2

ABSTRACT STRUCTURES

We begin with some remarks about category-theoretical definitions. These are characterizations of properties of objects and arrows in a category solely in terms of other objects and arrows, that is, just in the language of category theory. Such definitions may be said to be abstract, structural, operational, relational, or perhaps external (as opposed to internal). The idea is that objects and arrows are determined by the role they play in the category via their relations to other objects and arrows, that is, by their position in a structure and not by what they "are" or "are made of" in some absolute sense. The free monoid or category construction of the foregoing chapter was an example of one such definition, and we see many more examples of this kind later; for now, we start with some very simple ones. Let us call them *abstract characterizations*. We see that one of the basic ways of giving such an abstract characterization is via a Universal Mapping Property (UMP).

2.1 Epis and monos

Recall that in **Sets**, a function $f : A \to B$ is called

injective if $f(a) = f(a')$ implies $a = a'$ for all $a, a' \in A$,
surjective if for all $b \in B$ there is some $a \in A$ with $f(a) = b$.

We have the following abstract characterizations of these properties.

Definition 2.1. In any category **C**, an arrow

$$f : A \to B$$

is called

monomorphism, if given any $g, h : C \to A$, $fg = fh$ implies $g = h$,

$$C \underset{h}{\overset{g}{\rightrightarrows}} A \xrightarrow{f} B$$

epimorphism, if given any $i, j : B \to D$, $if = jf$ implies $i = j$,

$$A \xrightarrow{f} B \underset{j}{\overset{i}{\rightrightarrows}} D.$$

We often write $f : A \rightarrowtail B$ if f is a monomorphism and $f : A \twoheadrightarrow B$ if f is an epimorphism.

Proposition 2.2. *A function $f : A \to B$ between sets is monic just in case it is injective.*

Proof. Suppose $f : A \rightarrowtail B$. Let $a, a' \in A$ such that $a \neq a'$, and let $\{x\}$ be any given one-element set. Consider the functions

$$\bar{a}, \bar{a'} : \{x\} \to A$$

where

$$\bar{a}(x) = a, \qquad \bar{a'}(x) = a'.$$

Since $\bar{a} \neq \bar{a'}$, it follows, since f is a monomorphism, that $f\bar{a} \neq f\bar{a'}$. Thus, $f(a) = (f\bar{a})(x) \neq (f\bar{a'})(x) = f(a')$. Whence f is injective.

Conversely, if f is injective and $g, h : C \to A$ are functions such that $g \neq h$, then for some $c \in C$, $g(c) \neq h(c)$. Since f is injective, it follows that $f(g(c)) \neq f(h(c))$, whence $fg \neq fh$. □

Example 2.3. In many categories of "structured sets" like monoids, the monos are exactly the "injective homomorphisms." More precisely, a homomorphism $h : M \to N$ of monoids is monic just if the underlying function $|h| : |M| \to |N|$ is monic, that is, injective by the foregoing. To prove this, suppose h is monic and take two different "elements" $x, y : 1 \to |M|$, where $1 = \{*\}$ is any one-element set. By the UMP of the free monoid $M(1)$ there are distinct corresponding homomorphisms $\bar{x}, \bar{y} : M(1) \to M$, with distinct composites $h \circ \bar{x}, h \circ \bar{y} : M(1) \to M \to N$, since h is monic. Thus, the corresponding "elements" $hx, hy : 1 \to N$ of N are also distinct, again by the UMP of $M(1)$.

$$M(1) \underset{\bar{y}}{\overset{\bar{x}}{\rightrightarrows}} M \xrightarrow{\;h\;} N$$

$$1 \underset{y}{\overset{x}{\rightrightarrows}} |M| \xrightarrow{\;|h|\;} |N|$$

Conversely, if $|h| : |M| \to |N|$ is monic and $f, g : X \to M$ are any distinct homomorphisms, then $|f|, |g| : |X| \to |M|$ are distinct functions, and so $|h| \circ |f|, |h| \circ |g| : |X| \to |M| \to |N|$ are distinct, since $|h|$ is monic. Since therefore $|h \circ f| = |h| \circ |f| \neq |h| \circ |g| = |h \circ g|$, we also must have $h \circ f \neq h \circ g$.

A completely analogous situation holds, for example, for groups, rings, vector spaces, and posets. We shall see that this fact follows from the presence, in each of these categories, of certain objects like the free monoid $M(1)$.

Example 2.4. In a poset $\mathbf{P}$, every arrow $p \leq q$ is both monic and epic. Why?

Now, dually to the foregoing, the epis in **Sets** are exactly the surjective functions (exercise!); by contrast, however, in many other familiar categories they are not just the surjective homomorphisms, as the following example shows.

Example 2.5. In the category **Mon** of monoids and monoid homomorphisms, there is a monic homomorphism

$$\mathbb{N} \rightarrowtail \mathbb{Z}$$

where $\mathbb{N}$ is the additive monoid $(N, +, 0)$ of natural numbers and $\mathbb{Z}$ is the additive monoid $(Z, +, 0)$ of integers. We show that this map, given by the inclusion $N \subset Z$ of sets, is also epic in **Mon** by showing that the following holds:

Given any monoid homomorphisms $f, g : (\mathbb{Z}, +, 0) \to (M, *, u)$, if the restrictions to N are equal, $f\,|_N = g\,|_N$, then $f = g$.

Note first that

$$\begin{aligned} f(-n) &= f((-1)_1 + (-1)_2 + \cdots + (-1)_n) \\ &= f(-1)_1 * f(-1)_2 * \cdots * f(-1)_n \end{aligned}$$

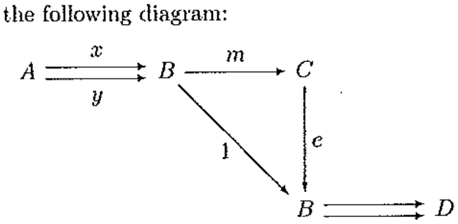
and similarly for g. It, therefore, suffices to show that $f(-1) = g(-1)$. But

$$\begin{aligned} f(-1) &= f(-1) * u \\ &= f(-1) * g(0) \\ &= f(-1) * g(1-1) \\ &= f(-1) * g(1) * g(-1) \\ &= f(-1) * f(1) * g(-1) \\ &= f(-1+1) * g(-1) \\ &= f(0) * g(-1) \\ &= u * g(-1) \\ &= g(-1). \end{aligned}$$

Note that, from an algebraic point of view, a morphism e is epic if and only if e cancels on the right: $xe = ye$ implies $x = y$. Dually, m is monic if and only if m cancels on the left: $mx = my$ implies $x = y$.

Proposition 2.6. *Every iso is both monic and epic.*

Proof. Consider the following diagram:



If m is an isomorphism with inverse e, then $mx = my$ implies $x = emx = emy = y$. Thus, m is monic. Similarly, e cancels on the right and thus is epic. □

In **Sets**, the converse of the foregoing also holds: every mono-epi is iso. But this is not in general true, as shown by the example in monoids above.

2.1.1 Sections and retractions

We have just noted that any iso is both monic and epic. More generally, if an arrow

$$f : A \to B$$

has a left inverse

$$g : B \to A, \quad gf = 1_A$$

then f must be monic and g epic, by the same argument.

Definition 2.7. A *split* mono (epi) is an arrow with a left (right) inverse. Given arrows $e : X \to A$ and $s : A \to X$ such that $es = 1_A$, the arrow s is called a *section* or *splitting* of e, and the arrow e is called a *retraction* of s. The object A is called a *retract* of X.

Since functors preserve identities, they also preserve *split* epis and *split* monos. Compare example 2.5 above in **Mon** where the forgetful functor

$$\mathbf{Mon} \to \mathbf{Set}$$

did not preserve the epi $\mathbb{N} \to \mathbb{Z}$.

Example 2.8. In **Sets**, every mono splits except those of the form

$$\emptyset \rightarrowtail A.$$

The condition that *every epi splits* is the categorical version of the axiom of choice. Indeed, consider an epi

$$e : E \twoheadrightarrow X.$$

We have the family of nonempty sets:

$$E_x = e^{-1}\{x\}, \quad x \in X.$$

A choice function for this family $(E_x)_{x\in X}$ is exactly a splitting of e, that is, a function $s : X \to E$ such that $es = 1_X$, since this means that $s(x) \in E_x$ for all $x \in X$.

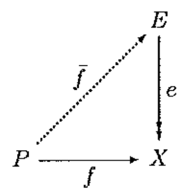
Conversely, given a family of nonempty sets,

$$(E_x)_{x\in X}$$

take $E = \{(x, y) \mid x \in X,\ y \in E_x\}$ and define the epi $e : E \twoheadrightarrow X$ by $(x, y) \mapsto x$. A splitting s of e then determines a choice function for the family.

The idea that a "family of objects" $(E_x)_{x \in X}$ can be represented by a single arrow $e : E \to X$ by using the "fibers" $e^{-1}(x)$ has much wider application than this, and is considered further in Section 7.10.

A notion related to the existence of "choice functions" is that of being "projective": an object P is said to be *projective* if for any epi $e : E \twoheadrightarrow X$ and arrow $f : P \to X$ there is some (not necessarily unique) arrow $\bar{f} : P \to E$ such that $e \circ \bar{f} = f$, as indicated in the following diagram:



One says that f *lifts across* e. Any epi into a projective object clearly splits. Projective objects may be thought of as having a more "free" structure, thus permitting "more arrows."

The axiom of choice implies that all sets are projective, and it follows that free objects in many (but not all!) categories of algebras then are also projective. The reader should show that, in any category, any retract of a projective object is also projective.

2.2 Initial and terminal objects

We now consider abstract characterizations of the empty set and the one-element sets in the category **Sets** and structurally similar objects in general categories.

Definition 2.9. In any category **C**, an object

0 is *initial* if for any object C there is a unique morphism

$$0 \to C,$$

1 is *terminal* if for any object C there is a unique morphism

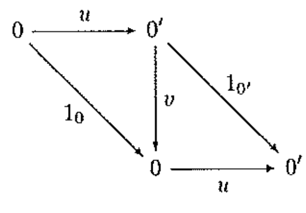
$$C \to 1.$$

As in the case of monos and epis, note that there is a kind of "duality" in these definitions. Precisely, a terminal object in **C** is exactly an initial object in $\mathbf{C}^{\mathrm{op}}$. We consider this duality systematically in Chapter 3.

First, observe that since the notions of initial and terminal object are simple UMPs, such objects are uniquely determined up to isomorphism, just like the free monoids were.

Proposition 2.10. *Initial (terminal) objects are unique up to isomorphism.*

Proof. In fact, if C and C' are both initial (terminal) in the same category, then there is a *unique* isomorphism $C \to C'$. Indeed, suppose that 0 and $0'$ are both initial objects in some category $\mathbf{C}$; the following diagram then makes it clear that 0 and $0'$ are uniquely isomorphic:



For terminal objects, apply the foregoing to $\mathbf{C}^{\mathrm{op}}$. □

Example 2.11.

1. In **Sets**, the empty set is initial and any singleton set $\{x\}$ is terminal. Observe that **Sets** has just one initial object but many terminal objects (answering the question of whether $\mathbf{Sets} \cong \mathbf{Sets}^{\mathrm{op}}$).
2. In **Cat**, the category $\mathbf{0}$ (no objects and no arrows) is initial and the category $\mathbf{1}$ (one object and its identity arrow) is terminal.
3. In **Groups**, the one-element group is *both* initial and terminal (similarly for the category of vector spaces and linear transformations, as well as the category of monoids and monoid homomorphisms). But in **Rings** (commutative with unit), the ring $\mathbb{Z}$ of integers is initial (the one-element ring with $0 = 1$ is terminal).
4. A *Boolean algebra* is a poset B equipped with distinguished elements $0, 1$, binary operations $a \vee b$ of "join" and $a \wedge b$ of "meet," and a unary operation $\neg b$ of "complementation." These are required to satisfy the conditions

$$\begin{aligned} & 0 \le a \\ & a \le 1 \\ a \le c \quad \text{and} \quad b \le c \quad & \text{iff} \quad a \vee b \le c \\ c \le a \quad \text{and} \quad c \le b \quad & \text{iff} \quad c \le a \wedge b \\ a \le \neg b \quad & \text{iff} \quad a \wedge b = 0 \\ & \neg\neg a = a. \end{aligned}$$

   There is also an equivalent, fully equational characterization not involving the ordering. A typical example of a Boolean algebra is the powerset $\mathcal{P}(X)$ of all subsets $A \subseteq X$ of a set X, ordered by inclusion $A \subseteq B$, and with the Boolean operations being the empty set $0 = \emptyset$, the whole set $1 = X$, union and intersection of subsets as join and meet, and the relative complement $X - A$ as $\neg A$. A familiar special case is the two-element

Boolean algebra $\mathbf{2} = \{0, 1\}$ (which may be taken to be the powerset $\mathcal{P}(1)$), sometimes also regarded as "truth values" with the logical operations of disjunction, conjunction, and negation as the Boolean operations. It is an initial object in the category **BA** of Boolean algebras. **BA** has as arrows the Boolean homomorphisms, that is, functors $h : B \to B'$ that preserve the additional structure, in the sense that $h(0) = 0$, $h(a \vee b) = h(a) \vee h(b)$, etc. The one-element Boolean algebra (i.e., $\mathcal{P}(0)$) is terminal.

5. In a poset, an object is plainly initial iff it is the least element, and terminal iff it is the greatest element. Thus, for instance, any Boolean algebra has both. Obviously, a category *need not* have either an initial object or a terminal object; for example, the poset $(\mathbb{Z}, \leq)$ has neither.
6. For any category $\mathbf{C}$ and any object $X \in \mathbf{C}$, the identity arrow $1_X : X \to X$ is a terminal object in the slice category $\mathbf{C}/X$ and an initial object in the coslice category $X/\mathbf{C}$.

2.3 Generalized elements

Let us consider arrows into and out of initial and terminal objects. Clearly only certain of these will be of interest, but those are often especially significant.

A set A has an arrow into the initial object $A \to 0$ just if it is itself initial, and the same is true for posets. In monoids and groups, by contrast, every object has a unique arrow to the initial object, since it is also terminal.

In the category **BA** of Boolean algebras, however, the situation is quite different. The maps $p : B \to \mathbf{2}$ into the initial Boolean algebra $\mathbf{2}$ correspond uniquely to the so-called *ultrafilters* U in B. A *filter* in a Boolean algebra B is a nonempty subset $F \subseteq B$ that is closed upward and under meets:

$$a \in F \text{ and } a \leq b \quad \text{implies} \quad b \in F$$
$$a \in F \text{ and } b \in F \quad \text{implies} \quad a \wedge b \in F$$

A filter F is *maximal* if the only strictly larger filter $F \subset F'$ is the "improper" filter, namely all of B. An *ultrafilter* is a maximal filter. It is not hard to see that a filter F is an ultrafilter just if for every element $b \in B$, either $b \in F$ or $\neg b \in F$, and not both (exercise!). Now if $p : B \to \mathbf{2}$, let $U_p = p^{-1}(1)$ to get an ultrafilter $U_p \subset B$. And given an ultrafilter $U \subset B$, define $p_U(b) = 1$ iff $b \in U$ to get a Boolean homomorphism $p_U : B \to \mathbf{2}$. This is easy to check, as is the fact that these operations are mutually inverse. Boolean homomorphisms $B \to \mathbf{2}$ are also used in forming the "truth tables" one meets in logic. Indeed, a row of a truth table corresponds to such a homomorphism on a Boolean algebra of formulas.

Ring homomorphisms $A \to \mathbb{Z}$ into the initial ring $\mathbb{Z}$ play an analogous and equally important role in algebraic geometry. They correspond to so-called *prime ideals*, which are the ring-theoretic generalizations of ultrafilters.

Now let us consider some arrows from terminal objects. For any set X, for instance, we have an isomorphism

$$X \cong \mathrm{Hom}_{\mathbf{Sets}}(1, X)$$

between elements $x \in X$ and arrows $\bar{x} : 1 \to X$, determined by $\bar{x}(*) = x$, from a terminal object $1 = \{*\}$. We have already used this correspondence several times. A similar situation holds in posets (and in topological spaces), where the arrows $1 \to P$ correspond to elements of the underlying set of a poset (or space) P. In any category with a terminal object 1, such arrows $1 \to A$ are often called *global elements*, or *points*, or *constants* of A. In sets, posets, and spaces, the general arrows $A \to B$ are determined by what they do to the points of A, in the sense that two arrows $f, g : A \to B$ are equal if for every point $a : 1 \to A$ the composites are equal, $fa = ga$.

But be careful; this is not always the case! How many points are there for an object M in the category of monoids? That is, how many arrows of the form $1 \to M$ for a given monoid M? Just one! And how many points does a Boolean algebra have?

Because, in general, an object is not determined by its points, it is convenient to introduce the device of *generalized elements*. These are arbitrary arrows,

$$x : X \to A$$

(with arbitrary domain X), which can be regarded as *generalized* or *variable elements* of A. Computer scientists and logicians sometimes think of arrows $1 \to A$ as constants or closed terms and general arrows $X \to A$ as arbitrary terms. Summarizing:

Example 2.12.

1. Consider arrows $f, g : P \to Q$ in **Pos**. Then $f = g$ iff for all $x : 1 \to P$, we have $fx = gx$. In this sense, posets "have enough points" to separate the arrows.
2. By contrast, in **Mon**, for homomorphisms $h, j : M \to N$, we always have $hx = jx$, for all $x : 1 \to M$, since there is just one such point x. Thus, monoids do not "have enough points."
3. But in any category **C**, and for any arrows $f, g : C \to D$, we always have $f = g$ iff for all $x : X \to C$, it holds that $fx = gx$ (why?). Thus, all objects have enough generalized elements.
4. In fact, it often happens that it is enough to consider generalized elements of just a certain form $T \to A$, that is, for certain "test" objects T. We shall consider this presently.

Generalized elements are also good for "testing" for various conditions. Consider, for instance, diagrams of the following shape:

$$X \underset{x'}{\overset{x}{\rightrightarrows}} A \xrightarrow{f} B$$

The arrow f is monic iff $x \neq x'$ implies $fx \neq fx'$ for all x, x', that is, just if f is "injective on generalized elements."

Similarly, in any category $\mathbf{C}$, to test whether a square commutes,

$$\begin{array}{ccc} A & \xrightarrow{f} & B \\ {\scriptstyle g}\downarrow & & \downarrow{\scriptstyle \alpha} \\ D & \xrightarrow[\beta]{} & C \end{array}$$

we shall have $\alpha f = \beta g$ just if $\alpha f x = \beta g x$ for all generalized elements $x : X \to A$ (just take $x = 1_A : A \to A$).

Example 2.13. Generalized elements can be used to "reveal more structure" than do the constant elements. For example, consider the following posets X and A:

$$X = \{x \leq y, x \leq z\}$$
$$A = \{a \leq b \leq c\}$$

There is an order-preserving bijection $f : X \to A$ defined by

$$f(x) = a, \qquad f(y) = b, \qquad f(z) = c.$$

It is easy to see that f is both monic and epic in the category $\mathbf{Pos}$; however, it is clearly not an iso. One would like to say that X and A are "different structures," and indeed, their being nonisomorphic says just this. But now, how to *prove* that they are *not* isomorphic (say, via some other $X \to A$)? In general, this sort of thing can be quite difficult.

One way to prove that two objects are not isomorphic is to use "invariants": attributes that are preserved by isomorphisms. If two objects differ by an invariant they cannot be isomorphic. Generalized elements provide an easy way to define invariants. For instance, the number of global elements of X and A is the same, namely the three elements of the sets. But consider instead the "2-elements" $\mathbf{2} \to X$, from the poset $\mathbf{2} = \{0 \leq 1\}$ as a "test-object." Then X has 5 such elements, and A has 6. Since these numbers are invariants, the posets cannot be isomorphic. In more detail, we can define for any poset P the numerical invariant

$$|\mathrm{Hom}(\mathbf{2}, P)| \quad = \quad \text{the number of elements of } \mathrm{Hom}(\mathbf{2}, P).$$

Then if $P \cong Q$, it is easy to see that $|\mathrm{Hom}(\mathbf{2}, P)| = |\mathrm{Hom}(\mathbf{2}, Q)|$, since any isomorphism

$$P \underset{j}{\overset{i}{\rightleftarrows}} Q$$

also gives an iso

$$\mathrm{Hom}(\mathbf{2}, P) \underset{j_*}{\overset{i_*}{\rightleftarrows}} \mathrm{Hom}(\mathbf{2}, Q)$$

defined by composition:

$$i_*(f) = if$$
$$j_*(g) = jg$$

for all $f : \mathbf{2} \to P$ and $g : \mathbf{2} \to Q$. Indeed, this is a special case of the very general fact that $\mathrm{Hom}(X, -)$ is always a functor, and functors always preserve isos.

Example 2.14. As in the foregoing example, it is often the case that generalized elements $t : T \to A$ "based at" certain objects T are especially "revealing." We can think of such elements geometrically as "figures of shape T in A," just as an arrow $\mathbf{2} \to P$ in posets is a figure of shape $p \le p'$ in P. For instance, as we have already seen, in the category of monoids, the arrows from the terminal monoid are entirely uninformative, but those from the free monoid on one generator $M(1)$ suffice to distinguish homomorphisms, in the sense that two homomorphisms $f, g : M \to M'$ are equal if their composites with all such arrows are equal. Since we know that $M(1) = \mathbb{N}$, the monoid of natural numbers, we can think of generalized elements $M(1) \to M$ based at $M(1)$ as "figures of shape $\mathbb{N}$" in M. In fact, by the UMP of $M(1)$, the underlying set $|M|$ is therefore (isomorphic to) the collection $\mathrm{Hom}_{\mathbf{Mon}}(\mathbb{N}, M)$ of all such figures, since

$$|M| \cong \mathrm{Hom}_{\mathbf{Sets}}(1, |M|) \cong \mathrm{Hom}_{\mathbf{Mon}}(\mathbb{N}, M).$$

In this sense, a map from a monoid is determined by its effect on all of the figures of shape $\mathbb{N}$ in the monoid.

2.4 Products

Next, we are going to see the categorical definition of a product of two objects in a category. This was first given by Mac Lane in 1950, and it is probably the earliest example of category theory being used to define a fundamental mathematical notion.

By "define" here I mean an abstract characterization, in the sense already used, in terms of objects and arrows in a category. And as before, we do this by giving a UMP, which determines the structure at issue up to isomorphism, as usual in category theory. Later in this chapter, we have several other examples of such characterizations.

Let us begin by considering products of sets. Given sets A and B, the *cartesian product* of A and B is the set of ordered pairs

$$A \times B = \{(a, b) \mid a \in A,\ b \in B\}.$$

Observe that there are two "coordinate projections"

$$A \xleftarrow{\pi_1} A \times B \xrightarrow{\pi_2} B$$

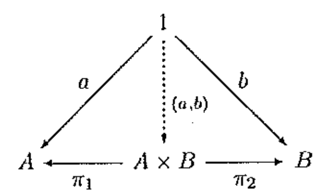
with

$$\pi_1(a, b) = a, \qquad \pi_2(a, b) = b.$$

And indeed, given any element $c \in A \times B$, we have

$$c = (\pi_1 c, \pi_2 c).$$

The situation is captured concisely in the following diagram:



Replacing elements by generalized elements, we get the following definition.

Definition 2.15. In any category $\mathbf{C}$, a *product diagram* for the objects A and B consists of an object P and arrows

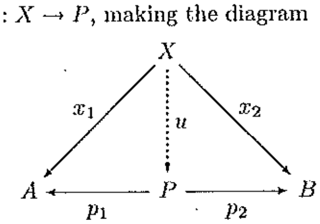
$$A \xleftarrow{p_1} P \xrightarrow{p_2} B$$

satisfying the following UMP:

Given any diagram of the form

$$A \xleftarrow{x_1} X \xrightarrow{x_2} B$$

there exists a unique $u : X \to P$, making the diagram



commute, that is, such that $x_1 = p_1 u$ and $x_2 = p_2 u$.

Remark 2.16. As in other UMPs, there are two parts:

> *Existence:* There is some $u : X \to U$ such that $x_1 = p_1 u$ and $x_2 = p_2 u$.
> *Uniqueness:* Given any $v : X \to U$, if $p_1 v = x_1$ and $p_2 v = x_2$, then $v = u$.

Proposition 2.17. *Products are unique up to isomorphism.*

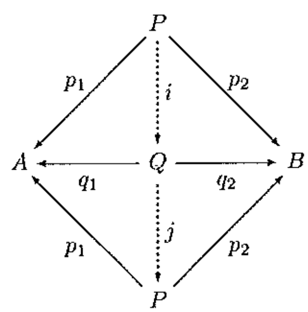
Proof. Suppose

$$A \xleftarrow{p_1} P \xrightarrow{p_2} B$$

and

$$A \xleftarrow{q_1} Q \xrightarrow{q_2} B$$

are products of A and B. Then, since Q is a product, there is a unique $i : P \to Q$ such that $q_1 \circ i = p_1$ and $q_2 \circ i = p_2$. Similarly, since P is a product, there is a unique $j : Q \to P$ such that $p_1 \circ j = q_1$ and $p_2 \circ j = q_2$.



Composing, $p_1 \circ j \circ i = p_1$ and $p_2 \circ j \circ i = p_2$. Since also $p_1 \circ 1_P = p_1$ and $p_2 \circ 1_P = p_2$, it follows from the uniqueness condition that $j \circ i = 1_P$. Similarly, we can show $i \circ j = 1_Q$. Thus, $i : P \to Q$ is an isomorphism. □

If A and B have a product, we write

$$A \xleftarrow{p_1} A \times B \xrightarrow{p_2} B$$

for one such product. Then given X, x_1, x_2 as in the definition, we write

$$\langle x_1, x_2 \rangle \ \text{for} \ u : X \to A \times B.$$

Note, however, that a pair of objects may have many different products in a category. For example, given a product $A \times B, p_1, p_2$, and any iso $h : A \times B \to Q$, the diagram $Q, p_1 \circ h, p_2 \circ h$ is also a product of A and B.

Now an arrow *into* a product

$$f : X \to A \times B$$

is "the same thing" as a pair of arrows

$$f_1 : X \to A, \qquad f_2 : X \to B.$$

So we can essentially forget about such arrows, in that they are uniquely determined by pairs of arrows. But something useful *is* gained if a category has products; namely, consider arrows *out of* the product,

$$g : A \times B \to Y.$$

Such a g is a "function in two variables"; given any two generalized elements $f_1 : X \to A$ and $f_2 : X \to B$, we have an element $g\langle f_1, f_2\rangle : X \to Y$. Such arrows $g : A \times B \to Y$ are not "reducible" to anything more basic, the way arrows into products were (to be sure, they are related to the notion of an "exponential" Y^B, via "currying" $\lambda f : A \to Y^B$; we discuss this further in Chapter 6).

2.5 Examples of products

1. We have already seen cartesian products of sets. Note that if we choose a different definition of ordered pairs $\langle a, b\rangle$, we get different sets

   $$A \times B \quad \text{and} \quad A \times' B$$

   each of which is (part of) a product, and so are isomorphic. For instance, we could set

   $$\langle a, b\rangle = \{\{a\}, \{a, b\}\}$$
   $$\langle a, b\rangle' = \langle a, \langle a, b\rangle\rangle.$$

2. Products of "structured sets" like monoids or groups can often be constructed as products of the underlying sets with *componentwise* operations: If G and H are groups, for instance, $G \times H$ can be constructed by taking the underlying set of $G \times H$ to be the set $\{\langle g, h\rangle \mid g \in G,\ h \in H\}$ and defining the binary operation by

   $$\langle g, h\rangle \cdot \langle g', h'\rangle = \langle g \cdot g', h \cdot h'\rangle$$

   the unit by

   $$u = \langle u_G, u_H\rangle$$

   and inverses by

   $$\langle a, b\rangle^{-1} = \langle a^{-1}, b^{-1}\rangle.$$

   The projection homomorphisms $G \times H \to G$ (or H) are the evident ones $\langle g, h\rangle \mapsto g$ (or h).

3. Similarly, for categories $\mathbf{C}$ and $\mathbf{D}$, we already defined the category of pairs of objects and arrows,

$$\mathbf{C} \times \mathbf{D}.$$

Together with the evident projection functors, this is indeed a product in $\mathbf{Cat}$ (when $\mathbf{C}$ and $\mathbf{D}$ are small). (Check this: verify the UMP for the product category so defined.)

As special cases, we also get products of posets and of monoids as products of categories. (Check this: the projections and unique paired function are always monotone and so the product of posets, constructed in $\mathbf{Cat}$, is also a product in $\mathbf{Pos}$, and similarly for $\mathbf{Mon}$.)

4. Let P be a poset and consider a product of elements $p, q \in P$. We must have projections

$$p \times q \leq p$$
$$p \times q \leq q$$

and if for any element x,

$$x \leq p, \quad \text{and} \quad x \leq q$$

then we need

$$x \leq p \times q.$$

Do you recognize this operation $p \times q$? It is just what is usually called the *greatest lower bound*: $p \times q = p \wedge q$. Many other order-theoretic notions are also special cases of categorical ones, as we shall see later.

5. (For those who know something about Topology.) Let us show that the product of two *topological spaces* X, Y, as usually defined, really is a product in $\mathbf{Top}$, the category of spaces and continuous functions. Thus, suppose we have spaces X and Y and the product spaces $X \times Y$ with its projections

$$X \xleftarrow{p_1} X \times Y \xrightarrow{p_2} Y.$$

Recall that $O(X \times Y)$ is generated by basic open sets of the form $U \times V$ where $U \in O(X)$ and $V \in O(Y)$, so every $W \in O(X \times Y)$ is a union of such basic opens.

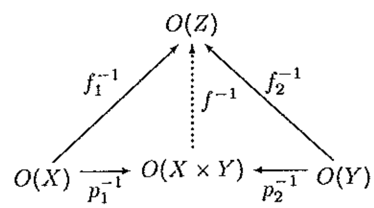
- Clearly p_1 is continuous, since $p_1^{-1}U = U \times Y$.
- Given any continuous $f_1 : Z \to X, f_2 : Z \to Y$, let $f : Z \to X \times Y$ be the function $f = \langle f_1, f_2 \rangle$. We just need to see that f is continuous.

- Given any $W = \bigcup_i (U_i \times V_i) \in O(X \times Y)$, $f^{-1}(W) = \bigcup_i f^{-1}(U_i \times V_i)$, so it suffices to show $f^{-1}(U \times V)$ is open. But

$$\begin{aligned} f^{-1}(U \times V) &= f^{-1}((U \times Y) \cap (X \times V)) \\ &= f^{-1}(U \times Y) \cap f^{-1}(X \times V) \\ &= f^{-1} \circ p_1^{-1}(U) \cap f^{-1} \circ p_2^{-1}(V) \\ &= (f_1)^{-1}(U) \cap (f_2)^{-1}(V) \end{aligned}$$

where $(f_1)^{-1}(U)$ and $(f_2)^{-1}(V)$ are open, since f_1 and f_2 are continuous.

The following diagram concisely captures the situation at hand:



6. (For those familiar with type theory.) Let us consider the *category of types* of the (simply typed) λ-calculus. The λ-calculus is a formalism for the specification and manipulation of functions, based on the notions of "binding of variables" and functional evaluation. For example, given the real polynomial expression $x^2 + 2y$, in the λ-calculus one writes $\lambda y.x^2 + 2y$ for the function $y \mapsto x^2 + 2y$ (for each fixed value x), and $\lambda x \lambda y.x^2 + 2y$ for the function-valued function $x \mapsto (y \mapsto x^2 + 2y)$.

   Formally, the λ-calculus consists of

   - Types: $A \times B,\ A \to B, \ldots$ (generated from some basic types)
   - Terms:

$$\begin{array}{rl} x, y, z, \ldots : A & \text{(variables for each type } A) \\ a : A,\ b : B, \ldots & \text{(possibly some typed constants)} \\ \langle a, b\rangle : A \times B & (a : A,\ b : B) \\ \mathrm{fst}(c) : A & (c : A \times B) \\ \mathrm{snd}(c) : B & (c : A \times B) \\ ca : B & (c : A \to B,\ a : A) \\ \lambda x.b : A \to B & (x : A,\ b : B) \end{array}$$

- Equations:

$$\begin{aligned}
\mathrm{fst}(\langle a, b\rangle) &= a \\
\mathrm{snd}(\langle a, b\rangle) &= b \\
\langle \mathrm{fst}(c), \mathrm{snd}(c)\rangle &= c \\
(\lambda x.b)a &= b[a/x] \\
\lambda x.cx &= c \quad (\text{no } x \text{ in } c)
\end{aligned}$$

The relation $a \sim b$ (usually called *$\beta\eta$-equivalence*) on terms is defined to be the equivalence relation generated by the equations, and renaming of bound variables:

$$\lambda x.b = \lambda y.b[y/x] \quad (\text{no } y \text{ in } b)$$

The category of types $\mathbf{C}(\lambda)$ is now defined as follows:

- Objects: the types,
- Arrows $A \to B$: closed terms $c : A \to B$, identified if $c \sim c'$,
- Identities: $1_A = \lambda x.x$ (where $x : A$),
- Composition: $c \circ b = \lambda x.c(bx)$.

Let us verify that this is a well-defined category:
Unit laws:

$$\begin{aligned}
c \circ 1_B &= \lambda x(c((\lambda y.y)x)) = \lambda x(cx) = c \\
1_C \circ c &= \lambda x((\lambda y.y)(cx)) = \lambda x(cx) = c
\end{aligned}$$

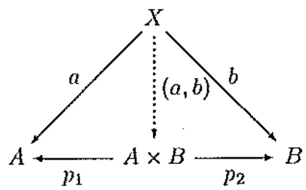
Associativity:

$$\begin{aligned}
c \circ (b \circ a) &= \lambda x(c((b \circ a)x)) \\
&= \lambda x(c((\lambda y.b(ay))x)) \\
&= \lambda x(c(b(ax))) \\
&= \lambda x(\lambda y(c(by))(ax)) \\
&= \lambda x((c \circ b)(ax)) \\
&= (c \circ b) \circ a
\end{aligned}$$

This category has binary products. Indeed, given types A and B, let

$$p_1 = \lambda z.\mathrm{fst}(z), \quad p_2 = \lambda z.\mathrm{snd}(z) \quad (z : A \times B).$$

And given a and b as in



let

$$(a, b) = \lambda x.\langle ax, bx\rangle.$$

Then

$$\begin{aligned} p_1 \circ (a, b) &= \lambda x(p_1((\lambda y.\langle ay, by\rangle)x)) \\ &= \lambda x(p_1\langle ax, bx\rangle) \\ &= \lambda x(ax) \\ &= a. \end{aligned}$$

Similarly, $p_2 \circ (a, b) = b$.

Finally, if $c : X \to A \times B$ also has

$$p_1 \circ c = a, \qquad p_2 \circ c = b$$

then

$$\begin{aligned} (a, b) &= \lambda x.\langle ax, bx\rangle \\ &= \lambda x.\langle (p_1 \circ c)x, (p_2 \circ c)x\rangle \\ &= \lambda x.\langle (\lambda y(p_1(cy)))x, (\lambda y(p_2(cy)))x\rangle \\ &= \lambda x.\langle (\lambda y((\lambda z.\mathrm{fst}(z))(cy)))x, (\lambda y((\lambda z.\mathrm{snd}(z))(cy)))x\rangle \\ &= \lambda x.\langle \lambda y(\mathrm{fst}(cy))x, \lambda y(\mathrm{snd}(cy))x\rangle \\ &= \lambda x.\langle \mathrm{fst}(cx), \mathrm{snd}(cx)\rangle \\ &= \lambda x.(cx) \\ &= c. \end{aligned}$$

Remark 2.18. The λ-calculus had another surprising interpretation, namely as a system of notation for proofs in propositional calculus; this is known as the "Curry–Howard" correspondence. Briefly, the idea is that one interprets types as propositions (with $A \times B$ being conjunction and $A \to B$ implication) and terms $a : A$ as proofs of the proposition A. The term-forming rules such as

$$\frac{a : A \qquad b : B}{\langle a, b\rangle : A \times B}$$

can then be read as annotated rules of inference, showing how to build up labels for proofs inductively. So, for instance, a natural deduction proof such as

$$\dfrac{\dfrac{\dfrac{[A] \qquad [B]}{A \times B}}{B \to (A \times B)}}{A \to (B \to (A \times B))}$$

with square brackets indicating cancellation of premisses, is labeled as follows:

$$\dfrac{\dfrac{\dfrac{[x : A] \qquad [y : B]}{\langle x, y \rangle : A \times B}}{\lambda y.\langle x, y \rangle : B \to (A \times B)}}{\lambda x \lambda y.\langle x, y \rangle : A \to (B \to (A \times B))}$$

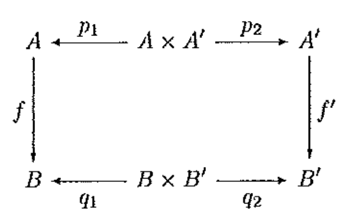
The final "proof term" $\lambda x \lambda y.\langle x, y \rangle$ thus records the given proof of the "proposition" $A \to (B \to (A \times B))$, and a different proof of the same proposition would give a different term.

Although one often speaks of a resulting "isomorphism" between logic and type theory, what we in fact have here is simply a functor from the category of proofs in the propositional calculus with conjunction and implication (as defined in example 10.), into the category of types of the λ-calculus. The functor will not generally be an isomorphism unless we impose some further equations between proofs.

Au: Please specify "example 10."

2.6 Categories with products

Let $\mathbf{C}$ be a category that has a product diagram for every pair of objects. Suppose we have objects and arrows



with indicated products. Then, we write

$$f \times f' : A \times A' \to B \times B'$$

for $f \times f' = \langle f \circ p_1, f' \circ p_2 \rangle$. Thus, both squares in the following diagram commute:

$$\begin{array}{ccccc} A & \xleftarrow{p_1} & A \times A' & \xrightarrow{p_2} & A' \\ {\scriptstyle f}\downarrow & & \vdots\, {\scriptstyle f \times f'} & & \downarrow{\scriptstyle f'} \\ B & \xleftarrow[q_1]{} & B \times B' & \xrightarrow[q_2]{} & B' \end{array}$$

In this way, if we choose a product for each pair of objects, we get a functor

$$\times : \mathbf{C} \times \mathbf{C} \to \mathbf{C}$$

as the reader can easily check, using the UMP of the product. A category which has a product for every pair of objects is said to *have binary products.*

We can also define ternary products

$$A_1 \times A_2 \times A_3$$

with an analogous UMP (there are three projections $p_i : A_1 \times A_2 \times A_3 \to A_i$, and for any object X and three arrows $x_i : X \to A_i$, there is a unique arrow $u : X \to A_1 \times A_2 \times A_3$ such that $p_i u = x_i$ for each of the three i's). Plainly, such a condition can be formulated for any number of factors.

It is clear, however, that if a category has binary products, then it has all finite products with two or more factors; for instance, one could set

$$A \times B \times C = (A \times B) \times C$$

to satisfy the UMP for ternary products. On the other hand, one could instead have taken $A \times (B \times C)$ just as well. This shows that the binary product operation $A \times B$ is associative up to isomorphism, for we must have

$$(A \times B) \times C \cong A \times (B \times C)$$

by the UMP of ternary products.

Observe also that a terminal object is a "nullary" product, that is, a product of no objects:

> Given no objects, there is an object 1 with no maps, and given any other object X and no maps, there is a unique arrow
>
> $$! : X \to 1$$
>
> making nothing further commute.

Similarly, any object A is the *unary product* of A with itself one time.

Finally, one can also define the product of a family of objects $(C_i)_{i \in I}$ indexed by *any* set I, by giving a UMP for "I-ary products" analogous to those for nullary, unary, binary, and n-ary products. We leave the precise formulation of this UMP as an exercise.

Definition 2.19. A category $\mathbf{C}$ is said to *have all finite products*, if it has a terminal object and all binary products (and therewith products of any finite cardinality). The category $\mathbf{C}$ *has all (small) products* if every set of objects in $\mathbf{C}$ has a product.

2.7 Hom-sets

In this section, we assume that all categories are locally small.

Recall that in any category $\mathbf{C}$, given any objects A and B, we write

$$\text{Hom}(A, B) = \{f \in \mathbf{C} \mid f : A \to B\}$$

and call such a set of arrows a *Hom-set.* Note that any arrow $g : B \to B'$ in $\mathbf{C}$ induces a function

$$\text{Hom}(A, g) : \text{Hom}(A, B) \to \text{Hom}(A, B')$$
$$(f : A \to B) \mapsto (g \circ f : A \to B \to B')$$

Thus, $\text{Hom}(A, g) = g \circ f$; one sometimes writes g_* instead of $\text{Hom}(A, g)$, so

$$g_*(f) = g \circ f.$$

Let us show that this determines a functor

$$\text{Hom}(A, -) : \mathbf{C} \to \mathbf{Sets},$$

called the (covariant) *representable functor* of A. We need to show that

$$\text{Hom}(A, 1_X) = 1_{\text{Hom}(A,X)}$$

and that

$$\text{Hom}(A, g \circ f) = \text{Hom}(A, g) \circ \text{Hom}(A, f).$$

Taking an argument $x : A \to X$, we clearly have

$$\begin{aligned} \text{Hom}(A, 1_X)(x) &= 1_X \circ x \\ &= x \\ &= 1_{\text{Hom}(A,X)}(x) \end{aligned}$$

and

$$\begin{aligned} \text{Hom}(A, g \circ f)(x) &= (g \circ f) \circ x \\ &= g \circ (f \circ x) \\ &= \text{Hom}(A, g)(\text{Hom}(A, f)(x)). \end{aligned}$$

We will study such representable functors much more carefully later. For now, we just want to see how one can use Hom-sets to give another formulation of the definition of products.

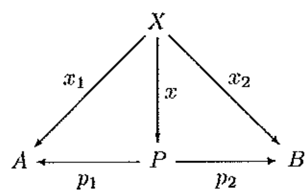
For any object P, a pair of arrows $p_1 : P \to A$ and $p_2 : P \to B$ determine an element (p_1, p_2) of the set

$$\mathrm{Hom}(P, A) \times \mathrm{Hom}(P, B).$$

Now, given any arrow

$$x : X \to P$$

composing with p_1 and p_2 gives a pair of arrows $x_1 = p_1 \circ x : X \to A$ and $x_2 = p_2 \circ x : X \to B$, as indicated in the following diagram:



In this way, we have a function

$$\vartheta_X = (\mathrm{Hom}(X, p_1), \mathrm{Hom}(X, p_2)) : \mathrm{Hom}(X, P) \to \mathrm{Hom}(X, A) \times \mathrm{Hom}(X, B)$$

defined by

$$\vartheta_X(x) = (x_1, x_2) \tag{2.1}$$

This function ϑ_X can be used to express concisely the condition of being a product as follows.

Proposition 2.20. *A diagram of the form*

$$A \xleftarrow[p_1]{} P \xrightarrow[p_2]{} B$$

is a product for A and B iff for every object X, the canonical function ϑ_X given in (2.1) *is an isomorphism,*

$$\vartheta_X : \mathrm{Hom}(X, P) \cong \mathrm{Hom}(X, A) \times \mathrm{Hom}(X, B).$$

Proof. Examine the UMP of the product: it says exactly that for every element $(x_1, x_2) \in \mathrm{Hom}(X, A) \times \mathrm{Hom}(X, B)$, there is a unique $x \in \mathrm{Hom}(X, P)$ such that $\vartheta_X(x) = (x_1, x_2)$, that is, ϑ_X is bijective. □

Definition 2.21. Let $\mathbf{C}$, $\mathbf{D}$ be categories with binary products. A functor $F : \mathbf{C} \to \mathbf{D}$ is said to *preserve binary products* if it takes every product diagram

$$A \xleftarrow[p_1]{} A \times B \xrightarrow[p_2]{} B \qquad \text{in } \mathbf{C}$$

to a product diagram

$$FA \xleftarrow[Fp_1]{} F(A \times B) \xrightarrow[Fp_2]{} FB \qquad \text{in } \mathbf{D}.$$

It follows that F preserves products just if

$$F(A \times B) \cong FA \times FB$$

"canonically," that is, iff the canonical "comparison arrow"

$$\langle Fp_1, Fp_2 \rangle : F(A \times B) \to FA \times FB$$

in $\mathbf{D}$ is an iso.

For example, the forgetful functor $U : \mathbf{Mon} \to \mathbf{Sets}$ preserves binary products.

Corollary 2.22. *For any object X in a category $\mathbf{C}$ with products, the (covariant) representable functor*

$$\mathrm{Hom}_{\mathbf{C}}(X, -) : \mathbf{C} \to \mathbf{Sets}$$

preserves products.

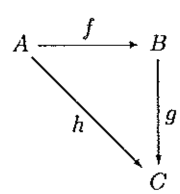
Proof. For any $A, B \in \mathbf{C}$, the foregoing proposition 2.20 says that there is a canonical isomorphism:

$$\mathrm{Hom}_{\mathbf{C}}(X, A \times B) \cong \mathrm{Hom}_{\mathbf{C}}(X, A) \times \mathrm{Hom}_{\mathbf{C}}(X, B)$$

□

2.8 Exercises

1. Show that a function between sets is an epimorphism if and only if it is surjective. Conclude that the isos in $\mathbf{Sets}$ are exactly the epi-monos.
2. Show that in a poset category, all arrows are both monic and epic.
3. (Inverses are unique.) If an arrow $f : A \to B$ has inverses $g, g' : B \to A$ (i. e., $g \circ f = 1_A$ and $f \circ g = 1_B$, and similarly for g'), then $g = g'$.
4. With regard to a commutative triangle,



   in any category $\mathbf{C}$, show

   (a) if f and g are isos (resp. monos, resp. epis), so is h;
   (b) if h is monic, so is f;
   (c) if h is epic, so is g;

(d) (by example) if h is monic, g need not be.

5. Show that the following are equivalent for an arrow
$$f : A \to B$$
in any category:
   (a) f is an isomorphism.
   (b) f is both a mono and a split epi.
   (c) f is both a split mono and an epi.
   (d) f is both a split mono and a split epi.
6. Show that a homomorphism $h : G \to H$ of graphs is monic just if it is injective on both edges and vertices.
7. Show that in any category, any retract of a projective object is also projective.
8. Show that all sets are projective (use the axiom of choice).
9. Show that the epis among posets are the surjections (on elements), and that the one-element poset **1** is projective.
10. Show that sets, regarded as discrete posets, are projective in the category of posets (use the foregoing exercises). Give an example of a poset that is not projective. Show that every projective poset is discrete, that is, a set. Conclude that **Sets** is (isomorphic to) the "full subcategory" of projectives in **Pos**, consisting of all projective posets and all monotone maps between them.
11. Let A be a set. Define an *A-monoid* to be a monoid M equipped with a function $m : A \to U(M)$ (to the underlying set of M). A morphism $h : (M, m) \to (N, n)$ of A-monoids is to be a monoid homomorphism $h : M \to N$ such that $U(h) \circ m = n$ (a commutative triangle). Together with the evident identities and composites, this defines a category A-**Mon** of A-monoids.

    Show that an initial object in A-**Mon** is the same thing as a free monoid $M(A)$ on A. (Hint: compare their respective UMPs.)
12. Show that for any Boolean algebra B, Boolean homomorphisms $h : B \to \mathbf{2}$ correspond exactly to ultrafilters in B.
13. In any category with binary products, show directly that
$$A \times (B \times C) \cong (A \times B) \times C.$$
14. (a) For any index set I, define the product $\prod_{i \in I} X_i$ of an I-indexed family of objects $(X_i)_{i \in I}$ in a category, by giving a UMP generalizing that for binary products (the case $I = 2$).
    (b) Show that in **Sets**, for any set X the set X^I of all functions $f : I \to X$ has this UMP, with respect to the "constant family" where $X_i = X$

for all $i \in I$, and thus

$$X^I \cong \prod_{i \in I} X$$

15. Given a category $\mathbf{C}$ with objects A and B, define the category $\mathbf{C}_{A,B}$ to have objects (X, x_1, x_2), where $x_1 : X \to A$, $x_2 : X \to B$, and with arrows $f : (X, x_1, x_2) \to (Y, y_1, y_2)$ being arrows $f : X \to Y$ with $y_1 \circ f = x_1$ and $y_2 \circ f = x_2$.
    Show that $\mathbf{C}_{A,B}$ has a terminal object just in case A and B have a product in $\mathbf{C}$.
16. In the category of types $\mathbf{C}(\lambda)$ of the λ-calculus, determine the product functor $A, B \mapsto A \times B$ explicitly. Also show that, for any fixed type A, there is a functor $A \to (-) : \mathbf{C}(\lambda) \to \mathbf{C}(\lambda)$, taking any type X to $A \to X$.
17. In any category $\mathbf{C}$ with products, define the *graph* of an arrow $f : A \to B$ to be the monomorphism

$$\Gamma(f) = \langle 1_A, f \rangle : A \rightarrowtail A \times B$$

    (Why is this monic?). Show that for $\mathbf{C} = \mathbf{Sets}$ this determines a functor $\Gamma : \mathbf{Sets} \to \mathbf{Rel}$ to the category $\mathbf{Rel}$ of relations, as defined in the exercises to Chapter 1. (To get an actual relation $R(f) \subseteq A \times B$, take the image of $\Gamma(f) : A \rightarrowtail A \times B$.)
18. Show that the forgetful functor $U : \mathbf{Mon} \to \mathbf{Sets}$ from monoids to sets is representable. Infer that U preserves all (small) products.

3

DUALITY

We have seen a few examples of definitions and statements that exhibit a kind of "duality," like initial and terminal object and epimorphisms and monomorphisms. We now want to consider this duality more systematically. Despite its rather trivial first impression, it is indeed a deep and powerful aspect of the categorical approach to mathematical structures.

3.1 The duality principle

First, let us look again at the formal definition of a category: There are two kinds of things, objects A, B, C and..., arrows $f, g, h, \ldots$; four operations $\mathrm{dom}(f)$, $\mathrm{cod}(f)$, 1_A, $g \circ f$; and these satisfy the following seven axioms:

$$\begin{aligned} \mathrm{dom}(1_A) = A \qquad & \mathrm{cod}(1_A) = A \\ f \circ 1_{\mathrm{dom}(f)} = f \qquad & 1_{\mathrm{cod}(f)} \circ f = f \\ \mathrm{dom}(g \circ f) = \mathrm{dom}(f) \qquad & \mathrm{cod}(g \circ f) = \mathrm{cod}(g) \\ h \circ (g \circ f) \ = & \ (h \circ g) \circ f \end{aligned} \tag{3.1}$$

The operation "$g \circ f$" is only defined where

$$\mathrm{dom}(g) = \mathrm{cod}(f),$$

so a suitable form of this should occur as a condition on each equation containing $\circ$, as in $\mathrm{dom}(g) = \mathrm{cod}(f) \Rightarrow \mathrm{dom}(g \circ f) = \mathrm{dom}(f)$.

Now, given any sentence Σ in the elementary language of category theory, we can form the "dual statement" Σ^* by making the following replacements:

$$\begin{array}{ccc} f \circ g & \text{for} & g \circ f \\ \mathrm{cod} & \text{for} & \mathrm{dom} \\ \mathrm{dom} & \text{for} & \mathrm{cod}. \end{array}$$

It is easy to see that then Σ^* will also be a well-formed sentence. Next, suppose we have shown a sentence Σ to entail one Δ, that is, $\Sigma \Rightarrow \Delta$, without using any of the category axioms, then clearly $\Sigma^* \Rightarrow \Delta^*$, since the substituted terms are treated as mere undefined constants. But now observe that the axioms (3.1) for category theory (CT) are themselves "self-dual," in the sense that we have,

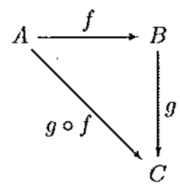
$$\mathrm{CT}^* = \mathrm{CT}.$$

We therefore have the following *duality principle.*

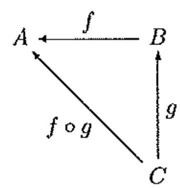
Proposition 3.1 (formal duality). *For any sentence Σ in the language of category theory, if Σ follows from the axioms for categories, then so does its dual Σ^*:*

$$\mathrm{CT} \Rightarrow \Sigma \quad \textit{implies} \quad \mathrm{CT} \Rightarrow \Sigma^*$$

Taking a more conceptual point of view, note that if a statement Σ involves some diagram of objects and arrows,



then the dual statement Σ^* involves the diagram obtained from it by reversing the direction and the order of compositions of arrows.



Recalling the opposite category $\mathbf{C}^{\mathrm{op}}$ of a category $\mathbf{C}$, we see that an interpretation of a statement Σ in $\mathbf{C}$ automatically gives an interpretation of Σ^* in $\mathbf{C}^{\mathrm{op}}$.

Now suppose that a statement Σ holds for all categories $\mathbf{C}$. Then it also holds in all categories $\mathbf{C}^{\mathrm{op}}$, and so Σ^* holds in all categories $(\mathbf{C}^{\mathrm{op}})^{\mathrm{op}}$. But since for every category $\mathbf{C}$,

$$(\mathbf{C}^{\mathrm{op}})^{\mathrm{op}} = \mathbf{C}, \tag{3.2}$$

we see that Σ^* also holds in all categories $\mathbf{C}$. We therefore have the following conceptual form of the duality principle.

Proposition 3.2 (Conceptual duality). *For any statement Σ about categories, if Σ holds for all categories, then so does the dual statement Σ^*.*

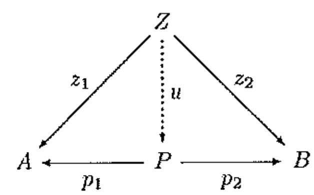
It may seem that only very simple or trivial statements, such as "terminal objects are unique up to isomorphism" are going to be subject to this sort of duality, but in fact this is far from being so. Categorical duality turns out to be a very powerful and a far-reaching phenomenon, as we see. Like the duality between points and lines in projective geometry, it effectively doubles ones "bang for the buck," yielding two theorems for every proof.

One way this occurs is that, rather than considering statements about all categories, we can also consider the dual of an abstract definition of a structure or property of objects and arrows, like "being a product diagram." The dual structure or property is arrived at by reversing the order of composition and the words "domain" and "codomain." (Equivalently, it results from interpreting the original property in the opposite category.) Section 3.2 provides an example of this kind.

3.2 Coproducts

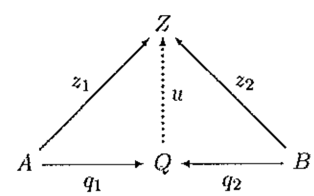
Let us consider the example of products and see what the dual notion must be. First, recall the definition of a product.

Definition 3.3. A diagram $A \xleftarrow{p_1} P \xrightarrow{p_2} B$ is a *product* of A and B, if for any Z and $A \xleftarrow{z_1} Z \xrightarrow{z_2} B$ there is a unique $u : Z \to P$ with $p_i \circ u = z_i$, all as indicated in



Now what is the dual statement?

A diagram $A \xrightarrow{q_1} Q \xleftarrow{q_2} B$ is a "dual-product" of A and B if for any Z and $A \xrightarrow{z_1} Z \xleftarrow{z_2} B$ there is a unique $u : Q \to Z$ with $u \circ q_i = z_i$, all as indicated in



Actually, these are called *coproducts*; the convention is to use the prefix "co-" to indicate the dual notion. We usually write $A \xrightarrow{i_1} A+B \xleftarrow{i_2} B$ for the coproduct and $[f, g]$ for the uniquely determined arrow $u : A+B \to Z$. The "coprojections" $i_1 : A \to A+B$ and $i_2 : B \to A+B$ are usually called *injections*, even though they need not be "injective" in any sense.

A coproduct of two objects is therefore exactly their product in the opposite category. Of course, this immediately gives lots of examples of coproducts. But what about some more familiar ones?

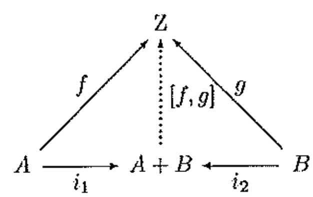
Example 3.4. In **Sets**, the coproduct $A + B$ of two sets is their disjoint union, which can be constructed, for example, as

$$A + B = \{(a, 1) \mid a \in A\} \cup \{(b, 2) \mid b \in B\}$$

with evident coproduct injections

$$i_1(a) = (a, 1), \qquad i_2(b) = (b, 2).$$

Given any functions f and g as in



we define

$$[f, g](x, \delta) = \begin{cases} f(x) & \delta = 1 \\ g(x) & \delta = 2. \end{cases}$$

Then, if we have an h with $h \circ i_1 = f$ and $h \circ i_2 = g$, then for any $(x, \delta) \in A + B$, we must have

$$h(x, \delta) = [f, g](x, \delta)$$

as can be easily calculated.

Note that in **Sets**, every finite set A is a coproduct:

$$A \cong 1 + 1 + \cdots + 1 \ \ (n\text{-times})$$

for $n = \text{card}(A)$. This is because a function $f : A \to Z$ is uniquely determined by its values $f(a)$ for all $a \in A$. So we have

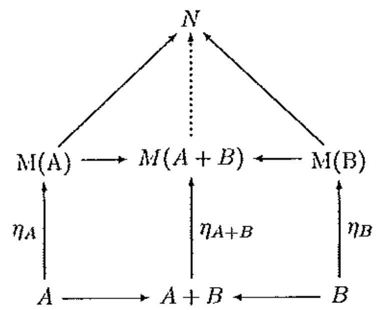
$$\begin{aligned} A &\cong \{a_1\} + \{a_n\} + \cdots + \{a_n\} \\ &\cong 1 + 1 + \cdots + 1 \ \ (n\text{-times}). \end{aligned}$$

In this spirit, we often write simply $2 = 1 + 1, 3 = 1 + 1 + 1$, etc.

Example 3.5. If $M(A)$ and $M(B)$ are *free* monoids on sets A and B, then in **Mon** we can construct their coproduct as

$$M(A) + M(B) \cong M(A + B).$$

One can see this directly by considering words over $A + B$, but it also follows abstractly by using the diagram



in which the η's are the respective insertions of generators. The universal mapping properties (UMPs) of $M(A)$, $M(B)$, $A + B$, and $M(A + B)$ then imply that the last of these has the required UMP of $M(A) + M(B)$. Note that the set of elements of the coproduct $M(A) + M(B)$ of $M(A)$ and $M(B)$ is *not* the coproduct of the underlying sets, but is only *generated by* the coproduct of their generators, $A + B$. We shall consider coproducts of arbitrary, that is, not necessarily free, monoids presently.

The foregoing example says that the free monoid functor $M : \mathbf{Sets} \to \mathbf{Mon}$ preserves coproducts. This is an instance of a much more general phenomenon, which we consider later, related to the fact we have already seen that the forgetful functor $U : \mathbf{Mon} \to \mathbf{Sets}$ is representable and so preserves products.

Example 3.6. In **Top**, the coproduct of two spaces

$$X + Y$$

is their disjoint union with the topology $O(X + Y) \cong O(X) \times O(Y)$. Note that this follows the pattern of discrete spaces, for which $O(X) = P(X) \cong 2^X$. Thus, for discrete spaces, we indeed have

$$O(X + Y) \cong 2^{X+Y} \cong 2^X \times 2^Y \cong O(X) \times O(Y).$$

A related fact is that the product of two powerset Boolean algebras $\mathcal{P}(A)$ and $\mathcal{P}(B)$ is also a powerset, namely of the coproduct of the sets A and B,

$$\mathcal{P}(A) \times \mathcal{P}(B) \cong \mathcal{P}(A + B).$$

We leave the verification as an exercise.

Coproducts of posets are similarly constructed from the coproducts of the underlying sets, by "putting them side by side." What about "rooted" posets, that is, posets with a distinguished initial element 0? In the category Pos_0 of such posets and monotone maps that preserve 0, one constructs the coproduct of two such posets A and B from the coproduct $A + B$ in the category Pos of posets, by "identifying" the two different 0s,

$$A +_{\mathrm{Pos}_0} B = (A +_{\mathrm{Pos}} B)/\text{“}0_A = 0_B\text{”}.$$

We shall soon see how to describe such identifications (quotients of equivalence relations) as "coequalizers."

Example 3.7. In a fixed poset P, what is a coproduct of two elements $p, q \in P$? We have

$$p \leq p + q \quad \text{and} \quad q \leq p + q$$

and if

$$p \leq z \quad \text{and} \quad q \leq z$$

then

$$p + q \leq z.$$

So $p + q = p \vee q$ is the *join*, or "least upper bound," of p and q.

Example 3.8. In the *category of proofs* of a deductive system of logic of example 10, Section 1.4, the usual natural deduction rules of disjunction introduction and elimination give rise to coproducts. Specifically, the introduction rules,

$$\frac{\varphi}{\varphi \vee \psi} \qquad \frac{\psi}{\varphi \vee \psi}$$

determine arrows $i_1 : \varphi \to \varphi \vee \psi$ and $i_2 : \psi \to \varphi \vee \psi$, and the elimination rule,

$$\frac{\varphi \vee \psi \qquad \begin{matrix}[\varphi] \\ \vdots \\ \vartheta\end{matrix} \qquad \begin{matrix}[\psi] \\ \vdots \\ \vartheta\end{matrix}}{\vartheta}$$

turns a pair of arrows $p : \varphi \to \vartheta$ and $q : \psi \to \vartheta$ into an arrow $[p, q] : \varphi \vee \psi \to \vartheta$. The required equations,

$$[p, q] \circ i_1 = p \qquad [p, q] \circ i_2 = q \tag{3.3}$$

will evidently not hold, however, since we are taking identity of proofs as identity of arrows. In order to get coproducts, then, we need to "force" these equations to hold by passing to equivalence classes of proofs, under the equivalence relation generated by these equations, together with the complementary one,

$$[r \circ i_1, r \circ i_2] = r \tag{3.4}$$

for any $r : A + B \to C$. (The intuition behind these identifications is that one should equate proofs which become the same when one omits such "detours.")

In the new category with equivalence classes of proofs as arrows, the arrow $[p, q]$ will also be the *unique* one satisfying (3.3), so that $\varphi \vee \psi$ indeed becomes a coproduct.

Closely related to this example (via the Curry–Howard correspondence of remark 2.18) are the sum types in the λ-calculus, as usually formulated using case terms; these are coproducts in the *category of types* defined in Section 2.5.

Example 3.9. Two monoids A, B have a coproduct of the form

$$A + B = M(|A| + |B|)/\sim$$

where, as before, the free monoid $M(|A|+|B|)$ is strings (words) over the disjoint union $|A| + |B|$ of the underlying sets—that is, the elements of A and B—and the equivalence relation $v \sim w$ is the least one containing all instances of the following equations:

$$\begin{aligned}(\dots x\, u_A\, y \dots) &= (\dots x\, y \dots)\\ (\dots x\, u_B\, y \dots) &= (\dots x\, y \dots)\\ (\dots a\, a' \dots) &= (\dots a \cdot_A a' \dots)\\ (\dots b\, b' \dots) &= (\dots b \cdot_B b' \dots).\end{aligned}$$

(If you need a refresher on quotienting a set by an equivalence relation, skip ahead and read the beginning of Section 3.4 now.) The unit is, of course, the equivalence class $[-]$ of the empty word (which is the same as $[u_A]$ and $[u_B]$). Multiplication of equivalence classes is also as expected, namely

$$[x \dots y] \cdot [x' \dots y'] = [x \dots y x' \dots y'].$$

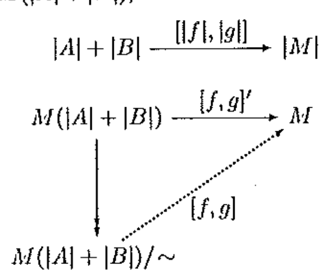
The coproduct injections $i_A : A \to A + B$ and $i_B : B \to A + B$ are simply

$$i_A(a) = [a], \qquad i_B(b) = [b],$$

which are now easily seen to be homomorphisms. Given any homomorphisms $f : A \to M$ and $g : B \to M$ into a monoid M, the unique homomorphism

$$[f, g] : A + B \longrightarrow M$$

is defined by first extending the function $[|f|, |g|] : |A| + |B| \to |M|$ to one $[f, g]'$ on the free monoid $M(|A| + |B|)$,



and then observing that $[f,g]'$ "respects the equivalence relation $\sim$," in the sense that if $v \sim w$ in $M(|A|+|B|)$, then $[f,g]'(v) = [f,g]'(w)$. Thus, the map $[f,g]'$ extends to the quotient to yield the desired map $[f,g] : M(|A|+|B|)/\sim \longrightarrow M$. (Why is this homomorphism the *unique* one $h : M(|A|+|B|)/\sim \longrightarrow M$ with $hi_A = f$ and $hi_B = g$?) Summarizing, we thus have

$$A + B \;\cong\; M(|A|+|B|)/\sim .$$

This construction also works to give coproducts in **Groups**, where it is usually called the *free product* of A and B and written $A \oplus B$, as well as many other categories of "algebras," that is, sets equipped with operations. Again, as in the free case, the underlying set of $A+B$ is *not* the coproduct of A and B as sets (the forgetful functor **Mon** $\to$ **Sets** does not preserve coproducts).

Example 3.10. For *abelian groups* A, B, the free product $A \oplus B$ need not be abelian. One could, of course, take a further quotient of $A \oplus B$ to get a coproduct in the category Ab of abelian groups, but there is a more convenient (and important) presentation, which we now consider.

Since the words in the free product $A \oplus B$ must be forced to satisfy the further commutativity conditions

$$(a_1 b_1 b_2 a_2 \ldots) \sim (a_1 a_2 \ldots b_1 b_2 \ldots)$$

we can shuffle all the a's to the front, and the b's to the back, of the words. But, furthermore, we already have

$$(a_1 a_2 \ldots b_1 b_2 \ldots) \sim (a_1 + a_2 + \cdots + b_1 + b_2 + \cdots).$$

Thus, we in effect have pairs of elements (a, b). So we can take the *product* set as the underlying set of the coproduct

$$|A + B| = |A \times B|.$$

As inclusions, we use the homomorphisms

$$i_A(a) = (a, 0_B)$$
$$i_B(b) = (0_A, b).$$

Then, given any homomorphisms $A \xrightarrow{f} X \xleftarrow{g} B$, we let $[f,g] : A + B \to X$ be defined by

$$[f,g](a,b) = f(a) +_X g(b)$$

which can easily be seen to do the trick (exercise!).

Moreover, not only can the underlying *sets* be the same, the product and coproduct of abelian groups are actually isomorphic as *groups*.

Proposition 3.11. *In the category* **Ab** *of abelian groups, there is a canonical isomorphism between the binary coproduct and product,*

$$A + B \cong A \times B.$$

ins: index entry: "product, of abelian groups"

Proof. To define an arrow $\vartheta : A + B \to A \times B$, we need one $A \to A \times B$ (and one $B \to A \times B$), so we need arrows $A \to A$ and $A \to B$ (and $B \to A$ and $B \to B$). For these, we take $1_A : A \to A$ and the zero homomorphism $0_B : A \to B$ (and $0_A : B \to A$ and $1_B : B \to B$). Thus, all together, we get

$$\vartheta = [\langle 1_A, 0_B\rangle, \langle 0_A, 1_B\rangle] : A + B \to A \times B.$$

Then given any $(a, b) \in A + B$, we have

$$\begin{aligned}
\vartheta(a, b) &= [\langle 1_A, 0_B\rangle, \langle 0_A, 1_B\rangle](a, b) \\
&= \langle 1_A, 0_B\rangle(a) + \langle 0_A, 1_B\rangle(b) \\
&= (1_A(a), 0_B(a)) + (0_A(b), 1_B(b)) \\
&= (a, 0_B) + (0_A, b) \\
&= (a + 0_A, 0_B + b) \\
&= (a, b).
\end{aligned}$$

□

This fact was first observed by Mac Lane, and it was shown to lead to a binary operation of addition on parallel arrows $f, g : A \to B$ between abelian groups (and related structures like modules and vector spaces). In fact, the group structure of a particular abelian group A can be recovered from this operation on arrows into A. More generally, the existence of such an addition operation on arrows can be used as the basis of an abstract description of categories like **Ab**, called "abelian categories," which are suitable for axiomatic homology theory.

Just as with products, one can consider the empty coproduct, which is an initial object 0, as well as coproducts of several factors, and the coproduct of two arrows,

$$f + f' : A + A' \to B + B'$$

which leads to a coproduct functor $+ : \mathbf{C} \times \mathbf{C} \to \mathbf{C}$ on categories $\mathbf{C}$ with binary coproducts. All of these facts follow simply by duality; that is, by considering the dual notions in the opposite category. Similarly, we have the following proposition.

Proposition 3.12. *Coproducts are unique up to isomorphism.*

Proof. Use duality and the fact that the dual of "isomorphism" is "isomorphism." □

In just the same way, one also shows that binary coproducts are associative up to isomorphism, $(A + B) + C \cong A + (B + C)$.

Thus is general, in the future it will suffice to introduce new notions once and then simply observe that the dual notions have analogous (but dual) properties. Sections 3.3 and 3.4 give another example of this sort.

3.3 Equalizers

In this section, we consider another abstract characterization; this time a common generalization of the kernel of a homomorphism and an equationally defined "variety," like the set of zeros of a real-valued function—as well as set theory's axiom of separation.

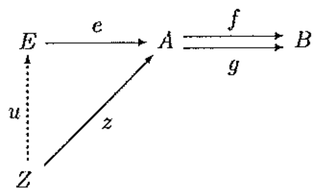
Definition 3.13. In any category **C**, given parallel arrows

$$A \underset{g}{\overset{f}{\rightrightarrows}} B$$

an *equalizer* of f and g consists of an object E and an arrow $e : E \to A$, universal such that

$$f \circ e = g \circ e.$$

That is, given any $z : Z \to A$ with $f \circ z = g \circ z$, there is a *unique* $u : Z \to E$ with $e \circ u = z$, all as in the diagram



Let us consider some simple examples.

Example 3.14. Suppose we have the functions $f, g : \mathbb{R}^2 \rightrightarrows \mathbb{R}$, where

$$f(x, y) = x^2 + y^2$$
$$g(x, y) = 1$$

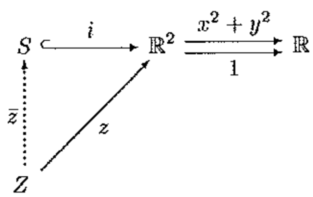
and we take the equalizer, say in **Top**. This is the subspace,

$$S = \{(x, y) \in \mathbb{R}^2 \mid x^2 + y^2 = 1\} \hookrightarrow \mathbb{R}^2,$$

that is, the unit circle in the plane. For, given any "generalized element" $z : Z \to \mathbb{R}^2$, we get a pair of such "elements" $z_1, z_2 : Z \to \mathbb{R}$ just by composing with the two projections, $z = \langle z_1, z_2 \rangle$, and for these we then have

$$\begin{aligned} f(z) = g(z) &\text{ iff } {z_1}^2 + {z_2}^2 = 1 \\ &\text{ iff } \text{“}\langle z_1, z_2 \rangle = z \in S\text{”}, \end{aligned}$$

where the last line really means that there is a factorization $z = \bar{z} \circ i$ of z through the inclusion $i : S \hookrightarrow \mathbb{R}^2$, as indicated in the following diagram:



Since the inclusion i is monic, such a factorization, if it exists, is necessarily unique, and thus $S \hookrightarrow \mathbb{R}^2$ is indeed the equalizer of f and g.

Example 3.15. Similarly, in **Sets**, given any functions $f, g : A \rightrightarrows B$, their equalizer is the inclusion into A of the equationally defined subset

$$\{x \in A \,|\, f(x) = g(x)\} \hookrightarrow A.$$

The argument is essentially the same as the one just given.

Let us pause here to note that in fact, every subset $U \subseteq A$ is of this "equational" form, that is, every subset is an equalizer for some pair of functions. Indeed, one can do this in a very canonical way. First, let us put

$$2 = \{\top, \bot\},$$

thinking of it as the set of "truth values." Then consider the *characteristic function*

$$\chi_U : A \to 2,$$

defined for $x \in A$ by

$$\chi_U(x) = \begin{cases} \top & x \in U \\ \bot & x \notin U. \end{cases}$$

Thus, we have

$$U = \{x \in A \,|\, \chi_U(x) = \top\}.$$

So the following is an equalizer:

$$U \longrightarrow A \underset{\chi_U}{\overset{\top!}{\rightrightarrows}} 2$$

where $\top! = \top \circ ! : U \xrightarrow{!} 1 \xrightarrow{\top} 2$.

Moreover, for every function,

$$\varphi : A \to 2$$

we can form the "variety" (i.e., equational subset)

$$V_\varphi = \{x \in A \mid \varphi(x) = \top\}$$

as an equalizer, in the same way. (Thinking of φ as a "propositional function" defined on A, the subset $V_\varphi \subseteq A$ is the "extension" of φ provided by the axiom of separation.)

Now, it is easy to see that these operations χ_U and V_φ are mutually inverse:

$$\begin{aligned} V_{\chi_U} &= \{x \in A \mid \chi_U(x) = \top\} \\ &= \{x \in A \mid x \in U\} \\ &= U \end{aligned}$$

for any $U \subseteq A$, and given any $\varphi : A \to 2$,

$$\begin{aligned} \chi_{V_\varphi}(x) &= \begin{cases} \top & x \in V_\varphi \\ \bot & x \notin V_\varphi \end{cases} \\ &= \begin{cases} \top & \varphi(x) = \top \\ \bot & \varphi(x) = \bot \end{cases} \\ &= \varphi(x). \end{aligned}$$

Thus, we have the familiar isomorphism

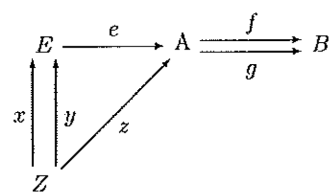
$$\mathrm{Hom}(A, 2) \cong P(A),$$

mediated by taking equalizers.

The fact that equalizers of functions can be taken to be subsets is a special case of a more general phenomenon.

Proposition 3.16. *In any category, if $e : E \to A$ is an equalizer of some pair of arrows, then e is monic.*

Proof. Consider the diagram



in which we assume e is the equalizer of f and g. Supposing $ex = ey$, we want to show $x = y$. Put $z = ex = ey$. Then $fz = fex = gex = gz$, so there is a *unique* $u : Z \to E$ such that $eu = z$. So from $ex = z$ and $ey = z$ it follows that $x = u = y$. □

Example 3.17. In many other categories, such as posets and monoids, the equalizer of a parallel pair of arrows $f, g : A \rightrightarrows B$ can be constructed by taking the equalizer of the underlying functions as above, that is, the subset $A(f = g) \subseteq A$ of elements $x \in A$ where f and g agree, $f(x) = g(x)$, and then restricting the structure of A to $A(f = g)$. For instance, in posets one takes the ordering from A restricted to this subset $A(f = g)$, and in topological spaces one takes the subspace topology.

In monoids, the subset $A(f = g)$ is then also a monoid with the operations from A, and the inclusion is therefore a homomorphism. This is so because $f(u_A) = u_B = g(u_A)$, and if $f(a) = g(a)$ and $f(a') = g(a')$, then $f(a \cdot a') = f(a) \cdot f(a') = g(a) \cdot g(a') = g(a \cdot a')$. Thus, $A(f = g)$ contains the unit and is closed under the product operation.

In abelian groups, for instance, one has an alternate description of the equalizer, using the fact that,

$$f(x) = g(x) \quad \text{iff} \quad (f - g)(x) = 0.$$

Thus, the equalizer of f and g is the same as that of the homomorphism $(f - g)$ and the zero homomorphism $0 : A \to B$, so it suffices to consider equalizers of the special form $A(h, 0) \rightarrowtail A$ for arbitrary homomorphisms $h : A \to B$. This subgroup of A is called the *kernel* of h, written $\ker(h)$. Thus, we have the equalizer

$$\ker(f - g) \hookrightarrow A \underset{g}{\overset{f}{\rightrightarrows}} B$$

The kernel of a homomorphism is of fundamental importance in the study of groups, as we consider further in Chapter 4.

3.4 Coequalizers

A coequalizer is a generalization of a quotient by an equivalence relation, so let us begin by reviewing that notion, which we have already made use of several times. Recall first that an *equivalence relation* on a set X is a binary relation $x \sim y$, which is

reflexive: $x \sim x$,
symmetric: $x \sim y$ implies $y \sim x$,
transitive: $x \sim y$ and $y \sim z$ implies $x \sim z$.

Given such a relation, define the *equivalence class* $[x]$ of an element $x \in X$ by

$$[x] = \{y \in X \mid x \sim y\}.$$

The various different equivalence classes $[x]$ then form a *partition* of X, in the sense that every element y is in exactly one of them, namely $[y]$ (prove this!).

One sometimes thinks of an equivalence relation as arising from the equivalent elements having some property in common (like being the same color). One can then regard the equivalence classes $[x]$ as the properties and in that sense as "abstract objects" (the colors red, blue, etc., themselves). This is sometimes known as "definition by abstraction," and it describes, for example, the way that the real numbers can be constructed from Cauchy sequences of rationals or the finite cardinal numbers from finite sets.

The set of all equivalence classes

$$X/\!\sim\; = \{[x] \mid x \in X\}$$

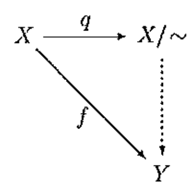
may be called the *quotient* of X by $\sim$. It is used in place of X when one wants to "abstract away" the difference between equivalent elements $x \sim y$, in the sense that in $X/\!\sim$ such elements (and only such) are identified, since

$$[x] = [y] \quad \text{iff} \quad x \sim y.$$

Observe that the *quotient mapping*,

$$q : X \longrightarrow X/\!\sim$$

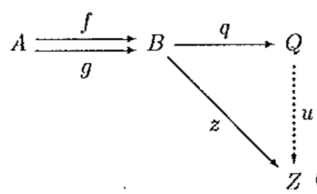
taking x to $[x]$ has the property that a map $f : X \to Y$ extends along q,



just in case f respects the equivalence relation, in the sense that $x \sim y$ implies $f(x) = f(y)$.

Now let us consider the notion dual to that of equalizer, namely that of a coequalizer.

Definition 3.18. For any parallel arrows $f, g : A \to B$ in a category $\mathbf{C}$, a *coequalizer* consists of Q and $q : B \to Q$, universal with the property $qf = qg$, as in



That is, given any Z and $z : B \to Z$, if $zf = zg$, then there exists a unique $u : Q \to Z$ such that $uq = z$.

First, observe that by duality, we know that such a coequalizer q in a category $\mathbf{C}$ is an equalizer in $\mathbf{C}^{\mathrm{op}}$, hence monic by proposition 3.16, and so epic in $\mathbf{C}$.

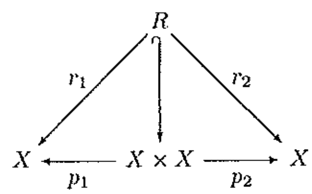
Proposition 3.19. *If $q : B \to Q$ is a coequalizer of some pair of arrows, then q is epic.*

We can therefore think of a coequalizer $q : B \twoheadrightarrow Q$ as a "collapse" of B by "identifying" all pairs $f(a) = g(a)$ (speaking as if there were such "elements" $a \in A$). Moreover, we do this in the "minimal" way, that is, disturbing B as little as possible, in that one can always map Q to anything else Z in which all such identifications hold.

Example 3.20. Let $R \subseteq X \times X$ be an equivalence relation on a set X, and consider the diagram

$$R \underset{r_2}{\overset{r_1}{\rightrightarrows}} X$$

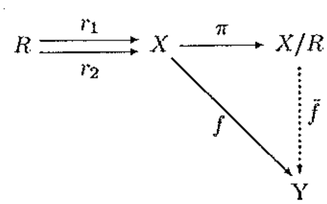
where the r's are the two projections of the inclusion $R \subseteq X \times X$,



The quotient projection

$$\pi : X \longrightarrow X/R$$

defined by $x \mapsto [x]$ is then a coequalizer of r_1 and r_2. For given an $f : X \to Y$ as in



there exists a function $\bar{f}$ such that

$$\bar{f}\pi(x) = f(x)$$

whenever f respects R in the sense that $(x, x') \in R$ implies $f(x) = f(x')$, as already noted. But this condition just says that $f \circ r_1 = f \circ r_2$, since $f \circ r_1(x, x') = f(x)$ and $f \circ r_2(x, x') = f(x')$ for all $(x, x') \in R$. Moreover, if it exists, such a function $\bar{f}$, is then necessarily unique, since π is an epimorphism.

The coequalizer in **Sets** of an arbitrary parallel pair of functions $f \cdot g : A \rightrightarrows B$ can be constructed by quotienting B by the equivalence relation generated by the equations $f(x) = g(x)$ for all $x \in A$. We leave the details as an exercise.

Example 3.21. In example 3.6, we considered the coproduct of *rooted* posets P and Q by first making $P+Q$ in posets and then "identifying" the resulting two different 0-elements 0_P and 0_Q (i.e., the images of these under the respective coproduct inclusions. We can now describe this "identification" as a coequalizer, taken in posets,

$$1 \underset{0_Q}{\overset{0_P}{\rightrightarrows}} P+Q \longrightarrow P+Q/(0_P = 0_Q)$$

This clearly has the right UMP to be the coproduct in *rooted* posets.

In topology one also often makes "identifications" of points (as in making the circle out of the interval by identifying the endpoints), of subspaces (making the torus from a region in the plane, etc.). These and many similar "gluing" constructions can be described as coequalizers. In **Top**, the coequalizer of a parallel pair of maps $f, g : X \to Y$ can be constructed as a *quotient space* of Y (see the exercises).

Example 3.22. *Presentations of algebras*
Consider any category of "algebras," that is, sets equipped with operations (of finite arity), such as monoids or groups. We shall show later that such a category has free algebras for all sets and coequalizers for all parallel pairs of arrows (see the exercises for a proof that monoids have coequalizers). We can use these to determine the notion of a *presentation* of an algebra by *generators* and *relations.* For example, suppose we are given

$$\begin{aligned} &\text{Generators:} \quad x,\ y,\ z \\ &\text{Relations:} \quad xy = z,\ y^2 = 1 \end{aligned} \tag{3.5}$$

To build an algebra on these generators and satisfying these relations, start with the free algebra,

$$F(3) = F(x, y, z),$$

and then "force" the relation $xy = z$ to hold by taking a coequalizer of the maps

$$F(1) \underset{z}{\overset{xy}{\rightrightarrows}} F(3) \xrightarrow{\ q\ } Q$$

We use the fact that maps $F(1) \to A$ correspond to elements $a \in A$ by $v \mapsto a$, where v is the single generator of $F(1)$. Now similarly, for the equation $y^2 = 1$, take the coequalizer

$$F(1) \underset{q(1)}{\overset{q(y^2)}{\rightrightarrows}} Q \longrightarrow Q'$$

These two steps can actually be done simultaneously. Let

$$F(2) = F(1) + F(1)$$

$$F(2) \underset{g}{\overset{f}{\rightrightarrows}} F(3)$$

where $f = [xy, y^2]$ and $g = [z, 1]$. The coequalizer $q : F(3) \to Q$ of f and g then "forces" both equations to hold, in the sense that in Q, we have

$$q(x)q(y) = q(z), \quad q(y)^2 = 1.$$

Moreover, no other relations among the generators hold in Q except those required to hold by the stipulated equations. To make the last statement precise, observe that given any algebra A and any three elements $a, b, c \in A$ such that $ab = c$ and $b^2 = 1$, by the UMP of Q there is a unique homomorphism $u : Q \to A$ such that

$$u(x) = a, \qquad u(y) = b, \qquad u(z) = c.$$

Thus, any other equation that holds among the generators in Q will also hold in any other algebra in which the stipulated equations (3.5) hold, since the homomorphism u also preserves equations. In this sense, Q is the "universal" algebra with three generators satisfying the stipulated equations; as may be written suggestively in the form

$$Q \cong F(x, y, z)/(xy = z,\ y^2 = 1).$$

Generally, given a finite presentation

$$\begin{aligned} &\text{Generators:} \quad g_1, \ldots, g_n \\ &\text{Relations:} \quad l_1 = r_1,\ \ldots,\ l_m = r_m \end{aligned} \tag{3.6}$$

(where the l_i and r_i are arbitrary terms built from the generators and the operations) the algebra determined by that presentation is the coequalizer

$$F(m) \underset{r}{\overset{l}{\rightrightarrows}} F(n) \longrightarrow Q = F(n)/(l = r)$$

where $l = [l_1, \ldots, l_m]$ and $r = [r_1, \ldots, r_m]$. Moreover, any such coequalizer between (finite) free algebras can clearly be regarded as a (finite) presentation by generators and relations. Algebras that can be given in this way are said to be *finitely presented.*

Warning 3.23. Presentations are not unique. One may well have two different presentations $F(n)/(l = r)$ and $F(n')/(l' = r')$ by generators and relations of the same algebra,

$$F(n)/(l = r) \ \cong\ F(n')/(l' = r').$$

For instance, given $F(n)/(l = r)$ just add a new generator g_{n+1} and the new relation $g_n = g_{n+1}$. In general, there are many different ways of presenting a given algebra, just like there are many ways of axiomatizing a logical theory.

We did not really make use of the finiteness condition in the foregoing considerations. Indeed, *any* sets of generators G and relations R give rise to an algebra in the same way, by taking the coequalizer

$$F(R) \underset{r_2}{\overset{r_1}{\rightrightarrows}} F(G) \longrightarrow F(G)/(r_1 = r_2).$$

In fact, every algebra can be "presented" by generators and relations in this sense, that is, as a coequalizer of maps between free algebras. Specifically, we have the following proposition for monoids, an analogous version of which also holds for groups and other algebras.

Proposition 3.24. *For every monoid M there are sets R and G and a coequalizer diagram,*

$$F(R) \underset{r_2}{\overset{r_1}{\rightrightarrows}} F(G) \longrightarrow M$$

with $F(R)$ and $F(G)$ free; thus, $M \cong F(G)/(r_1 = r_2)$.

Proof. For any monoid N, let us write $TN = M(|N|)$ for the free monoid on the set of elements of N (and note that T is therefore a functor). There is a homomorphism,

$$\pi : TN \to N$$
$$\pi(x_1, \ldots, x_n) = x_1 \cdot \ldots \cdot x_n$$

induced by the identity $1_{|N|} : |N| \to |N|$ on the generators. (Here we are writing the elements of TN as tuples $(x_1, \ldots, x_n)$ rather than strings $x_1 \ldots x_n$ for clarity.) Applying this construction twice to a monoid M results in the arrows π and ε in the following diagram:

Au: Please confirm if the change from "$x_1 \ldots x_n$" to "$x_1 \cdot \ldots \cdot x_n$" is OK as per the above equation.

GK

ins.

$$T^2M \underset{\mu}{\overset{\varepsilon}{\rightrightarrows}} TM \underset{\pi}{\longrightarrow} M \tag{3.7}$$

where $T^2M = TTM$ and $\mu = T\pi$. Explicitly, the elements of T^2M are tuples of tuples of elements of M, say $((x_1, \ldots, x_n), \ldots, (z_1, \ldots, z_m))$, and the homomorphisms ε and μ have the effect :

$$\varepsilon((x_1, \ldots, x_n), \ldots, (z_1, \ldots, z_m)) = (x_1, \ldots, x_n, \ldots, z_1, \ldots, z_m)$$
$$\mu((x_1, \ldots, x_n), \ldots, (z_1, \ldots, z_m)) = (x_1 \cdot \ldots \cdot x_n, \ldots, z_1 \cdot \ldots \cdot z_m)$$

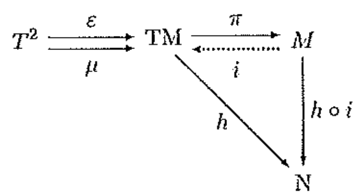
Briefly, ε uses the multiplication in TM and μ uses that in M.

Now clearly $\pi \circ \varepsilon = \pi \circ \mu$. We claim that (3.7) is a coequalizer of monoids. To that end, suppose we have a monoid N and a homomorphism $h : TM \to N$

with $h\varepsilon = h\mu$. Then for any tuple $(x, \dots, z)$, we have

$$\begin{aligned} h(x, \dots, z) &= h\varepsilon((x, \dots, z)) \\ &= h\mu((x, \dots, z)) \\ &= h(x \cdot \ldots \cdot z). \end{aligned} \tag{3.8}$$

Now define $\bar{h} = h \circ i$, where $i : |M| \to |TM|$ is the insertion of generators, as indicated in the following:



We then have

$$\begin{aligned} \bar{h}\pi(x, \dots, z) &= hi\pi(x, \dots, z) \\ &= h(x \cdot \ldots \cdot z) \\ &= h(x, \dots, z) \qquad \text{by (3.8)}. \end{aligned}$$

We leave it as an easy exercise for the reader to show that $\bar{h}$ is a homomorphism. □

3.5 Exercises

1. In any category $\mathbf{C}$, show that

$$A \xrightarrow{c_1} C \xleftarrow{c_2} B$$

is a coproduct diagram just if for every object Z, the map

$$\begin{aligned} \mathrm{Hom}(C, Z) &\longrightarrow \mathrm{Hom}(A, Z) \times \mathrm{Hom}(B, Z) \\ f &\longmapsto \langle f \circ c_1,\ f \circ c_2 \rangle \end{aligned}$$

is an isomorphism. Do this by using duality, taking the corresponding fact about products as given.

2. Show in detail that the free monoid functor M preserves coproducts: for any sets A, B,

$$M(A) + M(B) \cong M(A + B) \qquad \text{(canonically)}.$$

Do this as indicated in the text by using the UMPs of the coproducts $A + B$ and $M(A) + M(B)$ and of free monoids.

3. Verify that the construction given in the text of the coproduct of monoids $A+B$ as a quotient of the free monoid $M(|A|+|B|)$ really is a coproduct in the category of monoids.
4. Show that the product of two powerset Boolean algebras $\mathcal{P}(A)$ and $\mathcal{P}(B)$ is also a powerset, namely of the coproduct of the sets A and B,
$$\mathcal{P}(A) \times \mathcal{P}(B) \cong \mathcal{P}(A+B).$$
(Hint: determine the projections $\pi_1 : \mathcal{P}(A+B) \to \mathcal{P}(A)$ and $\pi_2 : \mathcal{P}(A+B) \to \mathcal{P}(B)$, and check that they have the UMP of the product.)
5. Consider the category of proofs of a natural deduction system with disjunction introduction and elimination rules. Identify proofs under the equations
$$[p,q] \circ i_1 = p, \qquad [p,q] \circ i_2 = q$$
$$[r \circ i_1, r \circ i_2] = r$$
for any $p : A \to C$, $q : B \to C$, and $r : A+B \to C$. By passing to equivalence classes of proofs with respect to the equivalence relation generated by these equations (i.e., two proofs are equivalent if you can get one from the other by removing all such "detours"). Show that the resulting category does indeed have coproducts.
6. Verify that the category of monoids has all equalizers and finite products, then do the same for abelian groups.
7. Show that in any category with coproducts, the coproduct of two projectives is again projective.
8. Dualize the notion of projectivity to define an *injective* object in a category. Show that a map of posets is monic iff it is injective on elements. Give examples of a poset that is injective and one that is not injective.
9. Complete the proof of proposition 3.24 in the text by showing that $\bar{h}$ is indeed a homomorphism.
10. In the proof of proposition 3.24 in the text it is shown that any monoid M has a specific presentation $T^2M \rightrightarrows TM \to M$ as a coequalizer of free monoids. Show that coequalizers of this particular form are preserved by the forgetful functor $\mathbf{Mon} \to \mathbf{Sets}$.
11. Prove that $\mathbf{Sets}$ has all coequalizers by constructing the coequalizer of a parallel pair of functions,
$$A \underset{g}{\overset{f}{\rightrightarrows}} B \longrightarrow Q = B/(f=g)$$
by quotienting B by a suitable equivalence relation R on B, generated by the pairs $(f(x), g(x))$ for all $x \in A$. (Define R to be the intersection of all equivalence relations on B containing all such pairs.)

12. Verify the coproduct–coequalizer construction mentioned in the text for coproducts of rooted posets, that is, posets with a least element 0 and monotone maps preserving 0. Specifically, show that the coproduct $P +_0 Q$ of two such posets can be constructed as a coequalizer in posets,

$$1 \underset{0_Q}{\overset{0_P}{\rightrightarrows}} P + Q \longrightarrow P +_0 Q.$$

(You may assume as given the fact that the category of posets has all coequalizers.)

13. Show that the category of monoids has all coequalizers as follows.
1. Given any pair of monoid homomorphisms $f, g : M \to N$, show that the following equivalence relations on N agree:
   (a) $n \sim n' \Leftrightarrow$ for all monoids X and homomorphisms $h : N \to X$, one has $hf = hg$ implies $hn = hn'$,
   (b) the intersection of all equivalence relations $\sim$ on N satisfying $fm \sim gm$ for all $m \in M$ as well as
   $$n \sim n' \text{ and } m \sim m' \Rightarrow n \cdot m \sim n' \cdot m'$$
2. Taking $\sim$ to be the equivalence relation defined in (1), show that the quotient set $N/{\sim}$ is a monoid under $[n] \cdot [m] = [n \cdot m]$, and the projection $N \to N/{\sim}$ is the coequalizer of f and g.

14. Consider the following category of sets:
   (a) Given a function $f : A \to B$, describe the equalizer of the functions $f \circ p_1, f \circ p_2 : A \times A \to B$ as a (binary) relation on A and show that it is an equivalence relation (called the *kernel* of f).
   (b) Show that the kernel of the quotient $A \to A/R$ by an equivalence relation R is R itself.
   (c) Given *any* binary relation $R \subseteq A \times A$, let $\langle R \rangle$ be the equivalence relation on A generated by R (the least equivalence relation on A containing R). Show that the quotient $A \to A/\langle R \rangle$ is the coequalizer of the two projections $R \rightrightarrows A$.
   (d) Using the foregoing, show that for any binary relation R on a set A, one can characterize the equivalence relation $\langle R \rangle$ generated by R as the kernel of the coequalizer of the two projections of R.

15. Construct coequalizers in **Top** as follows. Given a parallel pair of maps $f, g : X \rightrightarrows Y$, make a *quotient space* $q : Y \to Q$ by (i) taking the coequalizer of $|f|$ and $|g|$ in **Sets** to get the function $|q| : |Y| \to |Q|$, then (ii) equip $|Q|$ with the *quotient topology*, under which a set $V \subseteq Q$ is open iff $q^{-1}(V) \subseteq Y$ is open. This is plainly the finest topology on $|Q|$ that makes the projection $|q|$ continuous.