Contents

Chapter 1

Background

In this chapter the building blocks for this work, almost entirely based on categories, will be defined. The aim of what follows is not only to introduce concepts that will be used later, but also to understand how category theory is general enough to give the abstraction of known notions (mainly from set theory) to reuse them in different contexts. This is not a complete tutorial on categories, but instead a sufficient compendium of definitions to make clear what will be done in the next chapters.

1.1 Basic Notions

This section is all about basic definitions and examples, to get familiar with the formalism of categories.

1.1.1 Categories

Definition 1.1.1 (Category)**.** A *category* $\mathscr{C}$ comprises:

1. A collection of *objects* $\mathcal{O}b(\mathscr{C})$;
2. A collection of *arrows* (or *morphisms*) $\mathcal{H}om(\mathscr{C})$, often called *homset*.

For each morphism $f \in \mathcal{H}om(\mathscr{C})$, there are two operators *dom* and *cod* that map every morphism to two objects, respectively, its *domain* and its *codomain*. In case $dom\ f = A$ and $cod\ f = B$, we will write $f : A \to B$. The collection of morphisms from an object A to an object B is denoted as $\mathscr{C}(A, B)$. An operator $\circ$ of *composition* maps every couple of morphisms f, g with $cod\ f = dom\ g$ (in this case f and g are said to be composable) to a morphism $g \circ f : dom\ f \to cod\ g$. The composition operator is associative, i.e., for each composable arrows f, g and h, it holds that

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

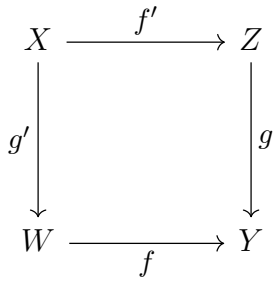
For each object A, an *identity* morphism $id_A : A \to A$ (or, when it is clear from the context, just denoted A) such that, for each $f : A \to B$:

$$id_B \circ f = f = f \circ id_A$$

The most important thing here is not the structure of the objects, but instead how this structure is preserved by the morphisms.

Example 1.1.2. The category with just one object and just one arrow, the identity arrow on that object, is denoted **1**. In particular, the only object of this category is $\bullet$, and the only arrow is $id_\bullet$.

To represent morphisms of a category $\mathscr{C}$ it is possible to use *diagrams*, as the one below, in which the vertices are objects of $\mathscr{C}$, and the edges are morphisms of $\mathscr{C}$.



The diagram is said to commute whenever $f \circ g' = g \circ f'$. Unique morphisms are represented with dashed arrows. A more rigorous definition of what a diagram is will be given later (Definition 1.2.3).

Example 1.1.3. It is easy to see that taking sets as objects and total functions as arrows, we obtain a category. In fact, given two functions $f : A \to B$ and $g : B \to C$, it is possible to compose them obtaining an arrow $g \circ f : A \to C$, and the composition is associative. For each set A there exists an identity function $id_A : A \to A$ such that $\forall a \in A : id_A(a) = a$. This category is denoted as **Set**.

Remark 1.1.4. It is important to note that the Definition 1.1.1 above does not specify what kind of collections $\mathcal{O}b(\mathscr{C})$ and $\mathcal{H}om(\mathscr{C})$ are. Taking **Set** as example, the collection $\mathcal{O}b(\mathbf{Set})$ cannot be a set itself, due to Russel's paradox. It would be more appropriate referring to a category $\mathscr{C}$ which $\mathcal{O}b(\mathscr{C})$ and $\mathcal{H}om(\mathscr{C})$ are both sets as a *small category*, but it is assumed in this work, except where it is made explicit, for a category to be small. Another clarification must to be done, still considering **Set**. Given two sets A and B, it is possible to construct the set B^A of all functions from A to B. This is isomorphic to $\mathbf{Set}(A, B)$, for each pair of sets A and B. rivedere A category $\mathscr{C}$ where, for each pair of objects A and B, $\mathscr{C}(A, B)$ is a set is said to be *locally small*.

1.1.2 Mono, Epi and Iso

Between the morphisms of a category, it is possible to distinguish some that have certain properties, as functions between sets can be surjective, injective or bijective.

Definition 1.1.5 (Monomorphism)**.** An arrow $f : B \to C$ in a category $\mathscr{C}$ is a *monomorphism* if, for any pair of arrows of $\mathscr{C}$ $g : A \to B$, $h : A \to B$, the equality $f \circ g = f \circ h$ implies $g = h$. The class of monomorphisms of $\mathscr{C}$ is denoted $\mathcal{M}ono(\mathscr{C})$.

Definition 1.1.6 (Epimorphism)**.** An arrow $f : A \to B$ in a category $\mathscr{C}$ is an *epimorphism* if, for any pair of arrows of $\mathscr{C}$ $g : B \to C$, $h : B \to C$, the equality $g \circ f = h \circ f$ implies $g = h$.

Definition 1.1.7 (Isomorphism)**.** An arrow $f : A \to B$ is an *isomorphism* if there is an arrow $f^{-1} : B \to A$, called the *inverse* of f, such that $f^{-1} \circ f = id_A$ and $f \circ f^{-1} = id_B$. Two objects are said to be *isomorphic* if there is an isomorphism between them.

Example 1.1.8. In **Set**, monomorphisms are injective functions, epimorphisms are surjective functions and isomorphisms are bijections.

Proposition 1.1.9. *The following statements hold for every pair of composable arrows f and g for any category $\mathscr{C}$:*

1. *if both f and g are mono, then $g \circ f$ is mono;*
2. *if $g \circ f$ is mono, then f is mono;*
3. *if both f and g are epi, then $g \circ f$ is epi;*
4. *if $g \circ f$ is epi, then g is epi.*

1.1.3 Categories from other categories

Starting from a category $\mathscr{C}$, it is possible to construct other categories with interesting property. An example is the *dual category* of a category $\mathscr{C}$, denoted $\mathscr{C}^{op}$, in which the objects are the same of $\mathscr{C}$, and the arrows are the opposite of the arrows in $\mathscr{C}$, i.e., if $f : A \to B$ is an arrow of $\mathscr{C}$, then $f : B \to A$ is an arrow of $\mathscr{C}^{op}$. Each definition in category theory has a dual form. In general, if a statement S is true in a category $\mathscr{C}$, then the opposite of the statement, S^{op}, obtained switching the words "domain" and "codomain" and replacing each composite $g \circ f$ into $f \circ g$, is still true in the category $\mathscr{C}^{op}$. Moreover, since every category is the opposite of its opposite, if a statement S is true for every category, then S^{op} is also true for every category [Pie91, pp8-9]. la discussione sul duale temo confonda

Another important notion is that of subcategory.

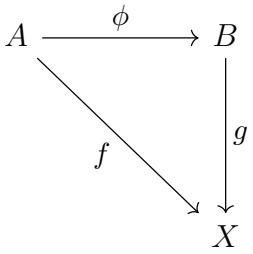
Definition 1.1.10 (Subcategory)**.** A category $\mathscr{D}$ is a *subcategory* of a category $\mathscr{C}$ if:

1. each object of $\mathscr{D}$ is an object of $\mathscr{C}$;
2. each morphism between two objects of $\mathscr{D}$ is a morphism of $\mathscr{C}$; and
3. composites and identities of $\mathscr{D}$ are the same of $\mathscr{C}$

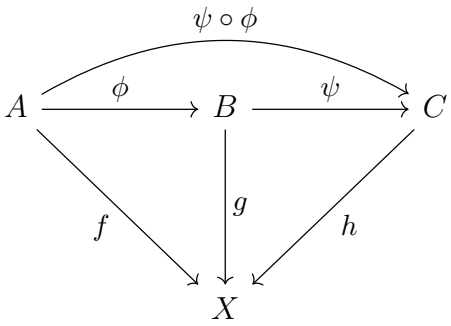
If the inclusion at 2 is an equality (i.e. $\mathscr{D}(A, B) = \mathscr{C}(A, B)$ for each couple of objects A, B of $\mathscr{D}$), then $\mathscr{D}$ is said to be a *full subcategory* of $\mathscr{C}$. Another way to express that composites are the same (point 3) is to say that if $f, g \in \mathcal{H}om(\mathscr{D})$ are composable, then $g \circ f \in \mathcal{H}om(\mathscr{D})$, i.e., $\mathcal{H}om(\mathscr{D})$ is *closed under composition*.

An object of a category marks out a category itself. This is the case of slice (and coslice) categories.

Definition 1.1.11 (Slice Category)**.** Given a category $\mathscr{C}$ and an object $X \in \mathcal{O}b(\mathscr{C})$, the *slice category* $\mathscr{C}/X$ is the category that has pairs (A, f) as objects, where A is an object of C and $f : A \to X$ is an arrow in $\mathscr{C}$, and arrows $\phi : (A, f) \to (B, g)$ are given by a morphism $\phi : A \to B$ of $\mathscr{C}$ such that the following diagram commutes:



– i.e, $g \circ \phi = f$. Composition between two arrows in $\mathscr{C}/X$ $\phi : (A, f) \to (B, g)$ and $\psi : (B, g) \to (C, h)$ is the arrow $\psi \circ \phi : (A, f) \to (C, h)$ obtained in the obvious way:



The dual definition of *coslice category*, noted $X/\mathscr{C}$ (where $X \in \mathcal{O}b(\mathscr{C})$), is obtained by taking as objects the morhisms of $\mathscr{C}$ with domain X and as arrows the morphisms $\phi : (A, f) \to (B, g)$ such that $f : X \to A, g : B \to X$ of $\mathscr{C}$ and $g = \phi \circ f$.

Furthermore, it is possible to raise a new category from two old ones by taking their product, as the following definition shows.

Definition 1.1.12 (Product category)**.** Given two categories $\mathscr{C}, \mathscr{D}$, the *product category* $\mathscr{C} \times \mathscr{D}$ has as objects pairs of objects (A, B), where $A \in \mathcal{O}b(\mathscr{C}), B \in \mathcal{O}b(\mathscr{D})$, and as arrows pairs of morphisms (f, g), where f is an arrow in $\mathscr{C}$ and g is an arrow in $\mathscr{D}$. Composition and identities are defined pairwise: $(f, g) \circ (h, k) = (f \circ h, g \circ k)$, and $id_{(A,B)} = (id_A, id_B)$.

1.2 Functors and Natural Transformations

bierei an-
ntro a questa
e DC

A functor is a structure preserving map between categories.

1.2.1 Functors

Definition 1.2.1 (Functor)**.** Let $\mathscr{C}$ and $\mathscr{D}$ be categories. A *functor* $F : \mathscr{C} \to \mathscr{D}$ is a map taking each object of $A \in \mathcal{O}b(\mathscr{C})$ to an object $F(A) \in \mathcal{O}b(\mathscr{D})$ and each arrow $f : A \to B$ of $\mathscr{C}$ to a arrow $F(f) : F(A) \to F(B)$ of $\mathscr{D}$, such that, for all objects $A \in \mathcal{O}b(\mathscr{C})$ and composable arrows f and g of $\mathscr{C}$:

- $F(id_A) = id_{F(A)}$;
- $F(g \circ f) = F(g) \circ F(f)$.

In this case, $\mathscr{C}$ is called *domain* and $\mathscr{D}$ is called *codomain* of the functor F.

Example 1.2.2. A first example of functor is the *identity functor*. Given a category $\mathscr{C}$, the identity functor $Id_{\mathscr{C}} : \mathscr{C} \to \mathscr{C}$ is the functor that maps each object on itself and each arrow onto itself.

Once defined what a functor is, we can give a more rigorous definition of diagram. Although this may seem extremely technical, it will be useful, especially in the definition of limits (Definition 1.3.18).

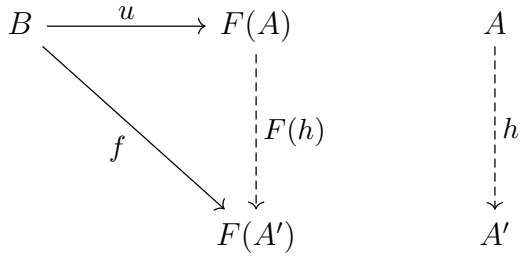
Definition 1.2.3 (Diagram)**.** A *diagram in a category* $\mathscr{C}$ *of shape* $\mathscr{I}$ is a functor $D : \mathscr{I} \to \mathscr{C}$. The category $\mathscr{I}$ can be considered as the category indexing the objects and the morphisms of $\mathscr{C}$ shaped in $\mathscr{I}$.

Example 1.2.4. A diagram of shape $\Lambda = (L \xleftarrow{l} X \xrightarrow{r} R)$ is said to be a *span*, and is denoted by $(l, X, r) : L \nrightarrow R$. A span can be viewed as the generalization of relations between sets. In fact, in **Set**, a relation $R \subseteq A \times B$ is a span, with the projections $\pi_A : R \to A$ and $\pi_B : R \to B$ as arrows.

The dual notion of span is a *cospan*, namely, a diagram of shape $\Lambda^{op} = (L \xrightarrow{l} X \xleftarrow{r} R)$, and is denoted by $(l, X, r) : L \nleftarrow R$.

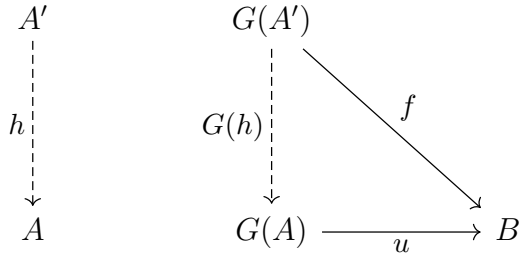
Functor are often used to generalize some structural behaviour that constructions in categories have. An important example of this fact is the universal property. The definition is not straightforward, but it gives the abstraction of a property that will be useful in further definitions [HS79].

Definition 1.2.5 (Universal property)**.** Let $F : \mathscr{C} \to \mathscr{D}$ be a functor, and let $B \in \mathcal{O}b(\mathscr{D})$. A pair (u, A), with $A \in \mathcal{O}b(\mathscr{C})$ and $u : B \to F(A)$ is said to be an *universal map for* B *with respect to* F if for each $A' \in \mathcal{O}b(\mathscr{C})$ and each $f : B \to F(A')$ there exists a unique morphism $h \in \mathscr{C}(A, A')$ such that the following triangle commute:



– i.e. there exists a unique h such that $F(h) \circ u = f$. In this case, (u, A) is said to have the *universal property*.

Dually, if $G : \mathscr{C} \to \mathscr{D}$ is a functor and $B \in \mathcal{O}b(\mathscr{D})$, then a pair (A, u) is a *co-universal map for B with respect to G* if $u : G(A) \to B$ and for each $A' \in \mathcal{O}b(\mathscr{C})$ and each $f : G(A') \to B$ there exists a unique morphism $h \in \mathscr{C}(A', A)$ such that the following diagram commutes:



Some interesting properties of certain functors depend strictly on how they behave on the homsets of the domain and the codomain categories. The following definitions are about this particular type of functors.

Definition 1.2.6 (Full functor, faithful functor, fully faithful functor)**.** Let $F : \mathscr{C} \to \mathscr{D}$ be a functor, and consider the inducted function

$$F_{A,B} : \mathscr{C}(A, B) \to \mathscr{D}(F(A), F(B))$$

If, for each A, B objects of $\mathscr{C}$, $F_{A,B}$ is surjective, then F is said to be *full*, if it is injective, F is said to be *faithful*, if it is both injective and surjective, F is said to be *fully faithful*.

Observation 1.2.7. Properties such as fullness and faithfulness are so called *self-dual*, because the dual notion coincide with the same notion. This fact can be advantageous because if for example the faithfulness implies the preservation of some property, then the dual property is implied at the same way.

Example 1.2.8. Let $\mathscr{C}$ be a category and $\mathscr{D}$ a subcategory. The inclusion functor $I : \mathscr{D} \to \mathscr{C}$, mapping each object and each arrow onto itself. I is a faithful functor, because, given any pair of objects A and B of $\mathscr{D}$, $I_{A,B}$ is injective. If $\mathscr{D}$ is a full subcategory, then I is fully faithful.

Having such classification among functors turns out to be useful in many contexts. For example, consider $F(m) : F(B) \to F(C)$ be a monomorphism in a category $\mathscr{D}$, where $F : \mathscr{C} \to \mathscr{D}$ is a faithful functor. Then, if $f, g : A \to B$ are two morphisms in $\mathscr{C}$ such that $m \circ f = m \circ g$, then $F(m \circ f) = F(m) \circ F(f) = F(m) \circ F(g) = F(m \circ g)$. Since $F(m)$ is mono, then $F(f) = F(g)$, and, since $F_{A,B}$ is injective, $f = g$. Together with the fact that faithfulness is a self-dual concept, we have a proof for what follows [HS79].

Proposition 1.2.9. *Let $F : \mathscr{C} \to \mathscr{D}$ be a faithful functor. Then F reflects monomorphisms and epimorphisms.*

1.2.2 Natural Transformations

Given two functors that share domain and codomain categories, it is possible to define a transformation between them, taking each object of the domain of the functors to an arrow in the codomain of the functors that represent the action of "changing the functor acting on that object".

Definition 1.2.10 (Natural transformation)**.** Let $F, G : \mathscr{C} \to \mathscr{D}$ be two functors. A *natural transformation* η between them, denoted $\eta : F \dot{\to} G$, is a function $\eta : \mathcal{O}b(\mathscr{C}) \to \mathcal{H}om(\mathscr{D})$ taking each $A \in \mathcal{O}b(\mathscr{C})$ to a morphism $\eta_A : F(A) \to G(A)$ in $\mathscr{D}$, such that, for each morphism $f : A \to B$ of $\mathscr{C}$, the following diagram commutes:

$$\begin{array}{ccc} F(A) & \xrightarrow{\eta_A} & G(A) \\ {\scriptstyle F(f)}\downarrow & & \downarrow{\scriptstyle G(f)} \\ F(B) & \xrightarrow[\eta_B]{} & G(B) \end{array}$$

– i.e., such that $G(f) \circ \eta_A = \eta_B \circ F(f)$.

We say that $\eta : F \dot{\to} G$ is a *natural isomorphism* if, for each $A \in \mathcal{O}b(\mathscr{C})$, η_A is an isomorphism in $\mathscr{D}$. In this case, F and G are said to be *naturally isomorphic*, and is denoted $F \cong G$.

Observation 1.2.11. It is easy to see that, given two natural transformations $\eta : F \dot{\to} G$, $\theta : G \dot{\to} H$, it is possible to compose them obtaining a new natural transformation $\xi = \theta \circ \eta : F \dot{\to} H$. This follows by the fact that the diagram

$$\begin{array}{ccccc} F(A) & \xrightarrow{\eta_A} & G(A) & \xrightarrow{\theta_A} & H(A) \\ {\scriptstyle F(f)}\downarrow & & \downarrow{\scriptstyle G(f)} & & \downarrow{\scriptstyle H(f)} \\ F(B) & \xrightarrow[\eta_B]{} & G(B) & \xrightarrow[\theta_B]{} & H(B) \end{array}$$

commutes because the two inner squares do. Sticking another diagram on the right of the one above, it is possible to show associativity of composition of natural transformations.

1.2.3 Functor Categories

The Observation 1.2.11 shows that natural transformations recreate on the functors the same structure that morphisms in a category have on objects. This leds us to define a particular kind of category, in which objects are functors between two categories, and arrow are natural transformations.

Definition 1.2.12 (Functor Category)**.** Let $\mathscr{C}$ and $\mathscr{D}$ be categories. The category whose objects are functors between $\mathscr{C}$ and $\mathscr{D}$ and whose arrows are natural transformations between them is said to be a *functor category*, and it is denoted by $[\mathscr{C}, \mathscr{D}]$.

A functor with a small category as domain (Remark 1.1.4) and **Set** as codomain is said to be a *presheaf* on that category. Given a category $\mathscr{C}$, it is possible to construct the functor category of the presheaves on $\mathscr{C}$, i.e. $[\mathscr{C}, \mathbf{Set}]$.

Remark 1.2.13. What we are calling here a presheaf is not totally accurate, because technically a presheaf on a small category $\mathscr{C}$ is a functor $F : \mathscr{C}^{op} \to \mathbf{Set}$. This technicality would bring more complexity, and it is beyond the scope of this work, so we will continue adopting the definition given above.

1.2.4 Comma Categories

Functor constructions allow us to generalise basic concept already seen for categories. An important example of this fact are comma categories, a more general notion of slice categories (Definition 1.1.11).

Definition 1.2.14 (Comma category)**.** Let $\mathscr{C}$, $\mathscr{D}$ and $\mathscr{E}$ be categories, and let $S : \mathscr{C} \to \mathscr{E}$, $T : \mathscr{D} \to \mathscr{E}$ be functors (source and target):

$$\mathscr{C} \xrightarrow{\quad S \quad} \mathscr{E} \xleftarrow{\quad T \quad} \mathscr{D}$$

Then, the *comma category* $(S \downarrow T)$ is the category in which:

- the objects are triples (A, f, B), where $A \in \mathcal{O}b(\mathscr{C})$, $B \in \mathcal{O}b(\mathscr{D})$ and $f : S(A) \to T(B)$ is an arrow of $\mathscr{E}$;
- the arrows are pairs $(c, d) : (A, f, B) \to (C, g, D)$, where $c \in \mathcal{H}om(\mathscr{C})$ and $d \in \mathcal{H}om(\mathscr{D})$, such that the square below commutes;

$$\begin{array}{ccc} S(A) & \xrightarrow{\quad f \quad} & G(B) \\ \Big\downarrow{\scriptstyle S(c)} & & \Big\downarrow{\scriptstyle T(d)} \\ T(C) & \xrightarrow[\quad g \quad]{} & T(B) \end{array}$$

- composition of morphisms is obtained via pairwise composition, i.e., $(a, b) \circ (c, d) = (a \circ c, b \circ d)$.

Thus, the slice category $\mathscr{C}/X$ is the comma category given by the two functors $Id_{\mathscr{C}}$ (the identity functor), and the functor $!_X : \mathbf{1} \to \mathscr{C}$, where $\mathbf{1}$ is the one-object category defined in Example 1.1.2, and $!_X$ sends the only object of $\mathbf{1}$ to X (then the only morphism of $\mathbf{1}$ to id_X of $\mathscr{C}$):

$$\mathscr{C} \xrightarrow{\quad Id_{\mathscr{C}} \quad} \mathscr{C} \xleftarrow{\quad !_X \quad} \mathbf{1}$$

It is easy to see that $(Id_{\mathscr{C}} \downarrow !_X)$ is exactly the same of $\mathscr{C}/X$.

In the same way, it is possible to define coslice categories in terms of comma categories: the category $(!_X \downarrow Id_{\mathscr{C}})$ is exactly the coslice $X/\mathscr{C}$.

1.3 Universal Constructions

1.3.1 Initial and Terminal Objects

The next definitions are about *universal constructions*. The simplest ones are the notion of initial and, dually, terminal objects.

Definition 1.3.1 (Initial and terminal object)**.** An object A of a category $\mathscr{C}$ is said to be *initial* if, for each other object B of $\mathscr{C}$, there exists a unique morphism from A to B. Dually, an object Z is said to be a terminal object in a category $\mathscr{C}$ if, for any other object X of $\mathscr{C}$, there exists a unique morphism from X to Z. An initial object is indicated by the symbol $\mathbf{0}$, and a terminal object by the symbol $\mathbb{1}$.

Observation 1.3.2. It makes sense to refer to an initial (and terminal) object as *the* initial (*the* terminal) object. Suppose that $\mathbf{0}$ and $\mathbf{0}'$ are distinct initial objects of a category $\mathscr{C}$. Then, there exists a unique morphism from $\mathbf{0}$ to $\mathbf{0}'$, say f. Likewise, it must exist a unique morphism from $\mathbf{0}'$ to $\mathbf{0}$, say g. Then, $g \circ f$ must be exactly $id_{\mathbf{0}}$, and $f \circ g = id_{\mathbf{0}'}$, and then they are isomorphic. The same argument works for the terminal object.

Example 1.3.3. In **Set**, the initial object is the empty set $\varnothing$, because, for each set S, there exists the empty function from $\varnothing$ to S. The terminal object of **Set** is the singleton $\{\bullet\}$, because there is exactly one function from a set S to $\{\bullet\}$, namely, the function which sends each $s \in S$ to $\bullet$. It is possible to visualize the Observation 1.3.2: given two singletons $\{\bullet\}$ and $\{\star\}$, the function between them is bijective.

We now illustrate a result on functor categories (Definition 1.2.12) that will be useful later.

Proposition 1.3.4. *Let $\mathscr{D}$ be a category. If $\mathscr{D}$ has an initial object, then, for any category $\mathscr{C}$, $[\mathscr{C}, \mathscr{D}]$ has an initial object. If $\mathscr{D}$ has a terminal object, then, for any category $\mathscr{C}$, $[\mathscr{C}, \mathscr{D}]$ has a terminal object.*

Proof. Let $\mathbf{0}_{\mathscr{D}}$ be the initial object of $\mathscr{D}$, and consider the constant functor $I(f) = id_{\mathbf{0}_{\mathscr{D}}}$ for all $f \in \mathcal{H}om(\mathscr{C})$. Then, for any $G : \mathscr{C} \to \mathscr{D}$, $\eta : I \to G$, defining η_A as the *unique morphism from $\mathbf{0}_{\mathscr{D}}$ to $G(A)$* for each $A \in \mathcal{O}b(\mathscr{C})$, is a natural transformation $I \dot{\to} G$, as the diagram below shows:

$$\begin{array}{ccc} I(A) = \mathbf{0}_{\mathscr{D}} & \xrightarrow{\eta_A} & G(A) \\ \Big\downarrow{\scriptstyle I(f) = id_{\mathbf{0}_{\mathscr{D}}}} & & \Big\downarrow{\scriptstyle G(f)} \\ I(A') = \mathbf{0}_{\mathscr{D}} & \xrightarrow[\eta_{A'}]{} & G(A') \end{array}$$

for each $f : A \to A'$, the square above must commute, since there is only one morphism from $\mathbf{0}_{\mathscr{D}}$ to $G(A')$. For the same reason, η is the only natural transformation from I to G, being indeed the initial object of $[\mathscr{C}, \mathscr{D}]$.

Defining $T(f) = id_{\mathbb{1}_{\mathscr{D}}}$ for each $f \in \mathcal{H}om(\mathscr{C})$. Then, $\theta : F \to T$, for any $F : \mathscr{C} \to \mathscr{D}$, defining θ_A as the *unique morphism from* $F(A)$ *to* $\mathbb{1}_{\mathscr{D}}$ is a natural transformation due to the commutativity of the following diagram for each $f : A \to A'$:

$$\begin{array}{ccc} F(A) & \xrightarrow{\theta_A} & T(A) = \mathbb{1}_{\mathscr{D}} \\ \Big\downarrow{\scriptstyle F(f)} & & \Big\downarrow{\scriptstyle T(f) = id_{\mathbb{1}_{\mathscr{D}}}} \\ F(A') & \xrightarrow[\theta_{A'}]{} & T(A') = \mathbb{1}_{\mathscr{D}} \end{array}$$

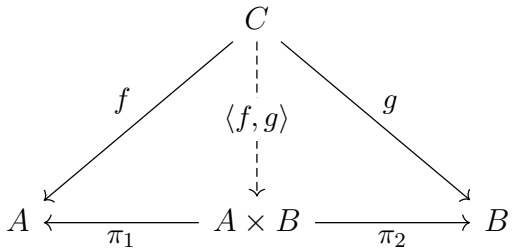
Hence, θ is the unique natural transformation from F to T, and T is the terminal object of $[\mathscr{C}, \mathscr{D}]$. □

In particular, every presheaf has an initial and a terminal object, because **Set** do (Example 1.3.3).

1.3.2 Products and Coproducts

More complex constructions are products (and, dually, coproducts).

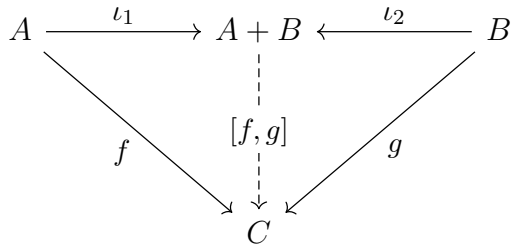
Definition 1.3.5 (Product)**.** A *product* of two objects A an B is an object $A \times B$ together with two *projection arrows* $\pi_1 : A \times B \to A$ and $\pi_2 : A \times B \to B$ such that, for every object C and pair of arrows $f : C \to A$, $g : C \to B$, there is exactly one arrow $\langle f, g \rangle : c \to A \times B$ making the diagram commutes



– i.e., such that $\pi_1 \circ \langle f, g \rangle = f$ and $\pi_2 \circ \langle f, g \rangle = g$.

Definition 1.3.6 (Coproduct)**.** The dual of the product is the *coproduct*. A coproduct of two objects A and B is an object $A + B$ together with two arrows $\iota_1 : A \to A + B$, $\iota_2 : B \to A + B$ such that, for every object C and pair of arrows $f : A \to C$, $g : B \to C$, there is a unique arrow $[f, g] : A + B \to C$ such that the

diagram commutes



– i.e., such that $[f, g] \circ \iota_1 = f$ and $[f, g] \circ \iota_2 = g$.

Example 1.3.7. Set has both products and coproduts. Given two sets A and B, the categorical product is the set-theoretic cartesian product $A \times B$, together with the two projections π_A and π_B, while the coproduct is the disjount sum $A \coprod B = \{(x, 0) \mid x \in A\} \cup \{(y, 1) \mid y \in B\}$, together with the two canonical injections ι_A and ιB, where $\iota_A(a) = (a, 0)$ and $\iota_B(b) = (b, 1)$.

The notions of product and coproduct can be easily generalized, extending the definition to the product (and coproduct) of a family of objects, together with appropriate arrows (e.g., the projection arrows for each object in the product). We will denote the product of a collection of objects indexed by a (finite) category $\mathscr{I}$ as $\left(\prod_{i \in \mathcal{O}b(\mathscr{I})} X_i, (\pi_i)_{i \in \mathcal{O}b(\mathscr{I})}\right)$, and the coproduct as $\left((\iota_i)_{i \in \mathcal{O}b(\mathscr{I})}, \coprod_{i \in \mathcal{O}b(\mathscr{I})} X_i\right)$.

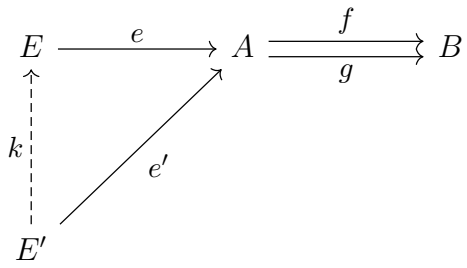
Again, the definition of these constructions is divided into two parts: one stating what the construction is, and another stating that the construction satifies the universal property.

1.3.3 Equalizers and Coequalzers

The next notion is about a construction that make parallel arrows (i.e., two possibly distinct morphisms with the same domain and codomain) equal. In **Set**, given two functions $f, g : A \to B$, such construction corresponds exactly to the restriction of the domain to the set of elements on which f and g are equal. Specifically, if we take the set $E = \{x \in A \mid f(x) = g(x)\}$, and we take the function $e : E \to A$ defined by $e(x) = x$, we obtain what is called an *equalizer* for f and g. In fact, e is that function that make f and g be the same function – i.e., $f \circ e = g \circ e$. This concept can be generalized as follows.

Definition 1.3.8 (Equalizer and Coequalizer)**.** Let $f, g : A \to B$ be two arrows of a category $\mathscr{C}$. An *equalizer* for f and g is pair (E, e), where E is an object and $e : E \to A$ is an arrow in $\mathscr{C}$ such that:

1. $f \circ e = g \circ e$; and
2. if (E', e') is another pair that satisfies 1, then there exists a unique $h : E' \to E$ such that $e \circ h = e'$.



A *coequalizer* of f and g, dually, is a pair (c, C), where C is an object and $c : B \to C$ such that $c \circ f = c \circ g$, with the universal property.

Proposition 1.3.9. *Let $e : E \to A$ be the arrow that equalizes $f, g : A \to B$ in a category $\mathscr{C}$. Then, e is a monomorphism.*

Proof. Suppose X be an object and $x, y : X \to E$ be two morphisms in $\mathscr{C}$ such that $e \circ x = e \circ y$, and let $z = e \circ x$. Then, since e is an equalizer, $f \circ e = g \circ e$, and $f \circ z = g \circ z$. But, for the universal property of equalizers, there must be exactly one $u : Z \to E$ such that $z = e \circ u$. It follow that $x = u$ and $y = u$, hence $x = y$. □

Of all monomorphism, an interesting subclass of them is the one that contains only the equalizers.

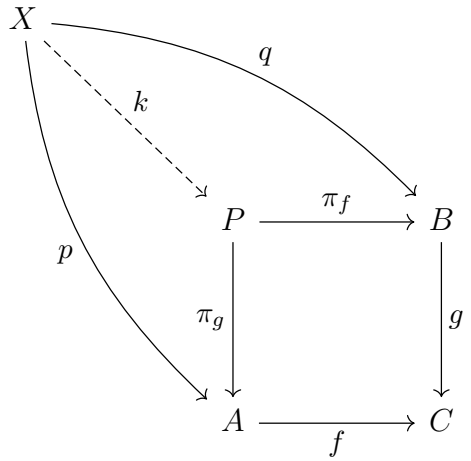
Definition 1.3.10 (Regular Monomorphism)**.** A monomorphism that is an equalizer for a pair of arrows is said *regular monomorphism*. The class of all regular monomorphisms of a category $\mathscr{C}$ is denoted $\mathcal{R}eg(\mathscr{C})$.

1.3.4 Pullbacks and Pushouts

Given two arrows, another pair of constructions one can find in a category is given by pullbacks and pushouts.

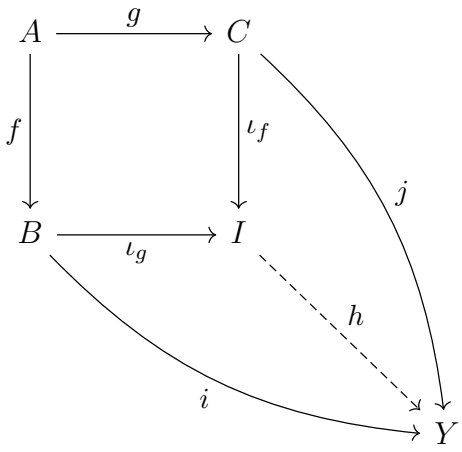
Definition 1.3.11 (Pullback, Pushout)**.** A *pullback* of two arrows $f : A \to C$ and $g : B \to C$ is a triple (P, π_g, π_f) containing an object P and a pair of arrows $\pi_g : P \to A$, $\pi_f : P \to B$ such that:

1. $f \circ \pi_g = g \circ \pi_f$; and
2. if (X, p, q), where $p : X \to A$ and $q : X \to B$, satisfies 1, then there is a unique $k : X \to P$ such that $p = \pi_g \circ k$ and $j = \pi_f \circ k$.



Dually, the *pushout* of two arrows $f : A \to B$ and $g : A \to C$ is a triple (ι_g, ι_f, I), where I is an object and $\iota_g : B \to I$ and $\iota_f : C \to I$ are morphisms such that:

1. $\iota_g \circ f = \iota_f \circ g$; and
2. if (i, j, Y), where $i : B \to Y$ and $j : C \to Y$, satisfies 1, then there is a unique $h : I \to Y$ such that $i = h \circ \iota_g$ and $j = h \circ \iota_f$.



Pullbacks (and, dually, pushouts) are a construction that is slightly more general than products and equalizers. An intuition of what they represent is given by considering what is concretely a pullback in the category of sets.

Example 1.3.12. In **Set**, given two functions $f : A \to C$ and $g : B \to C$, a pullback of f and g exists and is exactly the set $P = \{(x, y) \in A \times B \mid f(x) = g(y)\}$, with $\pi_f : P \to B$ and $\pi_g : P \to C$ defined, respectively, by $\pi_f((x, y)) = y$ and $\pi_g((x, y)) = x$. In this way, we have then, $\forall (x, y) \in P$:

$$\begin{aligned}
(f \circ \pi_g)((x, y)) &= f(\pi_g((x, y))) \\
&= f(x) && \textit{Definition of } \pi_g \\
&= g(y) && (x, y) \in P \\
&= g(\pi_f((x, y))) && \textit{Definition of } \pi_f \\
&= (g \circ \pi_f)((x, y))
\end{aligned}$$

thus, $f \circ \pi_g = g \circ \pi_f$.

Another important example to our aims is a concrete definition of what is a pushout in the category of sets, and why morally we can regard a pushout as *the way to identify part of an object with a part of another* [BW95].

Example 1.3.13. In **Set**, given two functions $f : A \to B$ and $g : A \to C$, the pushout of them is the set $X = (B \coprod C)/_\sim$, where $\sim$ is the least equivalence relation such that $f(a) \sim g(a)$ for each $a \in A$, with $\iota_g : B \to X$ and $\iota_f : C \to X$ as arrows, sending each element of the domain in the corresponding equivalence class in X. In

particular, for each $a \in A$:

$$\begin{aligned}
(\iota_g \circ f)(a) &= \iota_g(f(a)) \\
&= [(f(a), 0)] && \textit{Definition of } \iota_g \\
&= [(g(a), 1)] && f(a) \sim g(a) \\
&= \iota_f(g(a)) && \textit{Definition of } \iota_f \\
&= (\iota_f \circ g)(a)
\end{aligned}$$

When both f and g are monos (that is, injections), then we can construct the pushout taking at the same way we have done above, with $(f(a), 0) \sim (g(a), 1)$ when such a exists and $(b, 0) \sim (c, 1)$ on each b and c with no preimage in A, with ι_f and ι_g injective. An easy way to see this fact is considering the following situation: let $f : A \to A \cup B$ and $g : A \to A \cup C$, with A disjoint from B and C, $f(a) = a$ and $g(a) = a$. Then the pushout is the object $A \cup B \cup C$, with the inclusion as arrows, that are also injective. A more general case is what happens considering functions $f : A \to B$ and $g : A \to C$ injective. Differently from the previous example, in this case is not possible to take just the union of codomains as the pushout, but rather the disjoint union of them and then identify the elements $f(a)$ with $g(a)$, as we have done above. In the category of sets and functions, we have the certainty that the pullback arrows are injective. In fact, taking the equivalence relation $\sim$, we have that $f(a) \sim f(a')$ if and only if $a = a'$ by hypothesis, and then $x \sim x'$ if and only if $x = x'$, then the pushout morphisms sends each element in an equivalence class composed only by the element itself, thus are injective. This is an interesting property that in other categories may do not hold, and will be recalled later.

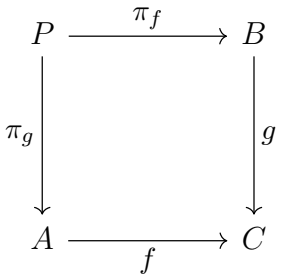
Given a subclass of morphisms of a category, an important property is *stability* under certain type of constructions. In our case, we are interested in stability under pullbacks and under pushouts.

$$\begin{array}{ccc}
A & \xrightarrow{f} & B \\
{\scriptstyle m}\downarrow & & \downarrow{\scriptstyle n} \\
C & \xrightarrow[g]{} & D
\end{array} \qquad (*)$$

Definition 1.3.14. [Stability under pullbacks, pushouts] Given a category $\mathscr{C}$, a subclass $\mathcal{A} \subseteq \mathcal{H}om(\mathscr{C})$ is said to be *stable under pullbacks* if, for every pullback square as the one in $(*)$, if $n \in \mathcal{A}$, then $m \in \mathcal{A}$. $\mathcal{A}$ is said to be *stable under pushouts* if, for every pushout square as the one in $(*)$, if $m \in \mathcal{A}$, then $n \in \mathcal{A}$.

Proposition 1.3.15. *Let $f : A \to C$, $g : B \to C$ arrows in any category $\mathscr{C}$, and*

consider the pullback square:



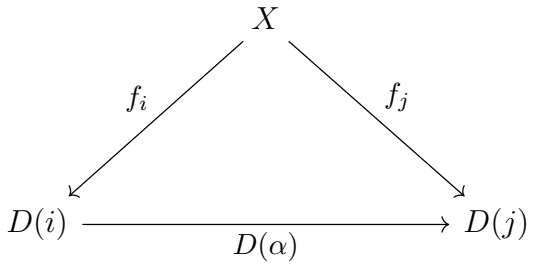
If g is mono, then so is π_g.

1.3.5 Limits and Colimits

We now introduce the notion that generalize all the universal constructions defined above. In fact, initial objects, products, equalizers an pullback (dually, terminal objects, coproducts, coequalizers and pushouts) can be seen as the particular case of a certain type of construction, called limit.

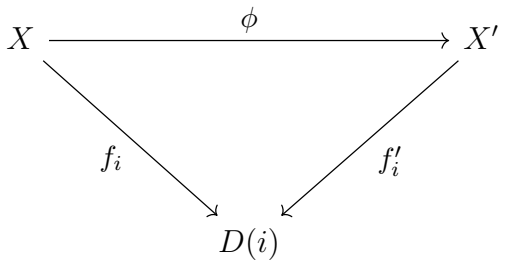
To define what a limit is, we first need to define cones.

Definition 1.3.16 (Cones)**.** Let $D : \mathscr{I} \to \mathscr{C}$ be a diagram in $\mathscr{C}$ of shape $\mathscr{I}$. A *cone* for D is an object X of $\mathscr{C}$, together with arrows $f_i : X \to D(i)$ indexed by $\mathscr{I}$ (i.e. one for each object i of $\mathscr{I}$), such that, for each morphism $\alpha : i \to j$ of $\mathscr{I}$, the following diagram commutes:

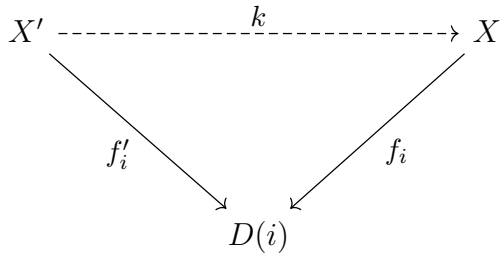


– i.e., $D(\alpha) \circ f_i = f_j$. We denote such cone as $\{f_i : X \to D(i)\}$.

Observation 1.3.17. Given a diagram D, the category of the cones for D, denoted **Cone**(D), is defined to have cones for D as objects and cone morphisms as arrows, where a cone morphism $\phi : C \to C'$ from $C = \{f_i : X \to D(i)\}$ to $C' = \{f'_i : X' \to D(i)\}$ is a morphism $\phi : X \to X'$ such that the following diagram commutes for each i:



Definition 1.3.18 (Limits)**.** Let $D : \mathscr{I} \to \mathscr{C}$ be a diagram in $\mathscr{C}$ of shape $\mathscr{I}$. A cone $\{f_i : X \to D(i)\}$ is a *limit* provided that, for any other cone $\{f'_i : X' \to D(i)\}$ for D, then there exists a unique morphism $k : X' \to X$ such that the following diagram commutes for each object i of $\mathscr{I}$:



– i.e., $f_i \circ k = f'_i$ for each object i of $\mathscr{I}$. Such limit is denoted as $(X, f_i)_{i \in \mathscr{I}}$

Observation 1.3.19. Given a diagram D, a limit for D is exactly the terminal object of the category $\mathbf{Cone}(D)$, defined in Observation 1.3.17.

The dual notions of cones and limits are that of cocones and colimits.

Definition 1.3.20. (Cocones, Colimits) A *cocone* for a diagram $D : \mathscr{I} \to \mathscr{C}$ is an object Y of $\mathscr{C}$ together with arrows $f_i : D(i) \to Y$ such that, for each $g : D(i) \to D(j)$ of $\mathscr{C}$, $f_j \circ g = f_i$. A cocone is denoted $\{f_i : D(i) \to Y\}$. A *colimit* for D is a cocone $C = \{f_i : D(i) \to Y\}$ with the universal property – i.e., if $C' = \{f'_i : D(i) \to Y'\}$ is another cone for D, then there exists a unique arrow $h : Y \to Y'$ such that, for each i, $h \circ f_i = f'_i$.

The following examples show how limits are general concepts for the constructions defined above.

Example 1.3.21. Let D be the empty diagram in the category $\mathscr{C}$. Then, a cone for D is any object of $\mathscr{C}$, while a limit is the terminal object of $\mathscr{C}$.

Example 1.3.22. Let D be the following diagram:

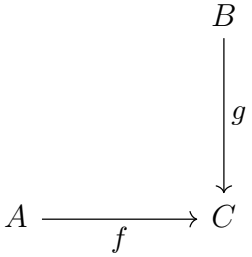
$$A \qquad\qquad B$$

Then, a cone for D is an object X and two arrows $f : X \to A$, $g : X \to B$ (i.e., a span, defined in Example 1.2.4):

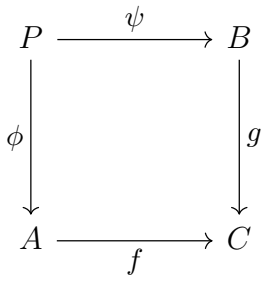
$$A \xleftarrow{\ f\ } X \xrightarrow{\ g\ } B$$

If it exists, a limit for D is the product of A and B.

Example 1.3.23. A pullback is the limit for the diagram below.



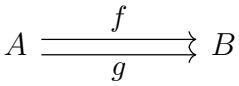
In fact, a cone for the diagram above is an object P and three arrows $\phi : P \to A$, $\psi : P \to B$ and $h : P \to C$, but the latter is uniquely determined by the other ones ($f \circ \phi = h = g \circ \psi$). Thus, the following diagram is a cone:



For (P, ϕ, ψ) to be a pullback, it must have the universal property. In other words, it has to be a limit.

This example show us that a pullback is a limit for a cospan (Example 1.2.4).

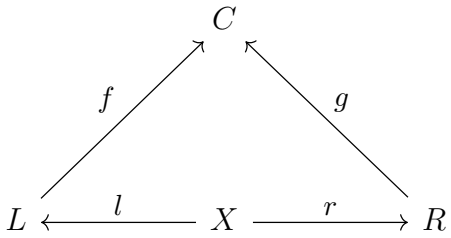
Example 1.3.24. A limit for the diagram below is an equalizer for f and g.



All the examples we provided about limits are still valid, in their dual form, for what regard colimits. In fact, initial objects, coequalizers, coproducts and pullbacks are particular cases of limits, and the way to see this fact is the same seen in previous examples.

Example 1.3.25. Given a span $S = (l, X, r) : L \rightharpoonup R$, shown in the diagram below,

a cocone for S is any commutative square of the form



(the morphism $X \to C$ is uniquely determined by the relation $f \circ l = g \circ r$). A colimit for S is then a pushout of l and r.

The connection between constructions as products and equalizers and limits is made clear by the following theorem. The idea behind the proof is the fact that, given a diagram $D : \mathscr{I} \to \mathscr{C}$, if each subset of objects $X = \{D(i) \mid i \in \mathcal{O}b(\mathscr{I})\} \subseteq \mathcal{O}b(\mathscr{C})$ has a product $(\prod_{i \in I} D(i), (\pi_i)_{i \in \mathcal{O}b(\mathscr{I})})$ and each pair of arrows $f, g \in \mathscr{C}(D(i), D(j))$ has an equalizer $Eq(f, g)$, then one can construct the cone taking the equalizer of the arrows that has as domain the product of the objects of the diagram, and as codomain the product of the codomains of the arrows of the diagram. This construction has the universal property because equalizers and products do. A detailed proof is in the appendix.

Theorem 1.3.26. *[Limit theorem] Let $\mathscr{C}$ be a category. Then $\mathscr{C}$ has all finite limits if and only if $\mathscr{C}$ has all finite products and all finite equalizers.*

Remark 1.3.27. The theorem above (and its relative proof) can be stated in its dual form leading to a theorem on existence of colimits, and a relative criterion to calculate them (taking the dual of the proof).

Example 1.3.28. Limit theorem gives us an easy way to calculate limits. An example of this fact is how limits are computed in **Set**. Given a diagram $D : \mathscr{I} \to \mathbf{Set}$, where $\mathscr{I}$ is a small category and $I = Ob(\mathscr{I})$, its limit is the set L defined as follows:

$$L = \{(d_i)_{i\in I} \in \prod_{i\in I} D(i) \mid \forall \phi \in \mathscr{I}(i,i'), D(\phi)(d_i) = d_{i'}\}$$

with projections as arrows.

Example 1.3.29. As we have done in Example 1.3.28 we illustrate how to construct colimits in the category of sets. Given a small category $\mathscr{I}$, $I = Ob(\mathscr{I})$, and a diagram $D : \mathscr{I} \to \mathbf{Set}$, consider the equivalence relation $\sim$ defined on $\coprod_{i\in I} D(i)$ such that $d_i \sim d_{i'}$ if $d_i \in D(i), d_{i'} \in D(i')$ and there exists some $\phi \in \mathscr{I}(i,i')$ such that $D(\phi)(d_i) = d_{i'}$. Then, a colimit for D is the set

$$C = \big(\coprod_{i\in I} D(i)\big)/\sim$$

with the inclusions as arrows.

Remark 1.3.30. Since a diagram is nothing more than a functor from a "shape" category to another, it makes sense to talk about limits of functors in general, even when they are not intended to be diagrams.

Observation 1.3.31. So far we introduced categories of presheaves. In these categories, an interesting fact is that limits and colimits are computed pointwise – i.e., the limit of a diagram in a category of presheaves is exactly the limit on each of its components.

In the next sections, we will work on a special kind of diagrams with certain properties. In particular, we are interested in how a functor behaves with respect to the constructions defined so far.

Definition 1.3.32. Let $D : \mathscr{I} \to \mathscr{C}$ be a diagram, and $F : \mathscr{C} \to \mathscr{D}$ a functor. We say that F:

1. *preserves limits* of D if, given a limit $(L, l_i)_{i\in\mathscr{I}}$ for D, then $(F(L), F(l_i))_{i\in\mathscr{I}}$ is a limit for $F \circ D$.

2. *reflects limits* of D if a cone $(L, l_i)_{i\in\mathscr{I}}$ is a limit for D whenever $(F(L), F(l_i))_{i\in\mathscr{I}}$ is a limit for $F \circ D$.

3. *lifts limits (uniquely)* of D if, given a limit $(L, l_i)_{i\in\mathscr{I}}$ for $F \circ D$, there exists a (unique) limit $(L', l'_i)_{i\in\mathscr{I}}$ for D such that $(F(L'), F(l'_i))_{i\in\mathscr{I}} = (L, l_i)_{i\in\mathscr{I}}$.

4. *creates limits* of D if D has a limit and F preserves and reflects limits along it.

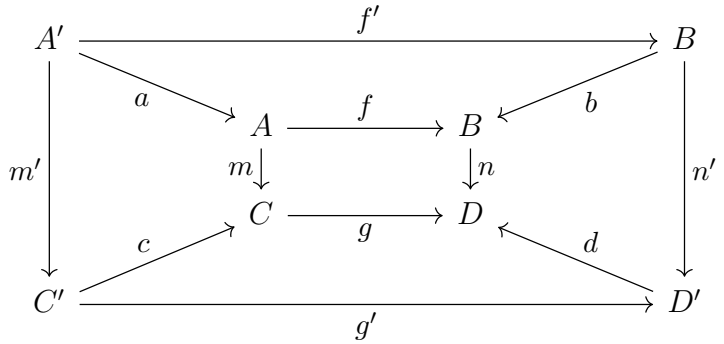
The dual notions are obtained in the obvious way, namely, substituting the words "limits" and "cones" with "colimits" and "cocones", respectively

Observation 1.3.33. It holds that if a functor creates limits, then lifts uniquely limits [AHS09].

1.4 Adhesivity

The next section is about adhesivity. An adhesive category is intuitively a category in which pushouts of (some) monomorphisms exist and they behave more or less as they do among sets.

Definition 1.4.1. (Van Kampen property) Let $\mathcal{A}$ be a subclass of $\mathcal{H}om(\mathscr{C})$, and consider the diagram below:



we say that the inner square is an $\mathcal{A}$-*Van Kampen* square if:

- it is a pushout;
- $a, b, c, d \in \mathcal{A}$;
- whenever the top and the left squares are pullbacks then the outer square is a pushout if and only of the right and the bottom squares are pullbacks.

We are now ready to give the notion of $\mathcal{M}$-adhesivity.

Definition 1.4.2. [$\mathcal{M}$-adhesivity] Let $\mathscr{C}$ be a category and $\mathcal{M} \subseteq \mathcal{M}ono(\mathscr{C})$ containing all isomorphisms, closed under composition and stable under pullbacks and pushouts (Definition 1.3.14). Then $\mathscr{C}$ is $\mathcal{M}$-*adhesive* if

1. every cospan $C \xrightarrow{g} D \xleftarrow{m} B$ with $m \in \mathcal{M}$ can be completed to a pullback (such pullbacks are called $\mathcal{M}$-pullbacks);
2. every span $C \xleftarrow{m} A \xrightarrow{f} B$ with $m \in \mathcal{M}$ can be completed to a pushout (such pushouts are called $\mathcal{M}$-pushouts);
3. pushouts along $\mathcal{M}$-arrows are $\mathcal{M}$-Van Kampen squares.

We also say that $\mathscr{C}$ is *adhesive* when it is $\mathcal{M}ono(\mathscr{C})$-adhesive, and *quasiadhesive* when it is $\mathcal{R}eg(\mathscr{C})$-adhesive.

Verifying $\mathcal{M}$-adhesivity using the definition above may turn out to be vary complex, so we can make use of the following result [CGM22].

Theorem 1.4.3. *Let $\mathscr{C}$ be a category, $\mathcal{M} \subseteq \mathcal{M}ono(\mathscr{C})$ containing all isomorphisms, closed under composition and stable under pullbacks and pushouts. Let now $F : \mathscr{C} \to \mathscr{D}$ be a functor with $\mathscr{D}$ $\mathcal{N}$-adhesive for some $\mathcal{N} \subseteq \mathcal{M}ono(\mathscr{D})$. If F is such that $F(\mathcal{M}) \subseteq \mathcal{N}$ and creates pullbacks and $\mathcal{M}$-pullbacks, then $\mathscr{C}$ is $\mathcal{M}$-adhesive.*

The idea behind this theorem is to simplify calculation to show that a certain category is adhesive for some subclass of monomorphisms, considering a functor from the category of which we want to prove adhesivity to a category we know it is adhesive, requiring that such functor has some properties.

Proof. In order to prove $\mathcal{M}$-adhesivity of $\mathscr{C}$, we have to verify the condition in Definition 1.4.2.

- Let $C \xrightarrow{g} D \xleftarrow{m} B$ with $m \in \mathcal{M}$ be a cospan in $\mathscr{C}$.Applying F, we obtain $F(C) \xrightarrow{F(g)} F(D) \xleftarrow{F(m)} B$, with $F(m) \in \mathcal{N}$ by hypothesis. Then, there exists a pullback $(P_F, p_{F(B)}, p_{(F(D))})$ in $\mathscr{D}$, which is an $\mathcal{N}$-pullback (Example 1.3.23). Since F creates pullbacks, hence lift them (Observation 1.3.33), there exist a pullback (P, p_B, p_D) in $\mathscr{C}$.

- Let $C \xleftarrow{m} A \xrightarrow{f} B$ with $m \in \mathcal{M}$ be a cospan in $\mathscr{C}$. Analogously to the previous point, applying the functor F we obtain $F(C) \xleftarrow{F(m)} F(A) \xrightarrow{F(f)} F(B)$ with $F(m) \in \mathcal{N}$, and there exists a $\mathcal{N}$-pushout $(q_{F(C)}, q_{F(B)}, F(Q))$ in $\mathscr{D}$. Since F reflects pushouts, (q_C, q_B, Q) is a $\mathcal{M}$-pushout in $\mathscr{C}$.

- the Van Kampen property of $\mathcal{M}$-pullbacks follows from the closure under pullbacks and pushouts of $\mathcal{M}$ and from the fact that F reflects pullbacks.

□

[DC]Alcuni commenti su questo capitolo. Non sono soddisfatto dall'ordine della presentazione. Io personalmente proporrei il seguente indice:

Sec. 1 Categorie e funtori

>def di categorie -menzionare che data una categoria c'è il duale -epi, mono e iso

>funtori e trasf naturali

>categorie comma (con slice e coslice come sottocaso)

Sec. 2: Limiti e colimiti Io sinceramente questa la riscriverei completamente. La definizione di (co)limite deve venire prima e solo dopo deve venire l'elenco degli esempi. Struttura che propongo:

>definizione, esempi e se avanza tempo come si calcolano (co)limiti generali a partire da quelli "base" (tipo prodotti da pb e equalizzatori) -epi e mono regolari

>Limiti nelle categorie di funtori e comma

Sec. 3: Adesività Unica osservazione che ho per ora è che terrei il cubo invece del diagramma che hai fatto

Chapter 2

Categories of Graphs

This chapter is about graphs, and how it is possible to formalize them using categories in order to point out their properties from an abstract point of view. Starting from the set-theoretical definition of graphs, we will give an abstraction via functor categories, in which a graph is nothing but a functor between a category to another.

2.1 Set-Theoretic Definitions of Graphs

First obvious definitions of graph are given via sets and set-theoretic tools, remarking what intuitively graphs are. We present in this section a few classes of graphs.

2.1.1 Graphs

Definition 2.1.1 (Graph)**.** A *graph* G is a tuple (V, E, s, t), where V and E are sets (respectively, the set of vertices, or nodes, and the set of edges, or arcs), and $s, t : E \to V$ are two functions (respectively, the source and the target of an edge). In general, given a graph G, we write $V(G)$ to denote the set of vertices and $E(G)$ the set of edges of G.

Definition 2.1.2. [Graph Homomorphism] Given two graphs $G = (V_G, E_G, s_G, t_G)$ and $H = (V_H, E_H, s_H, t_H)$, a *graph homomorphism* $h : G \to H$ is a pair of functions $h = (h_V : V_G \to V_H, h_E : E_G \to E_H)$ such that

$$h_V \circ s_G = s_H \circ h_E$$

and

$$h_V \circ t_G = t_H \circ h_E$$

that is, a structure preserving map. Given two graph homomorphisms $h = (h_V, h_E) : G \to H$, $k = (k_V, k_E) : H \to I$, the homomorphism $k \circ h = (k_V \circ h_V, k_E \circ h_E) : G \to I$ is the *composite* of h and k.

Graphs together with graph homomorphisms from a category.

Definition 2.1.3 (Category of Graphs)**. Graph** is the category in which objects are graphs and arrows are graph homomorphisms.

All the constructions defined in Section 1.3 exists in **Graph**, and they are very similar to construction in **Set**, intuitively because of the set-theoretic nature of graphs. The next examples try to make this point clear.

Example 2.1.4. The initial object in **Graph** is the empty graph, i.e., the graph with an empty set of vertices and an empty set of edges. The initial object instead is the graph with exactly one node and a single edge from that node to itself.

Example 2.1.5. Given two graphs $G = (V_G, E_G, s_G, t_G)$ and $H = (V_H, E_H, s_H, t_H)$, the graph $G \times H = (V_G \times V_H, E_G \times E_H, (s_G, s_H), (t_G, t_H))$, where $(s_G, s_H), (t_G, t_H) : V_G \times V_H \to E_G \times E_H$ are the pairwise sources and targets, is the categorical product in **Graph**, together with the two projections $\pi_G : G \times H \to G$, $\pi_H : G \times H \to H$ defined in the obvious way.

Example 2.1.6. The equalizer of two morphisms $h, k : G \to H$ in **Graph** is defined as in **Set**, that is, a graph Q together with a graph morphism q that is the restriction of G to all the vertices and all the arcs that are mapped on the same vertices and edges both from h and k. Formally, (Q, q) is an equalizer for $h, k : G \to H$, $h = (h_V, h_E), k = (k_V, k_E)$ where $V(Q) = \{n \in V(G) \mid h_V(n) = k_V(n)\}$, $E(Q) = \{e \in E(G) \mid h_E(e) = k_E(e)\}$, $s_Q = s_G \mid_{V(Q)}$, $t_E = t_G \mid_{V(Q)}$.

2.1.2 Graph with Equivalences

It is possible to endow the set of vertices of a graph with any sort of relation, requiring that such relation is preserved by homomorphisms. The ones we are interested in are *equivalence relations*. Recall that an equivalence relation R over a set A, $R \subseteq A \times A$, is a relation that is *reflexive* ($\forall a \in A, aRa$), *transitive* ($\forall a, b, c \in A, aRb$ and $bRc \Rightarrow aRc$) and *symmetric* ($\forall a, b \in A, aRb \Rightarrow bRa$). A graph with equivalence is a graph with such a relation defined over its set of vertices.

Definition 2.1.7 (Graph with Equivalence [BGM06])**.** A *graph with equivalence* is a pair $\mathcal{G} = (G, \sim_G)$ where G is a graph and $\sim_G \subseteq V(G) \times V(G)$ is an equivalence relation over its set of nodes. An homomorphism between two graphs with equivalences $h : \mathcal{G} = (G, \sim_G) \to \mathcal{H} = (H, \sim_H)$ is a graph homomorphism $h = (h_V, h_E) : G \to H$ such that, if $v_1 \sim_G v_2$ then $h_V(v_1) \sim_H h_V(v_2)$. Graphs with equivalences together with their homomorphism from a category, denoted **EqGrph**.

Remark 2.1.8. It is possible to define an equivalence relation $\sim \subseteq S \times S$ as a surjective function mapping each $s \in S$ to a partition of S in which each element is equivalent according to $\sim$, that is, its *equivalence class*. Formally, an equivalence relation $\sim \subseteq S \times S$ is fully described by a function $f_\sim : S \to \{[s]_\sim \mid s \in S\}$, where $[s]_\sim = \{t \in S \mid t \sim s\}$, defined by $f_\sim(s) = [s]_\sim$.

Observation 2.1.9. Graph, defined in Definition 2.1.3 is equivalent to the full subcategory of **EqGrph** where objects are graphs $(G, =)$, i.e., in which each node is equivalent only to itself.

Graphs with equivalences are alternative representation of *quotient graphs* over an equivalence relation. The graph with equivalence $\mathcal{G} = (G, \sim_G)$ is another representation of the graph $G/_{\sim_G}$.

Definition 2.1.10. [Quotient Functor] The *quotient functor* Q : **EqGrph** $\to$ **Graph** is defined as follows:

- on objects, given $\mathcal{G} = (G, \sim_G)$ as $Q(\mathcal{G}) = G/_{\sim_G} = (V/_{\sim_G}, E, s', t')$, where $s'(e) = [v]_{\sim_G}$ if $s(e) = v$, and $t'(e) = [v]_{\sim_G}$ if $t(e) = v$;
- on morphisms, given $h = (h_V, h_E) : \mathcal{G} \to \mathcal{H}$, as $Q(h_V)([v]_{\sim_G}) = [h_V(v)]_{\sim_H}$, and $Q(h_E) = h_E$.

In the category of graphs with equivalences, universal constructions are easy to obtain. The terminal object in **EqGrph** is the graph with only one node, only one edge and with the single node equivalent only to itself, while the initial object is the graph with no nodes and no edges. Given two graphs with equivalences $\mathcal{G} = (G, \sim_G)$ and $\mathcal{H} = (H, \sim_H)$, one can construct the product graph $\mathcal{G} \times \mathcal{H} = (G \times H, \sim_{G\times H})$, together with the two projections π_G and π_H, where $\sim_{G\times H}$ is the least equivalence relation on $G \times H$ such that, if w_1 and w_2 are nodes of $\mathcal{G} \times \mathcal{H}$, we have $w_1 \sim_{G\times H} w_2$ if $\pi_G(w_1) \sim_G \pi_G(w_2)$ and $\pi_H(w_1) \sim_H \pi_H(w_2)$. An equalizer in **EqGrph** for a pair of morphisms $f, g : \mathcal{G} \to \mathcal{H}$ is exactly the same as in **Graph**, namely, $(\mathcal{E} = (E, \sim_E), e)$, where E is the greatest subgraph of the underlying graph of $\mathcal{G}$ on which f and g are equal, together with the inclusion $e : \mathcal{E} \to \mathcal{G}$. The equivalence relation is preserved ($\sim_E$ is $\sim_G$ restricted to the set of vertices of E).

Previous examples show that the category **EqGrph** has both (co)equalizers and finite (co)products, so, by Theorem 1.3.26, we can conclude that **EqGrph** has finite limits and colimits.

An important characterization in the category of graphs with equivalences is the following.

Proposition 2.1.11. *In* **EqGrph***, a monomorphism is regular if it reflects equivalences. That is, if $h : \mathcal{G} \to \mathcal{H}$ is mono and such that $h_V(v_1) \sim_H h_V(v_2) \Rightarrow v_1 \sim_G v_2$, then h is regular mono.*

2.1.3 E-Graphs

E-Graphs are a particular type of graphs with equivalences, in which holds that

$$\frac{s(e) \sim s(e')}{t(e) \sim t(e')}$$

for each pair of edges e, e' of $\mathcal{G} = (G, \sim)$.

2.2 Graphs as Functors

Set-theoretical definitions of graphs are straightforward, but a further level of abstraction allows us to highlight some interesting properties of these objects. We can indeed consider a graph as a functor from a category of "primitive objects" (e.g., the object representing edges and the object representing vertices) onto a category which provides a representation of them. This concept is perfectly represented by

categories of presheaves, where an object is nothing than a way to interpret objects of a category as sets. An example will make clear what we are talking about.

Considering the category of graphs (as defined in Definition 2.1.3), it is possible to show that

$$\mathbf{Graph} \cong [E \overset{s}{\underset{t}{\rightrightarrows}} V, \mathbf{Set}]$$

where an object G is a functor, having $G(V)$ as the set of vertices, $G(E)$ as the set of edges, $G(s)$ and $G(t)$ as the source and target functions. A morphisms $\eta : G \dot{\to} H$ is a natural transformation, and the commutativity of the diagram below is given by definition of graph morphisms (Definition 2.1.2)

$$\begin{array}{ccc} G(E) & \xrightarrow{\eta_E} & H(E) \\ {\scriptstyle G(s)}\downarrow & & \downarrow{\scriptstyle H(s)} \\ G(V) & \xrightarrow[\eta_V]{} & H(V) \end{array}$$

(and the same for t).

Example 2.1.4 and following, in which limits and colimits are computed pointwise, are a concrete instance of Observation 1.3.31.

More complex objects are graphs with equivalence. We can think of **EqGrph** as a subcategory of the category $[V \rightrightarrows E \to Q, \mathbf{Set}]$, in which each object $\mathcal{G}$ is a functor mapping the arrow $q : E \to Q$ onto an epimorphism (i.e., a surjective function). The component $\mathcal{G}(q)$ here is intended to be the function discussed in Remark 2.1.8. In the following, we will refer with **Graph** and **EqGrph** to the category of presheaves we have just introduced.

Regular monos in **EqGrph** are then natural transformations $\eta : \mathcal{G} \dot{\to} \mathcal{H}$ such that η_Q is mono (Proposition 2.1.11). Since limits are computed pointwise, pullbacks preserve regular monomorphisms (Proposition 1.3.15). Regular monos are preserved also by pushouts. Again, since (co)limits are computed pointwise, and in **Set** monos are preserved by pushouts (since **Set** is adhesive), we can conclude what follows.

Lemma 2.2.1. *In* **EqGrph**, $\mathcal{R}eg(\mathbf{EqGrph})$ *is stable under pushouts and pullbacks (in the sense of Definition 1.3.14).*

Theorem 2.2.2. EqGrph *is quasiadhesive.*

Proof. In order to apply Theorem 1.4.3, we can consider the quotient functor defined in Definition 2.1.10 Q : **EqGrph** $\to$ **Graph**. We note that Q creates limits, and that regular monos in **EqGrph** are mappend onto monos in **Graph**. In addition to Lemma 2.2.1, we can conclude that **EqGrph** is $\mathcal{R}eg(\mathbf{EqGrph})$-adhesive. □

alizzare il
he i limiti
Grph ri-
ono limiti in
h

**

Appendix A

Omitted Proofs

Theorem 1.3.26. Let $\mathscr{C}$ be a category. Then $\mathscr{C}$ has all finite limits if and only if $\mathscr{C}$ has all finite products and all finite equalizers.

Proof. Let $D : \mathscr{I} \to \mathscr{C}$ a diagram, with $\mathscr{I}$ finite.

The *if* statement is easily satisfied as shown in Example 1.3.24 and Example 1.3.22.

To satisfy the *only if* statement, we want an object L together with morphisms $p_i : L \to D(j)$ such that:

1. $\{p_i : L \to D(i)\}$ is a cone – i.e., for each morphism of $\mathscr{I}$ $\alpha : i \to j$, $D(\alpha) \circ p_i = p_j$; and
2. for each E and $q_i : E \to D(j)$ in $\mathscr{C}$, with $D(\alpha) \circ q_i = q_j$ for each $\alpha : i \to j$ of $\mathscr{I}$, there exists a unique $f : E \to L$ such that $q_i = p_i \circ f$ for each $i \in \mathcal{O}b(\mathscr{I})$.

Consider the two products (which exist by hypothesis) $\prod_{j \in \mathcal{O}b(\mathscr{I})} D(j)$, the product of the objects of the diagram, and $\prod_{\alpha \in \mathcal{H}om(\mathscr{I})} D(cod\ \alpha)$, the product of the codomains of the morphisms in D, where π_x is the x-th projection of the product. Let now:

$$\gamma, \varepsilon : \prod_{j \in \mathcal{O}b(\mathscr{I})} D(j) \longrightarrow \prod_{\alpha \in \mathcal{H}om(\mathscr{I})} D(cod\ \alpha)$$

be defined by $\gamma_\alpha = \pi_{D(cod\ \alpha)}$ (the projection on the codomain of α) and $\varepsilon_\alpha = D(\alpha) \circ \pi_{D(dom\ \alpha)}$. Let now $e : L \to \prod_{j \in \mathcal{O}b(\mathscr{I})} D(j)$ the equalizer of γ and ε (which exists by hypothesis), and, for each $j \in \mathcal{O}b(\mathscr{I})$, $p_j : L \to D(j)$, defined by $p_j = \pi_{D(j)} \circ e$.

What we want now is to show that $(L, (p_i))_{i \in \mathscr{I}}$ is the limit of D, namely, to prove that the conditions given at the beginning are valid.

For condition 1, we have to show that, for each $\alpha : i \to j$ of $\mathscr{I}$, we have $D(\alpha) \circ p_i = p_j$:

$$\begin{aligned}
D(\alpha) \circ p_i &= D(\alpha) \circ \pi_{D(i)} \circ e && \textit{Definition of } p_j \\
&= \varepsilon_\alpha \circ e && \textit{Definition of } \varepsilon \\
&= \gamma_\alpha \circ e && e \textit{ is an equalizer of } \pi, \varepsilon \\
&= \pi_{D(j)} \circ e && \textit{Definition of } \pi \\
&= p_j && \textit{Definition of } p_j
\end{aligned}$$

For condition 2, suppose that $(E, (q_i)_{i \in \mathcal{O}b(\mathscr{I})})$ has the properties stated. By definition of product, there exists a (unique) arrow $q : E \to \prod_{j \in \mathcal{O}b(\mathscr{I})} D(j)$. For each arrow $\alpha : i \to j$, we have:

$$\begin{aligned}
\gamma_\alpha \circ q &= \pi_{D(j)} \circ q && \textit{Definition of } \pi \\
&= q_j && \textit{Definition of } q_j \\
&= D(\alpha) \circ q_i && \textit{Assumption on } q_j \\
&= D(\alpha) \circ \pi_{D(j)} \circ q && \textit{Definition of } q_i \\
&= \varepsilon_\alpha \circ q && \textit{Definition of } \varepsilon
\end{aligned}$$

Since e equalizes π and ε, there exists a unique $f : E \to L$ in $\mathscr{C}$ such that $q = e \circ f$. Then, for each $j \in \mathcal{O}b(\mathscr{I})$, we have $\pi_{D(j)} \circ q = \pi_{D(j)} \circ e \circ f$, hence, $q_i = p_i \circ f$. $\square$

Bibliography

[AHS09] J. Adámek, Horst Herrlich, and George E. Strecker. *Abstract and concrete categories: The joy of cats*. Dover Publications, 2009.

[BGM06] Paolo Baldan, Fabio Gadducci, and Ugo Montanari. Concurrent rewriting for graphs with equivalences. *CONCUR2006*, 2006.

[BW95] Michael Barr and Charles Wells. *Category theory for computing science*. Prentice Hall, 2 edition, 1995.

[CGM22] Davide Castelnovo, Fabio Gadducci, and Marino Miculan. A new criterion for $\mathcal{M},\mathcal{N}$-adhesivity, with an application to hierarchical graphs, 2022.

[HS79] Horst Herrlich and George E. Strecker. *Category Theory*, volume 1 of *Sigma Series in Pure Mathematics*. Heldermann Verlag Berlin, 2 edition, 1979.

[Pie91] Benjamin C. Pierce. *Basic Category Theory for Computer Scientists*. The MIT Press, 1991.