Chapter 3

FUNCTORS AND NATURAL TRANSFORMATIONS

The starting point of Category Theory is the premise that every kind of mathematically structured object comes equipped with a notion of “acceptable” transformation or construction, that is, a morphism that preserves the structure of the object. This premise holds for categories themselves: a **functor** is the “natural” notion of morphisms between categories. By a further step, the question about what a morphism between functors should look like suggests the notion of **natural transformation**.

Of course, this is not a matter of arbitrary generalizations: it is a habit of mathematics to build constructions on top of constructions, since one may understand in this way, by the uniformity of methods and language, the reasonableness of specific constructions. And this theoretical unification, suggesting power and intellectual unity, is one of the major merits of Category Theory and its applications.

Moreover, as we shall see in several examples, there are notions which are clarified in an essential way or even suggeted by the categorical language of functors and natural transformations. Indeed, there are even too many examples for our purposes, and we will be able to mention only a few that are easily met in computer science.

3.1 Functors

If a transformation F between two categories **C** and **D** must map the categorical structure of **C** to that of **D**, it must take objects and morphisms of **C** to objects and morphisms of **D**; moreover, it must preserve source, target, identities and composition. Such a transformation $F: \mathbf{C} \to \mathbf{D}$ is called a **functor**.

3.1.1 Definition *Let* $\boldsymbol{C}$ *and* $\boldsymbol{D}$ *be categories. A (covariant)* ***functor*** $F : \boldsymbol{C} \to \boldsymbol{D}$ *is a pair of operations* $F_{ob}: Ob_{\boldsymbol{C}} \to Ob_{\boldsymbol{D}}$, $F_{mor}: Mor_{\boldsymbol{C}} \to Mor_{\boldsymbol{D}}$ *such that, for each* $f: a \to b$, $g: b \to c$ *in* $\boldsymbol{C}$,

- $F_{mor}(f) : F_{ob}(a) \to F_{ob}(b)$
- $F_{mor}(g \circ f) = F_{mor}(g) \circ F_{mor}(f)$
- $F_{mor}(id_a) = id_{F_{ob}(a)}$.

It is usual practice to omit the subscripts “ob” and “mor” as it is always clear from the context whether the functor is meant to operate on objects or on morphisms.

1.2 Examples

1. If **C** and **D** are preorders, then a functor F: **C**→**D** is just a monotone function from the objects of **C** to the objects of **D**, indeed $a <_{\mathbf{C}} b \Leftrightarrow \exists f \in \mathbf{C}[a,b] \Rightarrow F(f) \in \mathbf{D}[F(a),F(b)] \Leftrightarrow F(a) <_{\mathbf{D}} F(b)$. Conversely, given a monotone function f between two partial orders C and D, there is of course only one way to extend f to a functor among the categories associated with C and D.

2. The identity functor **I**: **C**→**C** is defined by a pair of identity operations both on objects and on morphisms of **C**. Similarly, we define the **inclusion functor Inc**: **C**→**D** when **C** is a subcategory of **D**.

3. If **C** is a Cartesian category define **×**: **C×C**→**C** as the pair of operations that take $(a,b) \in Ob_{\mathbf{C\times C}}$ to $\times(a,b) = a\times b \in Ob_{\mathbf{C}}$, and $(f,g) \in \mathbf{C\times C}[(a,b),(c,d)]$ to $\times(f,g) = f\times g = <f\circ p_1, g\circ p_2> \in \mathbf{C}[a\times b, c\times d]$. **×** is a functor. Indeed **×** preserves sources and targets and, moreover,

$$\times(id_a, id_b) = id_a \times id_b = <p_a, p_b> = id_{a\times b} = id_{\times(a,b)}$$

$$\times(\ (f,g)\circ(h,k)\ ) = \times(f\circ h, g\circ k) = (f\circ h)\times(g\circ k) = (f\times g)\circ(h\times k) = \times(f,g) \circ \times(h,k)$$

× is usually called **product functor** (since products are only determined up to isomorphisms, we must assume here a canonical choiche, for each object a, b, of their product a×b).

4. The **power-set functor P**: **Set**→**Set** takes every set A to its power-set **P**(A) and every function f: A→B to the function **P**(f): **P**(A)→**P**(B) defined by

$$\forall A' \subseteq A \ \ \mathbf{P}(f)(A') = \{y \in B \ / \ \exists x \in A', y = f(x)\}.$$

Note that the set **P**(f)(A') is what is usually called f(A').

A functor F: **C**→**D** **preserves** a property P that an arrow f may have in **C**, if the arrow F(f) has the same property in **D**. For example, every functor preserves isomorphisms, or the property of being a component of a retraction pair, as it is stated in the following proposition:

3.1.3 Proposition *Let F: **C**→**D** be a functor. If a<b (a ≅ b) in **C**, then F(a) <F(b) (F(a) ≅ F(b)) in **D**.*

Proof. If $f\circ g = id$, then $F(f)\circ F(g) = F(f\circ g) = F(id) = id$. ♦

Exercise Does every functor F preserve the property of being monic or epic?

A functor F: **C**→**D** is **faithful** if for all $a,b \in Ob_{\mathbf{C}}$ and for all $f,g \in \mathbf{C}[a,b]$, F(f) = F(g) implies f = g; it is **full** if for all $a,b \in Ob_{\mathbf{C}}$ and every $h \in \mathbf{D}[F(a),F(b)]$ there is $g \in \mathbf{C}[a,b]$ such that h = F(g). A functor F: **C**→**D** is a **full embedding** iff it is full and faithful, and it is also injective for objects.

A functor that "forgets" (part of) the intended structure of the objects is called **forgetful**: typical examples are the functors from **Grp**, **Top**, or **PO** to **Set** which assign to each object its set of elements and to each arrow the function associated with it. This notion is not a precise one, but it is usually clear when a functor can be considered “forgetful”; note however that a forgetful functor should always be faithful (and very rarely full).

The definition of functor we have given preserves also the *direction* of the arrows in **C**. In some cases, it is interesting to study transformations among categories that reverse such directions, that is, by mapping sources to targets and vice versa. A transformation of this kind between two categories **C** to **D** may be considered as a functor from the dual category $\mathbf{C}^{\mathbf{op}}$ to **D**, and it is known as **contravariant functor** (from **C** to **D**). Explicitly, note the following definition

3.1.4 Definition *Let **C** and **D** be categories. A **contravariant functor** $F: \mathbf{C}\to\mathbf{D}$ is a pair of operations $F_{ob}: Ob_{\mathbf{C}}\to Ob_{\mathbf{D}}$, $F_{mor}: Mor_{\mathbf{C}}\to Mor_{\mathbf{D}}$ such that for each $f: a\to b$, $g: b\to c$ in **C**,*
- $F_{mor}(f) : F_{ob}(b)\to F_{ob}(a)$
- $F_{mor}(g \circ f) = F_{mor}(f) \circ F_{mor}(g)$
- $F_{mor}(id_a) = id_{F_{ob}(a)}$

Examples

1. Each functor $F: \mathbf{C}\to\mathbf{D}$ defines a contravariant functor $F^{op}: \mathbf{C}^{\mathbf{op}}\to\mathbf{D}$ that coincides with F on objects and such that $F^{op}(f) = F(f^{op})$, $F^{op}(g\circ f) = F^{op}(f)\circ F^{op}(g)$. Of course, $F = (F^{op})^{op}$.
2. The **duality** functor $(\ )^{op}: \mathbf{C}\to\mathbf{C}^{\mathbf{op}}$ is a contravariant functor such that $(a)^{op} = a$ for $a\in Ob_{\mathbf{C}}$ and $(f)^{op} = f$ for $f\in Mor_{\mathbf{C}}$.
3. The function that takes every set A to its power-set **P**(A) may be also extended in a natural way to a **contravariant power-set functor P: Set→Set**. Just define, for $f: A\to B$, $\mathbf{P}(f): \mathbf{P}(B)\to\mathbf{P}(A)$ as the function that assigns to each $B'\subseteq B$ the set $\mathbf{P}(f)(B') = f^{-1}(B') = \{x\in A \ /\ \exists y\in B',\ f(x)=y\}$.
4. Let **C** be a category with pullbacks, and let $f\in\mathbf{C}[a,b]$. Then f induces a **pullback functor** $f^*: \mathbf{C}\!\downarrow\! b\to\mathbf{C}\!\downarrow\! a$, whose action is described in the following diagram:

$$\begin{tikzcd}
c \arrow[rr] \arrow[dr, "f^*(k)"] \arrow[dd, "f^*(g)"'] & & d \arrow[dr, "k"] \arrow[dd, "g"' near end] & \\
 & c' \arrow[rr] \arrow[dl, "f^*(h)"] & & d' \arrow[dl, "h"] \\
a \arrow[rr, "f"'] & & b &
\end{tikzcd}$$

In this diagram, g and h are objects of $\mathbf{C}\!\downarrow\! b$, and $k\in\mathbf{C}\!\downarrow\! b[g,h]$; $f^*(g)$ and $f^*(h)$ are the pullbacks of g and h along f, yielding a unique arrow $f^*(k)$ making the above diagram commute. The proof that f^* is a functor is a consequence of elementary properties of pullback, and it is left as an exercise for the reader.

5. Let **C** be a CCC. For every $c \in Ob_{\mathbf{C}}$ the **exponent functor** $\mathbf{exp_c}$: $\mathbf{C} \rightarrow \mathbf{C}$ is defined as follows. For every $a \in Ob_{\mathbf{C}}$, set $\mathbf{exp_c}(a) = c^a$, and for every $f \in \mathbf{C}[a,b]$, $\mathbf{exp_c}(f) = \Lambda(eval_{b,c} \circ id \times f) : c^b \rightarrow c^a$. As a diagram,

$$\begin{tikzcd}
c^b \times a \arrow[r, "id \times f"] \arrow[d, "\Lambda(eval_{b,c} \circ id \times f) \times id_a"'] & c^b \times b \arrow[r, "eval_{b,c}"] & c \\
c^a \times a \arrow[urr, "eval_{a,c}"'] & &
\end{tikzcd}$$

In order to check that $\mathbf{exp_c}$ is actually a contravariant functor, we must take these steps:

$$\begin{aligned}
\mathbf{exp_c}(id) &= \Lambda(eval \circ id \times id) = \Lambda(eval) = id \\
\mathbf{exp_c}(f \circ g) &= \Lambda(eval \circ id \times (f \circ g)) \\
&= \Lambda(eval \circ id \times f \circ id \times g) \\
&= \Lambda(eval \circ \Lambda(eval \circ id \times f) \times id \circ id \times g) && \text{by the diagram} \\
&= \Lambda(eval \circ id \times g \circ \Lambda(eval \circ id \times f) \times id) \\
&= \Lambda(eval \circ id \times g) \circ \Lambda(eval \circ id \times f) \\
&= \mathbf{exp_c}(g) \circ \mathbf{exp_c}(f). \ \blacklozenge
\end{aligned}$$

Note that the composition of two functors is still a functor and that there exists the identity functor which is right and left invariant w.r.t. composition. Thus, it is sound to define the category **Cat** whose objects are categories and whose morphisms are functors. It is worth pointing out that this is a convenient and unusual algebraic property. The collection of groups is not (naturally) a group; similarly for topological spaces and so on. The definition of **Cat**, so similar to the paradoxical "set of all sets" of Set Theory, must be used with some caution, however. It is not our intention to engage in foundational questions in this book, but it is time to say a few more words about our "axiomatic" defintion of Category. So far we have made no assumption at all about the dimension of the classes $Ob_{\mathbf{C}}$ and $Mor_{\mathbf{C}}$; the word "class" itself has been used in an intuitive sense, without any reference to an underlying Set Theory. From now on, though, the reader may refer to the notions of set and class as used, say, in Gödel-Bernays Set Theory, in order to make this distinction clear. From this perspective, a Category **C** such that $Ob_{\mathbf{C}}$ and $Mor_{\mathbf{C}}$ are sets, is called **small**. In the sequel, when we refer to **Cat**, we mean the category of all small categories; of course, **Cat** itself is not a small category, and thus it is not among its own objects.

(*Question*: which of the categories mentioned so far are small, and which are not?)

If **C** is a small category, then, for all $a,b \in Ob_{\mathbf{C}}$, $\mathbf{C}[a,b]$ is a set, usually called **hom-set** of a and b (some authors use the word "hom-set" in an arbitrary category). We say that a category **C** is **locally small** when for all $a,b \in Ob_{\mathbf{C}}$, $\mathbf{C}[a,b]$ is a set and not a class. If **C** is a locally small category, it is possible to define some functors from **C** to **Set**, called **hom-functors**, that play a central role in the developments of Category Theory. We shall see later that most of this work may be done at a more abstract level, taking an arbitrary topos **D** instead of **Set**.

3.1.5 Definition *Let **C** be a locally small category. Given $a \in Ob_{\mathbf{C}}$, the **covariant hom-functor** **C**[a,_]: **C**→**Set** is given by:*

i. for every $b \in Ob_{\mathbf{C}}$ $\mathbf{C}[a,\_](b) = \mathbf{C}[a,b] \in Ob_{\mathbf{Set}}$;

ii. for every $g \in \mathbf{C}[b,b']$, $\mathbf{C}[a,\_](g): \mathbf{C}[a,b] \to \mathbf{C}[a,b']$ is the function that takes $f \in \mathbf{C}[a,b]$ to $(g \circ f) \in \mathbf{C}[a,b']$ that is,

$$\begin{tikzcd} b \arrow[d, "g"] \\ b' \end{tikzcd} \qquad\qquad \begin{tikzcd} \mathbf{C}[a,b] \arrow[d, "{\mathbf{C}[a,\_](g)}"] \\ \mathbf{C}[a,b'] \end{tikzcd}$$

The function **C**[a,_](g) will be denoted in the sequel by "**C**[a,g]" or also by the suggestive "g ∘ _" (some authors use also g_*). Note that the drawing above is clearly an implication between diagrams. Their plain juxtaposition will be used often in the sequel with this meaning, as a special case of the "conditional equational reasoning" mentioned earlier in section 1.2.

3.1.6 Definition *Let **C** be a locally small category. Given $b \in Ob_{\mathbf{C}}$, the **contravariant hom-functor** **C**[_,b]: **C**→**Set** is given by:*

i. for every $a \in Ob_{\mathbf{C}}$ $\mathbf{C}[\_,b](a) = \mathbf{C}[a,b] \in Ob_{\mathbf{Set}}$

ii. for every $h \in \mathbf{C}[a',a]$, $\mathbf{C}[\_,b](h): \mathbf{C}[a,b] \to \mathbf{C}[a',b]$ is the function that takes $f \in \mathbf{C}[a,b]$ to $f \circ h \in \mathbf{C}[a',b]$ that is

$$\begin{tikzcd} a \\ a' \arrow[u, "h"'] \end{tikzcd} \qquad\qquad \begin{tikzcd} \mathbf{C}[a,b] \arrow[d, "{\mathbf{C}[\_,b](h)}"] \\ \mathbf{C}[a',b] \end{tikzcd}$$

The function **C**[_,b](h) will be denoted in the sequel by "**C**[h,b]" or by the suggestive "_ ∘ h" (some authors use also h^* .)

3.1.7 Definition *Let **C** be a locally small category. Given $b \in Ob_{\mathbf{C}}$, the **hom-functor** **C**[_,_]: $\mathbf{C}^{op} x \mathbf{C} \to \mathbf{Set}$ is given by:*

i. for every $(a,b)\in Ob_{\mathbf{C}x\mathbf{C}}$ $\mathbf{C}[\_,\_](a,b) = \mathbf{C}[a,b]\in Ob_{\mathbf{Set}}$
ii. for every $h\in\mathbf{C}[a',a]$, $g\in\mathbf{C}[b,b']$, $\mathbf{C}[\_,\_](h,g)$: $\mathbf{C}[a,b]\to\mathbf{C}[a',b']$ is the function that takes $f\in\mathbf{C}[a,b]$ to $g\circ f\circ h\in\mathbf{C}[a',b']$

C[_,_] is contravariant in the first argument and covariant in the second. One may understand how **C**[_,_] works on morphisms by the following commutative diagram:

$$\begin{tikzcd} a \arrow[r, "f"] & b \arrow[d, "g"] \\ a' \arrow[u, "h"] \arrow[r, "g\circ f\circ h"'] & b' \end{tikzcd}$$

The function **C**[_,_](h,k) will be denoted by "**C**[h,k]" or by "g ∘_ ∘ h."

Exercise: prove in details that **C**[a,_] and **C**[_,b] are a covariant and a contravariant functor, respectively.

Exercise A category **C** has enough points iff the functor **C**[t, -] is faithful, when t is terminal.

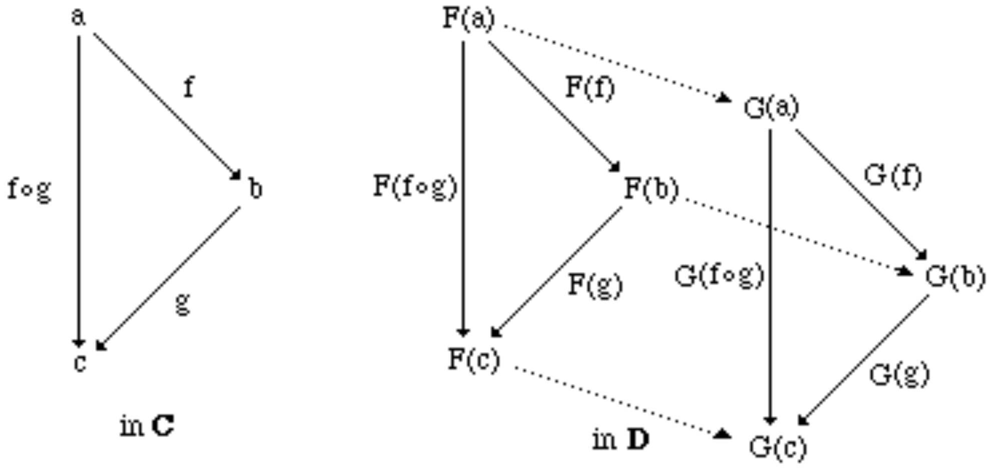
3.2 Natural Transformations

The fact that F is a functor from a category **C** to a category **D** may be equivalently expressed by F(id) = id and, for every f and g in $Mor_{\mathbf{C}}$, by the following (implication between) diagrams:

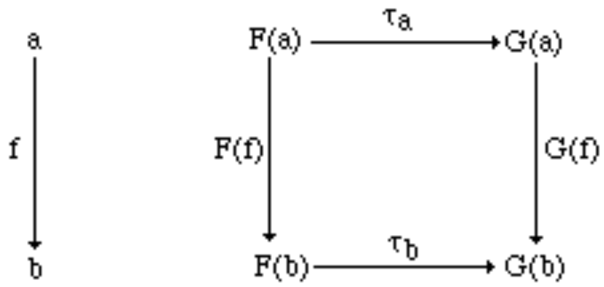
$$\begin{tikzcd} a \arrow[dr, "f"] \arrow[dd, "f\circ g"'] & \\ & b \arrow[dl, "g"] \\ c & \end{tikzcd} \qquad \begin{tikzcd} F(a) \arrow[dr, "F(f)"] \arrow[dd, "F(f\circ g)"'] & \\ & F(b) \arrow[dl, "F(g)"] \\ F(c) & \end{tikzcd}$$

in **C** in **D**

Consider now two functors F, G: **C**→ **D**. A quite reasonable idea of transformation from F to G is a "translation" as described in the following picture, where the dotted lines should yield commutative squares



Thus, the "translation" can be defined by assigning to each object $a \in Ob_{\mathbf{C}}$ an arrow τ_a: $F(a) \to G(a)$, with the only condition that, for every $f \in \mathbf{C}[a,b]$, the following diagram commutes:



The properties described in this diagram are equivalently formalized by the following definition.

3.2.1 Definition. *Let $F,G : \mathbf{C} \to \mathbf{D}$ be functors. Then $\tau : F \to G$ is a* ***natural transformation*** *from F to G iff:*

i. $\forall a \in Ob_{\mathbf{C}}\ \ \tau_a \in \mathbf{D}[F(a),G(a)]$

ii. $\forall f \in \mathbf{C}[a,b]\ \ \tau_b \circ F(f) = G(f) \circ \tau_a$.

3.2.2 Example Let $\mathbf{C}$ be a small category, and $h \in \mathbf{C}[a',a]$. The collection (in **Set**) of morphisms $\{\mathbf{C}[h,b] / \mathbf{C}[a,b] \to \mathbf{C}[a',b]\}_{b \in \mathbf{C}}$, defines a natural transformation $\mathbf{C}[h,\_]$ from the (contravariant) hom-functor $\mathbf{C}[a,\_]$ to the (contravariant) hom-functor $\mathbf{C}[a',\_]$. Note the following diagram:

$$
\begin{tikzcd}
 & a & & a' \arrow[ll, "h"'] \\
b \arrow[dd, "k"'] & \mathbf{C}[a,b] \arrow[rr, "{\mathbf{C}[h,b] = \_\circ h}"] \arrow[dd, "{\mathbf{C}[a,k] = k\circ\_}"'] & & \mathbf{C}[a',b] \arrow[dd, "{\mathbf{C}[a',k] = k\circ\_}"] \\
 & & & \\
b' & \mathbf{C}[a,b'] \arrow[rr, "{\mathbf{C}[h,b'] = \_\circ h}"] & & \mathbf{C}[a',b']
\end{tikzcd}
$$

The same diagram proves that, given $k \in \mathbf{C}[b,b']$, the collection of morphisms $\{\mathbf{C}[a,k] / \mathbf{C}[a,b] \to \mathbf{C}[a,b']\}_{a \in \mathbf{C}}$ defines a natural transformation $\mathbf{C}[\_,k]$ from the hom-functor $\mathbf{C}[\_,b]$ to the hom-functor $\mathbf{C}[\_,b']$.

It is easy to close up natural transformations under composition by setting $(\tau\circ\beta)_a = (\tau_a)\circ(\beta_a)$. This composition of natural transformation is usually called **vertical**, as opposed to the **horizontal composition**, defined at the end of this section. Since the identity transformation from a functor F to itself is defined in the obvious way, we have actually constructed a new category, starting from any two given categories **C** and **D** .

The new category is called the **category of functors** from **C** to **D**, **(C→D) = Funct(C,D)**; its objects are functors and the morphisms are natural transformations. In particular, if F,G: **C→D**, **Funct(C,D)**[F,G] is the collection of all the natural transformations from F to G; in the following we shall use the abbreviation **Nat**(F,G) instead of **Funct(C,D)**[F,G].

Two functors from **C** to **D** are equivalent (or **naturally isomorphic**) iff they are isomorphic as objects of **Funct(C,D)**. For example, it is well understood that any set A is isomorphic to A×1, where 1 is a singleton. For arbitrary cartesian categories, this corresponds to saying that the functor _×1 and the identity functor Id are naturally isomorphic. If **F**: **C→D** is a full embedding, then **C** is isomorphic to a full subcategory of **D**. The next section will present further examples of natural isomorphisms.

The concept of natural isomorphism of functors also allows us to define a notion of "equivalence" between categories, which captures better than the notion of isomorphism the sense that two categories can be said to be "essentially the same." Two categories **C** and **D** are **equivalent** if and only if there are two functors F: **C→D** and G : **D→C** such that $G \circ F \cong id_{\mathbf{C}}$ and $F \circ G \cong id_{\mathbf{D}}$ (note that **C** is isomorphic to **D** iff $G \circ F = id_{\mathbf{C}}$ and $F \circ G = id_{\mathbf{D}}$).

3.2.3 Proposition *Let $F,G : \mathbf{C}\to\mathbf{D}$ be functors and $\tau : F\to G$ be a natural transformation from F to G. Assume that, for each $a \in Ob_{\mathbf{C}}$, $\tau_a \in \mathbf{D}[F(a),G(a)]$ is an isomorphism. Then τ is a natural isomorphism.*

Proof Define $\tau^{-1} : G \to F$ by $\tau^{-1}{}_a = (\tau_a)^{-1}$. τ^{-1} is natural, since $\forall f \in \mathbf{C}[a,b]$

$$\begin{aligned} \tau^{-1}{}_b \circ G(f) &= \tau_b{}^{-1} \circ G(f) \circ \tau_a \circ \tau_a{}^{-1} \\ &= \tau_b{}^{-1} \circ \tau_b \circ F(f) \circ \tau_a{}^{-1} \\ &= G(f) \circ \tau^{-1}{}_a. \ \blacklozenge \end{aligned}$$

Examples A simple example of natural transformation may be given by studying "liftings" (see section 2.6), in various categories. One may actually understand the general notion better, by completing a little exercise on natural transformations. Indeed, what is hidden behind definition 2.6.2, is the "naturality" of τ_c. When writing this down explicitly, the definition is tidier and more expressive. Let $\mathbf{C}$ be a Category of partial maps, $\mathbf{C_t}$ be the associated category of total maps, and **Inc**: $\mathbf{C_t} \to \mathbf{C}$ be the obvious inclusion. Thus, 2.6.2 may be simply restated as

*The **lifting** of $a \in Ob_{\mathbf{C}}$ is the object $a°$ such that the functors $\mathbf{C}[\_,a] \circ \mathbf{Inc}$, $\mathbf{C_t}[\_,a°] : \mathbf{C_t} \to \mathbf{Set}$ are naturally isomorphic.*

Then, by definition of natural transformation and hom-functor, this requires the existence of a function τ such that the following diagram commutes, for any object b and c in $\mathbf{C}$ and $f \in \mathbf{C_t}[c,b]$

$$\begin{tikzcd} \mathbf{C}[b,a] \arrow[r, "\tau_b"] \arrow[d, "\_\circ f"'] & \mathbf{C_t}[b,a°] \arrow[d, "\_\circ f"] \\ \mathbf{C}[c,a] \arrow[r, "\tau_c"'] & \mathbf{C_t}[c,a°] \end{tikzcd}$$

That is, $\tau_c(g \circ f) = \tau_b(g) \circ f$ and $(\tau_c)^{-1}(h \circ f) = (\tau_b)^{-1}(h) \circ f$, for any total f, since τ is an isomorphism.

With this definition, to prove unicity of liftings is even smoother than in section 2.5. Indeed, let τ be the given natural isomorphism, and let a' and β be an alternative lifting and natural isomorphism. Set $\phi = \tau \circ \beta^{-1}$. Then $\mathbf{C_t}[\_,a']$ and $\mathbf{C_t}[\_,a°]$ are naturally isomorphic via ϕ and, for $f = \phi_{a'}(id): a' \to a°$ and $g = (\phi_{a°})^{-1}(id): a° \to a'$, one has:

$$\begin{aligned} g \circ f &= (\phi_{a°})^{-1}(id) \circ f \\ &= (\phi_{a'})^{-1}(id \circ f) && \text{by naturality} \\ &= (\phi_{a'})^{-1}(\phi_{a'}(id)) \\ &= id. \end{aligned}$$

Similarly for $f \circ g = id$ (on $a°$).

Since we are now familiar with functors, we may look also at lifting as a functor.

Let **C** be a pC and assume that for each $a \in Ob_{\mathbf{C}}$ there exists the lifting a°. Then there is a (unique) extension of the map $a \vdash a^\circ$ to a functor $\_^\circ : \mathbf{C} \to \mathbf{C_t}$ (the **lifting functor**).

The reader may check this as an exercise (*hint:* observe that $ex_a = (\tau_{a^\circ})^{-1}(id): a^\circ \to a$ and use the naturality of τ) .

As for specific examples, the lifting functor for **pSet** is obvious. It can be easily guessed also for the category **pPo** of p.o.sets and partial monotone functions with upward closed domains: just add a fresh least element and the rest is easy for the reader who has completed the last exercise. Note, and it is crucial, that by monotonicity the lifting functor does not exist if one doesn't assume that the domains are upward closed.

The category **pCPO** is given by defining complete partial orders under the assumption that directed sets are not empty. Thus, the objects of **pCPO** do not need to have a bottom element. As for morphisms, take the partial continuous functions with open subsets as domains. Clearly, the lifting functor is defined as it is for **pPo.**

A more complex example is given by **EN**, the category of numbered sets in section 2.2. Let **pEN** be the partial category of numbered sets in the example before 2.5.2. Given $\underline{a} = (a,e) \in Ob_{\mathbf{pEN}}$, define $\underline{a^\circ} = (a^\circ,e^\circ)$ by adding a new element $\perp$ to the set a and by defining $e^\circ(n) =$ <u>if</u> $\phi_n(0)$ converges <u>then</u> $e(\phi_n(0))$ <u>else</u> $\perp$. Clearly, $e^\circ : \omega \to a^\circ$ is surjective. Let now $\underline{b} = (b,e')$ and $f \in \mathbf{pEN}[\underline{b},\underline{a}]$. By definition, there exists $f' \in PR$ such that $f \circ e' = e \circ f'$. We define the extension $\underline{f} \in \mathbf{EN}[\underline{b},\underline{a^\circ}]$ of f by giving $\underline{f}' \in R$ which represents $\underline{f}$. That is, set $\phi_{\underline{f}'(n)}(0) = f(n)$. Such an $\underline{f}' \in R$ exists by the s-m-n (iteration) theorem. Thus,

$$\begin{aligned} \underline{f}(e'(n)) &= e\circ(\underline{f}'(n)) \\ &= \underline{\text{if}}\ \ \phi_{\underline{f}'(n)}(0) \text{ converges } \underline{\text{then}}\ \ e(\phi_{\underline{f}'(n)}(0))\ \ \underline{\text{else}}\ \perp . \end{aligned}$$

Therefore, if $f(e'(n)) = e(f'(n))$ is defined, then $\phi_{\underline{f}'(n)}(0)$ converges and, hence, $\underline{f}(e'(n)) = e(\phi_{\underline{f}'(n)}(0)) = f(e'(n))$. Finally, set $\tau_{\underline{a}\underline{b}}(f) = \underline{f}$. For each $\underline{a}$, $\tau_{\underline{a}}$ gives the required natural isomorphism, as $\forall \underline{g} \in \mathbf{EN}[\underline{b},\underline{a^\circ}]\ \exists! f \in \mathbf{pEN}[\underline{b},\underline{a}]\ \ f'(n) = \phi_{g'(n)}(0)$. (**Exercise**: check the due diagram). By the fact above, this defines the lifting functor in **pEN**.

Exercise Define the category **ER** of equivalence relations on ω and effective maps (*hint:* the objects are quotient sets on ω, and the morphisms are induced by total recursive functions similarly as for **EN**). Observe that **ER** and **EN** are equivalent, but not isomorphic. Indeed, one is small, while the other is not.

We next discuss ways to derive natural transformations from a given one and, finally, the notion of horizontal composition.

Let $H: A \to B$, $F: B \to C$, $G: B \to C$, $K: C \to D$ be functors, and let $\tau: F \to G$ be a natural transformation as shown in the following diagram:

$$\begin{tikzcd} \mathbf{A} \arrow[r, "H"] & \mathbf{B} \arrow[r, bend left, "F"] \arrow[r, bend right, "G"'] \arrow[r, phantom, "\Downarrow\tau"] & \mathbf{C} \arrow[r, "K"] & \mathbf{D} \end{tikzcd}$$

τ induces two natural transformations $K\tau: KF \to KG$, and $\tau H: FH \to GH$, respectively defined by

$$(K\tau)_b = K(\tau_b) : KF(b) \to KG(b);$$
$$(\tau H)_a = \tau_{H(a)} : FH(a) \to GH(a)$$

We have, for every $f \in \mathbf{B}[b,b']$ and every $g \in \mathbf{A}[a,a']$,

$$(K\tau)_{b'} \circ KF(f) = K(\tau_b) \circ K(F(f)) = K(\tau_b \circ F(f)) = K(G(f) \circ \tau_b) = KG(f) \circ K\tau_b = KG(f) \circ (K\tau)_b$$
$$(\tau H)_{a'} \circ FH(g) = \tau_{H(a')} \circ F(H(g)) = G(H(g)) \circ \tau_{H(a)} = GH(g) \circ (\tau H)_a$$

that proves the naturality of $K\tau$ and τH .

$$\begin{tikzcd} b \arrow[d, "f"] & KF(b) \arrow[r, "K\tau_b"] \arrow[d, "KF(f)"'] & KG(b) \arrow[d, "KG(f)"] \\ b' & KF(b') \arrow[r, "K\tau_{b'}"] & KG(b') \end{tikzcd} \qquad \begin{tikzcd} a \arrow[d, "g"] & FH(a) \arrow[r, "\tau_{H(a)}"] \arrow[d, "FH(g)"'] & GH(a) \arrow[d, "GH(g)"] \\ a' & FH(a') \arrow[r, "\tau_{H(a')}"] & GH(a') \end{tikzcd}$$

in **B**   in **D**   in **A**   in **C**

Consider now categories, functors, and natural transformations as described in the following diagram:

$$\begin{tikzcd} \mathbf{B} \arrow[r, bend left, "F"] \arrow[r, bend right, "G"'] \arrow[r, phantom, "\Downarrow\tau"] & \mathbf{C} \arrow[r, bend left, "H"] \arrow[r, bend right, "K"'] \arrow[r, phantom, "\Downarrow\sigma"] & \mathbf{D} \end{tikzcd}$$

Then, for the naturality of σ with respect to the arrow τ_b, the following diagram (in **D**) commutes for every $b \in Ob_{\mathbf{B}}$:

$$\begin{tikzcd} HF(b) \arrow[r, "H(\tau_b)"] \arrow[d, "\sigma_{F(b)}"'] & HG(b) \arrow[d, "\sigma_{G(b)}"'] \\ KF(b) \arrow[r, "K(\tau_b)"] & KG(b) \end{tikzcd}$$

The **horizontal composition** of σ and τ is the natural trasformation $\sigma\tau$: HF→KG defined by, for every $b \in Ob_{\mathbf{B}}$, $\sigma\tau_b = \sigma_{G(b)} \circ H(\tau_b) = K(\tau_b) \circ \sigma_{F(b)}$.
We check the naturality of $\sigma\tau$. Let $f \in \mathbf{B}[b,b']$, then

$$\begin{aligned}
\sigma\tau_{b'} \circ HF(f) &= \sigma_{G(b')} \circ H(\tau_{b'}) \circ HF(f) \\
&= \sigma_{G(b')} \circ H(\tau_{b'} \circ F(f)\,) \\
&= \sigma_{G(b)} \circ H(\,G(f) \circ \tau_b\,) && \text{by the naturality of } \tau \\
&= \sigma_{G(b)} \circ H(\,G(f) \circ \tau_b\,) && \text{by the diagram} \\
&= K(G(f) \circ \tau_b) \circ \sigma_{F(b)} \\
&= KG(f) \circ K(\tau_b) \circ \sigma_{F(b)} \\
&= KG(f) \circ \sigma\tau_b
\end{aligned}$$

Note that if we identify the functors K and H with the identity natural transformation id_K and id_H, $K\tau$ and τH may be understood as particular cases of the horizontal application between natural transformations (why?).

Exercise Prove the following equality among natural transformations (**interchange law**):

$$(\nu \circ \mu)(\tau \circ \sigma) = (\nu\tau) \circ (\,\mu\sigma)\,.$$

3.3 Cartesian and Cartesian Closed Categories Revisited

By definition, a category **C** is Cartesian iff it contains a terminal object t, and for every $a,b \in Ob_{\mathbf{C}}$ there is an object $a \times b$ together with two morphisms $p_1: a\times b \to a$, $p_2: a\times b \to b$, and for every object c an isomorphism $<\,,\,>_c : \mathbf{C}[c,a]\times\mathbf{C}[c,b] \to \mathbf{C}[c,a\times b]$ such that for all morphisms $f: c\to a$, $g: c\to b$, $h: c\to a\times b$, the following equations hold:

i. $p_1 \circ <f,g>_c = f$

ii. $p_2 \circ <f,g>_c = g$

We want now to show that $<\,,\,>_c$ is also "natural in c", i.e. it satisfies the property:

(nat) for every $k: c\to c'$ $<f,g>_{c'} \circ k = <f \circ k, g \circ k>_c$

Indeed, we have

$$\begin{aligned}
<f,g>_{c'} \circ k &= <p_1 \circ <f,g>_{c'} \circ k, p_2 \circ <f,g>_{c'} \circ k>_c && \text{by ii)} \\
&= <f \circ k, g \circ k>_c && \text{by i)}
\end{aligned}$$

The previous "naturality" property suggests that $<\,,\,>$ is actually a natural isomorphism. Indeed, let $\Delta: \mathbf{C}\to\mathbf{C}\times\mathbf{C}$ the functor defined by $\Delta(c) = (c,c)$, $\Delta(f) = (f,f)$ (Δ is called the **diagonal functor**); then $<\,,\,>$ is a natural isomorphism from $\mathbf{C}\times\mathbf{C}[\_,(a,b)] \circ \Delta$ to $\mathbf{C}[\_,a\times b]$. Note that $\mathbf{C}\times\mathbf{C}[\_,(a,b)] \circ \Delta$ and $\mathbf{C}[\_,a\times b]: \mathbf{C}\to\mathbf{Set}$ are contravariant functors. Conversely suppose to have for all objects a and b an object $a\times b$ and a natural isomorphism $\tau : \mathbf{C}\times\mathbf{C}[\_,(a,b)] \circ \Delta \cong \mathbf{C}[\_,a\times b]$. The naturality of τ^{-1} is expressed by the following commutative diagram:

$$\begin{tikzcd}
\mathbf{C}\times\mathbf{C}[(c,c),(a,b)] \arrow[d, "(_,_)\circ(f,f)"'] & \mathbf{C}[c,a\times b] \arrow[l, "\tau^{-1}_c"'] \arrow[d, "_\circ f"] & c \\
\mathbf{C}\times\mathbf{C}[(c',c'),(a,b)] & \mathbf{C}[c',a\times b] \arrow[l, "\tau^{-1}_{c'}"] & c' \arrow[u, "f"']
\end{tikzcd}$$

Let $(q_1,q_2) = \tau^{-1}{}_{a\times b}(id_{a\times b})$. Note that $q_1: a\times b\to a$, $q_2: a\times b\to b$. We want to prove that $a\times b$ is a product with projections q_1 and q_2. Indeed, let $f = \tau_c(h,g): c\to a\times b$ in the above diagram, then we have $(q_1,q_2) \circ (\tau_c(h,g), \tau_c(h,g)) = \tau^{-1}{}_c(id_{a\times b} \circ \tau_c(h,g)) = (h,g)$ and, in particular, $q_1 \circ \tau_c(h,g) = h$; $q_2 \circ \tau_c(h,g) = g$.

The previous considerations suggest another characterization of Cartesian Category:

3.3.1 Proposition *A category* **C** *is Cartesian iff it contains a terminal object t, and for every* $a,b\in Ob_{\mathbf{C}}$ *there is an object* $a\times b$ *and a natural isomorphism* $<\ ,\ > : \mathbf{C}\times\mathbf{C}[\_,(a,b)] \circ \Delta \to \mathbf{C}[\_,a\times b]$.

The situation is quite similar, and perhaps even simpler, in the case of exponents. Remember that a category **C** is a CCC iff it is Cartesian and has exponents for every pair of objects, i.e. for every a and b in **C** there is an object b^a together with a morphism $eval_{a,b}: b^a\times a\to b$ (evaluation map) and, for every object c , a function $\Lambda_c : \mathbf{C}[c\times a,b]\to\mathbf{C}[c,b^a]$ such that, for all morphisms $f: c\times a\to b$, $h: c\to b^a$, one has

$$\beta.\ \ eval_{a,b} \circ (\Lambda(f)\times id_a) = f$$
$$\eta.\ \ \Lambda(eval\circ(h\times id)) = h$$

It turns out that Λ_c is "natural in c", in the sense that for any $f\in\mathbf{C}[c\times a,b]$, $g\in\mathbf{C}[d,c]$

$$\Lambda_{c'}(f) \circ g = \Lambda_c(f \circ g\times id) ,$$

In order to check this, recall that (β) corresponds to:

(Diag.Eval)
$$\begin{tikzcd}
c\times a \arrow[r, "f"] \arrow[d, "\Lambda(f)\times id"'] & b \\
b^a\times a \arrow[ur, "eval"'] &
\end{tikzcd}$$

Thus, by twice (Diag.Eval), the following diagram commutes

$$\begin{tikzcd}
d\times a \arrow[r, "g\times id"] \arrow[dr, "\Lambda(f\circ(g\times id))\times id"'] & c\times a \arrow[r, "f"] \arrow[d, "\Lambda(f)\times id"] & b \\
& b^a\times a \arrow[ur, "eval"'] &
\end{tikzcd}$$

Moreover, set $\Lambda^{-1} = \mathrm{eval}\circ(\_\times id)$.

Then $\qquad \Lambda^{-1}\circ\Lambda = id \qquad$ by (β)

and $\qquad \Lambda\circ\Lambda^{-1} = id \qquad$ by (η).

That is, for each $c\in Ob_{\mathbf{C}}$, Λ_c: $\mathbf{C}[c\times a,b]\to\mathbf{C}[c,b^a]$ is an isomorphism. Thus Λ is a natural isomorphism, by proposition 3.2.3.

Note that, formally, Λ is a natural transformation from the contravariant functor $\mathbf{C}[\_,b] \circ \_\times a$ to the contravariant functor $\mathbf{C}[\_,b^a]$ (where $\_\times a$ is the product functor defined in the obvious way). We abbreviate $\mathbf{C}[\_,b] \circ \_\times a$ as $\mathbf{C}[\_\times a,b]$. Note also that $\mathbf{C}[\_\times a,b]$, $\mathbf{C}[\_,b^a]$: $\mathbf{C}\to\mathbf{Set}$.

Conversely, suppose that, for all objects a and b, there is an object b^a and a natural isomorphism Λ: $\mathbf{C}[\_\times a,b] \cong \mathbf{C}[\_,b^a]$. Then the naturality of Λ^{-1} is expressed by the following commutative diagram:

$$\begin{tikzcd}
\mathbf{C}[c\times a,b] \arrow[d, "_\circ f\times id"'] & \mathbf{C}[c,b^a]_{b^a} \arrow[l, "\Lambda^{-1}_c"'] \arrow[d, "_\circ f"] & c \\
\mathbf{C}[c'\times a,b] & \mathbf{C}[c',b^a] \arrow[l, "\Lambda^{-1}_{c'}"] & c' \arrow[u, "f"']
\end{tikzcd}$$

Set now $\mathrm{eval}_{a,b} = \Lambda^{-1}{}_{b^a}(id_{b^a})$ and note that $\mathrm{eval}_{a,b}$: $b^a\times a\to b$. We want to prove that b^a is an exponent with $\mathrm{eval}_{a,b}$ as evaluation map . Indeed, let $f = \Lambda_c(g)$: $c\to b^a$ in the above diagram. Then we have:

$$\mathrm{eval}_{a,b} \circ (\Lambda_c(g)\times id) = \Lambda^{-1}{}_c(id_{b^a} \circ \Lambda_c(g)) = g$$

This argument gives the following equivalent characterization of CCC:

3.3.2 Proposition *C is a* ***Cartesian closed category*** *iff it is Cartesian and for every $a,b\in Ob_{\mathbf{C}}$ there is an object b^a and a natural isomorphism $\Lambda : \mathbf{C}[\_\times a,b] \to \mathbf{C}[\_,b^a]$.*

Observe that definitions 3.3.1 and 3.3.2 require that the category **C** be locally small, since they are based on hom-functors. Thus, in a sense, the equational definition is more general.

3.3.3 Remark It is easy to prove that the following (natural) isomorphisms hold in all CCC's, for any object A, B, and C:

1. $A \cong A$;
2. $t \times A \cong A$;
3. $A \times B \cong B \times A$;
4. $(A \times B) \times C \cong A \times (B \times C)$;
5. $(A \times B) \to C \cong A \to (B \to C)$;
6. $A \to (B \times C) \cong (A \to B) \times (A \to C)$;
7. $t \to A \cong A$;
8. $A \to t \cong t$.

What is worth mentioning, though, is that these are exactly the isomorphisms that hold in all CCC's, i.e., no other isomorphism is valid in all CCC's. The proof of this fact is nontrivial and is a nice application of lambda calculus to categories (an application in the other direction from what is meant by the title of this book!). Its key idea will be mentioned in chapter 9.

3.4 More Examples of CCC's

Both examples here derive from bordering areas of (generalized) computability and Proof Theory. The first, in particular, like many aspects of these theories, is widely used in the type theoretical understanding of programming languages constructs.

3.4.1 Partial Equivalence Relations

A very relevant example of CCC is suggested by a long story. It began with Kleene's realizability interpretation of intuitionistic logic and continued with the work of Kreisel, Girard and Troelstra in Proof Theory. The idea is to look at functions as computable ones, as in the case of the category **EN**, but by a simple and insightful way cartesian closedness is obtained. The approach is also used in the "quotient-set" semantics of types, in functional programming. It will give us a paradigmatic structure in the categorical semantics of polymorphism in PART II.

As usual, let $\varphi: \omega \to PR$ an acceptable goedel numbering of the partial recursive functions. Let $K = (\omega, \cdot)$ be **Kleene's applicative structure**, where $\cdot : \omega \times \omega \to \omega$ is the partial application defined by: $m \cdot n = \varphi_m(n)$. A **partial equivalence relation** R on a set V is a symmetric and transitive relation, not necessarely reflexive, on V.

(**Notation**: $n R m$ iff n relates to m in R; $\{p\}_R = \{q \mid q R p\}$; $Q(R) = \{\{p\}_R \mid p R p\}$) .

3.4.1.1 Definition *The category* ***PER*** *of partial equivalence relations on ω has as objects the symmetric and transitive relations on ω. Morphisms are defined by*

$$f \in \mathbf{PER}[A,B] \quad \text{iff} \quad f : Q(A) \to Q(B) \ \text{ and } \ \exists n \ \forall p \ (pAp \ \Rightarrow f(\{p\}_A) = \{n \cdot p\}_B\ ) .$$

Thus, the morphisms in **PER** are "computable" because they are fully described by partial recursive functions, which are total on the domain of the source relation.

Let now $<\ ,\ >: \omega\times\omega\to\omega$ be an effective and bijective pairing. For $A,B\in Ob_{\mathbf{PER}}$, the **product** $A\times B$ is defined by

$$\forall m,n,m',n' \ \ <m,n> A\times B <m',n'> \ \Leftrightarrow \ (\ m\ A\ m' \ \text{ and } \ n\ B\ n').$$

It is easy to check that this actually defines a product functor in **PER**.

The **exponent object** B^A is defined by

$$\forall m,n, \ \ m\ (B^A)\ n \ \Leftrightarrow \ \forall p,q \ \ (p\ A\ q \ \Rightarrow \ m\cdot p\ B\ n\cdot q\ ) .$$

Cartesian closedness follows by giving a natural isomorphism $\Lambda : \mathbf{PER}[A\times C,B] \cong \mathbf{PER}[A,B^C]$. Let s be the recursive function of the s-m-n (or iteration) theorem, i.e. $\varphi_{s(n,p)}(q) = \varphi_n(p,q)$. Then set $\Lambda(f)(\{p\}_A) = \{s(n,p)\}_{C\to B}$, where n is an index for f. In other words $\{n\cdot<p,q>\}_B = \{s(n,p)\cdot q\}_B$. Observe that Λ is a well-defined function from $\mathbf{PER}[A\times C,B]$ to $\mathbf{PER}[A,B^C]$, since s is computable and, then, any index for the recursive function $p \vdash s(n,p)$, computes $\Lambda(f)\in\mathbf{PER}[A,B^C]$. As for the naturality of Λ , one may prove it by the argument above, which also gives some information on the evaluation map.

Let $\mathbf{eval_{A,B}}$: $B^A\times A\to B$ be defined by $eval(\ \{<m,n>\}_{B^A\times A}\ ) = \{m\cdot n\}_B$.
In order to prove that $eval_{A,B}$ is a morphism in the category we must find $e_{A,B}\in\omega$ such that

$$eval(\{<m,n>\}_{B^A\times A}\ ) = \{m\cdot n\}_B = \ \{e_{A,B}\cdot(<m,n>)\}_B$$

Let u be the "universal" function, i.e., the partial recursive function such that $u(<m,n>) = m\cdot n$, and let e be an index for it, i.e., $u = \varphi_e$. It is easy to observe that one can set $e_{A,B} = e$ for all A, B in $Ob_{\mathbf{PER}}$, since

$$\{e\cdot(<m,n>)\}_B = \{u(<m,n>)\}_B = \{m\cdot n\}_B \ .$$

Then, (β) is simply

$$eval(\{<s(n,p),q>\}_{B^A\times A}\ ) = \{n\cdot<p,q>\}_B, \ \text{ by definition of } \ s\ .$$

Similarly for (η).

3.4.2 Limit and Filter Spaces

There is an elegant, unifying way to understand the various approaches to generalized computability proposed in the 60's and 70's. The connecting point is the construction of categories of sets where a suitable notion of limit gives an abstract notion of computability. The idea is to generalize the technique used when defining the computable elements in Scott domains **D** (see 2.4.1), in the way explained below (in short, the computable elements are the limits of recursively enumerable indexed sequences.) This suggests several CCC's, such as limit spaces (**L-spaces**) and filter spaces (**FIL**) with their relevant subcategories. They will be introduced here and discussed also in the examples in section 5.3 and section 8.4, toghether with other ideas for higher type computations.

3.4.2.1 Definition *A **limit space** (L-space) $(X,\downarrow)$ is a set X and a relation (convergence) between countable sequences $\{x_i\}_{i\in\omega} \subseteq X$ and elements $x\in X$ (notation: $\{x_i\}\downarrow x$) such that*

1. *if all but finitely many x_i are x, i.e., $\{x_i\}$ is eventually x, then $\{x_i\}\downarrow x$;*
2. *if $\{x_i\}\downarrow x$ and $k(0) < k(l) < \ldots < k(n) < \ldots$, then $\{x_{k(i)}\}\downarrow x$;*
3. *if $not(\{x_i\}\downarrow x)$, then there is $k(0) < k(l) < \ldots < k(n) < \ldots$ such that for no subsequence $h(0) < h(l) < \ldots < h(n) < \ldots$ one has $\{x_{(i)}\}\downarrow x$.*

An L-space $(X,\downarrow)$ has a **countable basis** (is **separable**) iff for some given countable $X_o \subseteq X$, $\forall x\in X\ \exists\{x_i\}\subseteq X_o\ \{x_i\}\downarrow x$.

From now on we assume that each countably based L-spaces $(X,\downarrow)$ comes with a given surjective enumeration $e: \omega\to X_o$ of the base. An immediate example of separable L-spaces is the set of real numbers endowed with the usual notion of sequence converge (Cauchy).

3.4.2.2 Definition *The hom-set $\mathbf{L}[X,Y]$ between L-spaces $(X,\downarrow)$ and $(Y,\downarrow)$ consists of all continuous functions, i.e.,*

$$f\in\mathbf{L}[X,Y] \quad iff \quad \forall x\in X\ \ \forall\{x_i\}\downarrow x\ \ \{f(x_i)\}\downarrow f(x),$$

where convergence is given in the intended spaces.

This category has exponents and products, as $(\mathbf{L}[X,Y],\downarrow)$ also is an L-space by

$$\{f_i\}\downarrow f \quad \text{iff} \quad \forall x\in X, \forall\{x_i\}\downarrow x\ \ \{f_i(x_i)\}\downarrow f(x)\ ,$$

while products are given by componentwise convergence.

Finally, eval: $\mathbf{L}[X,Y]\times X\to Y$, with eval(f,x) = f(x), is continuous. As a matter of fact, $(\mathbf{L}[X,Y],\downarrow)$ is the coarsest limit structure (i.e., with more converging sequences) such that eval is continuous.

One can also show that if $(X,\downarrow)$ and $(Y,\downarrow)$ are separable, then also $(\mathbf{L}[X,Y],\downarrow)$ is separable. Indeed, L-spaces and separable L-spaces, with continuous maps as morphisms, form Cartesian closed categories.

Yet another CCC of limit spaces may be given by the Moore-Smith net-convergence or, equivalently, by filter-convergence. Just recall that a **filter** Φ (on a given set X) is a set of (sub)sets closed by intersection and such that $A\in\Phi$ and $A\subseteq B$ imply $B\in\Phi$ (e.g., the collection of subsets of A which contain a given $x\in A$ is the (ultra)filter generated by x). A **filter base** is a non empty collection of non empty subsets of X such that $A\in\Phi$ and $B\in\Phi$ imply $\exists C\in\Phi\ C\subseteq A\cap B$. A filter base Φ generates a unique filter $[\Phi] = \{B\subseteq X \mid \exists A\in\Phi\ A\subseteq B\}$. Define then

3.4.2.3 Definition *(X, F) is a **filter space** iff $\forall x \in X$ $F(x)$ is a filter of filters such that the ultrafilter generated by x is in F(x). Given a filter base Φ, we write $\Phi{\downarrow}x$ iff $[\Phi] \in F(x)$.*

Exercise Prove that the category **FIL** of filter spaces with continuous maps (where f is continuous iff $\Phi{\downarrow}x$ implies $f(\Phi){\downarrow}x$) is a CCC. Give a full and faithful functor F : **FIL→L-spaces**. (*Hint*: a filter structure on **FIL**[X,Y] is given by $\Xi{\downarrow}f$ iff $\Phi{\downarrow}x$ implies $\Xi(\Phi){\downarrow}f(x)$, where $\Xi(\Phi)$ is the set of all W(U) with $W \in \Xi$ and $U \in \Phi$ and $W(U) = \cup\{f(U) \mid f \in W\}$; moreover, given a filter Φ and a sequence $\{x_i\}$, define $Con(\Phi,\{x_i\})$ iff $\forall U \in \Phi\ \exists k\ \forall n \geq k\ x_n \in U$ and set $\{x_i\}{\downarrow}x$ iff $\exists \Phi{\downarrow}x\ Con(\Phi,\{x_i\})$).

A notion of separable filter space is easily given, by taking a countable base of filters (i.e., a countable collection of sets such that each converging filter is generated by elements of the base).

Clearly, each (separable) topological space (X, top) may be turned into a (separable) filter space: just take, for each point x , the collection F(x) of all filters containing the filter of neighborhoods of that point. The reader will have a better insight into the “injections”

Top → **FIL** → **L-spaces**

when looking at examples of adjunctions, in section 5.3. As for now, it may suffice to say that there exist filter spaces whose limit structure is not topological. Some of these filter spaces are among those needed for the study of the total computable functionals.

In the very general setting of L-spaces we can now hint a notion of computability which specializes to the one given over Scott domains, when the intended limit structure is derived from the Scott topology.

Let R be the total recursive functions and $(X, X_0, e, \downarrow)$ a separable L-space, where e is a given enumeration of the base X_0. Define then, in a slightly incomplete way (see the comment below), the collection $X_c \subseteq X$ of **computable elements** by $x \in X_c$ iff $\exists f \in R\ \{e_{f(i)}\}{\downarrow}x$.
In other words, given a countably based limit structure, an element is computable (or recursive) when it is the limit of a countable sequence indexed over an r.e. set.

Comment L-spaces actually carry too little structure to yield a good definition of “recursive” just by taking arbitrary limits of recursively enumerable converging sequences. Their simplicity and generality, though, should give an immediate intuition of what is going on. Indeed, the technique in the definition of computability tidily borrows from similar methods in mathematics. Consider, say, “smooth manifolds.” They are defined on the base of the familiar notion of differentiability in $\mathbb{R}^n$, which is extended by a system of local coordinates to abstract spaces. Similarly here, one takes for granted the recursive functions and extends computability to an abstract setting (and higher types, in particular) by "local" properties of convergence as in the equation above. As we shall see later, too, the categorical language relates and unifies the various classes of structures where this is done.

L-spaces suggest how to express computability by limits very simply; however, the weakness of these structures is that limits are far away from being uniquely determined and that there is no obvious way to characterize "interesting" sequences and limit points.

The point then is to take only "some" limits. This is done by directed sets in Scott domains (see 2.4.1), which are particular converging sequences with a privileged limit, the least upper bound. It will be described for **FIL** in the examples at the end of section 5.3.

3.5 Yoneda's Lemma

Let **C** be a Cartesian closed category. Let $\mathbf{F} = \mathbf{C}\times\mathbf{C}[\_,(a,b)] \circ \Delta : \mathbf{C}\to\mathbf{Set}$, $\mathbf{G} = \mathbf{C}[\_\times a,b]: \mathbf{C}\to\mathbf{Set}$. In §3.3 we proved that for both **F** and **G** there exists an object, respectively called $a\times b$ and b^a, such that $\mathbf{F} \cong \mathbf{C}[\_,a\times b]$ and $\mathbf{G} \cong \mathbf{C}[\_,b^a]$. In general, functors which enjoy this property are called (co-)**representable**. The formal definition, in case of covariant functors, is the following (note that **F** and **G** are contravariant; we leave as an exercise for the reader to derive the dual definition) :

3.5.1 Definition *A functor K: $\mathbf{C}\to\mathbf{Set}$ is representable iff there exist an object r in **C** and a natural isomorphism $\phi: K \cong \mathbf{C}[r,\_]$.*

Exercise Give another definition of CC and CCC by using the previous notion.

Note that, if $\mathbf{K}: \mathbf{C}\to\mathbf{Set}$, every natural transformation $\varphi: \mathbf{C}[r,\_]\to\mathbf{K}$ is fully determined by the image of the identity id_r . Indeed, for every $f\in\mathbf{C}[r,d]$, $\varphi_d(f) = \varphi_d(f \circ id_r) = \mathbf{K}(f) \circ \varphi_r(id_r)$, by the following diagram:

(Yoneda diagram)

$$\begin{tikzcd} \mathbf{C}[r,r] \arrow[r, "\varphi_r"] \arrow[d, "f\circ\_"'] & \mathbf{K}(r) \arrow[d, "\mathbf{K}(f)"] & r \arrow[d, "f"] \\ \mathbf{C}[r,d] \arrow[r, "\varphi_d"] & \mathbf{K}(d) & d \end{tikzcd}$$

3.5.2 Lemma (Yoneda) *Let $\mathbf{K}: \mathbf{C}\to\mathbf{Set}$ be a functor. The map $\psi_{r,\mathbf{K}}: Nat(\mathbf{C}[r,\_], \mathbf{K})\to\mathbf{K}(r)$ that takes every natural transfomation $\varphi: \mathbf{C}[r,\_]\to\mathbf{K}$ to $\varphi_r(id_r)\in\mathbf{K}(r)$ is an isomorphism. Moreover, it is natural in r and **K** , that is,*

$$\begin{tikzcd}
r \arrow[d, "f"'] & \mathbf{Nat}(\mathbf{C}[r,_],\mathbf{K}) \arrow[r, "\psi_{r,\mathbf{K}}"] \arrow[d, "_\circ\mathbf{C}[f,_]"'] & \mathbf{K}(r) \arrow[d, "\mathbf{K}(f)"] \\
d & \mathbf{Nat}(\mathbf{C}[d,_],\mathbf{K}) \arrow[r, "\psi_{d,\mathbf{K}}"] & \mathbf{K}(d)
\end{tikzcd}$$

(where $\mathbf{C}[f,\_] : \mathbf{C}[d,\_] \to \mathbf{C}[r,\_]$ *is the natural transformation defined by* $\mathbf{C}[f,\_]_c = \mathbf{C}[f,c] = \_\circ f : \mathbf{C}[d,c]\to\mathbf{C}[r,c]$*; see example 3.3.2), and*

$$\begin{tikzcd}
\mathbf{K} \arrow[d, "\mu"'] & \mathbf{Nat}(\mathbf{C}[r,_],\mathbf{K}) \arrow[r, "\psi_{r,\mathbf{K}}"] \arrow[d, "\mu\circ_"'] & \mathbf{K}(r) \arrow[d, "\mu_r"] \\
\mathbf{H} & \mathbf{Nat}(\mathbf{C}[r,_],\mathbf{H}) \arrow[r, "\psi_{r,\mathbf{H}}"] & \mathbf{H}(r)
\end{tikzcd}$$

Proof By the Yoneda diagram above and a routine verification of the commutativity of the diagrams in the definition. ♦

If we take $\mathbf{K}$ in the previous lemma to be $\mathbf{C}[d,\_]$, where d is a generic object in $\mathbf{C}$, this results in the statement that there is a natural bijection between arrows $g\in\mathbf{C}[d,r]$ (that is, elements in $\mathbf{C}[d,\_](r)$) and natural transformations from $\mathbf{C}[r,\_]$ to $\mathbf{C}[d,\_]$.

3.5.3 Proposition (Yoneda embedding)
i. Let Y *be the map which takes every* $r\in Ob_{\mathbf{C}}$ *to the hom-functor* $\mathbf{C}[r,\_]$*, and every* $g\in\mathbf{C}[d,r]$ *to the natural transformation* $\mathbf{C}[g,\_]: \mathbf{C}[r,\_]\to\mathbf{C}[d,\_]$ *defined by* $\mathbf{C}[g,\_]_c = \mathbf{C}[g,c] = \_\circ g : \mathbf{C}[r,c]\to\mathbf{C}[d,c]$*. Then* Y *is a full embedding from* $\mathbf{C}^{op}$ *to* ***Funct(C,Set)*** *.*
ii. Let Y *be the map which takes every* $r\in Ob_{\mathbf{C}}$ *to the hom-functor* $\mathbf{C}[\_,r]$ *, and every* $g\in\mathbf{C}[d,r]$ *to the natural transformation* $\mathbf{C}[\_,g]: \mathbf{C}[\_,d]\to\mathbf{C}[\_,r]$ *defined by* $\mathbf{C}[\_,g]_c = \mathbf{C}[c,g] = g\circ\_ : \mathbf{C}[c,d]\to\mathbf{C}[c,r]$ *. Then* Y *is a full and faithful covariant functor from* $\mathbf{C}$ *to* $\mathbf{Funct}(\mathbf{C}^{op},\mathbf{Set})$*.*
Proof We prove only (i), since the other proof is dual.

$Y(id_r) = id_{\mathbf{C}[r,\_]}$ is immediate. Given $g\in\mathbf{C}[d,r]$, $f\in\mathbf{C}[s,r]$, $Y(f\circ g): \mathbf{C}[s,\_]\to\mathbf{C}[d,\_]$ is defined as follows: for every $c\in Ob_{\mathbf{C}}$ $Y(g\circ f)_c = \_\circ f\circ g : \mathbf{C}[s,c]\to\mathbf{C}[d,c]$. Y is a functor, as, for every $h\in\mathbf{C}[s,c]$,

$$\begin{aligned}
Y(g\circ f)_c(h) &= h\circ f\circ g \\
&= Y(g)_c(f\circ h) \\
&= Y(g)_c(\,Y(f)_c(h)\,)
\end{aligned}$$

$= (Y(g)_c \circ Y(f)_c)(h).$

The fact that Y is full and faithful follows, as already observed, from the Yoneda lemma with **C**[d,_] instead of K. ♦

3.6 Presheaves

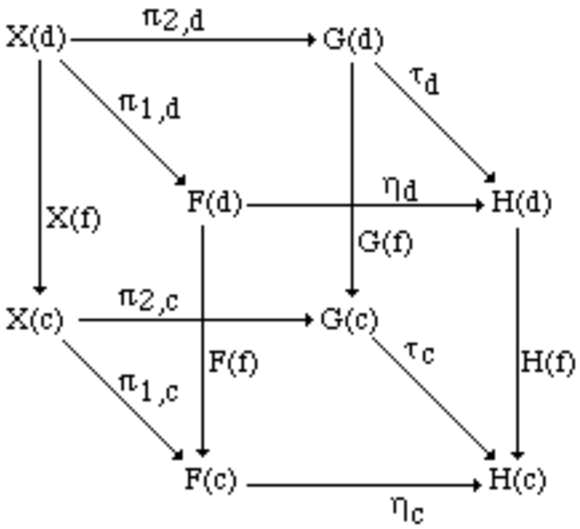
In the last section we proved that every small category **C** is isomorphic to a full subcategory of **Funct**($\mathbf{C}^{op}$,**Set**) through the Yoneda embedding Y. For its relevance, the category **Funct**($\mathbf{C}^{op}$,**Set**) has its own name: a functor F: $\mathbf{C}^{op}$□→**Set** is called **presheaf** on **C**, and **Funct**($\mathbf{C}^{op}$,**Set**) is the category of **presheaves** on **C**.

The category of presheaves inherits many interesting properties from **Set**, and in particular it is itself a topos. For the moment, we only prove the following properties:

3.5.1 Theorem *Given a category **C**, the category of presheaves on **C** has pullbacks for every pair of morphisms and is Cartesian closed.*

Proof. The terminal object is the functor T : $\mathbf{C}^{op}\to$**Set**, which takes every object c in $Ob_{\mathbf{C}}$ to the single set {*} and every arrow to the identity on this set.

Given two natural transformations η: F→H and τ: G→H , their pullback is defined objectwise: for every c in $Ob_{\mathbf{C}}$ let ($\pi_{1,c}$: $X_c \to F(c)$, $\pi_{2,c}$: $X_c \to G(c)$) be the pullback in **Set** of η_c: F(c)→H(c) and τ_c: G(c)→H(c) . Then define a functor X: $\mathbf{C}^{op}\to$**Set** where $X(c) = X_c$, and for every f:c→d, X(f) is the only arrow that makes the following diagram commute:



In the same way we define two natural transformations π_1: X→F, π_2: X→F, where for every c in $Ob_{\mathbf{C}}$, $\pi_1(c) = \pi_{1,c}$ and $\pi_2(c) = \pi_{2,c}$. Then it is easily verified that (π_1: X→F, π_2: X→F) is the pullback of η: F→H and τ: G→H.

As always, the pullback of two functors F and G on the terminal T gives the product $F\times G$. It is easy to verify that $F\times G(c) = F(c)\times G(c)$ and $F\times G(f) = F(f)\times G(f)$.

Exponents are defined by using the Yoneda lemma.

If G^F is the exponent of F and G, we must have an isomorphism $\mathbf{Nat}[H\times F,G] \cong \mathbf{Nat}[H,G^F]$ for every H. In particular, if H is $\mathbf{C}[\_,c]$, and since by the Yoneda lemma, $\mathbf{Nat}[\mathbf{C}[\_,c],K] \cong K(c)$ for every K, we have: $\mathbf{Nat}[\mathbf{C}[\_,c]\times F,G] \cong \mathbf{Nat}[\mathbf{C}[\_,c],G^F] \cong G^F(c)$. Thus, we *define* $G^F(c) = \mathbf{Nat}[\mathbf{C}[\_,c]\times F,G]$.

Given an arrow $f: c\to d$, we must now define $G^F(f): \mathbf{Nat}[\mathbf{C}[\_,d]\times F,G]\to\mathbf{Nat}[\mathbf{C}[\_,c]\times F,G]$. Let σ be a natural transformation from $\mathbf{C}[\_,d]\times F$ to G; then $G^F(f)(\sigma)$ must be a natural transformation from $\mathbf{C}[\_,c]\times F$ to G. We define $G^F(f)(\sigma) = \sigma \circ \mathbf{C}[\_,f]\times id_F$ (see example 3.2.2 for the definition of the natural transformation $\mathbf{C}[\_,f] : \mathbf{C}[\_,c]\to\mathbf{C}[\_,d]$).

G^F is a functor, indeed $G^F(id)(\sigma) = \sigma$. Moreover :

$$\begin{aligned} G^F(f \circ g)(\sigma) &= \sigma \circ \mathbf{C}[\_, f \circ g]\times id_F \\ &= \sigma \circ \mathbf{C}[\_,f]\times id_F \circ \mathbf{C}[\_,g]\times id_F \\ &= G^F(g)(\sigma \circ \mathbf{C}[\_,f]\times id_F) \\ &= G^F(g)(G^F(f)(\sigma)). \end{aligned}$$

Let us define the natural transformation of evaluation $\varepsilon_{F,G}: G^F\times F\to G$. For every d in $Ob_{\mathbf{C}}$, $\varepsilon_{F,G}(d): \mathbf{Nat}[\mathbf{C}[\_,d]\times F,G]\times F(d)\to G(d)$ is defined by $\varepsilon_{F,G}(d)(\sigma,n) = \sigma(d)(id_d,n)$.

We prove now that for any $H: \mathbf{C}^{op}\to\mathbf{Set}$ we have an isomorphism $\Theta: \mathbf{Nat}[H\times F,G] \cong \mathbf{Nat}[H,G^F]$. Let $\tau: H\times F\to G$ be a natural transformation. For any c in $Ob_{\mathbf{C}}$, $\Theta(\tau)(c)$ ought to be a function from $H(c)$ to $G^F(c) = \mathbf{Nat}[\mathbf{C}[\_,c]\times F,G]$. By the Yoneda lemma, we have for every c in $Ob_{\mathbf{C}}$ an isomorphism $\gamma_c: \mathbf{Nat}[\mathbf{C}[\_,c],H] \cong H(c)$; thus, if $m\in H(c)$,

$$\gamma_c^{-1}(m)\in\mathbf{Nat}[\mathbf{C}[\_,c],H]$$
$$\tau \circ (\gamma^{-1}(m)\times id_F) \in\mathbf{Nat}[\mathbf{C}[\_,c]\times F,G].$$

Define then $\Theta(\tau)(c) = \lambda m.\ \tau \circ (\gamma_c^{-1}(m)\times id_F): H(c)\to G^F(c) = \mathbf{Nat}[\mathbf{C}[\_,c]\times F,G]$.

We must prove that, for every $\tau: H\times F\to G$, $\mu: H\to G^F$,

1. $\varepsilon_{F,G} \circ (\Theta(\tau)\times id_F) = \tau$
2. $\Theta(\varepsilon_{F,G} \circ (\sigma\times id_F)) = \sigma$

For (1) we have, for every $d\in Ob_{\mathbf{C}}$, $m\in H(d)$, $n\in F(d)$, the following:

$$\begin{aligned} &(\varepsilon_{F,G}(d) \circ (\Theta(\tau)\times id_F)(d))(m,n) = \\ &= \varepsilon_{F,G}(d)\ (\Theta(\tau)(d)(m),n) \\ &= \varepsilon_{F,G}(d)\ (\tau \circ (\gamma_d^{-1}(m)\times id_F),n) && \text{by def. of } \Theta \\ &= (\tau \circ (\gamma_d^{-1}(m)\times id_F))(d)(id_d,n) && \text{by def. of } \varepsilon_{F,G} \\ &= \tau\ (d)(m,n) && \text{as } (\gamma_d^{-1}(m))(d)(id_d) = m \end{aligned}$$

For (2) we have, for every $d\in Ob_{\mathbf{C}}$, $m\in H(d)$, $c\in Ob_{\mathbf{C}}$, $h\in\mathbf{C}[c,d]$, $n\in F(d)$, the following:

$$\begin{aligned} &(\Theta(\varepsilon_{F,G} \circ (\mu\times id_F))(d)(m)(c)(h,n) = \\ &= (\varepsilon_{F,G} \circ (\mu\times id_F) \circ (\gamma_d^{-1}(m)\times id_F))(c)(h,n) && \text{by def. of } \Theta \end{aligned}$$

$= \varepsilon_{F,G}(c)\ (\ (\mu\times id_F)(c)(\ \gamma_d{}^{-1}(m)(c)(h),\ n\ )\ )$

$= \varepsilon_{F,G}(c)\ (\ (\mu\times id_F)(c)(\ H(h)(m),\ n\ )$ by def of $\gamma_d{}^{-1}$

$= \varepsilon_{F,G}(c)\ (\mu(c)(H(h)(m)),\ n)$

$= \mu(c)(H(h)(m))(c)\ (id_c,n)$ by def. of $\varepsilon_{F,G}(c)$

$= G^F(h)(\mu(d)(m))(c)\ (id_c,n)$ by naturality of μ

$= (\mu(d)(m) \circ \mathbf{C}[\_,h]\times id_F)\ (c)(id_c,n)$ by def. of $G^F(h)$

$= (\mu(d)(m)(c)\ ((\mathbf{C}[c,h]\times id_{F(c)})(id_c,n))$

$= (\mu(d)(m)(c)(h,n).$ ♦

References We only give some references for the examples we mentioned, as they are not usually presented in other books and are mostly endebted to the theory of computing. Partial equivalence relations as a model for higher type computations were given in Kreisel (1959) and later applied to the semantics of higher order intuitionistic logic in Girard (1972) and Troelstra (1973c). They recently came again to the limelight as relevant structures for the semantics of polymorphism in functional languages (see chapter 12). Limit spaces and their closure properties can be found in Kuratowski (1952). With our perspective, filter spaces are used in Hyland (1979).

For (pre)sheaves and related notions, the reader should consult the previous references for Topos Theory (as well as Fourman (1977) and Lambek and Scott (1986), among others). See also Scott (1980) for an application of Yoneda's embedding of arbitrary CCC's (and their reflexive objects, if any) into topoi of presheaves.