Category Theory

A Gentle Introduction

Peter Smith
University of Cambridge

Version of September 14, 2015

This PDF is very much an early incomplete version of work still in progress. For the latest and most complete version of this Gentle Introduction and for related materials (including the earlier Notes on Basic Category Theory, which provide less polished continuation chapters) see the Category Theory page at the Logic Matters website.

Corrections, please, to ps218 at cam dot ac dot uk.

Contents

Preface

This Gentle Introduction is a radical revision, as yet very partial, of my earlier 'Notes on Basic Category Theory' (available here).

The gadgets of basic category theory fit together rather beautifully in multiple ways. Their intricate interconnections mean, however, that there isn't a single best route into the theory. Different lecture courses, different books, can quite reasonably take topics in very different orders, all illuminating in their different ways. In the earlier Notes, I roughly followed the order of somewhat over half of the Cambridge Part III course in category theory, as given in 2014 by Rory Lucyshyn-Wright (see also Julia Goedecke's notes from 2013). We now proceed rather differently. The old ordering has its rationale; but I think that the new ordering has a greater logical appeal. Hence the rewrite.

The topics we will eventually cover, in different arrangements, are also the topics of (for example) all but the last chapter of Awodey's good but uneven *Category Theory* and of the whole of Tom Leinster's terrific – and appropriately titled – *Basic Category Theory*. But then, if there are some rightly admired texts out there, not to mention various sets of notes on category theory available online (see here), why produce another introduction to category theory?

I didn't intend to! My goal all along has been to get to understand what light category theory throws on logic, set theory, and the foundations of mathematics. But I realized that I needed to get a lot more securely on top of basic category theory if I was eventually to pursue these more philosophical issues. So my earlier Notes began life as detailed jottings for myself, to help really fix ideas: and then – as can happen – the writing has taken on its own momentum.

What remains distinctive about this Gentle Introduction, for good or ill, is that it is written by someone who doesn't pretend to be a fully-formed expert who usually operates at the frontiers of research in category theory. I hope, however, that this makes me rather more attuned to the likely needs of (at least some) beginners. That's why I go rather slowly over ideas that once gave me pause, and I have generally tried to be as clear as possible. Despite the length, however, the coverage in this present version still falls somewhat short even of the books by Awodey and Leinster, let alone the full Part III syllabus. However, I hope what *is* here will prove useful to others starting to get to grips with category theory. My own experience certainly suggests that initially taking things at a gentle pace as you work into a familiarity with categorial ways of thinking makes later adventures exploring beyond the basics so very much more manageable.

1 Introduction

This is in lieu of more expansive introductory chapters. I really ought one day to provide a lot more preliminary scene-setting. In particular, I eventually want to write at some length about the concept of a mathematical structure, and also about the non-set-theorist's everyday working conception of sets. But that will have to wait. In putting together the present version of this Gentle Introduction, my first concern is to get the presentation of some basic category theory into better shape. So I'm going to be very brisk in this chapter – but I hope forgivably so for now.

1.1 Why categories?

Here is a fundamental insight: we can think of a family of mathematical structures equipped with structure-preserving maps/morphisms/functions between them as itself forming a further mathematical structure.

For example: a particular group is a structure which comprises some objects equipped with an operation on them obeying certain familiar axioms. But we can also think of a family of such groups, taken together with maps between them (the homomorphisms which preserve group structure) as forming a further structure-of-structures.

We can then investigate such structures-of-structures, and indeed go on to think about structure-preserving maps – or as they say, functors – between *them*. Going up another level, we can talk in turn about maps between such functors.

These layers of increasing abstraction are the topic of category theory. So if modern mathematics already abstracts (moving, to take another kind of example, from concrete geometries to abstract metric and topological spaces), category theory abstracts again, and then again. What, if anything, do we gain by going up another level or two of abstraction? To quote Tom Leinster,

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

Which is already reason enough for mathematicians, or at least those with a certain generalizing cast of mind, to be interested in category theory.

What about others? Well, category theory can be thought of as a kind of generalized theory of maps or functions. It is no surprise then that it should turn out to have close links to the lambda calculus, which is another kind of generalized theory of functions. So that's one reason – among a number – why category theory turns out to be of considerable interest to logicians and theoretic computer scientists (who are already concerned with matters around and about the lambda calculus).

Again, philosophers – or at least those interested in questions about the sense in which modern pure mathematics is a study of abstract structures – will want to know what light category theory sheds on these questions. Rather more generally, the theory should perhaps appeal to a certain philosophical temperament. Many philosophers, pressed for a short characterization of their discipline, like to quote a famous remark by the profound American philosopher 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)

This conception sees philosophy, not as the self-indulgent pursuit of linguistic games or as providing obscurely edifying discourse, but as continuous with other serious enquiries, seeking to develop an overview of how their various subject-matters are interrelated. Category theory has a claim to provide part of that overview, as it explores the logical geography of aspects of abstract mathematics.

There are solid reasons, then, for theorists pursuing various disciplines to take an interest in category theory. However, some advocates have made much grander claims for the novelty and fundamental importance of categorial insights. For example, some proponents have suggested that category theory should replace set theory in the role of, so to speak, the official foundations of mathematics. I certainly make no such claims here at the outset. After all, we can only be in a position to assess the ambitions of such enthusiasts after we have made a decent start on understanding category theory for its own sake.

1.2 What do you need to bring to the party?

Legend has it that over the doorway to Plato's Academy was written 'Let no one ignorant of mathematics enter here'. The doorway to category theory should perhaps be similarly inscribed. After all, you will not be well placed to appreciate how category theory gives us a story about the way different parts of modern abstract mathematics hang together if you don't already know *some* modern abstract mathematics.

But in fact, in this Gentle Introduction, we need not presuppose very much background. We'll simply take it that you have some very fragmentary knowledge e.g. of what a group is, perhaps of what a topological space is, and some similar bits and pieces. And if a few later illustrative examples pass you by, don't panic.

I usually try to give multiple examples of important concepts and constructs; so simply skip those examples that don't work for you.

In due course, I hope to say rather more about the background that here we have to take for granted. For example, it would be illuminating and relevant to explore just how much set theory is or isn't presupposed by mathematicians working on (say) group theory or topology. After all, a group is said to be a *set* equipped with an operation obeying certain axioms (and conventionally, operations or functions are also identified with sets): but just how seriously do we need to take this conventional set talk? However, such questions – as I have already indicated – are for a later version of this Gentle Introduction. For now, we will just have to assume that groups, topological spaces and the rest are abstractions in good standing and see where we get.

1.3 Theorems as exercises

There are currently no exercises in what follows – or at least, there are none explicitly labeled as such. Maybe that's another omission to be rectified later.

However, almost all the proofs of theorems in basic category theory are easy, and often they are very easy indeed. Even when a result like the Yoneda Lemma takes a bit more work, I try to break things down using intermediate lemmas so that each stage is as straightforward to prove as possible.

Few of our theorems, then, require much more than straightforwardly applying category-theoretic definitions and/or earlier results in the book (perhaps together with tiny amounts of background knowledge). There's little ingenious trickery. So as a reader you can think of almost every theorem as in fact presenting you with an exercise which you could, even should, attempt in order to fix ideas. For a few tougher theorems, I sometimes give hints about how the argument goes. The ensuing proof which I spell out is then the answer (or at least, *an* answer) to the exercise. I almost always take these proofs at a gentle pace, as befits a Gentle Introduction.

1.4 Notation and terminology

I try to keep notation and terminology standard, and where there are standard alternatives often mention them.

But I should mention at the outset one minor idiosyncracy. As well as using the familiar $\square$ as an end-of-proof marker, I use $\triangleleft$ as an end-of-definition marker.

2 Categories defined

We are going to be investigating the general idea of a family of structures with structure-preserving maps between them: such a family is the paradigm of a category.

But what can we say about a family-of-structures in the abstract? One general thought is this: if, within a family, we have a structure-preserving map from A to B, and another structure-preserving map from B to C, then we should be able to compose these maps, follow the first by the second, to give a structure-preserving map from A to C. What principles should govern such composition of maps? Associativity, surely: given maps

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

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

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

That would seem not to give us very much to go on: but in fact it is already enough to guide our initial very abstract definition of categories as families-of-structures in this chapter. We then present some initial examples, and introduce categorial diagrams.[1]

2.1 The very idea of a category

A category $\mathscr{C}$ consists in certain data (in the sense of 'givens', rather than of 'information'!), with the data governed by a couple of axioms.

Definition 1. The data for a category $\mathscr{C}$ come in two distinct, non-overlapping, sorts:

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

(1) *Objects* (which we will typically notate by 'A', 'B', 'C', ...).

(2) *Arrows* (which we typically notate by 'f', 'g', 'h', ...).

Further,

(3) For each arrow f, there are unique associated objects $src(f)$ and $tar(f)$, respectively the *source* and *target* of f. We write '$f\colon A \to B$' or '$A \xrightarrow{f} B$' to notate that f is an arrow with $src(f) = A$ and $tar(f) = B$.

(4) 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.

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

The axioms then require

Identity. The identity arrows do behave like identities! So for any $f\colon A \to B$ we have $f \circ 1_A = f = 1_B \circ f$.

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

Five quick remarks on our terminology and notation:

(a) The labels 'objects' and 'arrows' for the two kinds of data are standard. But note that the objects of a category (the first type of data) are very often not 'naked' objects, so to speak, but rather – as promised – are typically entities that come equipped with certain operations and relations etc., i.e. can be thought of as structures.

(b) It should also be noted, though, that the 'objects' in categories needn't always be objects at all in the logician's strict sense, i.e. in the sense which contrasts objects with entities like relations or functions. There are categories whose objects – in the sense of the first type of data – are actually relations, and other categories whose objects are functions.

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

(d) In keeping with the functional paradigm, the source and target of an arrow are very often called, respectively, the 'domain' and 'codomain' of the arrow. But again that usage has the potential to mislead when arrows aren't (the right kind of) functions, which is again why I prefer our terminology.

(e) Finally, note the order in which we write the components of a composite arrow (some from computer science do things the other way about). Our standard notational convention is again suggested by the functional paradigm. In a category where $f\colon A \to B$, $g\colon B \to C$ are both functions, then $g \circ f(x) = g(f(x))$.

Now, our axioms suffice to ensure our first mini-result:

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

Proof. Suppose A has identity arrows 1_A and $1'_A$. Then applying the identity axioms, $1_A = 1_A \circ 1'_A = 1'_A$.

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

(As this illustrates, I will cheerfully call the most trivial of lemmas, the run-of-the-mill propositions, the interesting corollaries, and the weightiest of results all 'theorems' without distinction.)

So every object in a category is associated with one and only one identity arrow. And we can in fact pick out such identity arrows by the special way they interact with all the other arrows. Hence we could in principle give a variant definition of categories which initially deals just in terms of arrows. For an account of how to do this, see Adámek et al. (2009, pp. 41–43). But I find this technical trickery rather unhelpful. (The central idea of category theory is perhaps best understood as the idea that we should probe objects by considering the morphisms between them; but that surely need not mean writing the objects out of the story altogether!)

2.2 Examples

Naively, we might think of a structure as comprising *some objects* (plural) equipped with functions and/or relations on them. But a structure nowadays is much more commonly thought of as being *a set* (a single thing) equipped with functions and/or relations on it. But is there a difference? If so, does it matter?

As we said in the Introduction, we are having to shelve such questions for now. So, going along with the usual modern line on structures, we will expect some paradigm examples of categories to have as objects *sets*-equipped-with-some-functions/relations, and then the arrows between such objects will then be suitable functions between the carrier sets which in a good sense 'preserve structure'. Or perhaps we should say 'respect structure', for 'preservation' sounds like a matter of producing a full copy, which is more than we usually require.

We start with an extremal case, where the sets in fact come with no additional structure:

(1) Set is a category with

> Objects: all the sets (just naked sets, equipped with no extra structure).
>
> Arrows: given sets X, Y, any (total) set-function $f \colon X \to Y$ is an arrow.

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

Four comments on this initial case. (i) Annoyingly, there is already a substantive issue about just what category we have in mind here when talking about Set. What *kind* of sets are its objects? Pure sets, or do we allow sets with urelements (members which aren't themselves sets)? Are these sets as constrained by e.g. the axioms of ZFC or are we thinking of a universe of sets better described by a rival set theory like NFU? How safe is it to talk about *all* the sets?

Again, perhaps we should not pause over such questions now, or we'll never get started! Take your favoured conception of the universe of sets (or it you think that universe is indeterminately open-ended, consider levels of the set universe up to some suitable 'inaccessible' rank): then its objects and functions should constitute a category. So for the moment, you can just interpret talk of sets in your preferred way. For some more introductory remarks, see here. And do eventually take a look at Leinster 2014, Ch. 3, 'Interlude on sets'. But here let's simply press on!

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

Perhaps set theorists themselves ought to identity a set-function $f \colon A \to B$ with a triple $(A, \hat{f}, B)$. But be that as it may, that's how category theorists can officially regard arrows $f \colon A \to B$ in Set. (Still, having made the point, it will be harmless for most purposes to casually talk of set-functions, arrows in Set, as if they are just sets of pairs.)

(iii) We should 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 $1_\emptyset \colon \emptyset \to \emptyset$.

Note that in Set, the empty set is in fact the *only* set such that there is exactly one arrow from it to any other set. This gives us a nice first 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. We will return to this.

(iv) The function $id_A \colon A \to A$ defined by $id_A(x) = x$ will evidently serve in the category Set as the (unique) identity arrow 1_A.

We can't say that, however, in pure category-speak. Still, we can do something that comes to the same. Glancing ahead to ideas which we'll again return to, note first that we can define singletons in Set by relying on the observation that there is exactly one arrow from any object *to* a singleton. So fix on a singleton, call it simply '1'. Then consider the possible arrows (i.e. set-functions) $\vec{x} \colon 1 \to A$. There is a one-one correspondence between these arrows and the elements $x \in A$. So we can think of talk of arrows $\vec{x} \colon 1 \to A$ as our category-speak surrogate for talking about elements x of A. Then instead of saying $id_A(x) = x$ for all members x of A, we can say that for any arrow $\vec{x} \colon 1 \to A$, we have $1_A \circ \vec{x} = \vec{x}$. (More on this sort of thing in due course: but it gives us another glimpse ahead of how we might trade in talk of sets-and-their-elements for categorial talk of sets-and-arrows-between-them.)

(2) To continue with our examples of categories, there is also a category FinSet of hereditarily finite sets (i.e. sets with at most finite numbers of members, these members in turn having at most finite numbers of members, which in turn ...) and the functions between *them*.

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

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

> objects: all the non-empty sets, each set A equipped with a zero-place function that picks out a distinguished object $\star_A$ in the set,
>
> arrows: for A, B among the sets, any (total) function $f \colon A \to B$ which maps the distinguished object $\star_A$ to the distinguished object $\star_B$ is an arrow.

As we'll show later, Pfn and $\mathsf{Set}_\star$ are in a good sense equivalent categories (can you already see why we should expect that)?

Now, picking out a single distinguished object, as in $\mathsf{Set}_\star$, is about the least structure we can impose on a set. We turn next to consider cases of sets with rather more structure:

(5) Recall the usual definition of a monoid $(M, \cdot, 1_M)$. This is a pointed set M with a distinguished element 1_M, equipped with a two-place 'multiplication' function mapping elements to elements. It is required that (i) this

function is associative, i.e. for all elements $a, b, c \in M$, $(a \cdot b) \cdot c = a \cdot (b \cdot c)$, and (ii) the designated element 1_M acts as a unit, i.e. is such that for any $a \in M$, $1_M \cdot a = a = a \cdot 1_M$. Then Mon is a category with

objects: all the monoids,

arrows: for $(M, \cdot, 1_M), (N, \times, 1_N)$ among the monoids, any monoid homomorphism $f\colon M \to N$ is an arrow. Recall, $f\colon M \to N$ is a monoid homomorphism when it preserves monoidal structure, i.e. for any $x, y \in M$, $f(x \cdot y) = f(x) \times f(y)$, and f preserves identity elements, i.e. $f(1_M) = 1_N$.

In this category, the identity arrow on $(M, \cdot, 1_M)$ is the identity function on the carrier set M (which is trivially a homomorphism from the monoid to itself); and composition of arrows is, of course, composition of homomorphisms.

(6) Mon is just the first of a family of similar algebraic cases, where the objects are sets equipped with some functions and the arrows are maps preserving that structure. We also have:

a) Grp, the category of groups (the family of structures we mentioned at the very outset – except we now officially think of a group as comprising a *set* of objects equipped with a suitable operation), with
   objects of Grp: all the groups,
   arrows: group homomorphisms (preserving group structure)

b) Ab, the category with
   objects: all abelian groups,
   arrows: group homomorphisms.

c) Rng, the category with
   objects: all rings,
   arrows: ring homomorphisms.

d) Bool, the category with
   objects: all Boolean algebras,
   arrows: structure preserving maps between these algebras.

And so it goes!

(7) 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 the graph $\hat{R} \subset A \times B$ is the set of pairs (a, b) such that aRb.

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

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

(8) The monoids as objects together with all the monoid homomorphisms as arrows form a (very large!) category Mon. But now observe the important fact that any single monoid taken just by itself can be thought of as corresponding to a (perhaps very small) category.

Thus take the monoid $(M, \cdot, 1_M)$. Define $\mathscr{M}$ to be the corresponding category with data

i. the sole object of $\mathscr{M}$: just some object – choose whatever you like, and dub it '$\star$'.

ii. arrows of $\mathscr{M}$: the elements of the monoid M.

iii. the identity arrow $1_\star$ of $\mathscr{M}$ is the identity element 1_M of the monoid M.

iv. the composition $m \circ n \colon \star \to \star$ of two arrows $m \colon \star \to \star$ and $n \colon \star \to \star$ (those arrows being just the elements $m, n \in M$), is $m \cdot n$.

It is trivial that the axioms for being a category are satisfied. So we can think of a monoid as a one-object category. (Conversely, then, we can think of categories as, in a sense, generalized monoids.)

Note in this case, unless the elements of the original monoid M are themselves functions, the arrows of the associated category $\mathscr{M}$ are again not themselves functions or morphisms in any ordinary sense.

(9) Ord is a category, with

objects: the pre-ordered sets. Recall, a set S is pre-ordered iff equipped with an order $\preccurlyeq$ where for all $x, y \in S$, $x \preccurlyeq x$, and $x \preccurlyeq y \wedge y \preccurlyeq z \to x \preccurlyeq z$. We represent the resulting pre-ordered set $(S, \preccurlyeq)$.

arrows: monotone maps – i.e. maps $f \colon S \to T$, from the carrier set of $(S, \preccurlyeq)$ to the carrier set of $(T, \sqsubseteq)$, such that if $x \preccurlyeq y$ then $f(x) \sqsubseteq f(y)$.

(10) Note too that any single pre-ordered set can also itself be regarded as a category (this time, a category with at most one arrow between any two objects). For corresponding to the pre-ordered set $(S, \preccurlyeq)$, we will have a category $\mathscr{S}$, where

i. The objects of $\mathscr{S}$ are just the members of S.

ii. An arrow from $\mathscr{S}$ from source C to target D is just an ordered pair of objects (C, D) such that $C \preccurlyeq D$.

iii. $1_C = (C, C)$.

iv. Composition is defined by setting $(D, E) \circ (C, D) = (C, E)$.

It is easily checked that this satisfies the identity and associatively axioms. Conversely, any category $\mathscr{S}$ whose objects form a set S and where there is at most one arrow between objects can be regarded as a pre-ordered set $(S, \preccurlyeq)$ where for $C, D \in S$, $C \preccurlyeq D$ just in case there is an arrow from C to D of $\mathscr{S}$.

We can thus call a category with at most one arrow between objects a *pre-order category*.

(11) A closely related case to Ex. (9): Pos is a category with

objects: the posets – S is a poset iff equipped with a partial order $\preccurlyeq$, i.e. with a pre-order which satisfies the additional anti-symmetry constraint, i.e. for $x, y \in S$, $x \preccurlyeq y \wedge y \preccurlyeq x \to x = y$.

arrows: monotone maps.

And exactly as each pre-ordered set can be regarded as category, so each individual poset can be regarded as a category, a *poset category*.

(12) Top is a category with

objects: all the topological spaces,

arrows: the continuous maps.

(13) Met is a category with

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

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

(14) Vect_k is a category with

objects: vector spaces over the field k (each such space is a set of vectors, equipped with vector addition and multiplication by scalars in the field k),

arrows: linear maps between the spaces.

For those who already know about such beasts as e.g. sheaves, schemes, or simplicial sets, there are categories of those too, in each case with the relevant objects equipped with predictable structure-preserving maps as arrows. But we won't pause over such exotica: instead let's finish with a few much simpler cases:

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

(16) The smallest discrete category is $\mathsf{1}$ which has exactly one object and one arrow (the identity arrow on that object). Let's picture it in all its glory!

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

But should we talk about *the* category $\mathsf{1}$? Won't different choices of object make for different one-object categories? Well, yes and no! We can have, in our mathematical universe, different cases of single objects equipped with an identity arrow – *but they will be indiscernible from within* *category theory*. So as far as category theory is concerned, they are all 'the same'.

Compare a familiar sort of case from elsewhere in mathematics. There will be many concrete groups which have the right structure to be e.g. a Klein four-group. But they are group-theoretically indiscernible by virtue of being isomorphic. So we take them, for many purposes, to be 'the same' and talk of *the* Klein four-group. (We'll say more about this shortly.)

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

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

Call this category $\mathsf{2}$. We can think of the von Neumann ordinal 2, i.e. the set $\{\emptyset, \{\emptyset\}\}$, as giving rise to this category when it is considered as a poset with the subset relation as the ordering relation. Other von Neumann ordinals, finite and infinite, similarly give rise to other poset categories.

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

Indeed we might well begin to wonder whether it is just *too* easy to be a category. If such very different sorts of structures as a particular poset on the one hand and the whole universe of topological spaces on the other hand equally count as categories, how much mileage can there be general theorizing about categories and their interrelations? Well, that's exactly what we hope to see over the coming chapters.

2.3 Diagrams

We can graphically represent categories – and in particular, represent facts about the equality of arrows – in a very natural way, using so-called commutative diagrams. We've just seen a couple of trivial mini-examples. We'll be using diagrams a great deal: so we'd better say something about them straight away.

Talk of diagrams is in fact used by category theorists in three different (but very closely related) ways. For the moment, we'll mention two.

Definition 2. A *representational diagram* is a 'graph' with nodes representing objects from a given category $\mathscr{C}$, and drawn arrows between nodes representing arrows of $\mathscr{C}$. Nodes and drawn arrows are usually labelled.

Two nodes in a diagram can be joined by zero, one or more drawn arrows. A drawn arrow labelled 'f' from the node labeled 'A' to the node labeled 'B' of course represents the arrow $f\colon A \to B$ of $\mathscr{C}$. There can also be arrows looping from a node to itself, representing the identity arrow on an object or some other 'endomorphism' (i.e. other arrow whose source and target is the same). $\triangleleft$

Definition 3. A *diagram in a category* $\mathscr{C}$ is what is represented by a representational diagram – i.e. is some $\mathscr{C}$-objects and $\mathscr{C}$-arrows between them. $\triangleleft$

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

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

Now, within a drawn diagram, we may be able to follow a directed path through more than two nodes, walking along the connecting drawn arrows (from source to target, of course). So a path in a category diagram from node A to node E (for example) might look like this

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

And then we'll call the composite arrow $j \circ h \circ g \circ f$ the *composite along the path*. (We know that the composite must exist, and also that because of the associativity of composition we needn't worry about the order of bracketing here. Indeed, henceforth we freely insert or omit brackets, doing whatever promotes local clarity.) Then:

Definition 4. A category diagram *commutes* if (i) for any two directed paths along edges in the diagram from a node X to the distinct node Y, the composite arrow along the first path is equal to the composite arrow along the second path, and (ii) for any closed path with more than one node that loops around from a node X to itself, the composite arrow along that path is equal to the identity arrow on the object at X. $\triangleleft$

The main clause here is (i): it is useful to add (ii) to cover a few additional special cases.

Hence, for example, the associativity law can be represented by saying that the following diagram commutes:

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

Each triangle commutes by definition; and the commutativity axiom amounts then to the claim that we can paste such triangles together to get a larger commutative diagram.

(Note, however, that just drawing a diagram with different routes from e.g. A to D doesn't always mean that we have a commutative diagram – the equality of the composites along the paths in each case has to be argued for!)

3 Categories beget categories

We have already seen that categories are plentiful. This chapter describes a number of methods for constructing new categories from old. (We'll meet further methods later, but these first ones will be enough to be going on with.)

3.1 Subcategories

(a) Let's start with a particularly simple way of getting a new category – just slim down an old one.

Definition 5. Given a category $\mathscr{C}$, if $\mathscr{S}$ consists of the data

(1) Objects: some or all of the $\mathscr{C}$-objects,
(2) Arrows: some or all of the $\mathscr{C}$-arrows,

subject to the conditions

(3) for each $\mathscr{S}$-object C, the $\mathscr{C}$-arrow 1_C is also an $\mathscr{S}$-arrow,
(4) for any $\mathscr{S}$-arrows $f\colon C \to D$, $g\colon D \to E$, the $\mathscr{C}$-arrow $g \circ f\colon C \to E$ is also an $\mathscr{S}$-arrow,

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

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

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

(1) **Set** is a subcategory of **Pfn**,

(2) **FinSet** is a subcategory of **Set**,

(3) **Ab** is a subcategory of **Grp**,

(4) The discrete category on the objects of $\mathscr{C}$ is a subcategory of $\mathscr{C}$ for any category.

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

Definition 6. If $\mathscr{S}$ is a subcategory of $\mathscr{C}$ where, for all $\mathscr{S}$-objects A and B, the $\mathscr{S}$-arrows from A to B are all the $\mathscr{C}$-arrows from A to B, then $\mathscr{S}$ is said to be a *full subcategory* of $\mathscr{C}$. $\triangleleft$

We'll meet more cases of full subcategories later.

3.2 Duality

Another easy, but particularly important, way of getting one category from another is to reverse all the arrows. More carefully:

Definition 7. Given a category $\mathscr{C}$, then its *opposite* or *dual* $\mathscr{C}^{op}$ is the category with the data

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

It is trivial to check that this definition is in good order and that $\mathscr{C}^{op}$ is indeed a category. And it is trivial to check that $(\mathscr{C}^{op})^{op}$ is $\mathscr{C}$. So *every* category is the opposite of some category.

Do be careful here, however. Take for example Set^{op}. 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! (This is one of those cases where talking of 'domains' and 'codomains' instead of 'sources' and 'targets' could initially encourage confusion, since the domain of an arrow in Set^{op} is its codomain as a function.)

Take $\mathscr{L}$ to be the elementary pure language of categories. This will be 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 maps from arrows to objects), a built-in relation '$\ldots$ is the identity arrow for $\ldots$', and a two place function-expression '$\ldots \circ \ldots$' which expresses the function which takes two composable arrows to another arrow.

Definition 8. Suppose φ is a wff of $\mathscr{L}$. Then its *dual* φ^{op} is the wff you get by swapping (i) '*src*' and '*tar*' and swapping (ii) '$f \circ g$' for '$g \circ f$', etc. $\triangleleft$

Now, the claim that $\mathscr{C}^{op}$ is a category just reflects the fact that the duals of the axioms for a category are also axioms. And *that* observation gives us the following *duality principle*:

Theorem 2. *Suppose φ is an $\mathscr{L}$-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 of category theory. But those duals of axioms are themselves axioms, so we have a proof of φ^{op} from the axioms of category theory. $\square$

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

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

3.3 Arrow categories and slice categories

(a) For the moment, we will mention just two more ways of deriving a new category from a given category $\mathscr{C}$. The first again reinforces the point that the objects of category need not be really object-like:

Definition 9. Given a category $\mathscr{C}$, the derived *arrow category* $\mathscr{C}^{\to}$ has the following data:

(1) $\mathscr{C}^{\to}$'s objects, its first sort of data, are simply the *arrows* of $\mathscr{C}$,
(2) Given $\mathscr{C}^{\to}$-objects f_1, f_2 (i.e. $\mathscr{C}$-arrows $f_1 \colon X_1 \to Y_1$, $f_2 \colon X_2 \to Y_2$), a $\mathscr{C}^{\to}$-arrow $a \colon f_1 \to f_2$ is a pair (j, k) of $\mathscr{C}$-arrows such that the following diagram commutes:

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

The identity arrow on $f\colon X \to Y$ is defined to be the pair $(1_X, 1_Y)$. And composition of arrows $(j,k)\colon f_1 \to f_2$ and $(j',k')\colon f_2 \to f_3$ is then defined in the obvious way to be $(j' \circ j, k' \circ k)\colon f_1 \to f_3$ (just think of pasting together two of those commuting squares). $\triangleleft$

It is easily checked that this does indeed define a category. We'll give examples of naturally arising arrow categories later.

(b) Suppose next that $\mathscr{C}$ is a category, and I a particular $\mathscr{C}$-object. We will define a new category from $\mathscr{C}$, this time the so-called 'slice' category $\mathscr{C}/I$ where – as in an arrow category – the new category's *objects* are again (some of) the original category's *arrows*.

Her's the definition (and to keep things clear but brief, let's use 'arrow$_\mathscr{C}$' to refer to the old arrows and reserve plain 'arrow' for the new arrows to be found in $\mathscr{C}/I$):

Definition 10. Let $\mathscr{C}$ be a category, and I be a $\mathscr{C}$-object. Then the new category $\mathscr{C}/I$, the *slice category over* I, has the following data:

(1) The $\mathscr{C}/I$-objects are all the arrows$_\mathscr{C}$ $f\colon A \to I$, for any $\mathscr{C}$-object A.
(2) For $\mathscr{C}/I$-objects $f\colon A \to I$, $g\colon B \to I$, the associated $\mathscr{C}/I$-arrows from f to g are just the arrows$_\mathscr{C}$ $j\colon A \to B$ such that $g \circ j = f$ in $\mathscr{C}$.
(3) For every $\mathscr{C}/I$-object $f\colon A \to I$, there is an identity arrow, namely the arrow$_\mathscr{C}$ $1_A\colon A \to A$.
(4) For any $\mathscr{C}/I$-objects $f\colon A \to I$, $g\colon B \to I$, $h\colon C \to I$, the composition of $j\colon f \to g$ and $k\colon g \to h$ is defined as follows: $k \circ j\colon f \to h$ is the arrow$_\mathscr{C}$ $k \circ j\colon A \to C$. $\triangleleft$

Of course, we need to check that these data do indeed together satisfy the axioms for constituting a category. So let's do that.

$$\begin{tikzcd} A \arrow[d, "j"'] \arrow[dr, "f"] & \\ B \arrow[r, "g"'] & I \end{tikzcd}$$

Diagrams will help. Take the arrows$_\mathscr{C}$ $f\colon A \to I$, $f\colon B \to I$. They will be objects f, g of $\mathscr{C}/I$. And the arrows of $\mathscr{C}/I$ from f to g will be the arrows$_\mathscr{C}$ like $j\colon A \to B$ which make our first diagram commute. (Note: the domain and co-domain of j as an arrow$_\mathscr{C}$ are respectively A, B. But the domain and co-domain of j as an arrow in the slice category $\mathscr{C}/I$ are respectively f and g.)

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

We now need to confirm that our definition of $k \circ j$ for composing $\mathscr{C}/I$-arrows works. We are given in $\mathscr{C}$ that $j\colon A \to B$ is such that $g \circ j = f$, and also that $h \circ k = g$. Putting things together we get the second commutative diagram.

Or in equations, we have $(h \circ k) \circ j = f$ in $\mathscr{C}$, and therefore $h \circ (k \circ j) = f$. So $(k \circ j)$ does indeed count as an arrow in $\mathscr{C}/I$ from f to h, as we require.

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

(c) There's a dual notion we can define here, namely the idea of a *co-slice category*. This category $I/\mathscr{C}$ has as objects the arrows$_{\mathscr{C}}$ $f\colon I \to A$, for any $\mathscr{C}$-object A. And the rest of the definition is as you would predict given our explanation of duality: just go through the definition a slice category reversing arrows and the order of composition. (Check that this works!)

(d) Here are a couple of quick examples of slice and co-slice categories, one of each kind:

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

So now think about the co-slice category $1/\mathsf{Set}$. Its objects are all the morphisms $\vec{x}\colon 1 \to X$ (for any set X, one morphism for every point in $x \in X$).

We can think of each such morphism as in effect assigning a set X a special 'basepoint' x. Thought of like that, each object in $1/\mathsf{Set}$ encodes a pointed set. And the arrows in $1/\mathsf{Set}$ from some $\vec{x}\colon 1 \to X$ to some $\vec{y}\colon 1 \to Y$ (in effect, from X-with-basepoint-x to Y-with-basepoint-y) are all the maps $f\colon X \to Y$ in Set such that $f \circ \vec{x} = \vec{y}$: so we can think of such maps as the maps which preserve basepoints.

Hence we can intuitively think of $1/\mathsf{Set}$ as being, in some strong sense, 'the same as' the category Set_* of pointed sets (we'll return later to explain what 'being the same as' comes to for categories).

(2) Second, take an n-membered index set $I_n = \{c_1, c_2, c_3, \ldots, c_n\}$. Think of the members of the set as 'colours'. Then a morphism $S \to I_n$ is an n-colouring of the set S. So we can think of $\mathbf{FinSet}/I_n$ as the category of n-coloured finite sets, which is exactly the sort of thing that combinatorialists are interested in.

More generally, we can think of a slice category $\mathbf{Set}/I$ as a category of 'indexed' or 'typed' sets, with I providing the indices/types.

4 Kinds of arrows

This chapter characterizes a number of kinds of arrows in terms of how they interact with other arrows. This will give us some elementary but characteristic examples of categorial, arrow-theoretic, (re)definitions of familiar notions.

4.1 Monomorphisms, epimorphisms

(a) Take a set-function $f\colon A \to B$ living as an arrow in Set: how could we say that it is injective, i.e. one-one, using just category-speak about arrows?

We noted that we can think of elements x of f's domain A as arrows $\vec{x}\colon 1 \to A$ (where 1 is some singleton). Injectiveness then comes to this: $f \circ \vec{x} = f \circ \vec{y}$ implies $\vec{x} = \vec{y}$, for any element-arrows $\vec{x}, \vec{y}$. Hence if a function is more generally 'left-cancellable' in Set – meaning that, for any g, h, $f \circ g = f \circ h$ implies $g = h$ – then it certainly has to be an injection.

Conversely, if f is injective as a set-function, then for all x, $f(g(x)) = f(h(x))$ implies $g(x) = h(x)$ – which is to say that if $f \circ g = f \circ h$ then $g = h$, i.e. f is left-cancellable.

So that motivates introducing a notion with the following definition (the new-fangled terminology comes from abstract algebra):

Definition 11. An arrow $f\colon C \to D$ in the category $\mathscr{C}$ is a *monomorphism* (is *monic*) if and only if it is left-cancellable, i.e. for every pair of maps $g\colon B \to C$ and $h\colon B \to C$, if $f \circ g = f \circ h$ then $g = h$. $\triangleleft$

We have just proved

Theorem 3. *The monomorphisms in* Set *are exactly the injective functions.*

The same applies in many, but not all, other categories where arrows are functions. For example, we have:

Theorem 4. *The monomorphisms in* Grp *are exactly the injective group homomorphisms.*

Proof. We can easily show as before that the injective group homomorphisms are monomorphisms in Grp.

For the other direction, suppose $f\colon C \to D$ is a group homorphism between the groups $(C, \cdot, e_C)$ and $(D, \star, e_D)$ but is *not* an injection.

We must have $f(c) = f(c')$ for some $c, c' \in C$ where $c \neq c'$. Note then that

$$f(c^{-1} \cdot c') = f(c^{-1}) \star f(c') = f(c^{-1}) \star f(c) = f(c^{-1} \cdot c) = f(e_C) = e_D.$$

So $c^{-1} \cdot c'$ is an element in $K \subseteq C$, the kernel of f (the set of elements that f sends to the unit of $(D, \star, e_D)$). Since $c \neq c'$, we have $c^{-1} \cdot c' \neq e_C$, and hence K has more than one element.

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

Hence, contraposing, if f is monic in Grp it is injective. □

(b) Next, here is a companion definition:

Definition 12. An arrow $f\colon C \to D$ in the category $\mathscr{C}$ is an *epimorphism* (is *epic*) if and only if it is right-cancellable, i.e. for every pair of maps $g\colon D \to E$ and $h\colon D \to E$, if $g \circ f = h \circ f$ then $g = h$. ◁

Evidently, the notion of an epimorphism is dual to that of a monomorphism. Hence f is right-cancellable and so epic in $\mathscr{C}$ if and only if it is left-cancellable and hence monic in $\mathscr{C}^{op}$. And, again predictably, just as monomorphisms in the category Set are injective functions, we have:

Theorem 5. *The epimorphisms in* Set *are exactly the surjective functions.*

Proof. Suppose $f\colon C \to D$ is surjective. And consider two functions $g, h\colon D \to E$ where $g \neq h$. Then for some $d \in D$, $g(d) \neq h(d)$. But by surjectivity, $d = f(c)$ for some $c \in C$. So $g(f(c)) \neq h(f(c))$, whence $g \circ f \neq h \circ f$. So contraposing, the surjectivity of f in Set implies that if $g \circ f = h \circ f$, then $g = h$, i.e. f is epic.

Conversely, 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 $f[C] \subset D$ but disagree on the rest of D. Then $g \neq h$, even though by hypothesis $g \circ f$ and $h \circ f$ will agree everywhere on C, so f is not epic. Contraposing, if f is epic in Set, it is surjective. □

A similar result holds in many other categories, but in §4.3, Ex. (2), we'll encounter a case where we have an epic function which is *not* surjective.

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

Theorem 6. *(1) Identity arrows are always monic. Dually, they are always epic too.*

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

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

Proof. (1) is trivial.

Next, we need to show that if $(f \circ g) \circ j = (f \circ g) \circ k$, then $j = k$. So suppose the antecedent. By associativity, $f \circ (g \circ j) = f \circ (g \circ k)$. Whence, assuming f is monic, $g \circ j = g \circ k$. Whence, assuming g is monic, $j = k$. Which establishes that if f and g are monic, so is $(f \circ g)$.

Interchanging f and g, if f and g are monic, so is $(g \circ f)$: applying the duality principle it follows that f and g are epic, so is $(f \circ g)$.

For (3) assume $f \circ g$ is monic. Suppose $g \circ j = g \circ k$. Then $f \circ (g \circ j) = f \circ (g \circ k)$, and hence $(f \circ g) \circ j = (f \circ g) \circ k$, so $j = k$. Therefore if $g \circ j = g \circ k$ then $j = k$; i.e. g is monic. Dually again for epics. □

(c) We should note a common convention of using special arrows in representational diagrams, a convention which we will follow occasionally but not religiously:

$$f \colon C \rightarrowtail D \text{ or } C \overset{f}{\rightarrowtail} D \quad \text{represents a monomorphism } f,$$
$$f \colon C \twoheadrightarrow D \text{ or } C \overset{f}{\twoheadrightarrow} D \quad \text{represents an epimorphism.}$$

As a useful mnemonic (well, it works for me!), just think of the alphabetic ordering (ML) a *m*onomorphism is *l*eft cancellable and its representing arrow has an extra fletch on the *l*eft, while (PR) an e*p*imorphism is *r*ight cancellable and its representing arrow has an extra fletch on the *r*ight.

4.2 Inverses

(a) We next define some more types of arrow:

Definition 13. Given an arrow $f \colon C \to D$ in the category $\mathscr{C}$,

(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'. Note that if we represent the situation in (1) like this

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

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

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

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

It is obvious that an arrow f need not have a left inverse: just consider, for example, those arrows in Set which are many-one functions. An arrow f can also have many left inverses: for a toy example in Set again, consider $f\colon \{0\} \to \{0, 1\}$ where $f(0) = 0$. Then the map $g\colon \{0, 1\} \to \{0\}$ is a left inverse so long as $g(0) = 0$, which leaves us two choices for $g(1)$, and hence we have two left inverses.

By the duality principle, an arrow can also have zero or many right inverses. However,

Theorem 7. *If an arrow has both a right inverse and a left inverse, then these are the same and are the arrow's unique inverse.*

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

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

Or, in a commuting diagram,

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

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

Suppose now that f has inverses r and s. By definition r will be a right inverse and s a left inverse, so as before $r = s$. So in fact inverses are unique. $\Box$

(b) By way of an aside, let's remark that just as we can consider a particular monoid as a category, in the same way we can consider a particular group as a category. Take a group $(G, \cdot, e)$ and define $\mathscr{G}$ to be the corresponding category whose sole object is whatever you like, and whose arrows are the elements g of G, with e the identity arrow. Composition of arrows in $\mathscr{G}$ is defined as group-multiplication of elements in G. And since every element in the group has an inverse, it follows immediately that every arrow in the corresponding category has an inverse. So in sum, a group-as-a-category is a category with one object and whose every arrow has an inverse. (And generalizing, a category with perhaps more than one object but whose arrows all still have inverses is called a *groupoid*.)

(c) Now, how does talk of an arrow as a right inverse/left inverse hook up to talk of an arrow as monic/epic?

Theorem 8. *(1) In general, not every monomorphism is a right inverse; and dually, not every epimorphism is a left inverse.*

(2) But every right inverse is monic, and every left inverse is epic.

Proof. (1) can be shown by a toy example. Take the category 2 which we met back in §2.2, Ex. (17) – i.e. take as a category the two-element pre-ordered set which has just one non-identity arrow. That non-identity arrow is trivially monic and epic, but it lacks both a left and a right inverse.

For (2), suppose f is a right inverse for e, which means that $e \circ f = 1$ (suppressing unnecessary labellings of domains and codomains). 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 indeed f is monic. Similarly for the dual. □

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? Here's the answer:

Theorem 9. *In* Set*, every monomorphism is a right inverse apart from arrows of the form $\emptyset \to D$. Also in* Set*, the proposition that every epimorphism is a left inverse is (a version of) the Axiom of Choice.*

Proof. Suppose $f \colon C \to D$ in Set is monic. It is therefore one-to-one between C and $f[C]$, so consider a function $g \colon D \to C$ that reverses f on $f[C]$ and somehow or other maps $D - f[C]$ into C. Such a g is always possible to find in Set unless C is the empty set. So $g \circ f = 1_C$, and hence f is a right inverse.

Now suppose $f \colon C \to D$ in Set is epic, and hence a surjection. Assuming the Axiom of Choice, there will be a function $g \colon D \to C$ which maps each $d \in D$ to some chosen one of the elements c such that $f(c) = d$. And 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. 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 all epics have a left inverse gives us (one version of) the Axiom of Choice. □

(d) There is an oversupply of other jargon hereabouts, also in pretty common use. We should note the alternatives for the record.

Assume we have a pair of arrows in opposite directions, $f \colon C \to D$, and $g \colon D \to C$.

Definition 14. If $g \circ f = 1_C$, then f is also called a *section* of g, and g is a retraction of f. (In this usage, f *is* a section iff it *has* a retraction, etc.) ◁

Definition 15. If f has a left inverse, then f is a *split monomorphism*; if g has a right 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.) $\lhd$

Note that Theorem 8 tells us that right inverses are monic, so a split monomorphism is indeed properly called a monomorphism. Dually, a split epimorphism is an epimorphism.

4.3 Isomorphisms

(a) We are familiar, before we ever encounter category theory, with the notion of an isomorphism between structured sets (between groups, between topological spaces, whatever): it's a bijection between the sets which preserves all the structure. In the extremal case, in the category Set of sets with no additional structure, the arrows which are both monic and epic provide the bijective functions. 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 (we crucially want being isomorphic to be an equivalence relation, and hence in particular to be symmetric). But being monic and epic doesn't always imply having an inverse. We can use again the toy case of 2, or here's a generalized version of the same idea:

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

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

(2) Consider the category Mon of monoids. Among its objects are $N = (\mathbb{N}, +, 0)$ 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.

Consider the natural 'inclusion' map $i \colon N \to Z$ which sends a natural number to the corresponding positive integer (this is not a true inclusion map if we take it that naturals in $\mathbb{N}$ are a different type of thing to integers in $\mathbb{Z}$). This map obviously does not have an inverse in Mon. We can show, however, that it is both monic and epic.

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

Second, again take a monoid $\overline{M}$ and this time consider any two monoid homomorphisms $g, h \colon Z \to \overline{M}$ such that $g \circ i = h \circ i$. Then g and h must agree on all integers from zero up. But then note

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

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

$$g(-2) = g(-1+-1) = g(-1) \cdot g(-1) = h(-1) \cdot h(-1) = h(-1+-1) = h(-2),$$

and so it goes. So we have $g(j) = h(j)$ for all $j \in \mathbb{Z}$, positive and negative, and hence $g = h$ and i is right-cancellable, i.e. epic.

So in sum: we can't define an isomorphism as an epic monic if isomorphisms are to have the essential feature of invertibility.

(b) What to do? Build in that feature from the start, and say:

Definition 16. An *isomorphism* (in category $\mathscr{C}$) is an arrow which has an inverse. We conventionally represent isomorphisms by decorated arrows, thus: $\xrightarrow{\sim}$. ◁

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

Theorem 10. *(1) Identity arrows are isomorphisms.*

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

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

So let's immediately give some simple examples of isomorphisms in different categories:

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

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

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

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

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

(c) Isomorphisms are monic and epic by Theorem 8: but we now know that arrows which are monic and epic need not be isomorphisms as we have just defined them. However, it is worth remarking that we do have this:

Theorem 11. *If f is both monic and split epic (or both epic and split monic), then f is an isomorphism.*

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

And we'll also mention another easy result:

Theorem 12. *If f and g are both monic, and each factors through the other, i.e. there is an i such that $f = g \circ i$ and there a j such that $g = f \circ j$, then the factors i and j are isomorphisms.*

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

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

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

4.4 Isomorphic objects

Definition 17. If there is an isomorphism $f : C \xrightarrow{\sim} D$ in $\mathscr{C}$ then the objects C, D are said to be *isomorphic* in $\mathscr{C}$, and we write $C \cong D$. ◁

From the ingredients of Theorem 10, we immediately get the desirable result that

Theorem 13. *Isomorphism between objects in a category is an equivalence relation.*

Now, we might wonder how far this notion of isomorphism between objects captures the idea of two objects amounting to the same as far as their ambient category is concerned. We mentioned before the example where we have, living in Grp, lots of instances of a Klein four-group which are group-theoretically indiscernible by virtue of being isomorphic (indeed, between any two instances, there is a unique isomorphism). And yes, we then cheerfully talk about *the* Klein four-group.

There is a real question, however, about just what this way of talking amounts to, when we seemingly identify isomorphic objects. Some claim, indeed, that category theory itself throws a lot of light on this very issue (see e.g. Mazur 2008). Well certainly, category theory typically doesn't care about distinguishing isomorphic objects in a category. But note that it would, for example, initially strike us as odd to say that because all the instances of singleton sets are isomorphic (indeed, between any two instances, there is a unique isomorphism) we can talk about *the* singleton. After all, the pairwise distinctness of all the singletons $\{\emptyset\}, \{\{\emptyset\}\}, \{\{\{\emptyset\}\}\}, \{\{\{\{\emptyset\}\}\}\}, \ldots$ is precisely what allows us to use this sequence of distinct sets as one possible construction (Zermelo's) for the natural numbers.

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

5 Initial and terminal objects

Our definition of an isomorphism characterizes a type of arrow not by, so to speak, its internal workings, 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's Awodey, offering some similarly arm-waving

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

Over the next few chapters, then, we proceed to give further such external category-theoretic definitions of familiar notions. A prime exhibit will be the treatment of products, starting in the next chapter. In this chapter, we warm up by considering a particularly simple pair of cases.

5.1 Initial and terminal objects, definitions and examples

Definition 18. The object I is an *initial* object of the category $\mathscr{C}$ iff, for every $\mathscr{C}$-object X, there is a unique arrow $I \to X$.

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

It is common to use the likes of '$!\colon I \to X$' or '$!\colon X \to T$' for the unique arrow from an initial object or to a terminal object. ◁

Some examples:

[1] Warning: some call terminal objects *final*; and then that frees up 'terminal' to mean *initial or final*.

(1) In the poset $(\mathbb{N}, \leqslant)$ thought of as a category, zero is trivially the unique initial object and there is no terminal object. The poset $(\mathbb{Z}, \leqslant)$ has neither initial nor terminal objects.

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

(3) In Set, the empty set is an initial object (cf. the comment on Ex. 1 in §2.2).
And any singleton set $\{\star\}$ is a terminal object. (For if X has members, there's a unique Set-arrow which sends all the members to $\star$; while if X is empty, then there's a unique Set-arrow to any set, including $\{\star\}$).

(4) In $\mathsf{Set}_\star$ – the category of pointed sets, non-empty sets equipped with a distinguished member – each singleton is both initial *and* terminal. (A singleton's only member has to be the distinguished member in a pointed set. Arrows in $\mathsf{Set}_\star$ are functions which map distinguished elements to distinguished element. Hence there can be one and only one arrow from a singleton to some some other pointed set.)

(5) In Pos, the category of posets, the empty poset is initial, and any singleton equipped with the only possible order relation on it (the identity relation!) is terminal.

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

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

(8) In Grp, the trivial one-element group is an initial object (a group has to have at least one object, the identity; now recall that a group homomorphism sends identity elements to identity elements; so there is one and only one homomorphism from the trivial group to any given group G). The same one-element group is also terminal.

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

(10) Recall: in the slice category $\mathscr{C}/X$ an object is a $\mathscr{C}$-arrow like $f\colon A \to X$, and an arrow from $f\colon A \to X$ to $g\colon B \to X$ is a $\mathscr{C}$-arrow $j\colon A \to B$ such that $g \circ j = f$ in $\mathscr{C}$. Consider the $\mathscr{C}/X$-object $1_X\colon X \to X$. A $\mathscr{C}/X$ arrow from $f\colon A \to X$ to 1_X is a $\mathscr{C}$-arrow $j\colon A \to X$ such that $1_X \circ j = f$, i.e. such that $j = f$ – which exists and is unique! So 1_X is terminal in $\mathscr{C}/X$.

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

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

Definition 19. An object O in the category $\mathscr{C}$ is a *null object* of the category $\mathscr{C}$ iff it is both initial and terminal. $\triangleleft$

5.2 Uniqueness up to unique isomorphism

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

Theorem 14. *Initial objects, when they exist, are 'unique up to unique isomorphism': i.e. if the $\mathscr{C}$-objects I and J are both initial in the category $\mathscr{C}$, then there is a unique isomorphism $f\colon I \xrightarrow{\sim} J$ in $\mathscr{C}$. Dually for terminal objects.*

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

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

Now suppose I is initial, and that there is an isomorphism $i\colon I \to J$. Then for any X, there is a unique arrow $f\colon I \to X$, and hence there is an arrow $f \circ i^{-1}\colon J \to X$. Now suppose we also have $g\colon J \to X$. Then $g \circ i\colon I \to X$, and so $g \circ i = f$, hence $(g \circ i) \circ i^{-1} = f \circ i^{-1}$, hence $g = f \circ i^{-1}$. In sum, for any X there is a unique arrow from J to X, thus J is also initial.

Duals of these two arguments deliver, of course, the dual results. $\square$

It is standard to introduce notation for an arbitrary initial and terminal objects (since categorically, we usually don't care about distinctions among instances):

Definition 20. We use '0' to denote an initial object of $\mathscr{C}$ (assuming one exists), and likewise '1' to denote a terminal object. $\triangleleft$

Note that in Set, 0 is $\emptyset$, the only initial object – and $\emptyset$ is also the von Neumann ordinal 0. While the von Neumann ordinal 1 is $\{\emptyset\}$, i.e. a singleton, i.e. a terminal object 1. Which perhaps excuses the recycling of the notation.

By the way, null objects (objects which are both initial and terminal) are often alternatively called 'zero' objects. But that perhaps doesn't sit happily with 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. Hence our preference for 'null'.

5.3 Elements and generalized elements

(a) Consider the category Set again. And take a terminal object, a singleton 1. Now, an arrow $\vec{x}\colon 1 \to X$ is a set-function sending the member of the singleton 1 to some member $x \in X$. Trivially there is exactly one such arrow for any $x \in X$, with different arrows corresponding to different elements. So, as we've remarked before, in Set we can think of talk of such arrows $\vec{x}\colon 1 \to X$ as the categorial version of talking of elements of X.

This motivates the following more general definition:

Definition 21. In a category $\mathscr{C}$ with a terminal object 1, an *element* or *point* of the $\mathscr{C}$-object X is an arrow $\vec{x}\colon 1 \to X$. ◁

(In fact, the standard terminology for such an element is '*global* element', picking up from a paradigm example in topology – but we won't fuss about that.)

Theorem 15. *Elements $\vec{x}\colon 1 \to X$ in a category are monic.*

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

Definition 22. Suppose the category $\mathscr{C}$ has a terminal object, and for any object X, Y in $\mathscr{C}$, and arrows $f, g\colon X \to Y$, $f = g$ iff for all $\vec{x}\colon 1 \to X$, $f \circ \vec{x} = g \circ \vec{x}$. Then $\mathscr{C}$ is said to be *well-pointed.* ◁

Think of it this way: in a well-pointed category, there are enough elements (points) to ensure that arrows which act identically on all relevant elements are indeed identical.

Theorem 16. Set *is well-pointed.* Grp*, for example, is not.*

Proof. The claim about Set is immediate.

In Grp, for example, a homomorphism from 1 (the one-element group) to a group X has to send the only element of 1 to the identity element e of X: so there is only one possible homomorphism $\vec{e}\colon 1 \to X$ (so the 'elements'-as-arrows don't in general line up one-to-one with elements of the group).

Take any two group homomorphisms $f, g\colon X \to Y$ where $f \neq g$: yet for all possible $\vec{e}\colon 1 \to X$, both $f \circ \vec{e}$ and $g \circ \vec{e}$ send the sole element of 1 to the identity element of Y, so are equal. □

(b) We have just seen that, even when arrows in a category are functions, acting the same way on elements-as-arrows need not imply being the same arrow. But suppose we introduce *this* notion:

Definition 23. A *generalized element* (of shape S) of the object X in $\mathscr{C}$ is an arrow $e\colon S \to X$. ◁

(So a generalized element of X in $\mathscr{C}$ is in fact the same as an object of the slice category $\mathscr{C}/X$.) Generalized elements give us, so to speak, more ways of probing objects. And we will later see that, while in Set an object – i.e. a set – is determined by its point-elements, in a general category $\mathscr{C}$ we only have that an object is determined by its generalized elements.

For the moment, though, we just note that:

Theorem 17. *Parallel arrows in a category $\mathscr{C}$ are identical if and only if they act identically on all generalized elements.*

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

6 Products introduced

Our next main topic will be a categorial treatment of products (as in Cartesian products). The paradigm construction in Set takes sets X and Y and forms the set of ordered pairs of their elements. But what are ordered pairs? What is involved in pairing schemes more generally? We'll start by considering these sweeping questions as our route in to a categorial treatment of products.

6.1 Real pairs, virtual pairs

A word of caution first. We have fallen into the modern practice of using parentheses in multiple ways (and we didn't want to pause distractingly to discuss this before). But now we should draw an important distinction.

We have, as is standard, used parentheses as in '(x, y)' or '(j, k)' to refer to ordered pairs, and as in '$(A, \hat{f}, B)$' to refer to an ordered triple. Here, the parentheses do essential work, expressing constructors sending given objects to real pairs (triples, etc.) which are objects distinct from their components.

But we have also used parentheses in contexts where, at least initially, we can take them as simply helpful punctuation, not as a signifying a constructor for a new object. For example, when talking informally of the pre-ordered set '$(S, \preccurlyeq)$', we are initially just talking about the set S and about the ordering $\preccurlyeq$ defined over S, and we are not – or at least, not straight off – referring to some further pair-object. Here, we might say, the apparent reference to a pair is merely virtual, and can be translated away. Likewise, in talking of the monoid '$(M, \cdot, 1_M)$' we are initially talking about a set, an operation on it and a selected element and not – or at least, not straight off – about some further object. It may be that at some point we do need to regard pre-ordered sets or monoids as single objects, over and above the relevant sets and relations/functions, i.e. regard them as real rather than virtual pairs or triples. But that needn't be understood as built into the notation from the very start. Our principle should always be: read such notation as noncommittally as possible.

In this chapter, though, we are indeed concerned with real pairs. But what are they?

6.2 Pairing schemes

(a) Suppose for a moment that we are working in a theory of arithmetic and want to start talking about ordered pairs of natural numbers – perhaps we want to go on to use such pairs in constructing integers or rationals. Well, we can handle such pairs without taking on any new commitments by 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 $\langle m, n\rangle = \{(m + n)^2 + 3m + n\}/2$. Or, if we don't insist on every number coding a pair, we could adopt the simpler policy of using $\langle m, n\rangle =_{\text{def}} 2^m 3^n$. Relative to a given coding scheme, we can call such code-numbers $\langle m, n\rangle$ *pair-numbers*. Or, by a slight abuse of terminology, we can call them simply *pairs*, and we can refer to m as the first element of the pair, and n as the second element.

Why should this way of handling ordered pairs of natural numbers be regarded as somehow inferior to other, albeit more familiar, coding devices such as explicitly set-theoretic ones?

It might be said that (i) a single pair-number is really neither ordered nor a twosome; (ii) while a number m is a member of (or is one of) the pair of m with n, a number can't be a genuine member of a pair-number $\langle m, n\rangle$; and in any case (iii) such a coding scheme is pretty arbitrary (e.g. we could equally well have used $3^m 5^n$ as a code for the pair m, n).

Which is all true. But we can lay *exactly* analogous complaints against e.g. the familiar Kuratowski definition 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 (i) that set is not intrinsically ordered (after all, it is 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 pretty arbitrary choices: thus $\{\{n\}, \{m, n\}\}$ or $\{\{\{m\}\}, \{\{m, n\}\}\}$ etc., etc., would have done just as well. On these counts, at any rate, coding pairs of numbers by using pair-numbers involves no worse a trick than coding them using Kuratowski's standard gadget.

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

It might be retorted that the Kuratowski trick at least has the virtue of being an all-purpose device, available not just when you want to talk about pairs of *numbers*, while e.g. the powers-of-primes coding is of much more limited use. Again true. Similarly you can use sledgehammers to crack all sorts of things, while you can only use nutcrackers for nuts. But that's not particularly to the point if it happens to be nuts you currently want to crack, efficiently and with minimum resources. If we want to implement 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) – then pair-numbers are just the kind of thing we need.

(b) So what, more generally, does it take to have a way of pairing-up an object $x \in X$ with an object $y \in Y$?

We need some objects O to serve as ordered pairs, a pairing function that sends a given x and y to a pair-object $o \in O$, and (of course!) a couple of functions which allow us to recover x and y from o. And the point we've just been making is that maybe we shouldn't care too much about the 'internal' nature of the objects O, so long as we do have suitable associated pairing and unpairing functions which work in the right way (for example, pairing and then unpairing gets us back to where we started). Which motivates:

Definition 24. Suppose X, Y and O are sets of objects (these can be the same or different). Let $pr\colon X, Y \to O$ be a two-place function, while $\pi_1\colon O \to X$, and $\pi_2\colon O \to Y$, are one-place functions. Then $[O, pr, \pi_1, \pi_2]$ *form a pairing scheme for X with Y* iff

(a) $(\forall x \in X)(\forall y \in Y)(\pi_1(pr(x,y)) = x \wedge \pi_2(pr(x,y)) = y)$,
(b) $(\forall o \in O)\, pr(\pi_1 o, \pi_2 o) = o$.

The members of O will be said to be the *pair-objects* of the pairing scheme, with pr the associated *pairing function*, while π_1 and π_2 are unpairing or *projection* functions. $\triangleleft$

Evidently, if O is the set of naturals of the form $2^m 3^n$ and $pr(m,n) = 2^m 3^n$, with $\pi_1 o$ ($\pi_2 o$) returning the exponent of 2 (3) in the factorization of o, then $[O, pr, \pi_1, \pi_2]$ form a pairing scheme for $\mathbb{N}$ with $\mathbb{N}$. And if O' is the set of Kuratowski pairs $\{\langle m,n\rangle_K \mid m,n \in \mathbb{N}\}$, with $pr'(m,n) = \langle m,n\rangle_K$, and π_1 (π_2) taking a pair and returning its first (second) element, then $[O', pr', \pi'_1, \pi'_2]$ form another pairing scheme for $\mathbb{N}$ with $\mathbb{N}$

By the way, don't over-interpret the square brackets: they do need here to be read as no more than punctuation. After all, we are in the business of characterizing pairs-as-single-objects; so we don't want to presuppose e.g. that we already know about quadruples-as-single-objects!

Two simple facts about pairing schemes:

i. Different pairs of objects are sent by pr to different pair-objects. For suppose $pr(x, y) = pr(x', y')$. Then by (a) $x = \pi_1(pr(x, y)) = \pi_1(pr(x', y')) = x'$, and likewise $y = y'$.

ii. Note that by (b), pr is surjective. The 'unpairing' or 'projection' functions π_1 and π_2 are also surjective. For given $x \in X$, take any $y \in Y$ and put $o = pr(x, y)$. Then by (a), $x = \pi_1 o$. Likewise, given $y \in Y$ there is an $o \in O$ such that $y = \pi_2 o$.

So, in sum, pairing schemes basically work as you would expect.

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

Theorem 18. *(1) If $[O, pr, \pi_1, \pi_2]$ and $[O, pr, \pi_1', \pi_2']$ are both pairing schemes for X with Y, then $\pi_1 = \pi_1'$ and $\pi_2 = \pi_2'$.*

(2) If $[O, pr, \pi_1, \pi_2]$ and $[O, pr', \pi_1, \pi_2]$ are both pairing schemes, then $pr = pr'$.

Proof. For (1), take any $o \in O$. There is some (unique) x, y such that $o = pr(x, y)$. Hence, applying (a) to both schemes, $\pi_1 o = x = \pi_1' o$. Hence $\pi_1 = \pi_1'$, and similarly $\pi_2 = \pi_2'$.

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

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

Theorem 19. *If $[O, pr, \pi_1, \pi_2]$ and $[O', pr', \pi_1', \pi_2']$ are both schemes for pairing X with Y, then there is a unique bijection $f \colon O \to O'$ such that for all $x \in X, y \in Y$, $pr'(x, y) = f(pr(x, y))$.*

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

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

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

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

So f is a bijection with the right properties. And since every $o \in O$ is $pr(x, y)$ for some x, y, the requirement that $f(pr(x, y)) = pr'(x, y)$ fixes f uniquely. $\square$

(c) Here's another simple theorem, to motivate the final definition in this section:

Theorem 20. *Suppose X, Y, O are sets of objects, and the functions $\pi_1 \colon O \to X$, $\pi_2 \colon O \to Y$ are such that there is a* unique *function $pr \colon X, Y \to O$ such that (a) $(\forall x \in X)(\forall y \in Y)(\pi_1(pr(x, y)) = x \wedge \pi_2(pr(x, y)) = y)$. Then $[O, pr, \pi_1, \pi_2]$ form a pairing scheme.*

Proof. We argue that the uniqueness of pr ensures that the pairing function is surjective, and then that its surjectivity implies that condition (b) from Defn. 24 holds as well as the given condition (a).

Suppose pr is not surjective. Then for some $o \in O$, there is no $x \in X, y \in Y$ such that $pr(x, y) = o$. So $pr(\pi_1 o, \pi_2 o) = o' \neq o$. Consider then function pr' which agrees with pr on all inputs except that $pr'(\pi_1 o, \pi_2 o) = o$. Then for all cases other than $x = \pi_1 o, y = \pi_2 o$ we still have $\pi_1(pr'(x, y)) = x \wedge \pi_2(pr'(x, y)) = y$, and by construction for the remaining case $\pi_1(pr'(\pi_1 o, \pi_2 o)) = \pi_1 o \wedge \pi_2(pr'(\pi_1 o, \pi_2 o)) = \pi_2 o$. So condition (a) holds for pr', where $pr' \neq pr$. Contraposing, if pr uniquely satisfies the condition, it is surjective.

Because pr is surjective, every $o \in O$ is $pr(x, y)$ for some x, y. But then by (a) $\pi_1 o = x \wedge \pi_2 o = y$, and hence $pr(\pi_1 o, \pi_2 o) = pr(x, y) = o$. Generalizing gives us (b). $\square$

Pairing up X with Y through a pairing scheme, then, gives us a product-object O. But we don't want to identify the resulting product simply with the set O (for it depends crucially on the rest of the pairing scheme what role O plays). Our last theorem, however, makes the following an appropriate definition:

Definition 25. If X, Y are sets, then $[O, \pi_1, \pi_2]$ form *a product of X with Y*, where O is a set, and $\pi_1 \colon X \to O, \pi_2 \colon Y \to O$ are functions, so long as there is a unique two-place function $pr \colon X, Y \to O$ such that $(\forall x \in X)(\forall y \in Y)(\pi_1(pr(x, y)) = x \wedge \pi_2(pr(x, y)) = y)$. $\triangleleft$

6.3 Binary products, categorially

We have characterized pairing schemes and the resulting products they create in terms of a set of objects O being the source and target of some appropriate morphisms satisfying the principles in Defns. 24 and 25. Which all looks highly categorial in spirit (see the preamble to the previous chapter).

But our very natural story is *not* categorial quite as it stands. For a quite crucial ingredient, namely the pairing function $pr\colon X, Y \to O$, is a binary function (taking as input *two* objects, not a single pair-object). But the arrows in a category are always unary, i.e. have just a single domain. So how can we get a properly categorial version of our story about pairing schemes?

To work up to an answer, suppose for a moment we are working in a well-pointed category like Set, where 'elements' in the sense of Defn. 21 do behave sufficiently like how elements intuitively should behave. In this case, instead of talking informally of elements x of X and y of Y, we can talk of two arrows $\vec{x}\colon 1 \to X$ and $\vec{y}\colon 1 \to Y$. Suppose then that there is an object O and two arrows, $\pi_1\colon O \to X$ and $\pi_2\colon O \to Y$ such that for every $\vec{x}$ and $\vec{y}$ there is a *unique* arrow $\vec{u}\colon 1 \to O$ such the following commutes:

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

Our arrow $\vec{u}$ picks out an element in O to serve as the product-object $pr(x, y)$. And the requirement that, uniquely, $\pi_1 \circ \vec{u} = \vec{x} \wedge \pi_2 \circ \vec{u} = \vec{y}$ is an instance of the general condition in Defn. 25, re-written in terms of elements-as-arrows. So we can naturally say that $[O, \pi_1, \pi_2]$ here form a product of X with Y.

So far so good. But this will only give us what we want in well-pointed categories with 'enough' elements-as-arrows. However, we know how to generalize to other categories: replace talk about elements (points) with talk of generalized elements. Which motivates the following definition:

Definition 26. In any category $\mathscr{C}$, *a (binary) product* $[O, \pi_1, \pi_2]$ *for the objects* X *with* Y is an object O together with 'projection' arrows $\pi_1\colon O \to X, \pi_2\colon O \to Y$, such that for any object S and arrows $f_1\colon S \to X$ and $f_2\colon S \to Y$ there is always a unique 'mediating' arrow $u\colon S \to O$ such that the following diagram commutes:

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

This unique mediating arrow u can more helpfully be labelled '$\langle f_1, f_2 \rangle$'. $\triangleleft$

Note, by the way, that we are falling into the following common convention: in a category diagram, we use a dashed arrow $\dashrightarrow$ to indicate an arrow which is uniquely fixed by the requirement that the diagram commutes.

Let's have some examples.

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

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

Motivated by this paradigm case, we will henceforth often use the notation $X \times Y$ for the object O in a binary product $[O, \pi_1, \pi_2]$ for X with Y.

Continuing our examples:

(2) In Grp, you can construct a product of the groups $(G, \cdot)$ and $(H, \odot)$ as follows. Take the product object to be $(G \times H, \times)$, i.e. the usual Cartesian product of the underlying sets, and let the group operation be defined component-wise, so that $\langle g, h\rangle \times \langle g', h'\rangle = \langle g \cdot g', h \odot h'\rangle$. Now equip this group with the obvious projection functions from $G \times H$ to G (resp. to H) which send $\langle g, h\rangle$ to g (resp. to h).

(3) Take a poset $(P, \preccurlyeq)$ considered as a category (so there is an arrow $p \to q$ iff $p \preccurlyeq q$). Then a product of p and q would be an object c such that $c \preccurlyeq p, c \preccurlyeq q$ and such that for any object d with arrows from it to p and q, i.e. any d such that $d \preccurlyeq p, d \preccurlyeq q$, there is a unique arrow from d to c, i.e. $d \preccurlyeq c$. That means the product of p and q must be their greatest lower bound (equipped with the obvious two arrows).

Since pairs of objects in posets need not in general have greatest lower bounds, that goes to show that a category in general need not have products.

(4) Here's a new example of a category, call it Prop_L – its objects are propositions, wffs of a given first-order language $\mathscr{L}$, and there is a unique arrow from X to Y iff $X \vDash Y$, i.e. iff X semantically entails Y. The reflexivity and transitivity of semantic entailment means we get the identity and composition laws which ensure that this is a category.

In this case, one product of X with Y will be the conjunction $X \wedge Y$ (with the obvious projections $X \wedge Y \to X$, $X \wedge Y \to Y$).

6.4 Products as terminal objects

Here's a slightly different way of putting things. Let's say

Definition 27. A *wedge to X and Y* (in category $\mathscr{C}$) is an object S and a pair of arrows $f_1 : S \to X$, $f_2 : S \to Y$.

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

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

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

The definition of a product, now using the notion of wedges, can be recaptured as follows. First, we say:

Definition 28. Given a category $\mathscr{C}$ and objects $\mathscr{C}$-objects X, Y, then the derived wedge category $\mathscr{C}_{W(XY)}$ has as the following data. Its objects are all the wedges $[O, f_1, f_2]$ to X, Y.[1] And an arrow from $[O, f_1, f_2]$ to $[O', f_1', f_2']$ is a $\mathscr{C}$-arrow $g\colon O \to O'$ such that the two resulting triangles commute: i.e. $f_1 = f_1' \circ g$, $f_2 = f_2' \circ g$. The identity arrow on $[O, f_1, f_2]$ is 1_O, and the composition of arrows in $\mathscr{C}_{W(XY)}$ is the same as their composition as arrows of $\mathscr{C}$. $\triangleleft$

It is easily confirmed that $\mathscr{C}_{W(XY)}$ is indeed a category. And it is trivial to show, for future reference, that

Theorem 21. *If $g\colon [O, f_1, f_2] \to [O', f_1', f_2']$ is an isomorphism in $\mathscr{C}_{W(XY)}$, then $g\colon O \to O'$ is an isomorphism in $\mathscr{C}$.*

With our new definition of the derived category $\mathscr{C}_{W(XY)}$ to hand, our previous definition of a product can be recast like this:

Definition 29. A product of X with Y in $\mathscr{C}$ is a terminal object of the derived category $\mathscr{C}_{W(XY)}$. $\triangleleft$

[1] Does regarding $[O, f_1, f_2]$ as an object-in-a-category automatically mean treating it as single object in the logician's sense, a real triple, meaning something over an above its components? Why so? If the object-data in a category can – from a logical point of view – be as diverse as object, relations, functions, arrows from other categories, etc., it isn't obvious why they shouldn't comprise an object equipped with two related arrows.

6.5 Uniqueness up to unique isomorphism

As noted, products need not exist for arbitrary objects X and Y in a given category $\mathscr{C}$; and when they exist, they need not be strictly unique. However, when they do exist, they *are* 'unique up to unique isomorphism'. That is to say,

Theorem 22. *If both $[O, \pi_1, \pi_2]$ and $[O', \pi'_1, \pi'_2]$ are products for X with Y in the category $\mathscr{C}$, then there is a unique isomorphism $f\colon O \xrightarrow{\sim} O'$ commuting with the projection arrows (i.e. such that $\pi'_1 \circ f = \pi_1$ and $\pi'_2 \circ f = \pi_2$).*

Note the statement of the theorem carefully. It is *not* being baldly claimed that there is a unique isomorphism between any objects O and O' which are parts of different products for some given X, Y. That's false. For a very simple example, in Set, take the standard product object $X \times X$ comprising Kuratowski pairs: there are evidently two isomorphisms between it and itself, given by the maps $\langle x, x' \rangle \mapsto \langle x, x' \rangle$, and $\langle x, x' \rangle \mapsto \langle x', x \rangle$. The claim is, to repeat, that there is a unique isomorphism between any two product objects O and O' which respects their associated projection arrows.

Plodding proof from basic principles. Since $[O, \pi_1, \pi_2]$ is a product, every wedge factors uniquely through it, including itself. In other words, there is a unique u such that this diagram commutes:

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

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

(i) Given an arrow $u\colon O \to O$, if $\pi_1 \circ u = \pi_1$ and $\pi_2 \circ u = \pi_2$, then $u = 1_O$.

Now, since $[O', \pi'_1, \pi'_2]$ is a product, $[O, \pi_1, \pi_2]$ has to uniquely factor through it:

$$\begin{tikzcd}
& O \arrow[dl, "\pi_1"'] \arrow[d, dashed, "f"] \arrow[dr, "\pi_2"] & \\
X & O' \arrow[l, "\pi'_1"] \arrow[r, "\pi'_2"'] & Y
\end{tikzcd}$$

In other words, there is a unique $f\colon O \to O'$ commuting with the projection arrows, i.e. such that

(ii) $\pi'_1 \circ f = \pi_1$ and $\pi'_2 \circ f = \pi_2$.

And since $[O, \pi_1, \pi_2]$ is also a product, the other wedge has to uniquely factor through *it*. That is to say, there is a unique $g\colon O' \to O$ such that

(iii) $\pi_1 \circ g = \pi_1'$ and $\pi_2 \circ g = \pi_2'$.

Whence,

(iv) $\pi_1 \circ g \circ f = \pi_1' \circ f = \pi_1$ and $\pi_2 \circ g \circ f = \pi_2$.

From which it follows – given our initial observation (i) – that

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

The situation with the wedges is symmetric so we also have

(vi) $f \circ g = 1_{O'}$

Hence f is an isomorphism. $\square$

However, you'll recognize the key proof idea here is akin to the one we used in proving that initial/terminal objects are unique up to unique isomorphism. And we indeed can just appeal to that earlier result:

Proof using the alternative definition of products. $[O, \pi_1, \pi_2]$ and $[O', \pi_1', \pi_2']$ are both terminal objects in the wedge category $\mathscr{C}_{W(XY)}$. So by Theorem 14 there is a unique $\mathscr{C}_{W(XY)}$-isomorphism f between them. But, by definition, this has to be a $\mathscr{C}$-arrow $f\colon O \to O'$ commuting with the projection arrows. And by Theorem 21, f is an isomorphism in $\mathscr{C}$. $\square$

6.6 'Universal mapping properties'

Let's pause for a moment. We have defined a binary product for X with Y categorially as a special sort of wedge to X and Y.

Now, that doesn't fix a product absolutely; but we have now seen that products will be 'unique up to unique isomorphism'. And what makes a 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 *map*.

We can say, then, that products are defined by a *universal mapping property*. We've already met other examples of universal mapping properties: terminal and initial objects are defined by how any other object has a unique map to or from them. We will meet lots more examples.

It is perhaps too soon, however, to attempt a formal definition of what it is to be defined by a universal mapping property. So for the moment take the notion as an informal gesture towards a common pattern of definition which we can learn to recognize when we come across it.

7 Products explored

We continue to explore binary products, and then explain how to use them to categorially implement two-place functions. We show this implementation being put to work before going on to discuss products of more than two objects. The chapter finishes by considering the duals of products, namely coproducts.

7.1 More properties of binary products

(a) We start by verifying some more respects in which binary products behave just as you would hope:

Theorem 23. *In a category which has a terminal object* 1 *and has the relevant products,*

(1) $1 \times X \cong X \cong X \times 1$
(2) $X \times Y \cong Y \times X$
(3) $X \times (Y \times Z) \cong (X \times Y) \times Z$.

Proof. (1) Evidently the wedge $X \xleftarrow{1_X} X \xrightarrow{!} 1$ exists for some unique arrow ! since 1 is terminal. Hence if $X \times 1$ is a product, there is a unique u such that this diagram commutes:

$$\begin{tikzcd} & & X \\ X \arrow[rru, bend left, "1_X"] \arrow[r, dashed, "u"] \arrow[rrd, bend right, "!"'] & X \times 1 \arrow[ru, "\pi_1"] \arrow[rd, "\pi_2"'] & \\ & & 1 \end{tikzcd}$$

Hence, in particular, $\pi_1 \circ u = 1_X$.
Now consider the arrow $u \circ \pi_1 \colon X \times 1 \to X \times 1$. We have

(a) $\pi_1 \circ (u \circ \pi_1) = (\pi_1 \circ u) \circ \pi_1 = \pi_1$.
(b) $\pi_2 \circ (u \circ \pi_1) = \pi_2$ (both sides being arrows from $X \times 1$ to the terminal object 1, hence necessarily equal).

We now appeal to the principle (i) at the beginning of the plodding proof of Theorem 22 to conclude $u \circ \pi_1 = 1_{X \times 1}$. So u has a two-sided inverse, i.e. is an isomorphism, whence $X \cong X \times 1$.

Similarly, of course, for the other half of (1).

(2) If $[X \times Y, \pi_1 \colon X \times Y \to X, \pi_2 \colon X \times Y \to Y]$ is a product of X with Y, then $[X \times Y, \pi_2, \pi_1]$ is obviously a product of Y with X. Hence – by Theorem 22 – there is an isomorphism between the object in that product and the object $Y \times X$ of any other product of Y with X.

(3) It is a just-about-useful reality check to prove this by appeal to our initial definition of a product, using brute force. You are invited to try! But we give a slicker proof in §7.5. $\square$

(b) We next gather together, for later use, a natural definition and a handful of further simple results. First, it is worth noting that our product-style notation $\langle f_1, f_2 \rangle$ for mediating arrows to products doesn't mislead, because we have:

Theorem 24. *If $\langle f_1, f_2 \rangle = \langle g_1, g_2 \rangle$, then $f_1 = g_1$ and $f_2 = g_2$.*

Proof. Being equal, $\langle f_1, f_2 \rangle$ and $\langle g_1, g_2 \rangle$ must share as target the object in some product $[X \times Y, \pi_1, \pi_2]$. We therefore have $f_i = \pi_i \circ \langle f_1, f_2 \rangle = \pi_i \circ \langle g_1, g_2 \rangle = g_i$. $\square$

Here's a sort-of-converse:

Theorem 25. *Given a product $[X \times Y, \pi_1, \pi_2]$ and arrows $S \underset{v}{\overset{u}{\rightrightarrows}} X \times Y$, then, if $\pi_1 \circ u = \pi_1 \circ v$ and $\pi_2 \circ u = \pi_2 \circ v$, it follows that $u = v$.*

Proof. We have in fact already seen this result for the special case where v is the identity arrow. Another diagram shows all we need to prove the general case:

$$\begin{tikzcd}
 & S \arrow[dl, "\pi_1 \circ u / \pi_1 \circ v"'] \arrow[d, shift right, "u"'] \arrow[d, shift left, "v"] \arrow[dr, "\pi_2 \circ u / \pi_2 \circ v"] & \\
X & X \times Y \arrow[l, "\pi_1"] \arrow[r, "\pi_2"'] & Y
\end{tikzcd}$$

The same wedge $X \leftarrow S \rightarrow Y$ factors through $X \times Y$ both via u and v hence, by uniqueness of mediating arrows, $u = v$. $\square$

Definition 30. Suppose we are working in a category with the relevant products. Then the wedge $X \xleftarrow{1_X} X \xrightarrow{1_X} X$ must factor uniquely through the product $X \times X$ via an arrow $\delta_X \colon X \to X \times X$. That unique arrow δ_X is *the diagonal morphism* on X. $\lhd$

In Set, thinking of $X \times X$ in the usual way, δ_X sends an element $x \in X$ to $\langle x, x \rangle$ (imagine elements $\langle x, x \rangle$ lying down the diagonal of a two-dimensional array of pairs $\langle x, y \rangle$: hence the label 'diagonal' and the notation).

Theorem 26. *Given an arrow* $q\colon S \to X$, $\delta_X \circ q = \langle q, q\rangle$.

Proof. Consider the following diagram:

$$\begin{tikzcd}
 & S \arrow[d, "q"] \arrow[ddl, "q"'] \arrow[ddr, "q"] \arrow[dd, dashed, bend left, "{\langle q,q\rangle}"] & \\
 & X \arrow[d, "\delta_X"'] \arrow[dl, "1_X"'] \arrow[dr, "1_X"] & \\
X & X \times X \arrow[l, "\pi_1"] \arrow[r, "\pi_2"'] & X
\end{tikzcd}$$

The inner triangles commute, hence $\delta_X \circ q$ is a mediating arrow factoring the wedge $X \xleftarrow{q} S \xrightarrow{q} X$ through the product $X \times X$. But by definition, the unique mediating arrow which does that is $\langle q, q\rangle$. □

Theorem 27. *Assuming the* $\langle f, g\rangle$ *and* e *compose,* $\langle f, g\rangle \circ e = \langle f \circ e, g \circ e\rangle$.

Proof. Another, rather similar, diagram gives the proof:

$$\begin{tikzcd}
 & R \arrow[d, "e"] \arrow[ddl, "f\circ e"'] \arrow[ddr, "g\circ e"] \arrow[dd, dashed, bend left, "{\langle f\circ e,g\circ e\rangle}"] & \\
 & S \arrow[d, "{\langle f,g\rangle}"'] \arrow[dl, "f"'] \arrow[dr, "g"] & \\
X & X \times Y \arrow[l, "\pi_1"] \arrow[r, "\pi_2"'] & Y
\end{tikzcd}$$

Again the inner triangles commute, hence $\langle f, g\rangle \circ e$ is a mediating arrow factoring the wedge with apex R through the product $X \times Y$. But by definition, the unique mediating arrow is $\langle f \circ e, g \circ e\rangle$. □

Theorem 28. *Given parallel arrows* $S \overset{f_1}{\underset{f_2}{\rightrightarrows}} X$, *with* $f_1 \neq f_2$, *there are (at least) four distinct arrows* $S \to X \times X$.

Proof. By definition of the product, for each pair of indices $i, j \in \{1, 2\}$ there is a unique map $\langle f_i, f_j\rangle$ which makes the product diagram commute,

$$\begin{tikzcd}
 & S \arrow[dl, "f_i"'] \arrow[dr, "f_j"] \arrow[d, dashed, "{\langle f_i,f_j\rangle}"] & \\
X & X \times X \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & X
\end{tikzcd}$$

It is immediate from Theorem 24 that if $\langle f_i, f_j\rangle = \langle f_k, f_l\rangle$, then $i = k, j = l$. So each of the four different pairs of indices tally different arrows $\langle f_i, f_j\rangle$. □

(c) Finally in this section, again for future use, we should remark on a non-theorem. Suppose we have a pair of parallel composite arrows built up using the same projection arrow like this: $X \times Y \xrightarrow{\pi_1} X \overset{f}{\underset{g}{\rightrightarrows}} X'$. In Set, the projection arrow here just 'throws away' the second component of pairs living in $X \times Y$, and all the real action happens on X, so if $f \circ \pi_1 = g \circ \pi_1$, we should also have $f = g$. Generalizing, we might then suppose that, in any category, projection arrows in products are always right-cancellable, i.e. are epic.

This is wrong. Here's a brute-force counterexample. Consider the mini category with just four objects together with the following diagrammed arrows (labelled suggestively but noncommittally), plus all identity arrows, and the necessary two composites:

$$X' \overset{f}{\underset{g}{\leftleftarrows}} X \xleftarrow{\pi_1} V \xrightarrow{\pi_2} Y$$

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 $X \longleftarrow ? \longrightarrow Z$, so trivially all wedges of that shape uniquely factor through it. In other words, the wedge $X \xleftarrow{\pi_1} V \xrightarrow{\pi_2} Y$ is trivially a product and π_1 is indeed a projection arrow. But by construction it isn't epic.

7.2 Maps between two products

(a) Suppose we have two arrows $f\colon X \to X'$, $g\colon Y \to Y'$. Then we might want to characterize an arrow $f \times g\colon X \times Y \to X' \times Y'$ which works component-wise – i.e., putting it informally, the idea is that $f \times g$ sends the product of elements x and y to the product of $f(x)$ and $g(y)$.

In more categorial terms, we require $f \times g$ to be such that the following diagram commutes:

$$\begin{tikzcd}
X \arrow[d, "f"'] & X \times Y \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] \arrow[d, "f \times g"] & Y \arrow[d, "g"] \\
X' & X' \times Y' \arrow[l, "\pi_1'"'] \arrow[r, "\pi_2'"] & Y'
\end{tikzcd}$$

Note, however, that the vertical arrow is then a mediating arrow from the wedge $X' \xleftarrow{f \circ \pi_1} X \times Y \xrightarrow{g \circ \pi_2} Y'$ through the product $X' \times Y'$. Therefore $f \times g$ is indeed fixed uniquely by the requirement that that diagram commutes, and equals $\langle f \circ \pi_1, g \circ \pi_2 \rangle$. This warrants the following definition as in good order:

Definition 31. Given the arrows $f\colon X \to X'$, $g\colon Y \to Y'$, and the products $[X \times Y, \pi_1, \pi_2]$ and $[X' \times Y', \pi_1', \pi_2']$, then $f \times g\colon X \times Y \to X' \times Y'$ is the unique arrow such that $\pi_1' \circ f \times g = f \circ \pi_1$ and $\pi_2' \circ f \times g = g \circ \pi_2$. $\triangleleft$

(b) Here's a special case: sometimes we have an arrow $f: X \to X'$ 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.

It is tempting to suppose that if we have parallel maps $f, g: X \to X'$ 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 §7.1 (c), 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 $f \times 1_Y = g \times 1_Y$).

(c) Later, we will also need the following (rather predictable) general result:

Theorem 29. *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 Theorem 25. □

7.3 Two-place functions

Let's now return to the issue of two-place functions which we sidestepped in §6.3, and consider how such functions get implemented in category theory.

It might be helpful to recall first how a couple of other familiar frameworks manage to do without genuine multi-place functions by providing workable substitutes:

(1) Set-theoretic orthodoxy models a two-place total function from numbers to numbers (addition, say) as a function $f: \mathbb{N}^2 \to \mathbb{N}$. Here, $\mathbb{N}^2$ is the cartesian product of $\mathbb{N}$ with itself, i.e. is the set of ordered pairs of numbers. And an ordered pair is *one* thing not two things. So, in set-theory, a function $f: \mathbb{N}^2 \to \mathbb{N}$ is in fact strictly speaking a *unary function*, a function that

maps *one* argument, an ordered pair object, to a value, not a real binary function.

Of course, in set-theory, for any two things there is a pair-object that codes for them – we usually choose a Kuratowski pair – and so we can indeed trade in a function from two objects for a related function from the corresponding pair-object. And standard notational choices can indeed make the trade quite invisible. Suppose we adopt, as we earlier did, the modern convention of using '(m, n)' as our notation for the ordered pair of m with n. Then '$f(m, n)$' invites being parsed either way, as representing a two-place function $f(\cdot\,,\cdot)$ with arguments m and n or as a corresponding one-place function $f\cdot$ with the single argument, the pair (m, n). But note: the fact that the trade between the two-place and the one-place function is notationally glossed over doesn't mean that it isn't being made.

(2) Versions of type theory deal with two-place functions in a different way, by a type-shifting trick. Addition for example – naively a binary function that just deals in numbers – is traded in for a function of the type $N \to (N \to N)$. This is a unary function which takes one number (of type N) to outputs something of a higher type, i.e. a unary *function* (of type $N \to N$). We then get from two numbers as input to a numerical output in two steps, by feeding the first number to a function which delivers another function as output and then feeding the second number to the second function.

This so-called 'currying' trick of course is also perfectly adequate for certain formal purposes. But again a trade is being made. Here's a revealing quote from *A Gentle Introduction to Haskell* on the haskell.org site (Haskell being one those programming languages where what we might think of naturally as binary functions are curried):

> Consider this definition of a function which adds its two arguments:
>
> $\mathsf{add} :: \mathsf{Integer} \to \mathsf{Integer} \to \mathsf{Integer}$
>
> $\mathsf{add\ x\ y} = \mathsf{x} + \mathsf{y}$

So we have the declaration of type – we are told that add sends a number to a function from numbers to numbers. We are then told how this curried function acts ... but how? By appeal, of course, to our prior understanding of the familiar school-room two-place addition function! The binary function remains an essential rung on the ladder by which we climb to an understanding of what's going on in the likes of Haskell (even if we propose to throw away the ladder after we've climbed it).

Now back to categories. We don't have native binary morphisms in category theory. Nor do we get straightforward currying within a category, at least in the sense that we won't have arrows inside a category whose target is another arrow of that category (though we will meet a version of currying when we get round to talking about exponentials in categories). So what about using a version of the set-theoretic trick?

This, as we've seen, is the line to follow. We can in a noncircular way give a categorial treatment of pair-objects as ingredients of products. And with such objects now to hand, an arrow of the kind $f\colon X \times Y \to S$ is indeed available do duty for a two-place function from an object in X and an object in Y to a value in S. So this will be our implementation device. Let's pause to illustrate it being put to work together with our account of maps between two products.

7.4 Example: groups in categories

(a) We have already seen that there is a category Grp comprising all groups, and that any individual group can also be considered as a category in its own right. In this section we return to talk about groups again, and show how to characterize groups categorically.

We informally think of a group as some objects G equipped with a binary operation m – group multiplication – and with a designated element e which is an identity for the group operation. The group operation is associative, and every element has a two-sided inverse.

Now, how can we characterise such a structure living in Set? We need a set G of elements, and three arrows (which are functions in this category):

(i) $m\colon G \times G \to G$ (so here we are, as announced, trading in the informal binary function for an arrow from a binary product),

(ii) $e\colon 1 \to G$ (we've met before the trick of using this sort of arrow as a way of giving a designated element e),

(iii) $i\colon G \to G$ (this is going to be the arrow that sends an element to its inverse).

(We'll let context, informal vs. categorial, disambiguate the notation!) And to get a group, we need to impose constraints on these arrows corresponding to the usual group axioms:

(1) Informally, we require the group operation m to be associative. Categorially, consider the following diagram:

(G1)
$$\begin{tikzcd}
(G \times G) \times G \arrow[rr, "\cong"] \arrow[d, "m \times 1_G"'] & & G \times (G \times G) \arrow[d, "1_G \times m"] \\
G \times G \arrow[r, "m"] & G & G \times G \arrow[l, "m"']
\end{tikzcd}$$

Here the arrow at the top represents the naturally arising isomorphism between the two triple products that is (or rather, will be) established by the proof of Theorem 23 (3).

Remembering that we are working in Set, an element $\langle\langle j,k\rangle,l\rangle \in (G\times G)\times G$. Going round on the left, that gets sent to $\langle m(j,k),l\rangle$ and then to $m(m(j,k),l)$. Going round the other direction we get to $m(j,m(k,l))$. So requiring the diagram to commute captures the associativity of m.

(2) Informally, we next require e to act as a multiplicative identity: i.e., for all $g\in G$, $m(g,e)=g=m(e,g)$.

To get a categorial equivalent to that condition, start by defining the map $e!\colon G\to G$ by composing $G \xrightarrow{!} 1 \xrightarrow{e} G$. In Set we can think of $e!$ as the function which sends anything in G to the identity element of the group. We then have the following product diagram:

$$\begin{tikzcd}
 & G \arrow[dl, "1_G"'] \arrow[d, dashed, "\langle 1_G, e!\rangle"] \arrow[dr, "e!"] & \\
G & G\times G \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & G
\end{tikzcd}$$

and we can think of the mediating arrow $\langle 1_G, e!\rangle$ as sending an element $g\in G$ to the pair $\langle g,e\rangle$. The element e then behaves like a multiplicative identity on the right if m sends this pair $\langle g,e\rangle$ back to g – i.e. if the top triangle in the following diagram commutes:

(G2)
$$\begin{tikzcd}
G \arrow[r, "\langle 1_G, e!\rangle"] \arrow[d, "\langle e!, 1_G\rangle"'] \arrow[dr, "1_G"] & G\times G \arrow[d, "m"] \\
G\times G \arrow[r, "m"'] & G
\end{tikzcd}$$

Similarly the lower triangle commutes just if e behaves as an identity on the left. So, for e to behave as an identity element, it is enough that the whole diagram commutes.

(3) Finally, we informally require that every element $g\in G$ has an inverse g^{-1} or $i(g)$ such that $m(g,i(g))=e=m(i(g),g)$. Categorially, we can express this by requiring that the following commutes:

(G3)
$$\begin{tikzcd}
G\times G \arrow[d, "1_G\times i"] & G \arrow[l, "\delta_G"'] \arrow[r, "\delta_G"] \arrow[d, "e!"] & G\times G \arrow[d, "i\times 1_G"] \\
G\times G \arrow[r, "m"] & G & G\times G \arrow[l, "e"']
\end{tikzcd}$$

For take an element $g\in G$. Going left, the diagonal arrow δ_G (from Defn. 30) maps it to the pair $\langle g,g\rangle$, which is mapped in turn by $1_G\times i$ to $\langle g,i(g)\rangle$ and then by m to $m(g,i(g))$. Therefore, the requirement that the left square commutes tells us, as we want, that $m(g, i(g)) = e$. Similarly the requirement that the right square commutes tells us that $m(i(g), g) = e$

In summary then, the informal group axioms correspond to the commutativity of our last three diagrams.

But note immediately that this categorial treatment of groups only requires that we are working in a category with binary products and a terminal object. So it is natural to generalize, as follows:

Definition 32. Suppose $\mathscr{C}$ is a category which has the relevant binary products and a terminal object. Let G be a $\mathscr{C}$-object, and $m \colon G \times G \to G$, $e \colon 1 \to G$ and $i \colon G \to G$ be $\mathscr{C}$-arrows. Then $[G, m, e, i]$ is a *group-object* in $\mathscr{C}$ iff the three diagrams (G1), (G2), (G3) commute, where $e!$ in the latter two diagrams is the composite map $G \xrightarrow{!} 1 \xrightarrow{e} G$.

Here, 'group object' (rather than plain 'group') is the standard terminology: some alternatively say 'internal group'.

Then, if we don't fuss about the difference between an arrow $e \colon 1 \to G$ (in a group object) and a designated element e (in a group), we have established the summary result

Theorem 30. *In the category* Set*, a group object is a group.*

And conversely, every group – or to be pernickety, every group which hasn't got too many elements to form a set – can be regarded as a group object in Set.

(b) Here are just a few more examples of group objects living in different categories:

Theorem 31. *(1) In the category* Top*, which comprises topological spaces with continuous maps between them, a group object is a topological group in the standard sense.*

(2) In the category Man*, which comprises smooth manifolds with smooth maps between them, a group object is a Lie group.*

(3) In the category Grp*, a group object is an abelian group.*

Proof. The proofs of (1) and (2) are predictably straightforward if you know the usual definitions of topological groups and Lie groups, and we won't pause over them.

Claim (3), however, is more unexpected. But the proof is relatively straightforward, rather cute, and a rather useful reality-check. So let's present it.

Suppose $[G, m, e, i]$ is a group-object in Grp. Then the object G is already a group of objects $\dot{G}$ equipped with a group operation and an identity element. We'll use ordinary multiplication notation for the operation, as in '$x \cdot y$', and we'll dub the identity '$\dot{1}$' (so the group G is notated with dots!). The arrow

$e\colon 1 \to G$ in the group object also picks out a distinguished element of $\dot{G}$, call it '$\underline{1}$', an identity for m.

Now, each arrow in the group-object $[G, m, e, i]$ lives in Grp, so is a group homomorphism. That means in particular m is a homomorphism from $G \times G$ (the product group, with group operation $\times$) to G. So take the elements $x, y, z, w \in \dot{G}$. Then,

$$m\langle x \cdot z, y \cdot w\rangle = m(\langle x, y\rangle \times \langle z, w\rangle) = m\langle x, y\rangle \cdot m\langle z, w\rangle$$

The first equation holds because of how the operation $\times$ is defined for the product group; the second equation holds because m is a homomorphism.

For vividness, let's rewrite $m\langle x, y\rangle$ as $x \star y$ (so $\underline{1}$ is the unit for $\star$). Then we have established the interchange law

$$(x \cdot z) \star (y \cdot w) = (x \star y) \cdot (z \star w).$$

We will now use this law twice over. First, we have

$$\dot{1} = \dot{1} \cdot \dot{1} = (\underline{1} \star \dot{1}) \cdot (\dot{1} \star \underline{1}) = (\underline{1} \cdot \dot{1}) \star (\dot{1} \cdot \underline{1}) = \underline{1} \star \underline{1} = \underline{1}$$

We can therefore just write 1 for the shared unit, and show secondly that

$$\begin{aligned} x \cdot y &= (x \star 1) \cdot (1 \star y) = (x \cdot 1) \star (1 \cdot y) = x \star y \\ &= (1 \cdot x) \star (y \cdot 1) = (1 \star y) \cdot (x \star 1) = y \cdot x. \end{aligned}$$

We have shown, then, that if $[G, m, e, i]$ is a group object in Grp, G's own group operation commutes, and m is the same operation so that must also commute. Therefore the group object is indeed an abelian group. (We can, but won't, also show that every abelian group can be regarded as a group object in Grp.) $\Box$

(c) We could continue the story. We could categorially define a group homomorphism between two group objects in a category $\mathscr{C}$, and that would in turn enable us to define a category of groups living in $\mathscr{C}$. However, we won't pursue this further in this chapter. We've done enough for our present purposes; we have given an initial illustration of how products in categories can be put to work in defining structures within categories. More on this and related themes in due course. But for now, it's back to the more basic discussion of products. Two major themes remain – generalization and dualization.

7.5 Products generalized

(a) So far we have talked of binary products. But we can generalize in obvious ways. For example,

Definition 33. In any category $\mathscr{C}$, *a ternary product* $[O, \pi_1, \pi_2, \pi_3]$ *for the objects* X_1, X_2, X_3 is an object O together with projection arrows $\pi_i\colon O \to X_i$ (for $i = 1, 2, 3$) such that for any object S and arrows $f_i\colon S \to X_i$ there is always a unique arrow arrow $u\colon S \to O$ such that $f_i = \pi_i \circ u$. $\triangleleft$

And then, exactly as we would expect, using just the same proof ideas as in the binary case, we can prove

Theorem 32. *If both the ternary products* $[O, \pi_1, \pi_2, \pi_3]$ *and* $[O', \pi'_1, \pi'_2, \pi'_3]$ *exist for* X_1, X_2, X_3 *in the category* $\mathscr{C}$*, then there is a unique isomorphism* $f: O \xrightarrow{\sim} O'$ *commuting with the projection arrows.*

We now note that if $\mathscr{C}$ has binary products for all pairs of objects, then it has ternary products too, for

Theorem 33. $(X_1 \times X_2) \times X_3$ *together with the obvious projection arrows forms a ternary product of* X_1, X_2, X_3*.*

Proof. Assume $[X_1 \times X_2, \pi_1, \pi_2]$ is a product of X_1 with X_2, and $[(X_1 \times X_2) \times X_3, \rho_1, \rho_2]$ is a product of $X_1 \times X_2$ with X_3.

Take any object S and arrows $f_i: S \to X_i$. By our first assumption, (a) there is a unique $u: S \to X_1 \times X_2$ such that $f_1 = \pi_1 \circ u$, $f_2 = \pi_2 \circ u$. So by our second assumption (b) there is then a unique $v: S \to (X_1 \times X_2) \times X_3$ such that $u = \rho_1 \circ v$, $f_3 = \rho_2 \circ v$.

Therefore $f_1 = \pi_1 \circ \rho_1 \circ v$, $f_2 = \pi_2 \circ \rho_1 \circ v$, $f_3 = \rho_2 \circ v$

So now consider $[(X_1 \times X_2) \times X_3, \pi_1 \circ \rho_1, \pi_2 \circ \rho_1, \rho_2]$. This, we claim, is indeed a ternary product of X_1, X_2, X_3. We've just proved that S and arrows $f_i: S \to X_i$ factor through the product via the arrow v. It remains to confirm v's uniqueness in this new role.

Suppose we have $w: S \to (X_1 \times X_2) \times X_3$ 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 $\rho_1 \circ w: S \to X_1 \times X_2$ is such that $f_1 = \pi_1 \circ (\rho_1 \circ w)$, $f_2 = \pi_2 \circ (\rho_1 \circ w)$. Hence by (a), $u = \rho_1 \circ w$. But now invoking (b), that together with $f_3 = \rho_2 \circ w$ entails $w = v$. □

Evidently, an exactly similar argument will show that $X_1 \times (X_2 \times X_3)$ together with the obvious projection arrows forms a ternary product of X_1, X_2, X_3. Hence we are now in a position to neatly prove

Theorem 23. (3) $X \times (Y \times Z) \cong (X \times Y) \times Z$.

Proof. Both $(X_1 \times X_2) \times X_3$ and $X_1 \times (X_2 \times X_3)$ (with their projection arrows) are ternary products of X_1, X_2, X_3. So Theorem 32 entails that $X_1 \times (X_2 \times X_3) \cong (X_1 \times X_2) \times X_3$. □

(b) What goes for ternary products goes for n-ary products defined in a way exactly analogous to Defn. 33. If $\mathscr{C}$ has binary products for all pairs of objects it will have quaternary products such as $((X_1 \times X_2) \times X_3) \times X_4$, quinary products, and n-ary products more generally, for any finite $n \geq 2$.

To round things out, how do things go for the nullary and unary cases?

Following the same pattern of definition, a *nullary* product in $\mathscr{C}$ would be an object O together with *no* projection arrows, such that for any object S there is a unique arrow $u\colon S \to O$. Which is just to say that a nullary product is a terminal object of the category.

And a unary product of X would be an object O and a single projection arrow $\pi_1\colon O \to X$ such that for any object S and arrow $f\colon S \to X$ there is a unique arrow $u\colon S \to O$ such that $\pi \circ u = f$. Putting $O = X$ and $\pi = 1_X$ evidently fits the bill. So the basic case of a unary product of X is not quite X itself, but rather X equipped with its identity arrow (and like any product, this is unique up to unique isomorphism). Trivially, all unary products exist in all categories.

In sum, suppose we say

Definition 34. A category $\mathscr{C}$ *has all binary products* iff for all $\mathscr{C}$-objects X and Y, there exists a binary product of X with Y in $\mathscr{C}$.

$\mathscr{C}$ *has all finite products* iff $\mathscr{C}$ has n-ary products for any n objects, for all $n \geq 0$. ◁

Then our preceding remarks establish

Theorem 34. *A category $\mathscr{C}$ has all finite products iff $\mathscr{C}$ has a terminal object and has all binary products.*

(c) We can generalize further in the obvious way, beyond finite products to infinite cases. We will however be interested in cases where we are taking a product of objects which are not too many to form a set – i.e. we can take the objects to be indexed by the elements of some index set I.

Definition 35. In any category $\mathscr{C}$, the product of the $\mathscr{C}$-objects X_i (for indices $i \in I$), if it exists, is an object O together with projection arrows $\pi_i\colon O \to X_i$ (for $i \in I$) such that for any object S and arrows $f_i\colon S \to X_i$ (again for $i \in I$), there is always a unique arrow arrow $u\colon S \to O$ such that $f_i = \pi_i \circ u$. (We can use '$\prod\limits_{i \in I} X_i$' to notate such an O.) ◁

As before, these generalized products will be unique up to unique isomorphism. And we will say

Definition 36. A category $\mathscr{C}$ *has all (small) products* iff for any set's worth of $\mathscr{C}$-objects – i.e. any objects $\mathscr{C}$-objects X_i, for $i \in I$, where I is some set – the objects in the set have a product. ◁

Here, 'small' doesn't mean small by any normal standards! – it just indicates that we are taking products over collections of objects that are not so big that they can't be indexed by a set living somewhere in the usual universe of sets. We'll be returning to such issues of size in due course.

7.6 Coproducts

(a) Let's note a common terminological device:

Definition 37. For very many kinds of categorially defined widget, a *co-widget* of the category $\mathcal{C}$ is a widget of $\mathcal{C}^{op}$: co-widgets are dual to widgets. $\triangleleft$

For example, we have met co-slice categories, the duals of slice categories. We could (and a few do) call initial objects 'co-terminal'. Likewise we could (and a few do) call sections 'co-retractions'. 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 very widely. In particular, it is absolutely standard to talk of the duals of products as 'co-products' – though in this case, 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 38. In any category $\mathscr{C}$, *a (binary) coproduct* $[O, \iota_1, \iota_2]$ *for the objects* X *with* Y is an object O together with 'injection' arrows $\iota_1 \colon X \to O, \iota_2 \colon Y \to O$, such that for any object S and arrows $f_1 \colon X \to S$ and $f_2 \colon Y \to S$ there is always a unique mediating arrow $v \colon O \to S$ such that the following diagram commutes:

$$\begin{tikzcd}
 & S & \\
 & & \\
X \arrow[uur, "f_1"] \arrow[r, "\iota_1"'] & O \arrow[uu, dashed, "v"'] & Y \arrow[uul, "f_2"'] \arrow[l, "\iota_2"]
\end{tikzcd}$$

The object O in a coproduct for X with Y is often notated '$X \oplus Y$' or '$X \amalg Y$'; and we can notate the mediating arrow v by '$[f_1, f_2]$. $\triangleleft$

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

Let's say that objects and arrows arranged as $X \xrightarrow[\iota_1]{} O \xleftarrow[\iota_2]{} Y$ form a *corner* (or we could say 'co-wedge') from X and Y with vertex O. Then a coproduct of X with Y can be thought of as a corner from X and Y which factors through any other corner from X and Y via a unique map between the vertices of the corners.

We could now go on to define a category of corners from X and Y on the model of a category of wedges to X and Y, and then redefine a coproduct of X with Y as an initial object of this category. It is a useful reality check to work through the details.

(c) Let's have some examples of coproducts. Start with easy cases:

(1) For Set, the headline news is that disjoint unions are 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) Take a poset $(P, \preccurlyeq)$ considered as a category (so there is an arrow $p \to q$ iff $p \preccurlyeq q$). 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).

(3) In Prop_L (which we met in §6.3) 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.

However things soon get rather less obvious. We'll mention a couple more cases. But the details here aren't going to matter, so by all means skip:

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

Take the groups $G = (G, \cdot)$, $H = (H, \odot)$ (we abuse notation in a familiar way, recycling the label for a group's carrier set as the label for the group, and letting context disambiguate). If necessary, now doctor the groups to equate their identity elements while ensuring the sets G and H are otherwise disjoint. Form all the finite 'reduced words' $G \star H$ you get by concatenating elements from $G \cup H$, and then multiplying out neighbouring G-elements by $\cdot$ and neighbouring H-elements by $\odot$ as far as you can. Equip $G \star H$ with the operation $\diamond$ of concatenation-of-words-followed-by-reduction. Then $G \star H = (G \star H, \diamond)$ is a group – the free product of the two groups G and H – and there are obvious 'injection' group homomorphisms $\iota_1 \colon G \to G \star H$, $\iota_2 \colon H \to G \star H$.

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

$$\begin{tikzcd} & K & \\ G \arrow[ur, "f_1"] \arrow[r, "\iota_1"'] & G \star H \arrow[u, dashed, "v"'] & H \arrow[ul, "f_2"'] \arrow[l, "\iota_2"] \end{tikzcd}$$

Put $v\colon G\star H \to K$ to be the morphism that sends a word $g_1h_1g_2h_2\cdots g_r$ ($g_i \in G, h_i \in H$) to $j(g_1)*k(h_1)*j(g_2)*k(h_2)*\cdots*j(g_r)$. By construction, $v \circ \iota_1 = j$, $v \circ \iota_2 = k$. So that makes the diagram commute.

Let v' be any other candidate group homomorphism to make the diagram commute. Then, to take a simple example, consider $gh \in G \star H$. Then $v'(gh) = v'(g) * v'(h) = v'(i_1(g)) * v'(i_2(h)) = f_1(g) * f_2(g) = v(i_1(g)) * v(i_1(g)) = v(i_1(g) * i_1(g)) = v(gh)$. And by induction over the length of words we'll get $v' = v$. So, as required, v is unique.

(5) So what about coproducts in Ab, the category of abelian groups? Since the free product of two abelian groups need not be abelian, the same construction won't work again as it stands.

OK: hit the construction with the extra requirement that words in $G\star H$ be treated as the same if one can be shuffled into the other (in effect, further reduce $G \star H$ by quotienting out with the obvious equivalence relation). But that means that we can take a word other than the identity, bring all the G-elements to the front, followed by all the H elements: but now multiply out the G-elements and the H-elements and we are left with two-element word gh. So we can equivalently treat the members of our further reduced $G\star H$ as pairs $\langle g, h\rangle$ belonging to $G\times H$. Equip this with the group operation $\times$ defined component-wise as before (in §6.3): this gives us an abelian group if G and H are. Take the obvious injections, $g \xrightarrow{\iota_1} \langle g, 1\rangle$ and $h \xrightarrow{\iota_2} \langle 1, h\rangle$. Then we claim $[G \times H, \iota_1, \iota_2]$ is a coproduct for the abelian groups G and H.

Take any abelian group $K = (K, *)$ and morphisms $f_1\colon G \to K$, $f_2\colon H \to K$. Put $v\colon G\times H \to K$ to be the morphism that sends $\langle g, h\rangle$ to $f_1(g)*f_2(h)$. This evidently makes the coproduct diagram (with $G \times H$ for $G \star H$) commute. And a similar argument to before shows that it is unique.

So, in the case of abelian groups, the *same* objects can serve as both products and coproducts, when equipped with appropriate projections and injections respectively.

(d) By duality, we immediately know lots of properties of coproducts. In particular, we have the following composite theorem:

Theorem 35. *If both the coproducts $[O, \iota_1, \iota_2]$ and $[O', \iota_1', \iota_2']$ exist for X and Y in the category $\mathscr{C}$, then there is a unique isomorphism $f\colon O \xrightarrow{\sim} O'$ commuting with the injection arrows.*

And in a category with an initial object 0 and where the coproducts exist,

(1) $0 \oplus X \cong X \cong X \oplus 0$
(2) $X \oplus Y \cong Y \oplus X$
(3) $X \oplus (Y \oplus Z) \cong (X \oplus Y) \oplus Z$

Further, the notion of a coproduct generalizes beyond the binary case, just as with products. Thus, exactly as you would expect, we have:

Definition 39. In any category $\mathscr{C}$, the coproduct of the $\mathscr{C}$-objects X_i (for indices $i \in I$) is an object O together with injection arrows $\iota_i \colon X_i \to O$ (for $i \in I$) such that for any object S and suite of arrows $f_i \colon X_i \to S$, there is always a unique arrow $v \colon O \to S$ such that $f_i = v \circ \iota_i$. (We can use '$\coprod_{i \in I}$' to notate such an O.)

By dual arguments to those we've met for products, nullary coproducts are initial objects and unary coproducts are objects equipped with its identity arrow. And we can then define what it is for a category to have all binary coproducts and all finite coproducts in the obvious ways, and show that a category has all finite coproducts iff it has an initial object and all binary coproducts.

8 Equalizers

Terminal and initial objects, products and coproducts, are defined by universal mapping properties. In this chapter, we look at another pair of cases, so-called equalizers and their duals.

8.1 Equalizers

It was useful, when defining products, to introduce the idea of a 'wedge' (Defn. 27) for a certain small configuration of objects and arrows in a category. Here's a similar definition that is going to be useful in defining the equalizers:

Definition 40. A *fork* (from S through X to Y) consists of arrows $k\colon S \to X$ with $f\colon X \to Y$ and $g\colon X \to Y$, such that $f \circ k = g \circ k$. $\lhd$

So diagrammatically, a fork looks like this: $S \xrightarrow{k} X \overset{f}{\underset{g}{\rightrightarrows}} Y$, with the composite arrows from S to Y being equal.

Now, a product wedge from O to X and Y is a limiting case, a wedge such that any other wedge from S to X and Y uniquely factors through it. Likewise, an equalizing fork from E through X to Y is another limiting case, a fork such that any other fork from an object S through X to Y uniquely factors through it (so this is another definition via a universal mapping property). That is to say

Definition 41. Let $\mathscr{C}$ be a category and $f, g\colon X \to Y$ be a pair of parallel arrows in $\mathscr{C}$. Then the object E and arrow $e\colon E \to X$ form an *equalizer* in $\mathscr{C}$ for those arrows iff $f \circ e = g \circ e$ (so $E \xrightarrow{e} X \overset{f}{\underset{g}{\rightrightarrows}} Y$ is indeed a fork), and for any fork $S \xrightarrow{k} X \overset{f}{\underset{g}{\rightrightarrows}} Y$ there is a unique mediating arrow $u\colon S \to E$ such the following diagram commutes:

$$\begin{tikzcd}
S \arrow[d, dashed, "u"'] \arrow[dr, "k"] & & \\
E \arrow[r, "e"'] & X \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & Y
\end{tikzcd}$$

$\lhd$

Note that, just as with products (see Defn. 28), we can give an alternative definition which defines equalizers in terms of a terminal object in a category of forks. First we say

Definition 42. Given a category $\mathscr{C}$ and parallel arrows $f, g\colon X \to Y$, then the derived fork category $\mathscr{C}_{F(XY)}$ has as objects all forks $S \xrightarrow{k} X \overset{f}{\underset{g}{\rightrightarrows}} Y$. And an arrow from $S \xrightarrow{k} \cdots$ to $S' \xrightarrow{k'} \cdots$ in $\mathscr{C}_{F(XY)}$ is a $\mathscr{C}$-arrow $g\colon S \to S'$ such that the resulting triangle commutes: i.e. such that $k = k' \circ g$.

The identity arrow in $\mathscr{C}_{F(XY)}$ on the fork $S \xrightarrow{k} \cdots$ is the identity arrow 1_S in $\mathscr{C}$; and the composition of arrows in $\mathscr{C}_{F(XY)}$ is defined as the composition of the arrows as they feature in $\mathscr{C}$. $\lhd$

It is again easily checked that this indeed defines a category. Our definition of an equalizer then comes to this:

Definition 43. An equalizer of $f, g\colon X \to Y$ is some $[E, e]$ such the resulting fork $E \xrightarrow{e} X \overset{f}{\underset{g}{\rightrightarrows}} Y$ is terminal in $\mathscr{C}_{F(XY)}$. $\lhd$

Let's immediately give some examples of equalizers:

(1) Suppose in Set we have the parallel arrows $X \overset{f}{\underset{g}{\rightrightarrows}} Y$. Then consider the set $E \subseteq X$ such that $x \in E$ implies $fx = gx$, and let $e\colon E \to X$ be the obvious inclusion map. We show that $[E, e]$ is an equalizer for f and g.

By construction, $f \circ e = g \circ e$. So suppose $S \xrightarrow{k} X \overset{f}{\underset{g}{\rightrightarrows}} Y$ is any other fork through f, g. Since $f(k(s)) = g(k(s))$ for each $s \in S$, the set $k[S] = E$: so defining the mediating arrow $u\colon S \to E$ to agree with $k\colon S \to X$ on all inputs will make the diagram for equalizers commute.

It remains to show that this is the unique candidate for the function u. But note $k = e \circ u$, and e doesn't change the values of the function (only its codomain), so k and u must indeed agree on all inputs.

(2) Equalizers in categories whose objects are sets-with-structure behave similarly. Take the category Mon, for example. Given a pair of monoid homomorphisms $(X, \cdot) \overset{f}{\underset{g}{\rightrightarrows}} (Y, *)$, take the subset E of X on which the functions agree. Evidently E must contain the identity element of X (since f and g agree on this element: being homomorphisms, both must send it to the identity element of Y). And suppose $e, e' \in E$: then $f(e \cdot e') = f(e) * f(e') = g(e) * g(e') = g(e \cdot e')$, which means that E is closed under products of members.

So take E together with the monoid operation from $(X, \cdot)$ restricted to members of E. Then $(E, \cdot)$ is a monoid – for the identity element shared with $(X, \cdot)$ still behaves as an identity, and the operation is still associative.

And if we take $(E, \cdot)$ and equip it with the injection homomorphism into $(X, \cdot)$, this will evidently give us an equalizer for $(X, \cdot) \overset{f}{\underset{g}{\rightrightarrows}} (Y, *)$.

(3) Similarly, take Top. What is the equalizer for a pair of continuous maps $X \overset{f}{\underset{g}{\rightrightarrows}} Y$? Well, take the subset of (the underlying set of) X on which the functions agree, and give it the subspace topology. This topological space equipped with the injection into X is then the desired equalizer. (This works because of the way that the subspace topology is defined – we won't go into details).

(4) A special case. Suppose we are in Grp and have a group homomorphism, $f\colon X \to Y$. There is also another trivial homomorphism $o\colon X \to Y$ which sends any element of the group X to the identity element in Y, i.e. is the composite $X \to 1 \to Y$ of the only possible homomorphisms. Now consider what would constitute an equalizer for f and o.

Suppose K is the kernel of f, i.e. the subgroup of X whose objects are the elements which f sends to the identity element of Y, and let $i\colon K \to X$ be the inclusion map. Then $K \xrightarrow{i} X \overset{f}{\underset{o}{\rightrightarrows}} Y$ is a fork since $f \circ i = o \circ i$.

Let $S \xrightarrow{k} X \overset{f}{\underset{o}{\rightrightarrows}} Y$ be another fork. Now, $o \circ k$ sends every element of S to the unit of Y. Since $f \circ k = o \circ k$, k must send any element of S to some element in the kernel K. So let $k'\colon S \to K$ agree with $k\colon S \to X$ on all arguments.

Then the following commutes:

$$\begin{tikzcd} S \arrow[d, "k'"'] \arrow[dr, "k"] & & \\ & X \arrow[r, shift left, "f"] \arrow[r, shift right, "o"'] & Y \\ K \arrow[ur, "i"'] & & \end{tikzcd}$$

And evidently k' is the only possible homomorphism to make the diagram commute.

So the equalizer of f and o is f's kernel K equipped with the inclusion map into the domain of X. Or putting it the other way about, we can define kernels of group homorphisms categorially in terms of equalizers.

(5) Finally we remark that the equalizer of a pair of maps $X \overset{f}{\underset{g}{\rightrightarrows}} Y$ where in fact $f = g$ is simply $[X, 1_X]$.

Consider then a poset $(P, \preccurlyeq)$ considered as a category whose objects are the members of P and where there is a unique arrow $X \to Y$ (for $X, Y \in P$) iff $X \preccurlyeq Y$. So the only cases of parallel arrows from X to Y are cases of equal arrows which then, as remarked, have equalizers. So in sum, a poset category has all possible equalizers.

8.2 Uniqueness again

Just as products are unique up to unique isomorphism, equalizers are too. That is to say,

Theorem 36. *If both the equalizers $[E, e]$ and $[E', e']$ exist for* $X \overset{f}{\underset{g}{\rightrightarrows}} Y$, *then there is a unique isomorphism $j \colon E \xrightarrow{\sim} E'$ commuting with the equalizing arrows, i.e. such that $e = e' \circ j$.*

Plodding proof from first principles. We can use an argument that goes along exactly the same lines as the one we used to prove the uniqueness of products and equalizers. This is of course no accident, given the similarity of the definitions.

Assume $[E, e]$ equalizes f and g, and suppose $e \circ h = e$. Then observe that the following diagram will commute

$$\begin{tikzcd} E \arrow[d, "h"'] \arrow[dr, "e"] & & \\ E \arrow[r, "e"'] & X \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & Y \end{tikzcd}$$

So $h = 1_E$ makes that diagram commute. But by hypothesis there is a unique arrow $E \to E$ which makes the diagram commute. So we can conclude that if $e \circ h = e$, then $h = 1_E$.

Now suppose $[E', e']$ is also an equalizer for f and g. Then $[E, e]$ must factor uniquely through it. That is to say, there is a (unique) mediating $j \colon E \to E'$ such that $e' \circ j = e$. And since $[E, e]$ must factor uniquely though $[E', e']$ there is a unique k such that $e \circ k = e'$. So $e \circ k \circ j = e$, and hence by our initial conclusion, $k \circ j = 1_E$.

A similar proof shows that $j \circ k = 1_{E'}$. Which makes the unique j an isomorphism. ☐

Proof using the alternative definition of equalizers. $[E, e]$ and $[E', e']$ are both terminal objects in the fork category $\mathscr{C}_{F(XY)}$. So by Theorem 14 there is a unique $\mathscr{C}_{F(XY)}$-isomorphism j between them. But, by definition, this has to be a $\mathscr{C}$-arrow $j \colon E \xrightarrow{\sim} E'$ commuting with the equalizing arrows. And j is easily seen to be an isomorphism in $\mathscr{C}$ too. ☐

Let's add two further general results about equalizers. First:

Theorem 37. *If $[E, e]$ constitute an equalizer, then e is a monomorphism.*

Proof. Assume $[E, e]$ equalizes $X \overset{f}{\underset{g}{\rightrightarrows}} Y$, and suppose $e \circ g = e \circ h$, where $D \overset{g}{\underset{h}{\rightrightarrows}} E$. Then the following diagram commutes,

$$\begin{tikzcd}
D \arrow[d, shift right, "g"'] \arrow[d, shift left, "h"] \arrow[dr, "e\circ g = e\circ h"] & & \\
 & X \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & Y \\
E \arrow[ur, "e"'] & &
\end{tikzcd}$$

So $D \xrightarrow{e\circ g} X \overset{f}{\underset{g}{\rightrightarrows}} Y$ is a fork factoring uniquely through the equalizer, and hence there is only one arrow to do the factoring, i.e. $g = h$. So e is left-cancellable in the equation $e \circ g = e \circ h$; i.e. e is monic. □

Second, in an obvious shorthand,

Theorem 38. *In any category, an epic equalizer is an isomorphism*

Proof. Assume again that $[E, e]$ equalizes $X \overset{f}{\underset{g}{\rightrightarrows}} Y$, so that $f \circ e = g \circ e$. So if e is epic, it follows that $f = g$. Then consider the following diagram

$$\begin{tikzcd}
X \arrow[dd, dashed, "u"'] \arrow[dr, "1_X"] & & \\
 & X \arrow[r, "f"', "g"] & Y \\
E \arrow[ur, "e"'] & &
\end{tikzcd}$$

Because e equalizes, we know there is a unique u such that (i) $e \circ u = 1_X$.

But then also $e \circ (u \circ e) = 1_X \circ e = e = e \circ 1_E$. Hence, since equalizers are mono by the last theorem, (ii) $u \circ e = 1_E$.

Taken together, (i) and (ii) tell us that e has an inverse. Therefore e is an isomorphism. □

8.3 Subsets and subobjects

(a) In §8.1, we saw that in Set, given two parallel arrows from an object X, a certain subset of X (together with the trivial inclusion function) provides an equalizer for the arrows. And §8.2 tells us that this is the unique equalizer, up to isomorphism.

We now note that a reverse result holds too:

Theorem 39. *In* Set*, any subset S of X together with its natural inclusion map $i\colon S \to X$ form an equalizer for certain parallel arrows from X.*

Proof. A subset $S \subseteq X$ has an associated *characteristic function* $s\colon X \to \{0, 1\}$ which sends $x \in X$ to 1 ('true') if $x \in S$ and to 0 otherwise.

Another function from X to $\{0, 1\}$ is the completely indiscriminate map c which sends everything in X to 1.

We now show that $[S, i]$ is an equalizer for $X \overset{s}{\underset{c}{\rightrightarrows}} \{0, 1\}$.

First, it is trivial that $s \circ i = c \circ i$, so as required $S \xrightarrow{i} X \overset{s}{\underset{c}{\rightrightarrows}} \{0,1\}$ is indeed a fork.

So it remains to show that any upper fork in this next diagram factors through the lower fork via a unique mediating u:

$$\begin{tikzcd}
R \arrow[dr, "f"] \arrow[dd, dashed, "u"'] & & \\
& X \arrow[r, shift left, "s"] \arrow[r, shift right, "c"'] & \{0,1\} \\
S \arrow[ur, "i"'] & &
\end{tikzcd}$$

Since $s \circ f = c \circ f$ by assumption, it is immediate that $f[R] \subseteq S$. Hence, if we define $u\colon R \to S$ to agree with $f\colon R \to X$ on all inputs, then the diagram commutes. And this u is evidently the only possible candidate. $\square$

(b) Since $[S, i]$ is the unique equalizer for $X \overset{s}{\underset{c}{\rightrightarrows}} \{0,1\}$ up to isomorphism, this gives us a nice categorial way of characterizing subsets in Set as equalizers.

Here's a variant way of putting the same thought, taking a step towards further generalization. First, we remind ourselves that using the set $\{0,1\}$ as the set of truth-values is of course arbitrary. So let's now use Ω to denote some appropriate set. And let $t\colon 1 \to \Omega$ then be the map that picks out the truth-value 'true' from Ω. A characteristic function for a subset of X can now be treated as an arrow $X \to \Omega$.

Note next that the map $c \circ i\colon S \to \Omega$, which sends everything in S to the value 'true', is trivially equal to composite map

$$S \xrightarrow{\;!\;} 1 \xrightarrow{\;t\;} \Omega$$

with 1 a terminal object in the category. Similarly for the map $c \circ f\colon R \to \Omega$.

Now let us generalise by thinking not in terms of a strict inclusion map i but in terms of a monomorphism $i\colon S \rightarrowtail X$ which 'injects' S into X. Then, massaging our earlier diagram, the claim that $[S, i]$ is tantamount to a subset of X comes to this. There is a map $s\colon X \to \Omega$ (intuitively, the characteristic function for the subset S) such that for any $f\colon R \to X$ such that outer (bent) square commutes, there is unique u which makes the whole diagram commute:

$$\begin{tikzcd}
R \arrow[dr, dashed, "u"] \arrow[drr, bend left, "!"] \arrow[ddr, bend right, "f"'] & & \\
& S \arrow[r, "!"] \arrow[d, tail, "i"] & 1 \arrow[d, "t"] \\
& X \arrow[r, "s"] & \Omega
\end{tikzcd}$$

Now, this is perhaps not as transparent as our first way of putting things. But the attraction is that this categorial characterization of subsets in Set can be ported to given a useful general definition of subobjects in other categorties. More about all this in due course.

8.4 Co-equalizers

(a) We dualize our definition of an equalizer to get the notion of a co-equalizer. So, as a preliminary, we say

Definition 44. A *co-fork* (from X through Y to S) consists of parallel arrows $f\colon X \to Y$, $g\colon X \to Y$ and an arrow $k\colon Y \to S$, such that $k \circ f = k \circ g$. $\lhd$

(Actually, plain 'fork' is used for the dual too: but the barbarous 'co-fork' keeps things clear.) So diagrammatically, a co-fork looks like this: $X \overset{f}{\underset{g}{\rightrightarrows}} Y \xrightarrow{k} S$, with the composite arrows from X to S being equal.

Definition 45. Let $\mathscr{C}$ be a category and $f\colon X \to Y$ and $g\colon X \to Y$ be a pair of parallel arrows in $\mathscr{C}$. Then the object C and arrow $c\colon Y \to S$ form a *co-equalizer* in $\mathscr{C}$ for those arrows iff $c \circ f = c \circ g$, and for any co-fork from X through Y to S there is a unique arrow $u\colon C \to S$ such the following diagram commutes:

$$\begin{tikzcd} & & S \\ X \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & Y \arrow[ur, "k"] \arrow[dr, "c"'] & \\ & & C \arrow[uu, dashed, "u"'] \end{tikzcd}$$

$\lhd$

As we would now expect, by a dual argument, co-equalizers are unique up to a unique isomorphism; and since equalizers are monic, co-equalizers are epic. We won't pause to spell out those results, but turn immediately to give just one central example.

(b) What do co-equalizers look like? In Set, the parallel arrows $f, g\colon X \to Y$ determine a relation R on the members of Y where $y_1 R y_2$ holds when there is an $x \in X$ such that $f(x) = y_1 \wedge g(x) = y_2$.

Now if there is a co-fork $X \overset{f}{\underset{g}{\rightrightarrows}} Y \xrightarrow{k} S$, then if $y_1 R y_2$ then $k(y_1) = k(y_2)$ (and of course, being mapped to the same value by k is an equivalence relation). Having equal k-values is an equivalence relation. So, for every co-fork, there is a corresponding equivalence relation such that being R-related implies being in that equivalence relation. What's the limiting case of such an equivalence relation? It will have to be $R^{\sim}$, the smallest equivalence relation containing R.

That observation gives us the clue about how to proceed, still working in Set.

Theorem 40. *Given functions $f, g\colon X \to Y$ in* Set*, let $R^{\sim}$ be the smallest equivalence relation containing R – where y_1Ry_2 iff $(\exists x \in X)(f(x) = y_1 \wedge g(x) = y_2)$.*

Let C be $Y/R^{\sim}$, i.e. the set of $R^{\sim}$-equivalence classes of Y; and let c map $y \in Y$ to the $R^{\sim}$-equivalence class containing y. Then $[C, c]$, so defined, is a co-equalizer for f and g.

Proof. First we need to check that $c \circ f = c \circ g$. But the left-hand side sends $x \in X$ to the $R^{\sim}$-equivalence class containing $f(x)$ and the right-hand side sends x to the $R^{\sim}$-equivalence class containing $g(x)$. However, $f(x)$ and $g(x)$ are by definition R-related, and hence are $R^{\sim}$-related: so by construction they belong to the same $R^{\sim}$equivalence class. Hence $X \underset{g}{\overset{f}{\rightrightarrows}} Y \xrightarrow{c} C$ is indeed a co-fork.

Now suppose there is another co-fork $X \underset{g}{\overset{f}{\rightrightarrows}} Y \xrightarrow{k} S$ We need to show the first co-fork will factor through this via a unique mediating arrow u.

By assumption, $k \circ f = f \circ g$. And we first outline a proof that if $y_1R^{\sim}y_2$ then $k(y_1) = k(y_2)$.

Start with R defined as before, and let R' be its reflexive closure. Obviously we'll still have that if $y_1R'y_2$ then $k(y_1) = k(y_2)$. Now consider R'' the symmetric closure of R': again obviously, we'll still have that $y_1R''y_2$ then $k(y_1) = k(y_2)$. Now note that if $y_1R''y_2$ and $y_2R''y_3$, then $k(y_1) = k(y_3)$. So if we take the transitive closure of R'', we'll still have a relation which, when it holds between some y_1 and y_2, implies that $k(y_1) = k(y_2)$. But the transitive closure of R'' is $R^{\sim}$.

We have shown, then, that k is constant on members of a $R^{\sim}$-equivalence class, and so we can well-define a function $u\colon C \to S$ which sends an equivalence class to the value of k on a member of that class. This u is the desired mediating arrow which makes the diagram defining a co-equalizer commute. Moreover, since c is surjective and C only contains $R^{\sim}$-equivalence classes, u is the only function for which $u \circ c = k$. □

In a slogan then: *in* Set*, quotienting out by an equivalence relation is (up to unique isomorphism) the same as taking an associated co-equalizer*. In many other categories co-equalizers behave similarly, corresponding to 'naturally occurring' quotienting constructions. But we won't go into more detail here.

9 Limits and colimits defined

A terminal object is defined in terms of how all other objects in the category relate to it (by each sending it a unique arrow). A product wedge is defined in terms of how all other wedges in a certain family relate to it (each factoring through it via a unique arrow). An equalizing fork is defined in terms of how all other forks in a certain family relate to it (each factoring through it via a unique arrow). In an informal sense, terminal objects, products, and equalizers are limiting cases, defined in closely analogous ways. Likewise for their duals.

In this chapter, we now formally capture what's common to terminal objects, products and equalizers by defining a general class of *limits*, and confirming that terminal objects, products and equalizers are indeed examples. We also define a dual class of *co-limits*, which has initial objects, coproducts and co-equalizers as examples. We then give a new pair of examples, one for each general class, the so-called pullbacks and pushouts.

9.1 Defining limit cones

(a) Way back in Defn. 3, we characterized a diagram D in a category $\mathscr{C}$ as being simply a bunch of objects with some arrows between some of them. We need some way of indexing these objects to refer to them, so henceforth we'll refer to objects in D by terms like 'D_j' where j is in some appropriate index set. And for convenience, we'll allow double counting, permitting the case where $D_j = D_k$ for different indices. We will also allow the limiting cases where there are no arrows, and even the empty case where there are no objects.

Definition 46. Let D be a diagram in category $\mathscr{C}$. Then a *cone over* D comprises a $\mathscr{C}$-object C, the *vertex* of the cone, together with $\mathscr{C}$-arrows $c_j : C \to D_j$, one 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 diagram commutes:

$$\begin{tikzcd} & C \arrow[dl, "c_k"'] \arrow[dr, "c_l"] & \\ D_k \arrow[rr, "d"'] & & D_l \end{tikzcd}$$

We use '$[C, c_j]$' as our notation for such a cone. $\lhd$

Think of it diagrammatically(!) like this: arrange the objects in the diagram D in a plane, along with whatever arrows there are between them in D. Now sit the object C above the plane, with a quiverful of arrows from C zinging down, one to each object D_j in the plane. Those arrows form a skeletal cone. And the key requirement is that any triangles thus formed with C at the apex must commute.

We should note, by way of aside, that some authors prefer to say more austerely that a cone is not a vertex-object-with-a-family-of-arrows-from-that-vertex but simply a family of arrows from the vertex. Since we can read off the vertex of a cone as the common source of all its arrows, it is very largely a matter of convenience whether we speak austerely or explicitly mention the vertex. But for the moment, we'll take the less austere line.

There can be many cones, with different vertices, over a given diagram D. But, in the same spirit as our earlier definitions of products and equalizers, we can define a limiting case, by means of a universal mapping property:

Definition 47. A cone $[L, \pi_j]$ over a diagram D in $\mathscr{C}$ is a *limit (cone) over D* iff any cone $[C, c_j]$ over D uniquely factors through it, so there a unique mediating arrow $u\colon C \to L$ such that for each index j, $\pi_j \circ u = c_j$. In other words, for each D_j, D_k, D_l, $\ldots$, in D, the corresponding triangle with other vertices C and L commutes:

$$\begin{tikzcd}
 & & C \arrow[d, dashed, "u"] \arrow[dll, "c_j"'] \arrow[dr, "c_k"] \arrow[drr, "c_l"] & & & \\
 & & L \arrow[dll, "\pi_j"] \arrow[dr, "\pi_k"'] \arrow[drr, "\pi_l"] & & & \\
D_j & & & D_k & D_l & \cdots
\end{tikzcd}$$

$\lhd$

(b) Let's immediately confirm that our three announced examples of limits so far are indeed limit cones in the sense just defined.

(1) We start with the null case. Take the empty diagram in $\mathscr{C}$ – *zero* objects and so, necessarily, no arrows. Then a cone over the empty diagram is simply an object C, a lonely vertex (there is no further condition to fulfil), and an arrow between such cones is just an arrow between objects in $\mathscr{C}$. Hence L is a limit cone just if there is a unique arrow to it from any other object – i.e. just if L is a terminal object in $\mathscr{C}$!

(2) Consider now a diagram which is just *two* objects we'll call 'D_1', 'D_2', still with no arrow between them. Then a cone over such a diagram is just a wedge into D_1, D_2; and a limit cone is simply a product of D_1 with D_2.

(3) Next consider a diagram which again has just two objects, but now with two parallel arrows between them, which we can represent $D_1 \underset{d'}{\overset{d}{\rightrightarrows}} D_2$. Then a cone over this diagram is a commuting diagram like this:

$$\begin{tikzcd} & C \arrow[dl, "c_1"'] \arrow[dr, "c_2"] & \\ D_1 \arrow[rr, shift left, "d"] \arrow[rr, shift right, "d'"'] & & D_2 \end{tikzcd}$$

If there is such a diagram, then we must have $d \circ c_1 = d' \circ c_1$: and vice versa, if that identity holds, then we can put $c_2 = d \circ c_1 = d' \circ c_1$ to complete the commutative diagram. Hence we have a cone from the vertex C to our diagram iff $C \xrightarrow{c_1} D_1 \underset{d'}{\overset{d}{\rightrightarrows}} D_2$ is a fork. Since c_1 fixes what c_2 has to be to complete the cone, we can focus on the cut-down cone consisting of just $[C, c_1]$.

What is the corresponding cut-down limit cone? It consists in $[E, e]$ such there is a unique u such that $c_1 = e \circ u$. Hence $[E, e]$ is an equalizer of $D_1 \underset{d'}{\overset{d}{\rightrightarrows}} D_2$.

(c) We can now give a direct proof, along now familiar lines, for the predictable result

Theorem 41. *Limit cones over a given diagram D are unique up to a unique isomorphism commuting with the cones's arrows.*

Proof. As usual, we first note that a limit cone $[L, \pi_j]$ factors through itself via the mediating identity $1_L \colon L \to L$. But by definition, a cone over D uniquely factors through the limit, so that means that

(i) if $\pi_j \circ u = \pi_j$ for all j, then $u = 1_L$.

Now suppose $[L', \pi'_j]$ is another limit cone over D. Then $[L', \pi'_j]$ uniquely factors through $[L, \pi_j]$, via some f, so

(ii) $\pi_j \circ f = \pi'_j$ for all j.

And likewise $[L, \pi_j]$ uniquely factors through $[L', \pi'_j]$ via some g, so

(iii) $\pi'_j \circ g = \pi_j$ for all j.

Whence

(iv) $\pi_j \circ f \circ g = \pi_j$ for all j.

Therefore

(v) $f \circ g = 1_L$.

And symmetrically

(vi) $g \circ f = 1_{L'}$.

Whence f is not just unique (by hypothesis, the only way of completing the relevant diagrams to get the arrows to commute) but an isomorphism. □

9.2 Limit cones as terminal objects

We have already seen that

(1) A terminal object in $\mathscr{C}$ is ... wait for it! ... terminal in the given category $\mathscr{C}$.

(2) The product of X with Y in $\mathscr{C}$ is a terminal object in the derived category $\mathscr{C}_{W(X,Y)}$ of wedges to X and Y.

(3) The equalizer of parallel arrows through X to Y in $\mathscr{C}$ are (parts of) terminal objects in the derived category $\mathscr{C}_{F(XY)}$ of forks through X to Y.

Predictably, limit cones more generally are terminal objects in appropriate categories.

To spell this out, we first note that the cones $[C, c_j]$ over a given diagram D in $\mathscr{C}$ form a category in a very natural way:

Definition 48. Given a diagram D in category $\mathscr{C}$, the derived category $\mathscr{C}_{C(D)}$ – *the category of cones over D* – has the following data:

(1) Its objects are the cones $[C, c_j]$ over D.

(2) An arrow from $[C, c_j]$ to $[C', c'_j]$ is any $\mathscr{C}$-arrow $f\colon C \to C'$ such that $c'_j \circ f = c_j$ for all indices j. In other words, for each D_j, D_k, D_l, ..., in D, the corresponding triangle with remaining vertices C and C' commutes:

$$\begin{tikzcd}
& C \arrow[ddl, "c_j"'] \arrow[d, dashed, "f"] \arrow[ddr, "c_k"] \arrow[ddrr, "c_l"] & & & \\
& C' \arrow[dl, "c'_j"] \arrow[dr, "c'_k"'] \arrow[drr, "c'_l"] & & & \\
D_j & & D_k & D_l & \cdots
\end{tikzcd}$$

The identity arrow on a cone $[C, c_j]$ is the $\mathscr{C}$-arrow 1_C. And composition for arrows in $\mathscr{C}_{C(D)}$ is just composition of the corresponding $\mathscr{C}$-arrows. ◁

It is entirely routine to confirm that $\mathscr{C}_{C(D)}$ is indeed a category. We can then recast our earlier definition of a limit cone as follows:

Definition 49. A *limit cone for D* in $\mathscr{C}$ is a cone which is a terminal object in $\mathscr{C}_{C(D)}$. ◁

And we now have an alternative proof of our last theorem, that limit cones over a given diagram D are unique up to unique isomorphism commuting with .

Proof. Since a limit cone over D is terminal in $\mathscr{C}_{C(D)}$, it is unique in $\mathscr{C}_{C(D)}$ up to a unique isomorphism. But such an isomorphism in $\mathscr{C}_{C(D)}$ must be an isomorphism in $\mathscr{C}$ commuting with the cones's arrows. □

9.3 Results about limits

(a) Let's first prove two further simple theorems:

Theorem 42. *Suppose $[L, \pi_j]$ is a limit cone over a diagram D in $\mathscr{C}$, and $[L', \pi'_j]$ is another cone over D which factors through $[L, \pi_j]$ via an isomorphism f. Then $[L', \pi'_j]$ is also a limit cone.*

Proof. Take any cone $[C, c_j]$ over D. We need to show that (i) there is an arrow $v: C \to L'$ such that for all indexing objects j in D, $c_j = \pi'_j \circ v$, and (ii) v is unique.

But we know that there is a unique arrow $u: C \to L$ such that for j, $c_j = \pi_j \circ u$. And we know that $f: L' \to L$ and $\pi'_j = \pi_j \circ f$ (so $\pi_j = \pi'_j \circ f^{-1}$).

Therefore put $v = f^{-1} \circ u$, and that satisfies (i).

Now suppose there is another arrow $v': C \to L'$ such that $c_j = \pi'_j \circ v'$. Then we have $f \circ v': C \to L$, and also $c_j = \pi_j \circ f \circ v'$. Therefore $[C, c_j]$ factors through $[L, \pi_j]$ via $f \circ v'$, so $f \circ v' = u$. Whence $v' = f^{-1} \circ u = v$. Which proves (ii). □

Theorem 43. *Suppose $[L, \pi_j]$ is a limit cone over a diagram D in $\mathscr{C}$. Then the cones over D with vertex C correspond one-to-one with $\mathscr{C}$-arrows from C to L.*

Proof. Take any arrow $u: C \to L$. If there is an arrow $d: D_k \to D_l$ in the diagram D, then (since $[L, \pi_j]$ is a cone), $\pi_l = d \circ \pi_k$, whence $(\pi_l \circ u) = d \circ (\pi_k \circ u)$. Since this holds generally, $[C, \pi_j \circ u]$ is a cone over D. But (again since $[L, \pi_j]$ is a limit) every cone over D with vertex C is of the form $[C, \pi_j \circ u]$ for unique u. Hence there is indeed a one-one correspondence between arrows $u: C \to L$ and cones over D with vertex C. (Moreover, the construction is a natural one, involving no arbitrary choices.) □

(b) As a fun exercise and reality check, let's remark that the whole category $\mathscr{C}$ can be thought of as the limiting case of a diagram in itself, and then

Theorem 44. *A category $\mathscr{C}$ has an initial object if and only if $\mathscr{C}$, thought of as a diagram in $\mathscr{C}$, has a limit.*

Proof. Suppose $\mathscr{C}$ has an initial object I. Then for every $\mathscr{C}$-object C, there is a unique arrow π_C. $[I, \pi_C]$ is a cone (since for any arrow $f \colon C \to D$, the composite $f \circ \pi_C$ is an arrow from I to D and hence has to be equal to the unique π_D. Further, $[I, \pi_C]$ is a limit cone. For suppose $[A, a_C]$ is any other cone over the whole of $\mathscr{C}$. Then since it is a cone, the triangle

$$\begin{tikzcd} & A \arrow[dl, "a_I"'] \arrow[dr, "a_C"] & \\ I \arrow[rr, "\pi_C"'] & & C \end{tikzcd}$$

has to commute for all C. But that's just the condition for $[A, a_C]$ factoring through $[I, \pi_C]$ via a_I. And moreover, suppose $[A, a_C]$ also factors through by some u. Then in particular,

$$\begin{tikzcd} & A \arrow[dl, "u"'] \arrow[dr, "a_I"] & \\ I \arrow[rr, "1_I"'] & & I \end{tikzcd}$$

commutes, and so $u = a_C$. So the factoring is unique, and $[I, \pi_C]$ is a limit cone.

Now suppose, conversely, that $[I, \pi_C]$ is a limit cone over the whole of $\mathscr{C}$. Then there is an arrow $\pi_C \colon I \to C$ for each C in $\mathscr{C}$. If we can show it is unique, I will indeed be initial.

Suppose then that there is an arrow $k \colon I \to C$ for a given C. Then since $[I, \pi_C]$ is a cone, the diagram

$$\begin{tikzcd} & I \arrow[dl, "\pi_I"'] \arrow[dr, "\pi_C"] & \\ I \arrow[rr, "k"'] & & C \end{tikzcd}$$

has to commute. Considering the case where $k = \pi_C$, we see that $[I, \pi_C]$ factors through itself via π_I; but it also factors via 1_D, so the uniqueness of factorization entails $\pi_I = 1_D$. Hence the diagram shows that for any $k \colon I \to C$ has to be identical to π_C. So I is initial. $\Box$

(c) Before proceeding further, let's introduce some standard notation:

Definition 50. We denote the limit object at the vertex of a limit cone for the diagram D with objects D_j by '$\underset{\leftarrow j}{Lim}\, D_j$'. $\triangleleft$

Do note, however, that since limit cones are only unique up to isomorphism, different but isomorphic objects can be denoted in different contexts by '$\underset{\leftarrow j}{Lim}\, D_j$'.

The projection arrows from this limit object to the various objects D_j will then naturally be denoted '$\pi_i \colon \underset{\leftarrow j}{Lim}\, D_j \to D_i$', and the limit cone could therefore be represented by '$[\underset{\leftarrow j}{Lim}\, D_j, \pi_j]$'. (The direction of the arrow under '*Lim*' in this notation is perhaps unexpected, but we just have to learn to live with it.)

9.4 Colimits defined

The headline, and thoroughly predictable, story about duals is: reverse the relevant arrows and you get a definition of colimits.

So, dualizing §9.1 and wrapping everything together, we get:

Definition 51. Let D be a diagram in category $\mathscr{C}$. Then a *cocone under D* is a $\mathscr{C}$-object C, together with an arrow $c_j \colon D_j \to C$ 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 diagram commutes:

$$\begin{tikzcd} D_k \arrow[rr, "d"'] \arrow[dr, "c_k"'] & & D_l \arrow[dl, "c_l"] \\ & C & \end{tikzcd}$$

The cocones under D form a category with objects the cocones $[C, c_j]$ and an arrow from $[C, c_j]$ to $[C', c'_j]$ being any $\mathscr{C}$-arrow $f \colon C \to C'$ such that $c'_j = f \circ c_j$ for all indexes j. A colimit for D is an initial object in the category of cocones under D. It is standard to denote the object at the vertex of the colimit cocone for the diagram D by '$\underset{\rightarrow j}{Lim}\, D_j$'. $\lhd$

It is now routine to confirm that our earlier examples of initial objects, coproducts and co-equalizers do count as colimits.

(1) The null case where we start with the empty diagram in $\mathscr{C}$ gives rise to a cocone which which is simply an object in C. So the category of cocones over the empty diagram is just the category $\mathscr{C}$ we started with, and a limit cocone is just an initial object in $\mathscr{C}$!

(2) Consider now a diagram which is just *two* objects we'll call 'D_1', 'D_2', still with no arrow between them. Then a cocone over such a diagram is just a corner from D_1, D_2 (in the sense we met in §7.6); and a limit cocone in the category of such cocones is simply a coproduct.

(3) And if we start with the diagram $D_1 \underset{d'}{\overset{d}{\rightrightarrows}} D_2$ then a limit cocone over this diagram gives rise to a co-equalizer.

9.5 Pullbacks

(a) Let's illustrate all this by briefly exploring another kind of limit (in this section) and its dual (in the next section).

A co-wedge or, as I prefer to say, a corner D in category $\mathscr{C}$ is a diagram which can be represented like this:

$$\begin{tikzcd} & D_3 \arrow[d, "e"] \\ D_1 \arrow[r, "d"] & D_2 \end{tikzcd}$$

Now, a cone over our corner diagram has a rather familiar shape, i.e. it is a commutative square:

$$\begin{tikzcd} C \arrow[r, "c_3"] \arrow[d, "c_1"] \arrow[dr, "c_2"] & D_3 \arrow[d, "e"] \\ D_1 \arrow[r, "d"] & D_2 \end{tikzcd}$$

Though note, we needn't really draw the diagonal here, for if the sides of the square commute thus ensuring $d \circ c_1 = e \circ c_3$, then we know the diagonal c_2 exists making the triangles commute.

And a limit for this type of cone will be a cone with vertex $L = \varprojlim_j D_j$ and projections $\pi_j : L \to D_j$ such that for any cone $[C, c_j]$ over D, there is a unique $u : C \to L$ such that this whole diagram commutes:

$$\begin{tikzcd} C \arrow[dr, "u"] \arrow[drr, bend left, "c_3"] \arrow[ddr, bend right, "c_1"'] & & \\ & L \arrow[r, "\pi_3"] \arrow[d, "\pi_1"] & D_3 \arrow[d, "e"] \\ & D_1 \arrow[r, "d"] & D_2 \end{tikzcd}$$

(And note that if this commutes, there's just one possible $\pi_2 : L \to D_2$ and $c_2 : C \to D_2$ which makes the diagram still commute.)

Definition 52. A limit for a corner diagram is a *pullback*. The whole square formed by the original corner and its limit is a *pullback square*. $\lhd$

(b) Let's immediately have some examples of pullback squares living in the category Set.

(1) Changing the labelling, consider a corner comprising three sets X, Y, Z and a pair of functions which share the same codomain, thus:

$$\begin{tikzcd} & Y \arrow[d, "g"] \\ X \arrow[r, "f"] & Z \end{tikzcd}$$

We know from the previous diagram that the limit object L must be product-like (with any wedge over X, Y factoring through the wedge with vertex L). Hence to get the other part of the diagram to commute, the pullback square must have at its apex L something isomorphic to $\{\langle x, y\rangle \in X \times Y \mid f(x) = g(y)\}$ with the obvious projection maps to X and Y.

So suppose first that in fact both X and Y are subsets of Z, and the arrows into Z are both inclusion functions. And we then get a pullback square

$$\begin{tikzcd} L \arrow[r] \arrow[d] & Y \arrow[d, "i"] \\ X \arrow[r, "i"] & Z \end{tikzcd}$$

Then $L \cong \{\langle x, y\rangle \in X \times Y \mid x = y\} = \{\langle z, z\rangle \mid z \in X \cap Y\} \cong X \cap Y$. Hence, in Set, the intersection of a pair of sets is their pullback object (fixed, as usual, up to isomorphism).

(2) Take another case in Set. Suppose we have a corner as before but with $Y = Z$ and $g = 1_Z$. Then

$$L \cong \{\langle x, z\rangle \in X \times Z \mid f(x) = z\} \cong \{x \mid \exists z f(x) = z\} \cong f^{-1}[Z],$$

i.e. a pullback object for this corner is, up to isomorphism, the inverse image of Z, and we have a pullback square

$$\begin{tikzcd} f^{-1}[Z] \arrow[r] \arrow[d] & Z \arrow[d, "1_Z"] \\ X \arrow[r, "f"] & Z \end{tikzcd}$$

Hence in Set, the inverse image of a function is also a pullback object.

(3) Revisiting §8.3, we see that we in effect defined a subset $S \subseteq X$ in Set as a pullback object for a corner

$$\begin{tikzcd} & 1 \arrow[d, "t"] \\ X \arrow[r, "s"] & \{0, 1\} \end{tikzcd}$$

(c) Why 'pullback'? Look at e.g. the diagram in (2). We can say that we get to $f^{-1}[Z]$ from Z by pulling back along f – or more accurately, we get to the arrow $f^{-1}[Z] \to X$ by pulling back the identity arrow on Z along f.

In this sense,

Theorem 45. *Pulling back a monomorphism yields a monomorphism*

In other words, if we start with the same corner $X \xrightarrow{f} Z \xleftarrow{g} Y$ with g monic, and can pullback g along f to give a pullback square

$$\begin{tikzcd} L \arrow[r, "b"] \arrow[d, "a"] & Y \arrow[d, tail, "g"] \\ X \arrow[r, "f"] & Z \end{tikzcd}$$

then the resulting arrow a is monic. (Note, this does not depend on the character of f.)

Proof. Suppose, for some arrows $C \underset{k}{\overset{j}{\rightrightarrows}} L$, $a \circ j = a \circ k$. Then $g \circ b \circ j = f \circ a \circ j = f \circ a \circ k = g \circ b \circ k$. Hence, given that g is monic, $b \circ j = b \circ k$.

It follows that the two cones over the original corner, $X \xleftarrow{a \circ j} C \xrightarrow{b \circ j} Y$ and $X \xleftarrow{a \circ k} C \xrightarrow{b \circ k} Y$ are in fact the *same* cone, and hence must factor through the limit L via the same unique arrow $C \to L$. Which means $j = k$.

In sum, $a \circ j = a \circ k$ implies $j = k$, so a is monic. □

Here's another result about monomorphisms and pullbacks:

Theorem 46. *The arrow $f\colon X \to Y$ is a monomorphism in $\mathscr{C}$ if and only if the following is a pullback square:*

$$\begin{tikzcd} X \arrow[r, "1_X"] \arrow[d, "1_X"] & X \arrow[d, "f"] \\ X \arrow[r, "f"] & Y \end{tikzcd}$$

Proof. Suppose this is pullback diagram. Then any cone $X \xleftarrow{a} C \xrightarrow{b} Y$ over the corner $X \xrightarrow{f} Y \xleftarrow{f} X$ must uniquely factor through the limit with vertex X. That is to say, if $f \circ a = f \circ b$, then there is a u such that $a = 1_X \circ u$ and $b = 1_X \circ u$, hence $a = b$ – so f is monic.

Conversely, if f is monic, then given any cone $X \xleftarrow{a} C \xrightarrow{b} Y$ over the original corner, $f \circ a = f \circ b$, whence $a = b$. But that means the cone factors through the cone $X \xleftarrow{1_X} X \xrightarrow{1_X} X$ via the unique a, making that cone a limit and the square a pullback square. □

(d) We've explained, up to a point, the label 'pullback'. It should now be noted in passing that a pullback is sometimes called a *fibered product* (or fibre product) because of a construction of this kind on fibre bundles in topology. Those who know some topology can chase up the details.

But here's a way of getting products into the story, using an idea that we already know about. Recall the definition of the slice category $\mathscr{C}/Z$. Its objects are arrows$_{\mathscr{C}}$ $f\colon C \to Z$ where C is an object in $\mathscr{C}$, and an arrow from $f\colon X \to Z$ to $g\colon Y \to Z$ is an arrow$_{\mathscr{C}}$ $h\colon X \to Y$ such that $f = g \circ h$ in $\mathscr{C}$.

Now the pullback of the corner formed by f and g in $\mathscr{C}$ is a pair of arrows $a\colon L \to X$ and $b\colon L \to Y$ such that $f \circ a = g \circ b$ $(= k)$ and which form a wedge such that any other wedge $a'\colon L' \to X$, $b'\colon L' \to Y$ such that $f \circ a' = g \circ b'$ $(= k')$ factors uniquely through it.

Looked at as a construction in $\mathscr{C}/Z$, this means taking two $\mathscr{C}/Z$-objects f and g and getting a pair of $\mathscr{C}/Z$-arrows $a\colon k \to f$, $b\colon k \to g$ (check that a is indeed a $\mathscr{C}/Z$-arrow from $f \circ a$ to f!). And this pair of arrows forms a wedge such that any other wedge $a'\colon k' \to f$, $b'\colon k' \to g$ factors uniquely through it. In other words, the pullback in $\mathscr{C}$ is a product in $\mathscr{C}/Z$.

Product notation is often used for pullbacks, thus:

$$\begin{tikzcd}
X \times_Z Y \arrow[r] \arrow[d] \arrow[dr, phantom, "\lrcorner", very near start] & Y \arrow[d] \\
X \arrow[r] & Z
\end{tikzcd}$$

with the little corner conventionally indicating it is indeed a pullback square

9.6 Pushouts

As you would expect, the account of pullbacks dualizes. Suppose we take a wedge D, i.e. a diagram like this: $D_1 \xleftarrow{d} D_2 \xrightarrow{e} D_3$. A cocone under this diagram is another commutative square (omitting again the diagonal arrow which is fixed by the others).

$$\begin{tikzcd}
D_2 \arrow[r, "e"] \arrow[d, "d"'] & D_3 \arrow[d, "c_3"] \\
D_1 \arrow[r, "c_1"] & C
\end{tikzcd}$$

And a limit cocone of this type will be a cocone with apex $L = \underset{\to j}{Lim}\, D_j$ and projections $\pi_j\colon L \to D_j$ such that for any cocone $[C, c_j]$ under D, there is a unique $u\colon L \to C$ such that the obvious dual of the whole pullback diagram above commutes.

Definition 53. A limit for a wedge diagram is a *pushout*. $\lhd$

Now, in Set, we get the limit object for a corner diagram $X \xrightarrow{f} Z \xleftarrow{g} Y$ by taking a certain *subset* of a *product* $X \times Y$. Likewise we get the colimit object for a wedge diagram $X \xleftarrow{f} Z \xrightarrow{g} Y$ by taking a certain *quotient* of a *coproduct* $X \amalg Y$. (Recall from our discussion of (co)-equalizers that quotients are categorially dual to subsets in Set.) In fact, we need to quotient out by the smallest equivalence relation containing the relation R where $\langle x, 0\rangle R \langle y, 1\rangle$ iff there is a z such that $f(z) = x \wedge g(z) = y$. We won't, however, pause further over this now. Though it does illustrate how taking colimits can tend to beget messier constructions than taking limits.

10 The existence of limits

We have seen that a whole range of familiar constructions from various areas of ordinary mathematics can be regarded as instances of taking limits or colimits of (very small) diagrams in appropriate categories. Examples so far include: forming cartesian products or logical conjunctions, taking disjoint unions or free products, quotienting out by an equivalence relation, taking intersections, taking inverse images.

Not *every* familiar kind of 'universal' construction involves taking (co)limits. We'll see in the next chapter that exponentiation involves a different idea. But plainly we are mining a very rich seam here (and also beginning to make good on our promise to show how category theory helps reveal recurring patterns across different areas of mathematics).

It would get tedious, however, to explore what it takes for a category to have limits for various further kinds of diagram case-by-case, even if we just stick to considering limits over very small diagrams. But fortunately we don't need to do such a case-by-case examination. It turns out that if a category has certain basic limits, then it has, in a sense to be clarified, *all* limits.

10.1 Pullbacks, products and equalizers related

(a) Here's an obvious definition:

Definition 54. The category $\mathscr{C}$ *has all finite limits* if for any finite diagram D – i.e. for any diagram whose objects are D_j for indices $j \in I$, where I is a finite set – $\mathscr{C}$ has a limit over D. A category with all finite limits is said to be *finitely complete.* $\triangleleft$

Our main target theorems for this chapter are then as follows:

Theorem 47. *If $\mathscr{C}$ has a terminal object, and has all binary products and equalizers, it is finitely complete.*

Theorem 48. *If $\mathscr{C}$ has a terminal object, and has a pullback for any corner, it is finitely complete.*

Later, we will see how to get an analogous result for limits over infinite diagrams; but it will help fix ideas if we initially focus on the finite case. And of course, our theorems will have the predictable duals: we will return to them in §10.5.

We begin, in this section, by proving the following much more restricted versions of these theorems:

Theorem 49. *If a category $\mathscr{C}$ has binary products and equalizers, then it has a pullback for any corner.*

Theorem 50. *If $\mathscr{C}$ has a terminal object, and has a pullback for any corner, then it has all binary products and hence all equalizers.*

This will have a double pay-off. First, it means that we only need prove one of Theorems 47 and 48, since in the presence of Theorems 49 and 50 they imply each other: we will concentrate on proving Theorem 47. Second, our proof of Theorem 49 will provide a very instructive guide to how to do that.

(b) For those rather nobly trying, as we go along, to prove stated theorems before looking at the proofs, the results in this chapter do require a little more thought than what's gone before. Though a little exploration should reveal the only reasonable proof-strategies.

Proof for Theorem 49. Given an arbitrary corner $X \xrightarrow{f} Z \xleftarrow{g} Y$ we need to construct a pullback.

There is nothing to equalize yet. So our only option is try products. Because $\mathscr{C}$ has binary products, there will in particular be a product $X \times Y$ with the usual projections $\pi_1 \colon X \times Y \to X$ and $\pi_1 \colon X \times Y \to Y$. But that this immediately gives us parallel arrows $X \times Y \overset{f \circ \pi_1}{\underset{g \circ \pi_2}{\rightrightarrows}} Z$. Because $\mathscr{C}$ has equalizers, this parallel pair must have an equalizer $[E, e]$. So the square and its two internal triangles in the following diagram all commute:

$$\begin{tikzcd}
C \arrow[dr, dashed, "v"] \arrow[drr, bend left, "u"] \arrow[drrr, bend left, "c_2"] \arrow[ddr, bend right, "c_1"'] & & & \\
& E \arrow[rr, "\pi_2 \circ e"] \arrow[dr, "e"] \arrow[dd, "\pi_1 \circ e"'] & & Y \arrow[dd, "g"] \\
& & X \times Y \arrow[ur, "\pi_2"'] \arrow[dl, "\pi_1"] & \\
& X \arrow[rr, "f"] & & Z
\end{tikzcd}$$

Claim: the wedge formed by E with the projections $\pi_1 \circ e$, $\pi_2 \circ e$ is indeed a pullback of the original corner.

From this point, the argument is just fairly routine checking. Consider any other cone over the original corner, i.e. consider any wedge $X \xleftarrow{c_1} C \xrightarrow{c_2} Y$ with $fc_1 = gc_2$. We need to show that this factors uniquely through E.

Now, that wedge certainly uniquely factors through the product $X \times Y$, so there is a unique $u\colon C \to X \times Y$ such that $c_1 = \pi_1 \circ u$, $c_2 = \pi_2 \circ u$. Hence $f \circ \pi_1 \circ u = g \circ \pi_2 \circ u$. Therefore $C \xrightarrow{u} X \times Y \overset{f \circ \pi_1}{\underset{g \circ \pi_2}{\rightrightarrows}} Z$ is a fork, which must factor uniquely through the equalizer E via some v.

That is to say, there is a $v\colon C \to E$ such that $e \circ v = u$. Hence $\pi_1 \circ e \circ v = \pi_1 \circ u = c_1$. Similarly $\pi_2 \circ e \circ v = c_2$. Therefore the wedge with vertex C indeed factors through E, as we need.

To finish the proof, we have to establish the uniqueness of the mediating arrow v. Suppose $v'\colon C \to E$ also makes $\pi_1 \circ e \circ v' = c_1$, $\pi_2 \circ e \circ v' = c_2$. But $e \circ v'\colon C \to X \times Y$, and we know the wedge $X \xleftarrow{c_1} C \xrightarrow{c_2} Y$ factors uniquely through $X \times Y$, so $e \circ v' = u = e \circ v$. But equalizers are monic, so $v' = v$. □

(c) Next, to prove Theorem 50 on getting products from pullbacks, think first about a special case of $X \times_Z Y$. Then, to prove the theorem's claim about getting equalizers from products and pullbacks, try reconfiguring the parallel arrows you want to equalize as a wedge; take the product this wedge factors through, and use the fact that a certain corner involving this product has a pullback (the trick is to spot the right corner to use here).

Spelling things out, we have:

Proof for Theorem 50. Think about $X \times_1 Y$. In other words, note that since $\mathscr{C}$ has a terminal object, for any X, Y, there exists a corner $X \longrightarrow 1 \longleftarrow Y$. Applying the definition of a pullback, we find that a pullback for such a corner is indeed just the product $X \times Y$ with its usual projection arrows.

To show that $\mathscr{C}$ has equalizers, given that it has pullbacks and hence products, start by thinking of the parallel arrows we want to equalize, say $X \overset{f}{\underset{g}{\rightrightarrows}} Y$, as a wedge $Y \xleftarrow{f} X \xrightarrow{g} Y$. This wedge will factor uniquely via $\langle f, g\rangle$ through the product $Y \times Y$.

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. 30). This is nice to think about since (to arm-wave a bit!) the left arrow is evidently related to the parallel arrows we want to equalize, and the right arrow does some equalizing. Take this corner's pullback:

$$\begin{tikzcd} E \arrow[r, "q"] \arrow[d, "e"'] \arrow[dr, phantom, "\lrcorner", very near start] & Y \arrow[d, "\delta_Y"] \\ X \arrow[r, "\langle f,g\rangle"'] & Y \times Y \end{tikzcd}$$

Intuitively speaking, $E \xrightarrow{e} X \xrightarrow{\langle f,g\rangle} Y \times Y$ sends something in E to a pair of equals. So, morally, $[E, e]$ ought to be an equalizer for $X \underset{g}{\overset{f}{\rightrightarrows}} Y$. And, from this point on, it is a routine proof to check that it indeed is an equalizer. Here goes:

By the commutativity of the pullback square, $\delta_Y \circ q = \langle f, g\rangle \circ e$. Appealing to Theorems 24, 26 and 27, 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 $C \xrightarrow{c} X \underset{g}{\overset{f}{\rightrightarrows}} Y$. The wedge $X \xleftarrow{c} C \underset{g\circ c}{\overset{f\circ c}{\longrightarrow}} Y$ must factor through E (because E is the vertex of the pullback) via a unique mediating arrow v:

$$\begin{tikzcd}
C \arrow[dr, dashed, "v"] \arrow[drr, bend left, "f\circ c", "g\circ c"'] \arrow[ddr, bend right, "c"'] & & \\
& E \arrow[r, "q"] \arrow[d, "e"] & Y \arrow[d, "\delta_Y"] \\
& X \arrow[r, "d"] & Y \times Y
\end{tikzcd}$$

Rearranging part of this diagram, and unpacking again the composites $f \circ c, g \circ c$, this is just to say that there is a unique arrow $v \colon C \to E$ making this cut-down diagram commute:

$$\begin{tikzcd}
C \arrow[dd, dashed, "v"'] \arrow[dr, "c"] & & \\
& X \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & Y \\
E \arrow[ur, "e"'] & &
\end{tikzcd}$$

So $[E, e]$ is indeed an equalizer. $\square$

10.2 Set has all finite limits

Our target now is to show that if a category $\mathscr{C}$ has a terminal object and binary products (so has all finite products) and has equalizers, then it has a limit over *any* finite diagram D.

This is indeed our first Big Result in the book. To prove it, we are going to generalize the strategy pursued in proving the cut-down Theorem 49 that showed that having binary products and equalizers implies at least having pullbacks. So: we start with a product P of objects from D. We then find some appropriate parallel arrows out of this product P (here using arrows from D). Then we take an equalizer which will be a monic arrow from a subobject E of D, and use this subobject E as the vertex of the desired limit over D.

The devil, of course, is in the details! And in fact, you won't lose much if you skip past them. But assuming you want to understand the full proof, it might well help illuminate what's going on in the abstract general proof if we first consider how the proof works for a special category, Set. For here we can rely on a concrete understanding of finite products as sets of finite tuples, and of equalizers as subsets. So in this section, we will establish

Theorem 51. Set *has all finite limits.*

In this case, our proof suggestion comes to this. Take a product of the objects in a given diagram D, which we can take to be a set of tuples equipped with the trivial projection arrows which pick out components of the tuples. Then, we will directly find E, a subset of the product set, which can be used as the vertex of our limit cone, keeping only those projection arrows that commute with arrows in D. Working that idea through, we have:

Proof. We are given a finite diagram D in Set, whose objects are sets D_j for $j \in I$ (where the index set I is finite, so we might as well assume it to be the first n natural numbers for some n). There are perhaps functions as arrows between some or all of these D_j. The task is to construct a limit over D.

First take the finite product $[P, p_j]$ of all the sets D_j in the standard way. So we can take P to be the set of all n-tuples $\vec{x} = \langle x_j \rangle_{j \in I}$, where each tuple-component x_j is a member of the corresponding D_j. And then each function $p_j \colon P \to D_j$ will project out the j-th element of a tuple, so $p_j(\vec{x}) = x_j$.

In general, however, $[P, p_j]$ won't be a limit cone over D. For D may contain arrows between objects, and we've done nothing to ensure that the arrows p_j appropriately commute with the arrows in D in the way required for a cone.

Our announced strategy, then, is to look for a suitable subset of $E \subseteq P$. And if E is to be the vertex of a cone, we need to equip it with arrows $\pi_j \colon E \to D_j$. Since E is a set of n-tuples of elements of the D_j, the natural candidates will still simply project out the elements, so for $\vec{x} \in E$, $\pi_j(\vec{x}) = x_j$. In effect we just cut down each p_j from the domain P to the domain E. Or to put that more carefully, if $e \colon E \to P$ is the obvious injection map here, then $\pi_j = p_j \circ e$.

To get a cone, we need the relevant projection arrows to commute with any arrow in D. So if, in particular, we have $d \colon D_k \to D_l$ in D, then we require $\pi_l = d \circ \pi_k$; that is to say, for any tuple $\vec{x} \in E$, $d(x_k) = x_l$. And of course we require a similar condition for each arrow in the diagram D.

In summary so far, then, here's one way of getting a cone $[E, \pi_j]$ over D:

(1) Put E to be the set of tuples $\vec{x} = \langle x_j \rangle_{j \in I}$ such that $x_j \in D_j$ for all $j \in I$ and for every arrow $d \colon D_k \to D_l$ the corresponding condition $d(x_k) = x_l$ is satisfied.

(2) Put $\pi_j(\vec{x}) = x_j$ (i.e. put $\pi_j = p_j \circ e$, where $e\colon E \to P$ is the injection map).

Since we've constructed this cone by cutting down another limit in the most natural and economical way possible, it morally ought to be a limit cone.

Which it is! Suppose $[C, c_j]$ is any cone over D. Define the map $u\colon C \to E$ as sending $c \in C$ to the tuple $\langle c_j(c)\rangle_{j\in I}$. Then $c_j = \pi_j \circ u$ and so $[C, c_j]$ factors through $[E, \pi_j]$ via u. And u is unique. For if $[C, c_j]$ also factors through $[E, \pi_j]$ via u', then $\pi_j \circ u' = c_j$, therefore for any c the j-th component of $u'(c) = c_j(c)$, hence $u' = u$. $\square$

Note, since everything in that proof is finite it also establishes that FinSet also has all finite limits.

10.3 The existence of finite limits, more generally

The proof in the last section almost wrote itself once we had the key idea of first taking a product $[P, p_j]$ of the objects of D, and then (pretty much by brute force) extracting a subset of $E \subseteq P$ containing just those tuples which satisfy the condition necessary to get a cone with vertex E over D. However, our proof talked about tuples, subsets etc. in an old-school, non-categorial way. Let's see if we can rethink the proof, continuing to work for the moment in Set but now putting things in more purely categorial terms.

So again we take the product $[P, p_j]$ of the objects of D (we can do this since Set has all finite products). And in headline terms, we know what we need to do next – we must get the subset E by defining it as an equalizer. But an equalizer for what?

Take another look at the proof of Theorem 49. There we started with a corner diagram, i.e. with two arrows sharing a target, $f\colon X \to Z$, $g\colon Y \to Z$. We got parallel arrows which share a source as well as a target by taking a product, thereby getting $X \times Y \underset{g\circ\pi_2}{\overset{f\circ\pi_1}{\rightrightarrows}} Z$. And *then* we could look for an equalizer.

Now, in a diagram D there could be lots of arrows of the kind $d\colon D_k \to D_l$, and this time the arrows we need to deal with may well have different targets. But we still want to end up with a pair of parallel arrows with the same target if we are going to be able to take an equalizer. We'll evidently have to go from many targets to a single target by taking a product again. So let's define $[Q, q_l]$ as the product of all the objects D_l which are targets for arrows in D (again, we can do this since Set has all finite products).

The name of the game is now to define a pair of parallel arrows

$$P \underset{w}{\overset{v}{\rightrightarrows}} Q$$

which we are going to equalize by some $[E, e]$, and we aim to do this so that, in Set, E is the set of tuples which we met in the proof in the last section.

However, there are in fact only two naturally arising arrows from P to Q. Consider first the cone with vertex P and with an arrow $p_l\colon P \to D_l$ for each D_l which contributes to the product Q. This cone (by definition of the product $[Q, q_l]$) must factor through the product by a unique mediating arrow v, so that $p_l = q_l \circ v$ for each l.

Consider secondly the cone with vertex P and an arrow $d \circ p_k\colon P \to D_l$ for each arrow $d\colon D_k \to D_l$ in D. This cone too must factor through the product $[Q, q_l]$ by a unique mediating arrow w, so that $d \circ p_k = q_l \circ w$ for each arrow $d\colon D_k \to D_l$.

Since all parallel arrows have equalizers in Set, we can take the equalizer of v and w, namely $[E, e]$. Then E will be, as we want, the subset of tuples of objects from the D_j such that (for any arrow $d\colon D_k \to D_l$) $d(x_k) = x_l$.

And now the big claim: as before, $[E, p_j \circ e]$ will be a limit cone over D. Let's state this as a theorem:

Theorem 52. *Let D be a diagram in* Set. *Let $[P, p_j]$ be the product of the objects D_j in D, and $[Q, q_l]$ be the product of the objects D_l which are targets of arrows in D. Then there are arrows*

$$\begin{tikzcd} P \arrow[r, shift left, "v"] \arrow[r, shift right, "w"'] & Q \end{tikzcd}$$

such that the following diagrams commute for each $d\colon D_k \to D_l$:

$$\begin{tikzcd} P \arrow[r, "v"] \arrow[dr, "p_l"'] & Q \arrow[d, "q_l"] \\ & D_l \end{tikzcd} \qquad \begin{tikzcd} P \arrow[r, "w"] \arrow[d, "p_k"'] & Q \arrow[d, "q_l"] \\ D_k \arrow[r, "d"] & D_l \end{tikzcd}$$

Let the equalizer of v and w be $[E, e]$. Then $[E, p_j \circ e]$ will be a limit cone over D in Set.

Proof. We have already defined v and w so that the given diagrams commute. So next we confirm $[E, p_j \circ e]$ is a cone. Suppose there is an arrow $d\colon D_k \to D_l$. Then we require $d \circ p_k \circ e = p_l \circ e$.

But indeed $d \circ p_k \circ e = q_l \circ w \circ e = q_l \circ v \circ e = p_l \circ e$, where the inner equation holds because e is an equalizer of v and w and the outer equations are given by the commuting diagrams above.

Second we show that $[E, p_j \circ e]$ is a limit. So suppose $[C, c_j]$ is any other cone over D. Then there must be a unique $u\colon C \to P$ such that every c_j factors through the product and we have $c_j = p_j \circ u$.

Since $[C, c_j]$ is a cone, for any $d\colon D_k \to D_l$ in D we have $d \circ c_k = c_l$. Hence $d \circ p_k \circ u = p_l \circ u$, and hence for each q_l, $q_l \circ w \circ u = q_l \circ v \circ u$. But then we can apply the obvious generalized version of Theorem 25, and conclude that $w \circ u = v \circ u$. Which means that

$$\begin{tikzcd} C \arrow[r, "u"] & P \arrow[r, shift left, "p"] \arrow[r, shift right, "q"'] & Q \end{tikzcd}$$

is a fork, which must therefore uniquely factor through the equalizer $[E, e]$. That is to say, there is a unique $s\colon C \to E$ such that $u = e \circ s$, and hence for all j, $c_j = p_j \circ u = p_j \circ e \circ s$. That is to say, $[C, c_j]$ factors uniquely through $[E, p_j \circ e]$ via s. Therefore $[E, p_j \circ e]$ is indeed a limit cone. □

So that gives us the desired purely categorial proof, without mentioning subsets, of our result that Set has all finite limits. *But now note that we didn't depend on anything but the fact that* Set *has finite products and equalizers.* And since having a terminal object plus binary products sufficies for all finite products, we – at last! – have the originally promised sweeping result:

Theorem 47. If $\mathscr{C}$ has a terminal object, and has all binary products and equalizers, it is finitely complete.

Given ingredients from our previous discussions, since the categories in question have terminal objects, binary products and equalizers,

Theorem 53. *The categories of algebraic structured sets such as* Mon*,* Grp*,* Ab*,* Rng *are all finitely complete. Similarly* Top *is finitely complete.*

While e.g. a poset-as-a-category may lack many products and hence not be finitely complete.

10.4 Infinite limits

Now we extend our key Theorem 47 to reach beyond the finite case. First, we need:

Definition 55. The category $\mathscr{C}$ *has all (small) limits* if for any diagram D whose objects are D_j for indices $j \in I$, for some set I, then $\mathscr{C}$ has a limit over D. A category with all (small) limits is also said to be *complete*. ◁

We are no longer stipulating, then, that the collection of indices for an object in a diagram be finite. We only require now that it is no bigger than set-sized. So, to repeat a remark we've made before, 'small' doesn't mean small by any normal standards! – it just indicates that the relevant diagrams are not so mind-bogglingly enormous that they can't be indexed by a set.

Inspection of the proof in the last section shows that, so long as the diagram D is small (set-sized), then the argument will continue to go through if we are still dealing with a category like Set which has products for all set-sized collections of objects, or as we say, has all small products. Hence, without further ado, we get:

Theorem 54. *If $\mathscr{C}$ has all (small) products and has equalizers, then it has all small limits, i.e. is complete.*

We can similarly extend Theorem 53 to show that

Theorem 55. *The categories of structured sets* Mon*,* Grp*,* Ab*,* Rng *(among others) are all complete.* Top *too is complete.*

We have already met a category which, by contrast, is finitely complete but is evidently not complete, namely FinSet.

10.5 Dualizing again

Needless to say by this stage, our results in this chapter dualize in obvious ways. Thus we need not delay over the further explanations and proofs of

Theorem 56. *If $\mathscr{C}$ has initial objects, binary coproducts and co-equalizers, then it has all finite colimits, i.e. is finitely cocomplete. If $\mathscr{C}$ has all coproducts and has co-equalizers, then it has all (small) colimits, i.e. is cocomplete.*

Theorem 57. Set *is cocomplete – as are the categories of structured sets* Mon*,* Grp*,* Ab*,* Rng*.* Top *too is cocomplete.*

But note that a category can of course be (finitely) complete without being (finitely) cocomplete and vice versa. For a generic source of examples, take again a poset $(P, \preccurlyeq)$ considered as a category. This automatically has all equalizers (and coequalizers) – see §8.1 Ex. (5). But it will have other limits (colimits) depending on which products (coproducts) exists, i.e. which sets of elements have suprema (infima). For a simple case, take a poset with a maximum element and such that every pair of elements has a supremum: then considered as a category it has all finite limits (but maybe not infinite ones). But it need not have a minimal element and/or infima for all pairs of objects: hence it can lack some finite colimits despite having all finite limits.

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.

Awodey, S., 2006. *Category theory*, vol. 49 of *Oxford Logic Guides*. Oxford: Oxford University Press.

Goedecke, J., 2013. Category theory. URL https://www.dpmms.cam.ac.uk/~jg352/pdf/CategoryTheoryNotes.pdf.

Leinster, T., 2014. *Basic Category Theory*. Cambridge: Cambridge University Press.

Mazur, B., 2008. When is one think 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.

Sellars, W., 1963. Philosophy and the scientific image of man. In *Science, Perception and Reality*. Routledge & Kegan Paul.