrow $1 \to 0$.

But we know from the proof of Theorem 73 that the existence of such an arrow in a Cartesian closed category leads to degeneracy. □

(b) Degenerate cases apart, then, the truth-value object of a topos will have distinct elements corresponding to *true* and *false*, i.e. distinct arrows $\top : 1 \to \Omega$ and $\bot : 1 \to \Omega$. Must these two exhaust the possibilities? Must Ω have just two elements?

Take our example of the subobject classifier in Graph from §22.3. A terminal object 1 in that category is a graph with a single node and a single loop on that node. And that terminal graph can be sent by a graph homomorphism to the truth-value object Ω in *three* ways. As explained before, the arrow $\top : 1 \to \Omega$ sends the 1's node to the 'in' node $\bullet$ and 1's loop to the 'everything' loop. Similarly, $\bot : 1 \to \Omega$ should send 1's node to the 'out' node $\star$ and send 1's loop to to the 'nothing' loop. That leaves the third option of an arrow sending 1's node to the 'in' node and 1's loop to the unmarked loop.



23.4 Negation

Like any arrow from 1, the arrow $\bot : 1 \to \Omega$ is monic (see Theorem 28). So, assuming we are in a nice enough category like a topos, $(1, \bot)$ is a subobject of Ω with a classifying characteristic arrow. Here it is:

Definition 94. The arrow $\neg : \Omega \to \Omega$ is the characteristic arrow of $(1, \bot)$, making this is pullback:

$$\begin{array}{ccc} 1 & \longrightarrow & 1 \\ \downarrow & \lrcorner & \downarrow \top \\ \Omega & \xrightarrow{\neg} & \Omega \end{array}$$

△

The negation-like notation '$\neg$' for that characteristic arrows seems quite appropriate, for we have the following neat result:

Theorem 108. *$\neg \circ \bot = \top$ and $\neg \circ \top = \bot$.*

Proof. The first claim is true by definition. For the second, consider the left diagram below, which commutes because each square does:

$$\begin{array}{ccccc} 0 & \longrightarrow & 1 & \longrightarrow & 1 \\ \downarrow & \lrcorner & \downarrow & \lrcorner & \downarrow \top \\ 1 & \xrightarrow{\top} & \Omega & \xrightarrow{\neg} & \Omega \end{array} \qquad \begin{array}{ccc} 0 & \longrightarrow & 1 \\ \downarrow & \lrcorner & \downarrow \top \\ 1 & \xrightarrow{\neg \circ \top} & \Omega \end{array}$$

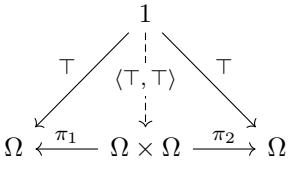
One square is the pullback defining $\bot$, reflected about the diagonal. The other square is the pullback which introduces $\neg$. So the overall rectangle, equivalently the right-hand diagram, is a pullback too by the pullback lemma Theorem 84.

But $\bot$ is by definition the sole arrow $1 \to \Omega$ which completes the pullback from $\top\colon 1 \to \Omega$ to $!\colon 0 \to 1$. Hence, $\neg \circ \top = \bot$. □

23.5 Conjunction, and more logic

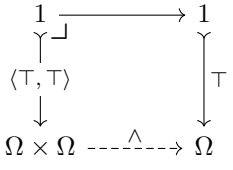
(a) We can easily take the next step in introducing logical operators.

Still working in topos (or another nice enough category), consider the arrow from 1 to the product $\Omega \times \Omega$ defined by the following product diagram:



This arrow $\langle \top, \top \rangle$, being monic, gives us a subobject of $\Omega \times \Omega$; and therefore this will have a classifying characteristic arrow:

Definition 95. The arrow $\wedge\colon \Omega \times \Omega \to \Omega$ is the characteristic arrow of the subobject $(1, \langle \top, \top \rangle)$, making this a pullback:



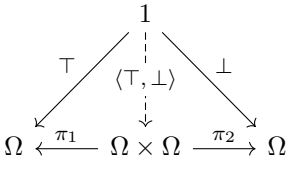
△

I now claim that, as the notation suggests, $\wedge$ acts like a conjunction:

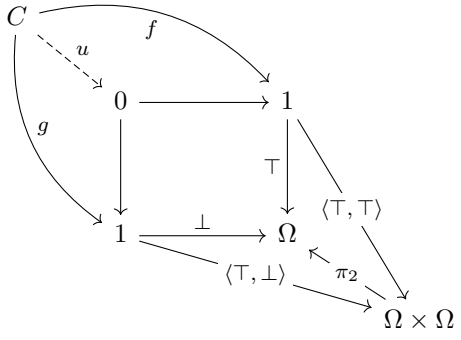
Theorem 109. $\wedge \circ \langle \top, \top \rangle = \top$, *while* $\wedge \circ \langle \top, \bot \rangle = \wedge \circ \langle \bot, \top \rangle = \wedge \circ \langle \bot, \bot \rangle = \bot$.

Proof. I'll spell out the proof that $\wedge \circ \langle \top, \bot \rangle = \bot$.

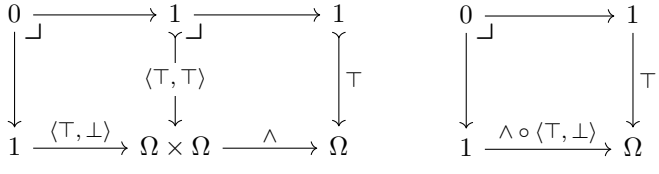
So first let's remind ourselves of the definition of the arrow $\langle \top, \bot \rangle\colon 1 \to \Omega \times \Omega$:



And now let's stare at the following diagram and chase some arrows around in the familiar sort of way:



By hypothesis, the inner square commutes, as do the two added lower triangles. Now suppose that f and g are such that $\langle \top, \top \rangle \circ f = \langle \top, \bot \rangle \circ g$. Then, of course, $\pi_2 \circ \langle \top, \top \rangle \circ f = \pi_2 \circ \langle \top, \bot \rangle \circ g$, and therefore $\top \circ f = \bot \circ g$. But then, because the inner square is a pullback, there must be a unique arrow u making the diagram commute. Which means, in sum, that the left-hand square in diagram (i) below is a pullback:



But the right-hand square in (i) is also a pullback, so the overall rectangle is a pullback. In other words, (ii) is a pullback square. But by definition $\bot$ is the unique lower arrow making that a pullback. So, as we wanted to show, $\wedge \circ \langle \top, \bot \rangle = \bot$.

The other three components of our Theorem 109 are proved similarly (exercise!). Hence we are done. □

(b) In a rich enough category like a topos, then, we can start to model some logical notions, and prove results tantamount to e.g. the truism that the conjunction of a truth and a falsehood is false.

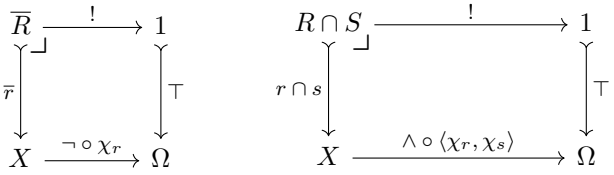
Ok, *that* result is pretty trivial. Fair point. But we have to begin somewhere, and we can in fact go on develop a lot more. For a start, we can similarly define arrows $\vee$ and $\Rightarrow$ that stand to disjunction and the conditional as our arrows $\wedge$ and $\neg$ stand to conjunction and negation – though this is actually a little more intricate. Then, with arrow versions of our four basic connectives in play, we can next ask: given a topos, what does its 'internal propositional logic' look like? (Spoiler: it will not in general be a classical two-valued logic.). And we can also then ask: is there an analogous way of getting something tantamount to quantifiers into play?

23.6 More about subobjects

(a) Those last questions are intriguing, but I don't want to pursue them further in Part I of these Notes. Instead, I just want to link our brief discussion of negation and conjunction to the earlier discussion of subobjects.

Suppose, then, that we are working in some topos. Take two subobjects of X, namely (R, r) and (S, s), with their respective characteristic arrows χ_r and χ_s. We can form two corners $X \xrightarrow{\neg \circ \chi_r} \Omega \xleftarrow{\top} 1$ and $X \xrightarrow{\wedge \circ \langle \chi_r, \chi_s \rangle} \Omega \xleftarrow{\top} 1$. And like any corners in a topos, these will have pullbacks.

So here are the relevant squares, with the pullback objects and arrows suggestively labelled, and remembering that pulling back a monic yields a monic:



And these diagrams can be associated with a couple of definitions:

Definition 96. Suppose χ_r is the characteristic arrow of a subobject of X, namely (R, r). Then a subobject which results from pulling back $\top \colon 1 \to \Omega$ along $\neg \circ \chi_r$ is a *(pseudo-)complement* of (R, r). △

Definition 97. Suppose χ_r and χ_s are the characteristic arrows of two subobjects of X, namely (R, r) and (S, s). Then a subobject which results from pulling back $\top \colon 1 \to \Omega$ along $\wedge \circ \langle \chi_r, \chi_s \rangle$ is an *intersection* of (R, r) and (S, s). △

Note: defining an arrow as the result of a pullback doesn't fix it uniquely. However, we can apply Theorems 78 and 97, which immediately give us

Theorem 110. *If $(\overline{R}, \overline{r})$ and $(\overline{R}', \overline{r}')$ are complements of (R, r), then $(\overline{R}, \overline{r}) \equiv (\overline{R}', \overline{r}')$. Likewise, if $(R \cap S, r \cap s)$ and $(R \cap S', r \cap s')$ are intersections of (R, r) and (S, s), then $(R \cap S, r \cap s) \equiv (R \cap S', r \cap s')$.* □

In short, complements and intersections are defined up to equivalence of subobjects.

And why are our definitions sensible? Consider the diagram on the left, and think about the situation in Set. Then the upper path in our diagram sends any element of the set $\overline{R}$ round to *true*. And so, to get a commuting square, the arrow $\overline{r}$ must send any element of $\overline{R}$ to some element of X that *isn't* mapped to true by χ_r. In other words, $\overline{r}$ must send an element of $\overline{R}$ into the *complement* of $r(R)$ in X. However, our square is a pullback, i.e. a limiting case. So in the limiting case $\overline{r}$ will map $\overline{R}$ not just to part but to the whole of that complement. In this sense, the subobject $(\overline{R}, \overline{r})$ behaves in Set like the complement of the subobject (R, r). Hence our chosen notation, of course. (It will emerge shortly why the qualification 'pseudo' is cautiously inserted into Defn. 96.)

Now consider the diagram on the right, and think again about the situation in Set. A moment's reflection tells us that the subobject $(R \cap S, r \cap s)$ behaves like the intersection of (R, r) and (S, s).

This is very nice, then. In a topos, we not only can talk of subobjects of a given object, where these can be interrelated by inclusion, but we further get the beginnings of an algebra of subobjects, akin to the familiar algebra of subsets of a given set in pre-categorial mathematics.

(b) "Hold on! Isn't there an alternative plausible definition for the complement of a subobject, as follows?"

Definition 96* Suppose χ_r is the characteristic arrow of a subobject of X, namely (R, r). Then the *(pseudo-)complement* of (R, r) is the subobject which results from pulling back $\perp\colon 1 \to \Omega$ along χ_r. △

A little reflection suggests that this is equally natural. However, there is a simple enough result:

Theorem 111. *$(\overline{R}, \overline{r})$ is the complement of (R, r) according to Defn. 96 if and only it is the complement by Defn. 96* too.*

But I'll leave the proof for now as an (easy) challenge to be taken up in the next section.

(c) "Hold on again! Isn't there an equally natural alternative definition for the intersection of two subobjects of a given object X? Recall the discussion of §19.2(c) where we defined the intersection of objects in Set using a pullback. Pursuing that line of thought, wouldn't the following make an equally sensible definition for the intersection of subobjects?"

Definition 97* If (R, r) and (S, s) are both subobjects of X, their *intersection* is $(R \cap S, r \cap s)$, where $R \cap S$ is the vertex of the pullback over the corner formed by r and s,

$$\begin{array}{ccc} R \cap S & \xrightarrow{\ i_R\ } & R \\ \downarrow{\scriptstyle i_S} & \lrcorner & \downarrow{\scriptstyle r} \\ S & \xrightarrow{\ s\ } & X \end{array}$$

and $r \cap s\colon R \cap S \to X$ is the diagonal, equalling the composite arrow round the square on either path. △

(The labels 'i_R' and 'i_S' are supposed to suggest injections of $R \cap S$ into R and S respectively – being pullbacks of monics, these arrows are indeed monics too.)

Another fair point. But fortunately we again don't have to choose between these definitions, by the next result:

Theorem 112. *$(R \cap S, r \cap s)$ is the intersection of (R, r) and (S, s) according to Defn. 97 if and only if it is the intersection by Defn. 97* too.*

Proving this is another challenge for the next section.

(d) By Defn 93 and Defn. 97*, the intersection in a topos of the relevant Ω's subobjects $(1, \top)$ and $(1, \bot)$ will be equivalent to $(0, 0_\Omega)$ – i.e. will be an initial object equipped with its unique arrow to Ω. In the snappier notation, then, we have $\top \cap \bot \equiv 0_\Omega$.

But applying Defn. 90, the characteristic of the arrow $\top$ is 1_Ω. Hence by Defn 96, we get a complement of $(1, \top)$ by pulling back $\top$ along $\neg \circ 1_\Omega = \neg$. Which of course just gives us $(1, \bot)$. Therefore – at least having, as usual, fixed on a choice of terminal object 1 in defining elements – we have $\overline{\top} = \bot$.

So, putting things together, we can conclude

Theorem 113. $\top \cap \overline{\top} \equiv 0_\Omega$. □

Now, this is in fact a (very) special case of a (very) much more general result:

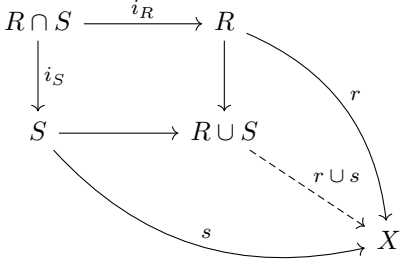
Theorem 114. *In a topos, if (R, r) is any subobject of some object X, then $r \cap \overline{r} \equiv 0_X$.*

Evidently, we want this result if $\overline{r}$ is to count in some sense as a complement of r. But again, partly not to break the flow, I'll leave the proof as a third challenge to be returned to.

(e) What about the *union* of subobjects? We could define this in the style of Defn. 97, if only we had already defined a disjunctive arrow $\vee\colon \Omega \times \Omega \to \Omega$ to put alongside the conjunctive $\wedge\colon \Omega \times \Omega \to \Omega$. However, to avoid unncessary complications, we haven't done that.

Still, we do have to hand the resources to introduce unions in the style of Defn. 97*. Recall from §19.6 how we can define unions in Set by pushouts. So carrying over this idea, we will say:

Definition 98. If (R, r) and (S, s) are both subobjects of X, their *union* is $(R \cup S, r \cup s)$, where $R \cup S$ is the vertex of the pushout from the wedge formed by $R \xleftarrow{i_S} R \cap S \xrightarrow{i_R} R$ as in Defn 97*,



and $r \cup s\colon R \cup S \to X$ is the unique arrow making the diagram commute. △

This definition is in good order because the outer paths from $R \cap S$ to X are equal by the definition of $R \cap S$. And hence by the definition of a pushout there must be a unique arrow from $R \cup S$ to X making the diagram commute. A little reflection confirms that this is a sensible enough definition e.g. in Set.

The union of a subset of X and its complement (in the ordinary sense) is of course X. What about the categorial union of a subobject of X and its categorial complement? Remembering Theorem 96 which tells us that $(X, 1_X)$ is the maximal subobject of X, let's say

Definition 99. A topos is *complemented* if for every subobject (R, r) of any object X, $r \cup \overline{r} \equiv 1_X$.

So a complemented topos is one where (pseudo-)complements do work like the real thing. However, being complemented doesn't come for free:

Theorem 115. *Not every topos is complemented.*

Proof sketch. Think, for example, about Graph. We noted in §23.3 that there are *three* arrows from the terminal object 1 to its truth-value object Ω, namely $\top$, $\bot$ (which we can now also think of as $\overline{\top}$), and a third one. So arm-waving just a bit, $\top \cup \overline{\top}$ won't cover all the cases and give us back 1_Ω. □

23.7 Meeting the challenges

That's the main business of the chapter done, and this concluding section really is eminently skippable!

But for completists – or for enthusiasts who just like this kind of thing – I'll walk through some proof ideas for Theorems 111, 112 and 114.

(a) *Proving our alternative definitions of complements are equivalent.* Let's begin by thinking about complements. Look at the following simple diagram:

$$\begin{array}{ccccc} \overline{R} & \xrightarrow{!} & 1 & \xrightarrow{!} & 1 \\ \downarrow{\scriptstyle \overline{r}} & \lrcorner & \downarrow{\scriptstyle \bot} & \lrcorner & \downarrow{\scriptstyle \top} \\ X & \xrightarrow{\chi_r} & \Omega & \xrightarrow{\neg} & \Omega \end{array}$$

The right-hand square is the pullback defining $\neg$; the whole rectangle is the pullback defining a complement of (R, r) according to Defn. 96. So according to Theorem 84 the left square is a pullback too. Which means that $(\overline{R}, \overline{r})$ as defined by Defn. 96 is also a complement as defined by Defn. 96*.

We can obviously enough also read the same diagram as showing that given $(\overline{R}, \overline{r})$ as defined by Defn. 96*, i.e. given the left pullback square, the whole rectangle is a pullback by Theorem 84, showing that this $(\overline{R}, \overline{r})$ is also a complement by Defn. 96. Which gives us Theorem 111. □

(b) Next we show that an intersection according to Defn. 97* is indeed an intersection by the lights of Defn. 97.

The proof idea. Obviously, we somehow need to end up with the arrow $\wedge \circ \langle \chi_r, \chi_s \rangle$ in play. In other words, we need to be looking at a diagram which contains the composite

$$X \xrightarrow{\langle \chi_r, \chi_s \rangle} \Omega \times \Omega \xrightarrow{\quad \wedge \quad} \Omega.$$

And, looking on the right, what else might we expect to find than the following?

$$\begin{array}{ccccc} & & 1 & \xrightarrow{\quad ! \quad} & 1 \\ & & \downarrow{\scriptstyle \langle \top, \top \rangle} & & \downarrow{\scriptstyle \top} \\ X & \xrightarrow{\langle \chi_r, \chi_s \rangle} & \Omega \times \Omega & \xrightarrow{\quad \wedge \quad} & \Omega \end{array}$$

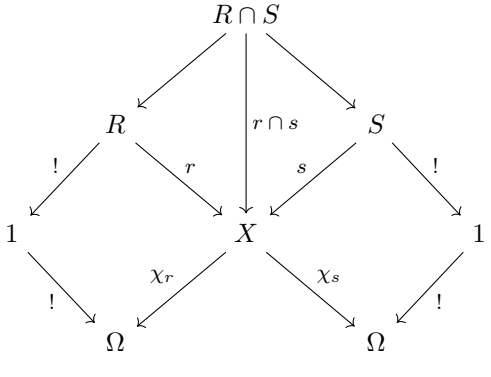
Then, to get $(R \cap S, r \cap s)$ from Defn. 97* into the picture, there seems to be nothing for it than to hope that this gives us a commuting diagram:

$$\begin{array}{ccccc} R \cap S & \xrightarrow{\quad ! \quad} & 1 & \xrightarrow{\quad ! \quad} & 1 \\ \downarrow{\scriptstyle r \cap s} & & \downarrow{\scriptstyle \langle \top, \top \rangle} & & \downarrow{\scriptstyle \top} \\ X & \xrightarrow{\langle \chi_r, \chi_s \rangle} & \Omega \times \Omega & \xrightarrow{\quad \wedge \quad} & \Omega \\ & \text{(L)} & & \text{(R)} & \end{array}$$

In fact we want more. We know the right-hand square (R) is a pullback. If the left-hand square (L) is a pullback too, then we are in business! For in that case, the outer rectangle will be a pullback. But that rectangle is trivially equivalent to the pullback square in Defn. 97.

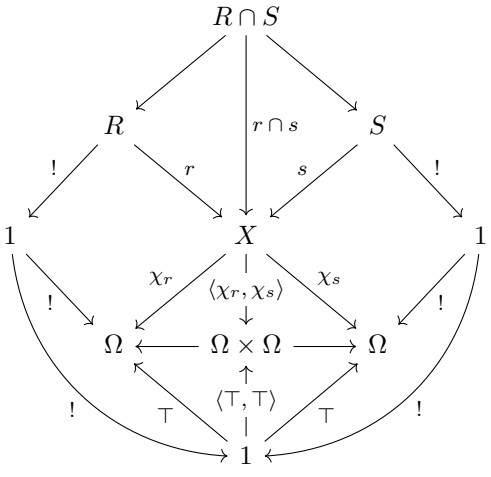
In sum: if $(R \cap S, r \cap s)$ from Defn. 97* makes (L) a pullback, then that subobject is an intersection of (R, r) and (S, s) as defined by Defn. 97. □

Proving (L) is a pullback. Starting from Defn. 97*, and getting χ_r and χ_s into the picture using their defining pullbacks, we get the three squares of following diagram.



We then complete the diagram in the obvious(?) way. The square (L) which we want to prove is a pullback involves two product arrows into $\Omega \times \Omega$; so let's now get *them* into the picture, in the added bottom square below. (The curved arrows linking the various '1's are there simply as a device to avoid identifying them

and then having to draw a three-dimensional diagram which is more difficult to read!)

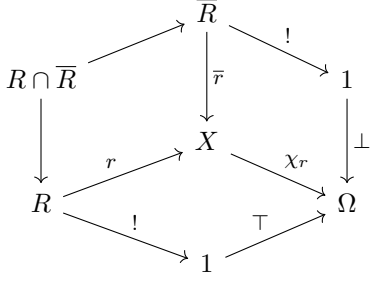


OK: we can now read off from this commuting diagram that the downward arrow $R \cap S \xrightarrow{r \cap s} X \xrightarrow{\langle \chi_r, \chi_s \rangle} \Omega \times \Omega$ equals the results of the long trip around (whether to the left or right) which equals $R \cap S \xrightarrow{!} 1 \xrightarrow{\langle \top, \top \rangle} \Omega \times \Omega$. Which means that the square (L) commutes.

Now, this isn't yet quite what we want: we need (L) to be a pullback. But having done the main work, let's allow ourselves to arm-wave: everything in the diagram is derived from limits – three pullback squares, and the newly added products – so the resulting square (L) will give us a limiting case too. □

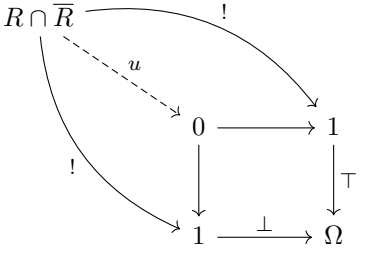
So that's one direction of the biconditional Theorem 112 done: the other direction follows using basically the same ideas.

(c) *Proving that for any of X's subobjects (R, r), $r \cap \overline{r} \equiv 0_X$.* This time, consider the following diagram:



The top left square is the pullback defining the intersection by Defn. 97*. The top right square is the pullback defining the complement by Defn. 96*. The bottom square is the pullback defining χ_r.

Since everything commutes, the two outer circuits from $R \cap \overline{R}$ to Ω must be equal. But then now consider



We've just seen that the outer paths are equal. But the inner square is a pullback, so there is a unique u making everything commute. And since a topos is Cartesian closed, Theorem 72(3) applies, and we know that u, having target 0, is an isomorphism.

Hence $R \cap \overline{R} \cong 0$, and so $R \cap \overline{R}$ is initial, and arrows from it to a given target are unique. So in particular, the arrow $r \cap \overline{r} \colon R \cap \overline{R} \to X$ must equal $0_X \circ u$. But this tells us that $(R \cap \overline{R}, r \cap \overline{r}) \leqslant (0, 0_X)$. However, $(0, 0_X)$ is already a minimum subobject by Theorem 96. Therefore $(R \cap \overline{R}, r \cap \overline{r}) \equiv (0, 0_X)$. □

24 Natural numbers objects

There are infinitely many natural numbers, but each natural number is itself a finite object. It is the same with some toposes. Take FinOrd for example. It has infinitely many objects, but these objects are finite objects, and can be thought of just as natural numbers. So it is interesting to ask: what nice condition can we put on a topos which will require it to contain some intuitively infinitary objects? Indeed, what could serve as a categorial axiom of infinity?

A neat formulation was proposed by F. W. Lawvere.[1] The idea is to require our topos to include a 'natural numbers object', an object which which behaves in a crucial way like the (infinite!) set of natural numbers. This penultimate chapter in Part I explains and explores.

24.1 NNOs defined

(a) Let's start with the familiar notion of a sequence. A sequence has a first member. And then each member is followed by a unique successor.

In the general case, a sequence of successors can circle round and repeat itself. But of course, the natural numbers form a special kind of sequence, an *ω-sequence*. No distinct two objects have the same successor – so the sequence plods on for ever without repetition.

This suggests, in a categorial spirit, we might first define a family of sequences and then locate the natural numbers (up to isomorphism, in the usual way) as forming a limiting case.

So, in arrow-speak, to handle sequences in general, we need an arrow-as-element to pick out an initial object, and then we need an arrow-as-operation which takes an element to its 'successor'. Let's officially say, then:

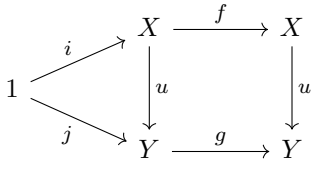
Definition 100. If C is a category with a terminal object, then (X, i, f) forms a *sequence* in C if and only if X is a C-object and i, f are C-arrows $i\colon 1 \to X$ and $f\colon X \to X$. △

If we are working in the category Set, for example, an arrow $i\colon 1 \to X$ is a function picking out the initial element of a sequence, call this element i too; and $f\colon X \to X$ then generates a sequence i, $f(i)$, $f^2(i)$, $f^3(i)$, ...

[1] In his classic paper 'An elementary theory of the category of sets' (Lawvere 1964).

As. noted, such a sequence could eventually start repeating; our task therefore is to categorially characterize the limiting case of sequence objects which generate non-repeating sequences $f^n(i)$. i.e. sequences which look like the natural numbers, i.e. which are ω-sequences. To do this, we start with another definition:

Definition 101. If C is a category with a terminal object, then the derived category C_{Seq} has the following data: an object is any of C's sequence objects (X, i, f), and an arrow $u\colon (X, i, f) \to (Y, j, g)$ is any C-arrow $u\colon X \to Y$ which makes this diagram commute in C:



C_{Seq}'s identity arrow on (X, i, f) is C's identity arrow on X, and arrows compose in C_{Seq} in the same way that the compose in C. △

Why is the definition of a C_{Seq}-arrow here a sensible one? Think what happens with Set's sequences. The idea is that an arrow u between the sequence i, $f(i)$, $f^2(i)$, $f^3(i)$, ... and the sequence j, $g(j)$, $g^2(j)$, $g^3(j)$ ... should match up the n-th member of the f-sequence to the n-th member of the g-sequence. The commuting triangle on the left ensures that u matches up the first members. And then the commuting square ensures that $f^n(i)$ gets sent by u to $g^n(j)$.[2]

It is routine, then, to check that our definition does characterize a category. Three observations about this:

(1) Think again of starting from sequences in a category like Set, with an arrow $u\colon (X, i, f) \to (Y, j, g)$ making the diagram commute. We said, u sends $f^n(i)$ to $g^n(j)$. Therefore, since u is functional, if $g^m(j) \neq g^n(j)$ then $f^m(i) \neq f^n(i)$. In other words, the sequence (X, i, f) can't be *more* constrained by equations of the form $f^m(i) = f^n(i)$ governing the sequence than (Y, j, g) is constrained by similar equations between *its* elements.

(2) So suppose C is a category rich in sequences like Set and the corresponding C_{Seq} has an initial object (N, z, s) – think 'zero' and 'successor'. Then because there is an arrow u from it to any other sequence), this initial object have to be as unconstrained a sequence as possible, governed by no additional equations of the form $s^m(z) = s^n(z)$ (where $m \neq n$), and so never repeating. So (N, z, s) will have to behave like an ω-sequence.

(3) Conversely, consider the standard implementation of the natural numbers $\mathbb{N} = \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}, \ldots\}$ in Set, together with the arrow $z\colon 1 \to \mathbb{N}$ which sends the object in the singleton to $\emptyset$, and the arrow $s\colon \mathbb{N} \to \mathbb{N}$ which

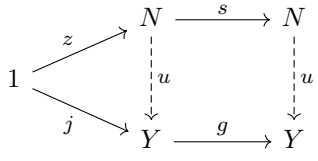
[2]Look at it this way. The commuting square tells us that it doesn't matter, from a given point in the f-sequence, whether we first update by f and then use u to jump to the other sequence, or first jump to the other sequence and then update by g.

sends a set $n \in \mathbb{N}$ to the set $n \cup \{n\}$. Then $(\mathbb{N}, z, s)$ form an initial object in C_{Seq}. Why? Given any other sequence (Y, j, g) in Set, setting u to be the function $n \mapsto g^n(j)$ will make the diagram commute. And evidently u is the unique function which ensures the diagram commutes.

Which all goes to motivate the following general definition:

Definition 102. If C is a category with a terminal object, then a *natural numbers object* (NNO) in C is an initial object of the derived category C_{Seq}.

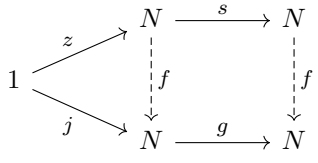
Equivalently, a natural numbers object (N, z, s) in C comprises an object N and two arrows $z\colon 1 \to N$ and $s\colon N \to N$ such that for any object Y and arrows $j\colon 1 \to Y$ and $g\colon Y \to Y$ there is a *unique* arrow u which makes the following diagram commute:



△

Being initial objects of the derived category C_{Seq}, it follows that if (N, z, s) and (N', z', s') are both NNOs in C then $N \cong N'$ (and indeed there is a unique isomorphism commuting with the arrows in the obvious way).

Note a special case, thinking in a category where arrows are functions. Suppose we fix an initial number j by $j\colon 1 \to N$, and are given a function $g\colon N \to N$. Then the claim that there is a unique function f making this commute



is just the claim that the pair of equations $f(0) = j$ and $f(sn) = gf(n)$ together define a unique function f by recursion.

24.2 Proving a NNO is Dedekind infinite

(a) A natural numbers object intuitively looks as if it should be infinite in a good sense – but perhaps we should prove that. OK: let's at least prove

Theorem 116. *If (N, z, s) is a natural numbers object in a non-degenerate topos, then (i) s is monic, and (ii) for every arrow $n\colon 1 \to N$, $s \circ n \neq z$.*

In appropriate categories, this is a version of the claim that N is Dedekind infinite: there is a function $s\colon N \to N$ which is an injection (s is monic) but not surjective (s never maps an element of N to the element picked out as the zero).

A neat route to establishing this claim starts by proving the following lemma:

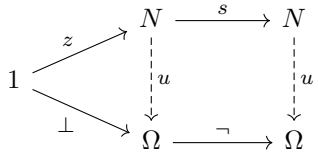
Theorem 117. *In a category with products and a NNO (N, z, s) we can define an arrow $p\colon N \to N$ such that (i) $p \circ z = z$ and (ii) $p \circ s = 1_N$.*

Intuitively, if s is thought of as a successor function, p is the corresponding predecessor function. So suppose for a moment that we are entitled to this lemma. Then we have:

A proof that the predecessor lemma implies Theorem 116. By the second part of the lemma, s is a right inverse, and so is monic by Theorem 17.

It remains to show that there can't be an arrow $n\colon 1 \to N$ such that $s \circ n = z$. Well, (1) suppose otherwise. Then invoking both parts of the predecessor lemma, $n = p \circ s \circ n = p \circ z = z$. So our supposition (1) is equivalent to (2) $s \circ z = z$. Which is intuitively ruled out.

But let's do better than intuition! Consider the sequence object $(\Omega, \bot, \neg)$ in our category (this must exist since the assumption is that we are working in a topos, a nice enough category to have a subobject classifier including Ω and an initial object which enables us to define $\bot$ and $\neg$). By the definition of our NNO, there must be a unique u making the following commute:

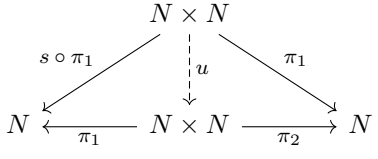


So from the outer paths, $u \circ s \circ z = \neg \circ \bot = \top$. Hence given our supposition (2) that $s \circ z = z$, it follows that $u \circ z = \top$. But the left triangle tells us that $u \circ z = \bot$. Hence, our supposition (2) implies $\top = \bot$ making our topos degenerate.

Contraposing, if our topos is not degenerate, then (2) is false, defeating supposition (1). □

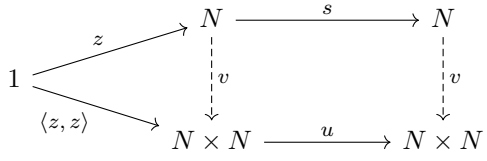
So it remains to establish the predecessor lemma:

Proof of Theorem 117. Start by considering the following product diagram, involving the product $N \times N$ and its two projection arrows $\pi_j\colon N \to N$. By definition of the product, there will be a unique mediating arrow u making the diagram commute:



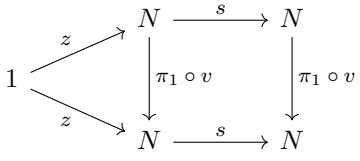
This might look a bit mysterious. But suppose we are in Set for example, and the members of $N \times N$ are good old-fashioned ordered pairs. Then if you trace things round you will see that u sends an initial pair $\langle 0, 0\rangle$ successively to $\langle 1, 0\rangle$, $\langle 2, 1\rangle$, $\langle 3, 2\rangle$, ... In other words, u will have exactly the extension of the predecessor function. So we should be able to work with this idea.

OK, take the sequence object formed from $N \times N$, the arrow $\langle z, z \rangle \colon 1 \to N \times N$ (which intuitively sends the initial object to a pair of zeros), and the arrow u. Then by the definition of our NNO there is a unique v such that the following commutes:



By our previous remark, in Set for example, v will send a number n to the pair $\langle n, \mathit{predecessor}(n) \rangle$. Hence, back to arrow talk, the predecessor arrow $p \colon N \to N$ we want should be given by $p = \pi_2 \circ v$.

We just need to check! So first, note that $p \circ z = \pi_2 \circ v \circ z = \pi_2 \circ \langle z, z \rangle = z$. Second, we need to show $p \circ s = 1_N$. Well, note that from the original product diagram $s \circ \pi_1 = \pi_1 \circ u$; hence $s \circ \pi_1 \circ v = \pi_1 \circ u \circ v = \pi \circ v \circ s$. But that means that the following commutes:



But if we replace the down arrow $\pi_1 \circ v$ by 1_N the diagram also trivially commutes: and by the definition of the NNO, the down arrow is unique. Therefore $\pi_1 \circ v = 1_N$.

We now have (by definition, then by appeal to the last-but-one diagram, then by the product diagram) $p \circ s = \pi_2 \circ v \circ s = \pi_2 \circ u \circ v = \pi_1 \circ v = 1_N$. So we are done! □

24.3 Induction

(a) Let's recall the informal Dedekind-Peano axioms 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 $A = N$.

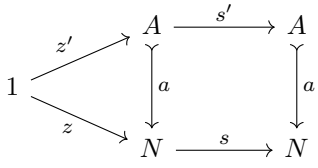
Now, thinking in categorial terms, the last section tells us that an NNO in a topos will give us categorial version of axioms (1) to (4). So it remains to show that we also get a categorial version of the induction axiom.

(b) First, what is the categorial version of (5′)?

(i) Taking some set A of natural numbers becomes starting with a subobject of N, $(A, a\colon A \rightarrowtail N)$.
(ii) The assumption that $0 \in A$ becomes the idea that there is some element of A which gets mapped by a to zero: i.e. there is some $z'\colon 1 \to A$ such that $a \circ z' = z$.
(iii) The assumption that $n \in A \Rightarrow sn \in A$ becomes the idea that there is some map on A which marches in step with the operation of successor applied to N: i.e. there is some $s'\colon S \to S$ such that $s \circ a = a \circ s'$.
(iv) The conclusion that $A = N$ becomes: $A \cong N$.

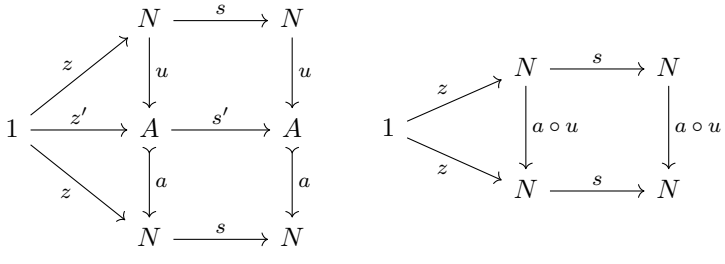
With that 'categorification', the claim that induction holds for natural numbers becomes this:

Theorem 118. *Let* E *be a topos with a NNO* (N, z, s)*. Then for any subobject* (A, a) *of* N*, and arrows* $z'\colon 1 \to A$*,* $s'\colon S \to S$ *such that this diagram commutes*



it follows that $A \cong N$ *in* E*.*

Proof. Given our assumption, there is a unique u such the left hand diagram commutes:



The bottom part commutes by assumption: that there is some arrow u which makes the top part commute follows from the assumption that (N, z, s) is a NNO. Hence the arrow $a \circ u$ makes the right hand diagram commute. But 1_N would also make that diagram commute. And by the uniqueness of the completing down arrows in NNO diagrams, $a \circ u = 1_N$.

Hence a is a left-inverse and so epic. But it is monic by assumption. And in a nice enough category which has a subobject classifier, epic monics are isomorphisms (by Theorem 104). Hence $A \cong N$. □

24.4 More on recursion

(a) In sum, we have shown that if a nice enough category such as a topos has an NNO, then it will satisfy categorified version of the Dedekind-Peano postulates for natural numbers and have a Dedekind-infinite object. Excellent! But there's still some work 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 $g, h\colon N \to N$:

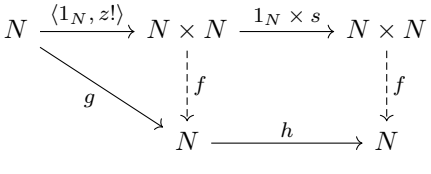
(1) $f(m, 0) = g(m)$

(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.

Call this 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.

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 C which has a natural numbers object (N, z, 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 C which makes this diagram commute

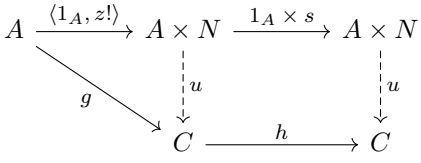


where $z!$ is the composite map $N \xrightarrow{!} 1 \xrightarrow{z} N$. Saying the triangle commutes is the categorial equivalent of saying that (1) holds (since the arrow $\langle 1_N, 0! \rangle$ sends m to the pair $m, 0$. And saying the square commutes is the equivalent of saying that (2) holds. Hence if a category C satisfies condition (P), then

in effect parameterized recursion well-defines functions in C, so we'll be able to define the primitive recursive functions in our category.

And in a topos with a NNO, (P) *does* hold. In fact, we can prove the following more general result:

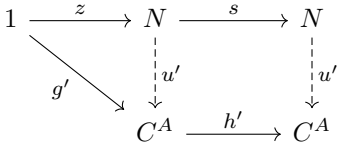
Theorem 119. *If* E *is a topos with a natural number object* $(N, 0, s)$*, then given any objects* A, C*, and arrows* $g\colon A \to C$ *and* $h\colon C \to C$*, then there is a unique* u *which makes the following diagram commute:*



where $z!$ *is now the composite map* $A \xrightarrow{!} 1 \xrightarrow{z} N$.

Our previous diagram of course illustrates the special case where $A = C = N$. So in a topos – or more generally, a Cartesian closed category – with a natural number object, we certainly can warrant parameterized recursive definition.

(b) How can we prove this last theorem? We note that arrows $A \times N \to C$ can easily be replaced by arrows $N \times A \to C$ (via the standard isomorphism between $A \times N$ and $N \times A$). And then, in a category with exponentials, arrows $N \times A \to C$ can be 'curried', i.e. in effect replaced by arrows $N \to C^A$. So the idea is to get ourselves from a diagram of the shape given in the statement of the theorem to a diagram of this shape, where g' and h' depend on g and h:



Then we use the assumption that our category has an NNO to prove that this derived diagram has a unique $u'\colon N \to C^A$ making it commute, and then we can construct an arrow $u\colon A \times N \to C$ in terms of u' which makes our original diagram commute.

And I think I'll rather meanly leave it as a final challenge to you to complete the proof – with the hints that, given the isomorphism i from $1 \times A$ to A, we want

$$g' = \widetilde{g \circ i}\colon 1 \to C^A, \quad h' = \widetilde{h \circ ev}\colon C^A \to C^A,$$

where the wavy overlining notates exponential transposes as usual.

Our theorem can now be extended in the same vein to cover not only definitions by recursion that carry along a single parameter but also the most general kind of definitions by primitive recursion. Therefore in a Cartesian closed category with a natural number object we can start doing some serious arithmetic.

25 A topos of 'abstract' sets?

Standard set theory describes a generous arena in which we can implement the natural numbers and then form products, quotients, exponentials, etc. and thereby construct versions of the reals, spaces of reals, function spaces over the reals, and so on and so forth. The story is so very familiar!

Now, set theory in its usual form is ultimately framed in terms of one sort of object and a single relationship – membership – defined over those objects. But we are getting a glimmering of how there could be another, more categorial way to go, where we deal instead with objects and arrows between them, yet can still reconstruct basic arithmetic and deploy versions of all of the familiar constructions too.

I'll finish Part I by pursuing this thought just a little further.

25.1 Well-pointedness

(a) Much earlier, in Defn. 43, we characterized a category as well-pointed if its arrows are function-like at least in this respect: the identity of arrows is fixed by how they act on elements (i.e. on arrows from an initial object). Carrying over the same idea, and adding a non-degeneracy condition for convenience, we'll say:

Definition 103. A topos is *well-pointed* iff it is non-degenerate and whenever parallel arrows $f, g\colon X \to Y$ agree on all elements – i.e. $f \circ \vec{x} = g \circ \vec{x}$ for all $\vec{x}\colon 1 \to X$ – then $f = g$. △

Now, it is not built in to the definition of a topos that its arrows need be function-like in this respect. But toposes which *are* well-pointed share a number of nice features. For a start, we have:

Theorem 120. *In a well-pointed topos,*

(1) any non-initial object has at least one element;

(2) but a truth-value object Ω has just two elements, $\top$ and $\bot$ (in a word, the topos is bivalent).

Try proving this pair of results before reading on!

Proof that non-initial objects have elements. An arbitrary object X is not guaranteed to be the source or target of many arrows. But at least we know that

for any non-initial object X, there are the arrows $0_X : 0 \to X$ and $1_X : X \to X$, which are distinct, having different sources. And these are both monic arrows (by Theorems 72 and 15).

So we know that in a topos there must be a couple of related parallel arrows, the subobject classifiers $\chi_0 : X \to \Omega$ and $\chi_1 : X \to \Omega$. Theorem 102 then tells us that these arrows must be distinct (otherwise the suobjects $(0, 0_X)$ and $(X, 1_X)$ would be equivalent, and so $0 \cong X$, contrary to hypothesis).

But in that case, by well-pointedness, the parallel arrows $\chi_0, \chi_1 : X \to \Omega$ must act differently on some element of X. Implying that X must indeed have at least one element! □

Proof that any well-pointed topos is bivalent. Take any truth-value-seeking arrow $v : 1 \to \Omega$. Make a corner with $\top : 1 \to \Omega$, and form its pullback:

$$\begin{array}{ccc} X & \overset{f}{\rightarrowtail} & 1 \\ \downarrow & \lrcorner & \downarrow v \\ 1 & \underset{\top}{\rightarrowtail} & \Omega \end{array}$$

If $X \cong 0$, we are back with the pullback defining *false*, and $v = \bot$.

Otherwise $X \ncong 0$, and by the previous result there is an arrow $\vec{x} : 1 \to X$. But then we note that for any parallel g, h such that $g \circ f = h \circ f$ we will have $g \circ f \circ \vec{x} = h \circ f \circ \vec{x}$ and hence $g = h$ (because $f \circ \vec{x} : 1 \to 1$ has to be the identity on the terminal object). Hence f is right-cancellable, so epic.

But f is also monic, being the pullback of a monic up along v. Hence f is an isomorphism by Theorem 104, and so $X \cong 1$, which makes $v = \top$. □

(b) Now, this latest theorem shows that well-pointedness puts puts a tight constraint on the nature of the subobject classifier of a topos. We can also derive what turns out to be a somewhat stronger result, that the subobject classifier will – up to isomorphism – be the two-element co-product $1 \oplus 1$ equipped with one of its injections from 1. And from *that* we can in fact conclude the following key result, recalling Defn. 99:

Theorem 121. *A well-pointed topos is complemented.*

I won't prove this here.[1] But it has a very important corollary, when combined with some other easier results about intersections and unions: the subobjects of a given object form a Boolean algebra.

In other words, complements, intersections and unions for subobjects of an object X in a well-pointed topos behave just like complements, intersections and unions in the familiar world of subsets of given set X.

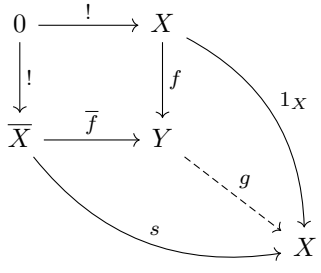
(c) Here's another corollary of our last theorem, which generalizes Theorem 18 which told us that, an exceptional case apart, monics are right inverses in Set:

[1] With some regret! But – a judgement call – pursuing the details would unnecessarily weigh down the discussion at this point, given the limited technical ambitions of this chapter. Compare for example Goldblatt (2006, Ch. 7).

Theorem 122. *In a well-pointed topos, any monic $f\colon X \rightarrowtail Y$ with $X \ncong 0$ is a right inverse, i.e. there is an arrow $g\colon Y \to X$ such that $g \circ f = 1_X$.*

And since the proof is quick and rather neat, this time I will give it!

Proof. Because our topos is complemented, and remembering that $X \cap \overline{X} \cong 0$ and $X \cup \overline{X} \cong Y$, the upper square here is a pushout:



X is non-initial, so Theorem 120 tells us that there is an arrow $\vec{x}\colon 1 \to X$; and trivially, there is an arrow $!\colon \overline{X} \to 1$. So there is an arrow $s = \vec{x} \circ !\colon \overline{X} \to X$.

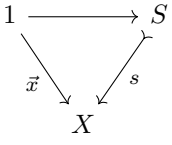
Hence the outer square with opposite vertices 0 and X exists and must commute since arrows from the initial 0 are unique. Hence, since the inner square is a pushout, there must be an arrow g making the whole diagram commute, giving us $g \circ f = 1_X$. □

25.2 Members of subobjects

(a) In a well-pointed topos, to stress the point again, the categorial subobjects of a given object X behave very nicely – just like subsets of some given set, as far as the algebra of their interrelations is concerned. Can we press the subobject/subset parallel further by defining a notion of membership, holding between elements of X and subobjects-of-X-treated-as-arrows?

Yes. For consider the following definition:

Definition 104. If $\vec{x}\colon 1 \to X$ is an element of X, and $(S, s\colon S \to X)$ is a subobject of X, then we'll say that $\vec{x}$ *is a member of* s – in symbols, $\vec{x} \in s$ – iff there is some arrow $1 \to S$ making this diagram commute:



△

Note that the arrow $1 \to S$, if it exists, is unique – for if both $f, g\colon 1 \to S$ make the triangle commute, we'd have $s \circ f = s \circ g$, and therefore $f = g$ since s is monic.

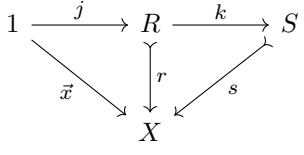
And let's immediately have two quick theorems which show that our definition behaves quite sensibly. The first follows immediately from the definitions:

Theorem 123. *The members of the maximum subobject of X, $(X, 1_X)$, are exactly all the elements $\vec{x}\colon 1 \to X$. The minimum subobject $(0, 0_X)$ has no members.* □

Second, and using the terse idiom for subobject inclusion (just because it makes the result look neat!), we have

Theorem 124. *If $\vec{x}$ is a member of r and r is included in s then $\vec{x}$ is a member of s.*

Proof. Just look at the diagram:



Since $\vec{x} \in r$ there is some j making the left triangle commute; and since $r \leqslant s$ there is some k making the right triangle commute. So $k \circ j$ makes the outer triangle commute, witnessing that $\vec{x} \in s$. □

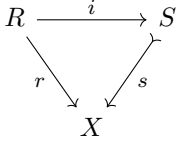
(b) Can we also show that if every member of r is a member of s then r is included in s? Well, that doesn't hold in *every* category. However, we do have:

Theorem 125. *Suppose (R, r) and (S, s) are subobjects of X in a well-pointed topos. Then: if every member of r is a member of s then r is included in s.*

Proof. First take the case where R is initial. Then since it is initial, there is a unique arrow $j\colon R \to S$, giving us an arrow $s \circ j\colon R \to X$. But then $s \circ j = r$, since there is exactly one arrow from $R \to X$. So $r \leqslant s$, and the theorem's conditional conclusion follows.

Now suppose R is not initial, and assume every member of r is a member of s. By Theorem 120 there is an arrow $k\colon 1 \to R$. Which makes $r \circ k$ a member of r. Therefore s has a member, and that requires there to be an arrow $1 \to S$, and hence S isn't initial.

Then since s is monic, and its source isn't initial, Theorem 122 tells us that there is an $u\colon X \to S$ such that $u \circ s = 1_S$. Now put $i = u \circ r$. I claim that, given our assumption,



commutes, showing that $r \leqslant s$.

Since we are dealing with a well-pointed category, to prove that last diagram commutes, i,e. to prove that $s \circ i = r$, it is enough to show that for every $p\colon 1 \to S$, $s \circ i \circ p = r \circ p$.

But note that $\vec{x} = r \circ p \colon 1 \to X$ is, trivially, a member of r, and hence by our assumption it is also a member of s. Hence there is some $j \colon 1 \to S$ such that $\vec{x} = s \circ j$. So putting everything together we have

$$s \circ i \circ p = s \circ u \circ r \circ p = s \circ u \circ \vec{x} = s \circ u \circ s \circ j = s \circ j = \vec{x} = r \circ p.$$

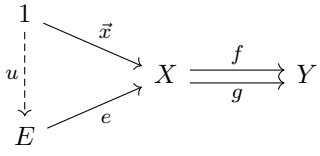
So we are done! □

(c) Let's finish this section with a little result which partially reverses the last theorem:

Theorem 126. *Suppose a category is such that if every member of r is a member of s then r is included in s. Then, assuming it has equalizers, the category is well-pointed.*

Proof. We need to show that parallel arrows $f, g \colon X \to Y$ will be equal if, for all $\vec{x} \colon 1 \to X$, $f \circ \vec{x} = g \circ \vec{x}$.

So suppose $f \circ \vec{x} = g \circ \vec{x}$. Then, in a category with equalizers, there must be an arrow u making the following commute, where (E, e) equalizes f and g:



But, remembering that equalizers are monic, that means $\vec{x}$ is a member of e. Hence, remembering that any $\vec{x}$ is a member of 1_X, it follows that every member of 1_X is a member of e. So, by our initial supposition, $1_X \leqslant e$. But that means there must be an arrow j such that $1_X = e \circ j$.

But since e equalizes f and g, we have $f \circ e = g \circ e$, hence $f \circ e \circ j = g \circ e \circ j$, hence $f \circ 1_X = g \circ 1_X$. Which, of course, gives us our desired result that f and g are equal. □

25.3 Classical arenas

(a) When we look at how set-talk is deployed in ordinary mathematics, we find that we are very typically working in some limited background universe, e.g. in the natural numbers, the reals, some given collection of points, etc. Then the sets that will concern us will be all be subsets of that background universe, and the members of these subsets will all be elements of that same universe. Here's Colin McLarty making the same point:

> Throughout mathematics it is crucial to know which elements $x \in A$ of a set A are members of which subsets $S \subseteq A$. We say this relation is *local* to elements and subsets of the ambient set A. For example, arithmeticians need to know which natural numbers $n \in \mathbb{N}$ are in the subset of primes $Pr \subset \mathbb{N}$. On the other hand, nearly no one ever

> asks whether the imaginary unit $i \in \mathbb{C}$ is also a member of the unit sphere $S^2 \subset \mathbb{R}^3$, because they do not lie inside of any one natural ambient. (McLarty 2017, p. 11)

Now, we haven't given anything akin to a global categorial definition of what it is e.g. for any one object to be a member of another. But we *have* got categorial definitions of what is for one element of X to be a member of a subobject of X, and for one subobject of X to be included in another. And the rather plausible suggestion is that this is in fact good enough for giving a categorial story echoing familiar mathematical constructions with sets, since – to repeat – the sets we are ordinarily working with in a particular context are (almost always?) subsets of some ambient set.

(b) So the situation is this. A topos with a natural numbers object (N, z, s) implements the numbers, and enables us to do arithmetic. And if the topos is also well-pointed, we get something like a set-theory-for-applications, because we can at least talk locally about the subobjects of a given object and about the members of these subobjects.[2] So, we can in particular talk about the subobjects of N – intuitively corresponding to sets of numbers – and about their members. And hence this local sort of set theory, if that's quite the right word for it, arguably suffices to enable all kinds of familiar mathematical constructions starting from naturals and sets of naturals. Let's say then (though this, for once, is not at all standard terminology)

Definition 105. A *classical arena* is a well-pointed topos with a natural numbers object. △

The thought, in short, is that an *arena* in this sense indeed provides a generous enough framework in which we should be able to implement much ordinary mathematics (compare §3.2). And it is *classical* because the collection of subobjects of a given object X forms a Boolean algebra just like the collection of subsets of a given set in standard set theory (and relatedly, in a classical arena, the internal propositional logic in the sense hinted at in §23.5(b) will be classical).

25.4 Choice

(a) We will be able to implement *more* ordinary mathematics in a classical arena if we make another familiar and very basic principle available, namely some version of the Axiom of Choice. "Much ink has been spilled over this axiom", to echo the laconic remark of Lawvere and Rosebrugh in their category-theoretic text *Sets for Mathematics*: and this certainly isn't the place to spill more. So I'll simply assume that you know something about choice and how it can have a role in fairly elementary arguments.

[2] It would be nice to call this simply a 'local set theory' – except that this term has come to denote a particular rich development of the basic idea, with a type-theoretic basis and a non-classical logic: see Bell (1988).

Now, we've already mentioned the topic much earlier: Theorem 18 tells us that the proposition (C1) *every epimorphism has a right inverse* (i.e. is a left inverse, or in other jargon 'splits') is a version of the Axiom of Choice for Set. And now (C1) can serve as our choice principle for classical arenas more generally.

(b) Interestingly, the original proposal for a categorial choice principle due to Lawvere was different: it was in effect the claim that (C2): *for any arrow* $f\colon X \to Y$ *(where* $X \ncong 0$*), there is a* $g\colon Y \to X$ *such that* $f \circ g \circ f = f$.

However, we have the following comforting result:

Theorem 127. *In a classical arena, the choice principles (C1) and (C2) are equivalent.*

Proof: (C1) implies (C2). We need another result, that in a classical arena we can complete the proof started in §19.4(b) and show that every arrow has an epi-mono factorization. Suppose we can help ourselves to that result.

So given an arrow $f\colon X \to Y$ (where $X \ncong 0$), there will be an epic $e\colon X \twoheadrightarrow Z$ and mono $m\colon Z \rightarrowtail Y$ such that $f = m \circ e$.

Given (C1), e has a right inverse, so there is an arrow $j\colon Z \to X$ such that $e \circ j = 1_Z$.

Now also assume $X \ncong 0$. Then we also have $Z \ncong 0$, or else by Theorem 72 the arrow e would be an isomorphism making $X \cong 0$ after all. Since $Z \ncong 0$, we can apply Theorem 122, and the monic m is a right inverse, so there is a $k\colon Y \to Z$ such that $k \circ m = 1_Z$.

Put $g = j \circ k$. And then we have

$$f \circ g \circ f = (m \circ e) \circ (j \circ k) \circ (m \circ e) = m \circ (e \circ j) \circ (k \circ m) \circ e = m \circ e = f$$

So in sum, given (C1), then (C2) follows.[3] □

Proof:(C2) implies (C1). Suppose $X \cong 0$. Then $f\colon X \to Y$ is monic by Theorem 72; so if it is epic too then it is an isomorphism by Theorem 104 and hence has a right inverse.

So suppose $X \ncong 0$, then by (C2) for some g, $f \circ g \circ f = 1_Y \circ f$. So if f is epic, we can right-cancel and derive $f \circ g = 1_Y$, so f has a right inverse, giving us (C1). □

25.5 ETCS

(a) Now, just as different categories can be (say) Cartesian closed, different categories can be classical arenas (with or without choice). We get the obvious example by taking the objects to be sets and the arrows to be set-functions, with the universe of sets understood in a standard way: Set as ordinarily conceived is

[3] I take this proof from Kim (1996). Does the argument have to be as involved? It seems so: see this Mathoverflow answer where Mike Shulman offers basically the same proof: tinyurl.com/ShulmanAC.

a classical arena. But again, just as that category has much more structure than is needed to satisfy the conditions for being Cartesian closed, it also has more structure than is needed to satisfy the conditions for being a classical arena.

For a start, note that there is nothing in the specification of a classical arena that tells us about the identity or non-identity of objects. Categorially, objects are only pinned down up to isomorphism, so there can be enriched arenas with multiple initial objects (as opposed to a single empty set). Or equally there can be sparser arenas of set-like objects but with just one set of any cardinality (take Set but then identify isomorphic sets). Again, the universe of Set on the familiar understanding has a hierarchical structure, with sets of sets of sets ... living at various levels; but it isn't built in to the nature of an arena that its objects have to be so arranged into levels (in a classical arena, elements of objects aren't required to be more objects). And so it goes.

Up to this point, when we have talked about specific categories like Grp or Top for example, we have for definiteness taken them to be living in a fixed universe of sets (so it makes determinate sense to talk about all the groups or all the topological spaces implemented there). And while we have officially remained neutral about the precise character of that background universe, very probably you assumed it to be much as described by ZFC or some close relative, so giving Set a rich structure. But now a really interesting question arises: if a classical arena suffices as a framework in which we can develop a lot of mathematics, but Set has much more structure than is required for being such an arena, what more modest category does the job – ideally not by being a mutilated, cut-down, version of Set-as-usually-conceived but by being an autonomous arena-for-doing-mathematics? In fact, can there be a structure which is (as it were) no more than such an arena?

(b) A positive answer has been defended.

Once upon a time, the story goes, F. W. Lawvere set out to characterize the axiomatic assumptions about sets and functions between sets that we actually require in ordinary mathematics. Put informally, in ordinary maths-speak, his proposed axioms come to these, in Tom Leinster's neat formulation in his very helpful 'Rethinking set theory' (2012):

1. Composition of functions is associative and has identities.
2. There is a set with exactly one element.
3. There is a set with no elements.
4. A function is determined by its effect on elements.
5. Given sets X and Y, one can form their cartesian product $X \times Y$.
6. Given sets X and Y, one can form the set of functions from X to Y.
7. Given $f\colon X \to Y$ and $y \in Y$, one can form the inverse image $f^{-1}(y)$.
8. The subsets of a set X correspond to the functions from X to $\{0, 1\}$.
9. The natural numbers form a set.

10. Every surjection has a right inverse.

Framed in categorial terms, versions of these axioms are set out in Lawvere's classic 1964 paper: they amount to the proposal that a suitable category of sets-for-applications is provided by a topos (cf. Leinster's 1–3, 5–8) which is well-pointed (4), with a natural numbers object (9), where every epimorphism splits (10). In another words, it is a classical arena with choice.

Now, Lawvere's title for his original paper is 'An elementary theory of the category of sets', hence ETCS. Note the 'the'! Left unqualified, that seems quite misplaced. On any sensible view, there are lots of interesting, differently structured, categories of sets – for a start, the topos of sets-living-in-the-usual-iterative-hierarchy-as-described-by-ZFC, the category of NF sets (not a topos because not Cartesian closed), categories of constructively-definable sets, etc. (and of course, lots of subcategories of each of those too). So it is notable that in Lawvere's later book with Rosebrugh, *Sets for Mathematics*, while the authors still talk about 'the category of sets', sometimes the axiomatic theory is more carefully said to characterize '*a* category of abstract sets and arbitrary mappings' (Lawvere and Rosebrugh 2003, p. 113).

And what is 'abstract' doing here? Here is Lawvere in an earlier paper:

> [A]n abstract set may be conceived of as a bag of dots which are devoid of properties apart from mutual distinctness. Further, the bag as a whole [is] assumed to have no properties except cardinality, which amounts to just the assertion that it might or might not be isomorphic to another bag. (Lawvere 1994, p. 5)

Compare, of course, the sets in the standard iterative hierarchy which typically have members which do have intrinsic properties and internal structure.

(c) Now, it is a nice question whether the metaphor of a 'bag of dots which are devoid of properties apart from mutual distinctness' really makes much sense. Certainly, it could hardly be said to encapsulate a notion of set that we deploy in everyday mathematics – when we talk in the ordinary way of a set of natural numbers or a set of reals, for example, we are surely not talking of items which are so devoid of properties! The metaphor overshoots: but then what exactly is its intended non-metaphorical content?

There are further issues too. For a start, does Lawvere's ETCS (or Leinster's 'ordinary maths' gloss) really gives us an arena which is rich enough for the construction of all the 'ordinary' classical, non-set-theoretic, mathematics we want? And most fundamentally, does saying 'take a well-pointed topos with a natural numbers object and choice' really in any sense suffice to pin down an autonomous classical arena, at least up to some sensible notion of equivalence of categories?

We will certainly want to return to those last questions.[4] But at least by now we have some initial sense of how John L. Bell can write

[4]Impatient readers might in the meantime enjoy looking at e.g. Linnebo and Pettigrew (2011) and McLarty (2017).

> The practice of topos theory quickly spawned an associated philosophy ... whose chief tenet is the idea that, like a model of set theory, any topos may be taken as an autonomous universe of discourse or 'world' in which mathematical concepts can be interpreted and constructions performed. (Bell 2001)

We have also glimpsed just the very beginning of a rather involved but fascinating story about how a topos will, in a sense, have its own internal logic. A great deal remains to be said even at our current fairly introductory level.

25.6 So where now?

But before we say much more about toposes, we really do need to backtrack to get our heads around some additional (though still very basic) general category theory.

P. T. Johnstone, in a note introducing his famed Cambridge course, says

> Category theory begins with the observation ... that the collection of all mathematical structures of a given type, together with all the maps between them, is itself an instance of a nontrivial structure which can be studied in its own right. In keeping with this idea, the real objects of study are not so much categories themselves as the maps between them – functors, natural transformations and (perhaps most important of all) adjunctions.

Taking this view, it could very well be complained that some of the most significant objects of study for category theory haven't yet really come into view in Part I of these Notes. An entirely fair point. As I said in §1.3, I do optimistically hope that something has been gained by taking our slow approach before getting round to functors, natural transformations, and the rest. However it is most certainly time to explore them. Onwards then to Part II, with the promissory note that eventually we will get back to toposes.

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.

Arbib, M. and Manes, E., 1975. *Arrows, Structures, and Functors: The Categorical Imperative*. Academic Press Rapid Manuscript Reproduction. New York: Academic Press.

Awodey, S., 2010. *Category theory*, vol. 49 of *Oxford Logic Guides*. Oxford: Oxford University Press, 2nd edn.

Barr, M. and Wells, C., 1995. *Category Theory for Computing Science*. Prentice-Hall International Series in Computer Science. Prentice Hall, 2nd edn. URL http://www.tac.mta.ca/tac/reprints/articles/22/tr22.pdf.

Beardon, A. F., 2005. *Algebra and Geometry*. Cambridge: Cambridge University Press.

Bell, J. L., 1988. *Toposes and local set theories: an introduction*, vol. 14 of *Oxford Logic Guides*. Oxford: Clarendon Press.

Bell, J. L., 2001. The development of categorical logic. In F. Gabbay, D.M.; Guenthner (ed.), *Handbook of Philosophical Logic*, vol. 12, pp. 279–361. Springer.

Benacerraf, P., 1965. What numbers could not be. *Philosophical Review*, 74: 47–73.

Bollobás, B., 1998. *Modern Graph Theory*. New York: Springer.

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.

Finsler, P., 1926. Über die Grundlagen der Mengenlehre, I. *Mathematische Zeitschrift*, 25: 683–713. Reprinted and translated in Booth and Ziegler 1996: 103–132.

Fong, B. and Spivak, D. I., 2019. *An Invitation to Applied Category Theory: Seven Sketches in Compositionality*. Cambridge: Cambridge University Press. URL http://arxiv.org/abs/1803.05316.

Goldblatt, R., 2006. *Topoi: The Categorial Analysis of Logic*. Mineola, New York: Dover Publications, revised edn. URL https://tinyurl.com/GoldblattTopoi.

Incurvati, L., 2020. *Conceptions of Set and the Foundations of Mathematics*. Cambridge: Cambridge University Press.

Johnstone, P., 1997. *Topos Theory*. New York: Academic Press.

Johnstone, P., 2002. *Sketches of an Elephant: A Topos Theory Compendium, Vol. 1*, vol. 43 of *Oxford Logic Guides*. Clarendon Press.

Kim, I. S., 1996. On the axiom of choice in a well-pointed topos. *J. Korea Soc. of Math. Edu.: Pure and Applied Mathematics*, 3: 131–139.

Kunen, K., 1980. *Set Theory: An Introduction to Independence Proofs*. Amsterdam: Elsevier.

Lawvere, F. W., 1964. An elementary theory of the category of sets. *Proceedings of the national academy of sciences*, 52: 1506–1511.

Lawvere, F. W., 1994. Cohesive toposes and cantor's 'lauter einsen'. *Philosophia Mathematica*, 2: 5–15.

Lawvere, F. W. and Schanuel, S. H., 2009. *Conceptual Mathematics: A first introduction to categories*. Cambridge: Cambridge University Press, 2nd edn.

Lawvere, W. and Rosebrugh, R., 2003. *Sets for Mathematics*. Cambridge: Cambridge University Press.

Leinster, T., 2012. Rethinking set theory. URL https://arxiv.org/abs/1212.6543.

Leinster, T., 2014. *Basic Category Theory*. Cambridge: Cambridge University Press. URL https://arxiv.org/abs/1612.09375.

Linnebo, Ø. and Pettigrew, R., 2011. Category theory as an autonomous foundation. *Philosophia Mathematica*, 19.

Mac Lane, S., 1997. *Categories for the Working Mathematician*. New York: Springer, 2nd edn.

Mac Lane, S. and Moerdijk, I., 1992. *Sheaves in Geometry and Logic: A First Introduction to Topos Theory*. New York, Heidelberg, Berlin: Springer.

Maddy, P., 2017. Set-theoretic foundations. *Contemporary Mathematics*, 690: 289–322.

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.

McLarty, C., 1992. *Elementary Categories, Elementary Toposes*. Oxford: Oxford University Press.

McLarty, C., 2017. The roles of set theories in mathematics. In E. Landry (ed.), *Categories for the Working Mathematician*, pp. 1–17. Oxford: Oxford University Press.

Oliver, A. and Smiley, T., 2006. What are sets and what are they for? *Philosophical Perspectives*, 20: 123–155.

Oliver, A. and Smiley, T., 2016. *Plural Logic*. Oxford: Oxford University Press, 2nd edn.

Pierce, B. C., 1991. *Basic category theory for computer scientists*. Cambridge, Mass.: MIT Press.

Quine, W. V. O., 1963. *Set Theory and Its Logic*. Cambridge, MA: The Belknap Press of Harvard University Press.

Riehl, E., 2017. *Category Theory in Context*. Aurora: Dover Modern Math Originals. Mineola, New York: Dover Publications. URL https://emilyriehl.github.io/files/context.pdf.

Roman, S., 2017. *An Introduction to the Language of Category Theory*. Compact Textbooks in Mathematics. Birkhäuser Verlag.

Russell, B. A. W., 1903. *The Principles of Mathematics*. Cambridge: Cambridge University Press.

Scott, D. S., 1980. Relating theories of the λ-calculus. In J. Hindley and J. Seldin (eds.), *To H. B. Curry, Essays on Combinatory Logic, Lambda Calculus and Formalism*, pp. 403–450. London: Academic Press.

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., 2009. *Subsystems of Second Order Arithmetic*. Cambridge: Cambridge University Press, 2nd edn.

Tao, T., 2016. *Analysis*. Springer, 3rd edn.

Index

Some special notation

Categories

Categorial definitions