Towards Categorical logic for Kleisli computations

Author: Sybren Boland
Supervisor: Prof. dr. Bart Jacobs

July 28, 2014
Contents

Introduction .................................................. 2
Preliminaries ..................................................... 4

1 Monadic computations ........................................ 8
   1.1 Notions of computations ................................. 8
   1.2 Kleisli triples ........................................... 9
   1.3 Monads and Kleisli triples ............................. 10

2 Relating the Algebraic structure ............................ 12
   2.1 The triangle ............................................. 12
   2.2 Category of algebras .................................... 13
   2.3 The comparison functor .................................. 15

3 The Basic Adjunction .......................................... 18
   3.1 The functors ............................................. 18
   3.2 The adjunction ........................................ 21
   3.3 The adjunction for Sets ............................... 24

4 Relating program logic ....................................... 26
   4.1 Beck’s theorem ......................................... 26
   4.2 The Pred functor ........................................ 32

5 Non-deterministic computations ............................ 34
   5.1 The desired situation ..................................... 34
   5.2 Constructing the category ............................... 36
   5.3 Constructing the functors ............................... 39

Conclusions and Future work .................................. 43
Introduction

The main subject of this thesis is monads, mathematical objects that take some time getting used to but prove themselves worthy in a vast number of applications in both mathematics and computer science. They come to pass in category theory, a mathematical foundation that yields excellent tools for exploring connections between different fields of mathematics. This is exactly what we will do in this thesis.

The article from Bart Jacobs where our study of the subject began shows that we need to view (monadic) computations, viewed as a simple computer program, in unity with the algebraic structure and the logic of the program. Separately the three have been studied intensively. Computations have been interpreted as maps in the Kleisli category of the monad, the so called denotational semantics first done by Moggi. The algebraic structure of programs can be seen as the category of algebras of the monad. Moreover the program logic, in the form of predicate transformers, have been studied by Hoare and Dijkstra. Now we unify the three approaches in the following picture. It is just to keep in mind, the technical aspects are to come.

The picture above is mainly based on the monad that we begin with. Hence natural questions that arise would be: What is the most general category on which the monad can be defined for this triangle to appear? Is there a general way, in which for any monad, this triangle can be constructed? How do the three vertices relate to each other? These are questions that we will make precise, elaborate and answer if possible.

The thesis is organised as follows. The preliminaries introduce the type of categories and monads we will be working with as well as other notions not present in an undergraduate course in category theory. Section 1 will discuss the bottom of the triangle, illustrating that monads are a natural way of looking at computations. We continue in section 2 with the algebraic structure and its relation to the previous work. In section 3 we start with our own contributions to the subject. We generalise the connection overlapping the top two vertices
of the triangle to a more general category. Section 4 shows how to apply Beck’s tripleability theorem to our situation to filter out the left vertex of the triangle. Leaving section 5 to justify the previous section by spelling out the details in one example.
Preliminaries

In this section we will introduce the type of categories and monads we will be working with. We will follow the work of MacLane [1] and Jacobs [2].

**Definition 0.1.** A *monoidal category* \( C = \langle C, \otimes, I, \alpha, \lambda, \rho \rangle \) is a category \( C \), a bifunctor \( \otimes : C \times C \to C \), an object \( I \) of \( C \) and three natural isomorphisms \( \alpha, \lambda, \rho \). Where

\[
\alpha_{X,Y,Z} : X \otimes (Y \otimes Z) \congto (X \otimes Y) \otimes Z
\]

natural in \( X, Y \) and \( Z \) such that the pentagonal diagram

\[
\begin{array}{ccc}
X \otimes (Y \otimes (Z \otimes W)) & \overset{\alpha}{\rightarrow} & (X \otimes Y) \otimes (Z \otimes W) \\
\downarrow_{id \otimes \alpha} & & \downarrow_{\alpha \otimes id} \\
X \otimes ((Y \otimes Z) \otimes W) & \overset{\alpha}{\rightarrow} & (X \otimes (Y \otimes Z)) \otimes W
\end{array}
\]

commutes for all \( X, Y, Z, W \) in \( C \). Moreover

\[
\lambda_X : I \otimes X \congto X, \quad \rho_X : X \otimes I \congto X
\]

are natural in \( X \) such that the triangular diagram

\[
\begin{array}{ccc}
X \otimes (I \otimes Z) & \overset{\alpha}{\rightarrow} & (X \otimes I) \otimes Z \\
\downarrow_{id \otimes \lambda} & & \downarrow_{\rho \otimes id} \\
X \otimes Z & \cong & X \otimes Z
\end{array}
\]

commutes for all \( X, Z \) in \( C \), and also

\[
\lambda_I = \rho_I : I \otimes I \to I.
\]

**Definition 0.2.** A *monoidal closed* category \( C \) is a monoidal category \( C \) such that for every object \( X \) the functor \( (-) \otimes X : C \to C \) has a right adjoint, notated as

\[
(-) \otimes X \dashv (-)^X
\]
**Definition 0.3.** A *symmetric monoidal* category $C$ is a monoidal category $C$ with additional isomorphisms

$$\gamma_{X,Y} : X \otimes Y \xrightarrow{\simeq} Y \otimes X$$

natural in $X,Y$, which are idempotent i.e.

$$\gamma_{Y,X} \circ \gamma_{X,Y} = id_{X \otimes Y}$$

and for $I$ satisfy the following commutativity

$$X \otimes I \xrightarrow{\gamma} I \otimes X$$

\[\begin{array}{ccc}
\rho & \downarrow & \lambda \\
X & \rightarrow & X
\end{array}\]

and moreover, together with the associativity of $\alpha$, make the following hexagonal diagram commute:

$$((X \otimes Y) \otimes Z) \xrightarrow{\gamma} Z \otimes (X \otimes Y)$$

\[\begin{array}{ccc}
\alpha^{-1} & \downarrow & \alpha \\
X \otimes (Y \otimes Z) & \rightarrow & (Z \otimes X) \otimes Y \\
\downarrow & \downarrow & \downarrow \\
X \otimes (Z \otimes Y) & \rightarrow & (X \otimes Z) \otimes Y
\end{array}\]

If we combine the two previous definitions we get a *symmetric monoidal closed category* which will be the type of category we will be working with from now on. We will now specify the type of monads we will use, namely strong monads, which are a special kind of strong endofunctors.

**Definition 0.4.** Let $C$ be a symmetric monoidal closed category. An endofunctor $F : C \rightarrow C$ is called *strong* if it comes with a *strength* natural transformation with components:

$$st_{X,Y} : F(X) \otimes Y \rightarrow F(X \otimes Y)$$

that commutes with the $\alpha, \rho$ in the following way

$$\begin{array}{ccc}
(F(X) \otimes Y) \otimes Z & \xrightarrow{st \otimes id} & F(X \otimes Y) \otimes Z \\
\alpha & \downarrow & \downarrow F(\alpha) \\
F(X) \otimes (Y \otimes Z) & \xrightarrow{st} & F(X \otimes (Y \otimes Z))
\end{array}$$

and

$$\begin{array}{ccc}
F(X) \otimes I & \xrightarrow{st} & F(X \otimes I) \\
\rho & \downarrow & \downarrow F(\rho) \\
F(X) & \rightarrow & F(X)
\end{array}$$
Sometimes we need the strength in its ‘swapped’ form, namely \( st' \) with components

\[
\begin{align*}
st'_{X,Y} = (X \otimes T(Y) \xrightarrow{\gamma} T(Y) \otimes X \xrightarrow{st} T(Y \otimes X) \xrightarrow{T(\gamma)} T(X \otimes Y))
\end{align*}
\]

Now automatically the following diagram commutes

\[
\begin{tikzcd}
I \otimes F(X) \arrow{rr}{st'} \arrow{dr}{\lambda} & & F(I \otimes X) \arrow{d}{F(\lambda)} \\
& F(X) &
\end{tikzcd}
\]

Note that every endofunctor on \( \text{Sets} \) is strong. For we have a natural strength function given by:

\[
st : T(X) \times Y \to T(X \times Y); (T(x), y) \mapsto T(x, y)
\]

for \( x \in X \) and \( y \in Y \).

**Definition 0.5.** A monad \((T, \eta, \mu)\) on a symmetric monoidal closed category \( \mathbb{C} \) is called strong if \( T \) is a strong endofunctor which commutes with unit \( \eta \) and multiplication \( \mu \) in the following way:

\[
\begin{tikzcd}
X \otimes Y \arrow{r}{\eta \otimes id} & T(X) \otimes Y \arrow{r}{st} & T(X \otimes Y) \arrow{d}{\eta}
\end{tikzcd}
\]

and

\[
\begin{tikzcd}
T^2(X) \otimes Y \arrow{r}{st} & T(T(X) \otimes Y) \arrow{r}{T(st)} & T^2(X \otimes Y) \arrow{d}{\mu}
\end{tikzcd}
\]

Sometimes we will need an equivalent strength map for exponentials. Therefore we define

\[
r = \Lambda(T(ev) \circ st) : T(Y^X) \to T(Y)^X
\]

where inside the \( \Lambda(\cdot) \) we use the composite

\[
\begin{tikzcd}
T(Y^X) \otimes X \arrow{r}{st} & T(Y^X \otimes X) \arrow{r}{T(ev)} & T(Y)
\end{tikzcd}
\]

Another bit of terminology that may be unknown to the reader is that of a reflexive pair of arrows.
Definition 0.6. Let $C$ be a category and $f, g : A \to B$ be a pair of parallel arrows. Then $f, g$ is called a reflexive pair if there exists a mutual section for $f$ and $g$ i.e. there exists an arrow $s : B \to A$ such that

$$f \circ s = id_B = g \circ s.$$
1. Monadic computations

All sorts of computations can be described as maps in the Kleisli category, \( K\ell(T) \), of a particular monad \( T \). More precisely, kleisli maps give a categorical semantics of computations. In this section we will describe the so-called denotational semantics of programming languages by Moggi [7]. We implicitly restrict ourselves to a monadic setting, as we will see later on, but this is of no concern now.

1.1 Notions of computations

The leading motivation in interpreting computations in a category \( C \), is that we need to distinguish the value of type \( A \) and the computation of type \( A \). We do so by identifying an object \( A \) of \( C \) with the value of type \( A \). And for a notion of computation \( T \), we identify an object \( T(A) \) of \( C \) with a computation of type \( A \). Later on we will further investigate this notion of computation, for now we content ourselves with the following examples.

Example 1.1.

- Non-deterministic: \( T(A) = \mathcal{P}(A) \)
- Partial: \( T(A) = A + \bot \), where \( \bot \not\in A \) is the diverging computation.
- Side-effects: \( T(A) = (A \times S)^S \), where \( S \) is a set of states.
- Distribution: \( T(A) = \{ \phi : X \rightarrow [0,1] \mid \text{supp}(\phi) \text{ is finite, and } \Sigma x \phi(x) = 1 \} \)
- Continuations: \( T(A) = R^{(R^A)} \), where \( R \) is a set of results.

There are many more examples but from now on we will work with the first two: non-deterministic and partial computations. In order to interpret these notions we will only impose that these computations form a category. A more systematic (but equivalent) approach imposes that \( T \) is part of a Kleisli triple.
1.2 Kleisli triples

Definition 1.2. A Kleisli triple over a category $\mathcal{C}$ is a triple $(T, \eta, (\cdot)^*)$, where $T : \text{Obj}(\mathcal{C}) \to \text{Obj}(\mathcal{C})$, $\eta_A : A \to T(A)$ for $A$ object of $\mathcal{C}$ and $f^* : T(A) \to T(B)$ for $f : A \to T(B)$ in $\mathcal{C}$, such that the following equations hold:

$$
\eta_A^* = id_{T(A)} \\
f^* \circ \eta_A = f \\
g^* \circ f^* = (g^* \circ f)^*
$$

for any $A, f : A \to T(B)$ and $g : B \to T(C)$ in $\mathcal{C}$.

Intuitively we interpret $\eta$ as the inclusion of values into computations. And $f^*$ as the extension of $f$ from a map from values to computations to a map from computations to computations, which first evaluates a computation and then applies $f$ to the resulting value. To justify the axioms for a Kleisli triple we define an associated category where the morphisms correspond to computations.

Definition 1.3. Let $(T, \eta, (\cdot)^*)$ be a Kleisli triple. We define the Kleisli category $\mathcal{K}(T, \eta, (\cdot)^*)$ of the triple as follows

- Object are objects of $\mathcal{C}$.
- Arrow $f : A \to B$ is arrow $f : A \to T(B)$ in $\mathcal{C}$.
- Identity map on $A$ is $\eta_A : A \to T(A)$.
- Composition of $f : A \to B$ and $g : B \to C$, notated as $g \bullet f$, is $g^* \circ f$.

There is now an easy justification of the axioms for a Kleisli triple because they correspond to the identity and associativity axioms for $\mathcal{K}(T, \eta, (\cdot)^*)$ being a category.

Example 1.4. We see that our two working examples are part of a Kleisli triple:

- Non-deterministic: $T(A) = \mathcal{P}(A)$
  
  Where $\eta_A(a) = \{a\}$
  
  and for $f : A \to \mathcal{P}(B)$ and $C \in \mathcal{P}(A)$ we have $f^*(C) = \bigcup_{c \in C} f(c)$

The Kleisli category can now be seen as that of Sets and non-deterministic functions. For a function $f : A \to \mathcal{P}(B)$ represents all possible output values $f(a) \subseteq B$, for $a \in A$.

If we wish to compose two of these functions $f : A \to \mathcal{P}(B)$ and $g : B \to \mathcal{P}(C)$ intuitively want the composition, for $a \in A$, to compute $f(a) \subseteq B$ and then for all $b \in f(a)$ to collect all $g(b)$. Hence we see this is exactly

$$
\bigcup_{b \in f(a)} g(b) = g^*(f(a)) = g \bullet f(a)
$$
- Partial: $T(A) = A + \bot$
  Where $\eta_A : A \to A + \bot$ is the inclusion of $A$
  and for $f : A \to T(B)$ we have $f^* (\bot) = \bot$ and $f^* (a) = a$ otherwise.

The Kleisli category can now be seen as that of Sets and \emph{partial functions}.
For a function $f : A \to B + \bot$ and $a \in A$, one says that $f$ has no value at $a$ if $f(a) = \bot$.
If we wish to compose two of these functions $f : A \to B + \bot$ and $g : B \to C + \bot$ then we want the composition to have no value on those $a$ such that either $f(a) = \bot$ or $f(a) \in B$ and $g(f(a)) = \bot$. This is exactly achieved by extending $g$ and then pre-composing with $f$ i.e. $g^* \circ f = g \bullet f$.

In functional programming languages, in particular in Haskell, these Kleisli triples appear in the form of a $T$ together with a \emph{return} function corresponding to $\eta$ and a \emph{bind} function corresponding to the $(\cdot)^*$-composition.

### 1.3 Monads and Kleisli triples

We will now relate to the title of this section by stating that a Kleisli triple is just another way of defining a monad. As we’ve seen the Kleisli triple has a strong connection to computations, but the monadic way will be more useful in our further work.

**Proposition 1.5.** There is a bijective correspondence between Kleisli triples and monads.

**Proof.** (sketch)
We will give the constructions without proving that the result is indeed of the desired form. These proofs are straightforward and only use axioms of a monad and Kleisli triple resp.
Let $(T, \eta, (\cdot)^*)$ be a Kleisli triple. Then we construct a associated monad by:
- Extend $T$ to a (endo)functor by defining $T$ on functions:
  $T(f : X \to Y) = (\eta \circ f)^*$
- For an object $A$ let $\mu_A = (id_{T(A)})^*$

Conversely, let $(T, \eta, \mu)$ be a monad. Then the associated Kleisli triple is given by:
- Let $T$ be the functor $T$ restricted to objects
- For $f : X \to Y$ let $f^* = \mu_Y \circ T(f)$

(In both constructions $\eta$ stays the same)
We will now prove that these constructions are mutually inverses of each other. Let \((T, \eta, (-)^\ast)\) be a KLeisli triple and \(f : X \to Y\) then
\[
\mu_Y \circ T(f) = (id \circ \eta_Y \circ f)^\ast = (id \circ \eta_Y \circ f)^\ast = (id \circ f)^\ast = f^\ast
\]
And if \((T, \eta, \mu)\) is a monad, then
\[
(id_T(X))^\ast = \mu_T(X) \circ T(id_T(X)) = \mu_T(X)
\]
Which concludes the proof.

\[\Box\]

**Example 1.6.** We will now see the monads are formed in our examples.

- **Non-deterministic:** We have \(\mu_A : \mathcal{P}^2(A) \to \mathcal{P}(A)\) where
  \(\mu_A(C) = (id_{\mathcal{P}(A)})^\ast(C) = \bigcup_{B \in C} B\)
  Hence the associated monad is \((\mathcal{P}, \{-\}, \cup)\).

- **Partial:** We have
  \(\mu_A : A + \{\perp\} + \{\perp'\} \to A + \{\perp\} : a \mapsto \begin{cases} a & \text{if } a \in A \\ \perp & \text{otherwise} \end{cases}\)
  so \(\mu_A(\perp_1) = \perp = \mu_A(\perp)\).

We can now define the Kleisli category associated with a monad \(T\).

**Definition 1.7.** Let \(T\) be a monad on \(C\). Then the Kleisli category associated with \(T\), \(\mathcal{K}\ell(T)\), is defined as:

- **Objects** are the objects of \(C\).
- **Morphisms** \(f : X \to Y\) are morphisms \(f : X \to T(Y)\) is \(C\).
- **Identity arrow** on object \(X\) is \(\eta_X\).
- **Composition** of \(f : X \to Y\) and \(g : Y \to Z\) is given by \(g \bullet f = \eta_Z \circ T(g) \circ f\).

Note that by the previous proposition this is just a reformulation of our former definition of a Kleisli category. Hence the morphisms of \(\mathcal{K}\ell(T)\) also correspond to computations associated with \(T\).
2. Relating the Algebraic structure

In this section we will introduce the interpretation of the triangle from the introduction. This diagram will haunt us the rest of this thesis. We go on exploring the upper right vertex of this diagram, dealing with the algebraic structure of the computations. We go on spelling out what this means in our examples and relating it to the previous section.

2.1 The triangle

Previously we have seen that monads are a good way of modelling computations. Recently they have been used for program verification. These monads not only describe the program but also their correctness assertions, hence combining semantics and logic of programs. Our motivating article [3], and also earlier work of Jacobs [4], [5], strongly suggest that we look at this in a unified way. Giving rise to a triangle of the following form:

\[
\begin{array}{c}
\text{Log}^{op} = \left\{ \text{predicate transformers} \right\} \\
\text{computations} \\
\text{Pred} \\
\text{Stat} \\
\end{array}\xrightarrow{\top} \begin{array}{c}
\text{transformers} \\
\text{state} \\
\end{array}
\]

Where the nodes are categories for which the morphisms are described. The arrows are functors of which the top two form an adjunction. Moreover the triangle needs to commute in both directions. This picture corresponds to that in the introduction, now with more technical details. We shall refer to it as the triangle.
2.2 Category of algebras

In our monadic setting we saw that the bottom vertex translates to the Kleisli category. The Kleisli category of $T$ can be related to the algebraic structure of the monad i.e. the category of Eilenberg-Moore algebras. In this category objects can be interpreted as states and morphisms as programs, yielding an output state for given input. The upper right vertex is therefore named with 'state transformers'.

Example 2.1. We will give alternative description of the $\mathcal{EM}$-category in our examples.

- Non-deterministic computations:
  We have already seen that the associated monad is given by $(\mathcal{P}, \{-\}, \cup)$. Claim: $\mathcal{EM}(\mathcal{P}) \cong \mathcal{C}l\bot$

  We define $F : \mathcal{EM}(\mathcal{P}) \to \mathcal{C}l\bot$ where $F(X,\alpha) = X$ such that for $A \subseteq X$ we have $\bigvee A = \alpha(A)$ and $F(f : (X,\alpha) \to (Y,\beta)) = f$. Hence we have

\[
x \leq y \iff \alpha(\{x\}) = y
\]

We have the following two equalities, for $x \in X$ and $U = \{U_i \mid i \in I\}$:

1. $\alpha(\{x\}) = \alpha \circ \eta_X(x) = x$
2. $\alpha(\bigcup_{i \in I} U_i) = \alpha \circ \mu_X(U) = \alpha \circ \mathcal{P}(\alpha)(U) = \alpha(\{\alpha(U_i) \mid i \in I\})$

Then $\leq$ defines a partial order on $X$. For $x, y, z \in X$:

- $x \leq x$ iff $\alpha(\{x\}) = \alpha \circ \eta_X(x) = x$
- Suppose $x \leq y$ and $y \leq x$ then $x = \alpha(\{x, y\}) = \alpha(\{y, x\}) = y$
- Suppose $x \leq y$ and $y \leq z$ then

\[
\alpha(\{x, z\}) = \alpha(\{\alpha(x), \alpha(z)\}) \quad (1)
\]
\[
= \alpha(\{x, \{x, z\}\}) \quad (2)
\]
\[
= \alpha(\{x, y\} \cup \{z\})
\]
\[
= \alpha(\{\alpha(x, y), \alpha(z)\}) \quad (2)
\]
\[
= \alpha(\{y, z\}) \quad (1)
\]
\[
= z
\]

Moreover $\bigvee$ indeed defines a join. Let $U \subseteq X$ and $x \in U$ then

\[
\alpha(\{x, \alpha(U)\}) = \alpha(\{\alpha(\{x\}), \alpha(U)\}) \quad (1)
\]
\[
= \alpha(\{x\} \cup \alpha(U)) \quad (2)
\]
\[
= \alpha(U)
\]
hence for all \( x \in U \) we have \( x \leq \bigvee U \). Now suppose that there exists a \( y \) such that \( x \leq y \) for all \( x \in U \), then
\[
\alpha(\{\alpha(U), y\}) = \alpha(\{\alpha(U), \alpha(\{y\})\}) = \alpha(U \cup \{y\}) = \alpha(\bigcup_{x \in U} \{x, y\}) = \alpha(\{\alpha(\{x, y\}) | x \in U\}) = \alpha(\{y | x \in U\}) = \alpha(\{y\}) = y \quad (1)
\]
Hence \( \bigvee \) defines a join. Also for \( f : X \to Y \) we have
\[
F(f)(\bigvee U) = f \circ \alpha(U) = \beta \circ P(f)(U) = \bigvee f(U) = \bigvee F(f)(U)
\]
Moreover \( F \) obviously preserves identity and composition, thus \( F \) defines a functor.

We also define \( G : Cl_\bigvee \to E\mathcal{M}(\mathcal{P}) \) where \( G(X) = (X, \alpha_X) \) such that \( \alpha_X(U) = \bigvee U \) and \( G(f : X \to Y) = f \). Then for \( y \in X \) and \( A = \{A_i | i \in I\} \) we have
\[
\alpha_X \circ \eta_X(y) = \alpha_X(\{y\}) = \bigvee \{y\} = y
\]
and
\[
\alpha_X \circ P(\alpha_X)(A) = \bigvee \{\alpha_X(A_i) | i \in I\} = \bigvee_{i \in I} \bigvee A_i = \bigvee \bigcup_{i \in I} A_i = \alpha_X \circ \mu_X(A)
\]
hence \( G(X) \) is a \( \mathcal{P} \)-algebra. And by definition \( G(f) \) is a \( \mathcal{P} \)-algebra morphism. Obviously \( G \) preserves identity and composition thus defines a functor. Now we will show that \( F \) and \( G \) are each others inverses.

We have \( G \circ F(X, \alpha) = G(X, \leq_X) = (X, \alpha_X) \) where for \( U \subseteq X \)
\[
\alpha_X(U) = \bigvee U( \text{ in } (X, \leq_X)) = \alpha(U)
\]
and \( F \circ G(X, \leq_X) = F(X, \alpha_X) = (X, \leq) \) where for \( x, y \in X \)
\[
x \leq y \iff y = \alpha_X(\{x, y\}) = \bigvee \{x, y\} \iff x \leq_X y
\]
Thus proving the isomorphism \( E\mathcal{M}(\mathcal{P}) \cong Cl_\bigvee \).

- Partial computations. Claim: \( E\mathcal{M}(\mathcal{L}) \cong Sets_\ast \)

We define \( F : E\mathcal{M}(\mathcal{L}) \to Sets_\ast \) where \( F(X, \alpha) = (X, \alpha(\bot)) \) and \( F(f : (X, \alpha) \to (Y, \beta)) = f \).
Then \( F(f)(\alpha(\bot)) = f(\alpha(\bot)) = \beta \circ (f + 1)(\bot) = \beta(\bot) \) so \( F(f) \) is well-defined. By a similar argument we see that \( F \) preserves composition. It obviously preserves identity thus defines a functor.

Now we define \( G : \text{Sets}_* \to \mathcal{E}M(L) \)

where \( G(X,x_0) = (X,\alpha_X) \) and \( \alpha_X : X+1 \to X \ ; \ x \mapsto \begin{cases} x_0 & \text{if } x = \bot \\ x & \text{otherwise} \end{cases} \)

and \( G(f : (X,x_0) \to (Y,y_0)) = f. \)

Then \( G(f) \circ \alpha_X(\bot) = G(f)(x_0) = f(x_0) = y_0 = \alpha_Y(\bot) = \alpha_Y \circ (G(f) + 1)(\bot) \) and for similar for \( x \in X. \) So \( G(f) \) is well-defined. Again \( G \) is obviously a functor. Now we will show that \( F \) and \( G \) are inverses of each other:

\[
F \circ G(Y,y_0) = F(Y,\alpha_Y) = (Y,\alpha_Y(\bot)) = (Y,y_0)
\]

Moreover \( G \circ F(Y,\beta) = G(Y,\beta(\bot)) = (Y,\alpha_Y) \) where \( \alpha_Y(\bot) = \beta(\bot) \) and \( \alpha_Y(y \in Y) = y = \beta \circ \eta_X(y) = \beta(y). \) Hence \( G \circ F(Y,\beta) = (Y,\beta). \)

On morphisms \( F \) and \( G \) are obviously inverses thus proving the isomorphism: \( \mathcal{E}M(L) \cong \text{Sets}_*. \)

### 2.3 The comparison functor

These categories of algebras have a special status. They are the terminal object in the category of adjunctions inducing that monad. We will not make all this precise, we will content ourselves with the following proposition.

**Proposition 2.2.** Let \( F \dashv G : \mathbb{C} \to \mathbb{A}. \) Then \( T = GF \) induces a monad and we have the following situation

\[
\begin{array}{ccc}
\mathbb{A} & \xrightarrow{\top} & \mathbb{C} \\
\downarrow{F} & & \downarrow{U^T} \\
\mathcal{E}M(T) & \xrightarrow{K} & \end{array}
\]

where the **comparison functor**

\[
K : \mathbb{A} \to \mathcal{E}M(T)
\]

is the unique functor such that \( K \circ F = F \) and \( U^T \circ K = G \) given by

\[
K(A) = (G(A),G(\epsilon_A)) \\
K(f : X \to Y) = G(f)
\]

15
Proof. We will first show that $K$ is well-defined. We have
\[ G(\epsilon_A) \circ \eta_{G(A)} = \text{id}_{G(A)} \]
by a triangular identity. Moreover
\[
G(\epsilon_A) \circ \mu_{G(A)} = G(\epsilon_A) \circ G(\epsilon_{FG(A)}) \\
= G(\epsilon_A \circ \epsilon_{FG(A)}) \\
= G(\epsilon_A \circ FG(\epsilon_A)) \\
= G(\epsilon_A) \circ TG(\epsilon_A)
\]
Hence $K(A)$ is a $T$-algebra. And for $f : A \to B$ we have that $K(f)$ is a $T$-algebra
morphism for
\[
K(f) \circ G(\epsilon_A) = G(f) \circ G(\epsilon_A) \\
= G(\epsilon_B \circ FG(f)) \\
= G(\epsilon_B) \circ TK(f)
\]
It is straightforward to check the commuting properties. Now suppose that $H : \mathbb{A} \to \mathcal{EM}(T)$ is a functor such that $H \circ F = FT$ and $U^T \circ H = G$. Then by the later equation we have on arrows $H(f) = G(f) = K(f)$ and on objects $H(A) = (G(A), \alpha)$ where $\alpha$ is the structure map of the algebra. But we can say for the arrow $\epsilon_A$
\[
G(\epsilon_A) = H(\epsilon_A) = \epsilon^T_{H(A)} = \epsilon^T_{G(A), \alpha} = \alpha
\]
where the second equality has to be proven to conclude the proof. Now both
adjunctions have the same unit. So if we denote the adjunction isomorphisms
by $\phi$ and $\phi^T$ we have for an arrow $g : F(C) \to A$
\[
\phi(g) = G(g) \circ \eta_C = U^T \circ H(g) \circ \eta_C = \phi^T(H(g))
\]
in particular for the arrow $g = \phi^{-1}(id_{G(A)})$ we have the third equality in
\[
H(\epsilon_A) = H(\phi^{-1}(id_{G(A)})) \\
= (\phi^T)^{-1} \circ \phi^T \circ H(\phi^{-1}(id_{G(A)})) \\
= (\phi^T)^{-1} \circ \phi \circ \phi^T(\epsilon_{G(A)}) \\
= (\phi^T)^{-1}(id_{U^T \circ H(A)}) \\
= \epsilon^T_{H(A)}
\]
Thus $H = K$ is the unique functor with these commuting properties. \( \square \)

In the case that $\mathbb{A} = \mathcal{K}(T)$, the comparison functor takes the following form
by definition of the functors associated with the Kleisli category.
\[
K : \mathcal{K}(T) \to \mathcal{EM}(T)
\]
Sending an object $X$ of $\mathbb{C}$ to the free algebra on $X$
\[
K(X) = \mu_X : T^2(X) \to T(X)
\]
And for a map \( f : X \to Y \) in \( \mathcal{K}\ell(T) \) we have
\[
K(f) = \mu_Y \circ T(f) : T(X) \to T(Y)
\]
which is a \( T \)-algebra morphism because \( \mu_Y \) and \( T(f) \) are. Functorality is obtained next, by recalling what identity and composition are in \( \mathcal{K}\ell(T) \):
\[
K(\eta_X) = \mu_X \circ T(\eta_X) = id_{T(X)}
\]
\[
K(g \bullet f) = \mu_Z \circ T(g \bullet f)
\]
\[
= \mu_Z \circ T(\mu_Z \circ T(g) \circ f)
\]
\[
= \mu_Z \circ T(\mu_Z) \circ T^2(g) \circ T(F)
\]
\[
= \mu_Z \circ T(g) \circ \mu_Y \circ T(f)
\]
\[
= K(g) \circ K(f)
\]
From now on we will reserve the letter \( K \) for this form of the comparison functor.

**Lemma 2.3.** The comparison functor \( K : \mathcal{K}\ell(T) \to \mathcal{E}\mathcal{M}(T) \) is full and faithful.

**Proof.** Let \( f, g : X \to Y \) in \( \mathcal{K}\ell(T) \) given by \( f, g : X \to T(Y) \) in \( \mathbb{C} \) such that
\[
\mu_Y \circ T(f) = K(f) = K(g) = \mu_Y \circ T(g)
\]
Then
\[
f = \mu_Y \circ \eta_{T(Y)} \circ f
\]
\[
= \mu_Y \circ T(f) \circ \eta_X
\]
\[
= \mu_Y \circ T(g) \circ \eta_X
\]
\[
= g
\]
Hence \( K \) is faithful. And if \( \alpha : \mu_X \to \mu_Y \) in \( \mathcal{E}\mathcal{M}(T) \) given by \( \alpha : T(X) \to T(Y) \) in \( \mathbb{C} \) then
\[
f = f \circ \mu_X \circ T(\eta_X)
\]
\[
= \mu_Y \circ T(f) \circ T(\eta_X)
\]
\[
= \mu_Y \circ T(f \circ \eta_X)
\]
\[
= K(f \circ \eta_X)
\]
Hence \( K \) is full. \( \square \)

Recapitulating we have introduced the triangle as a unified way of three approaches to monadic computations and explored its right hand side, dealing with the algebraic structure of these computations.
3. The Basic Adjunction

To further interpret the triangle from the previous section we need to find the appropriate adjunction. As we can see in [3] there is an adjunction $\text{Hom}(\cdot, \omega) \dashv \text{Hom}(\cdot, \Omega)$ for every monad $T$ on $\text{Set}$ and a fixed algebra $\omega : T(\Omega) \to \Omega$. We will now generalise this adjunction to a monad on a more general category, namely to a symmetric monoidal closed category. We will begin this section by defining the two functors which we will prove to be adjoints of each other towards the end.

3.1 The functors

Definition 3.1. Let $\mathcal{C}$ be symmetric monoidal closed category and let $\mathcal{E}\mathcal{M}(T)$ be the category of Eilenberg-Moore algebras of a strong monad $T$ on $\mathcal{C}$. Then for a fixed algebra $\omega : T(\Omega) \to \Omega$ define:

$$\omega(\cdot) : \mathcal{C}^{\text{op}} \to \mathcal{E}\mathcal{M}(T)$$

where for an object $X$

$$\omega_X = (T(\Omega^X) \xrightarrow{\tau} T(\Omega)^Y \xrightarrow{\omega^X} \Omega^X)$$

and for an arrow $f : X \to Y$

$$\omega_f = \Omega^f = \Lambda(ev \circ (id \otimes f)) : \Omega^Y \to \Omega^X$$

Lemma 3.2. The just defined $\omega(\cdot) : \mathcal{C}^{\text{op}} \to \mathcal{E}\mathcal{M}(T)$ defines a functor.

Proof. Note that we have the following identity for $X$ object in $\mathcal{C}$:

$$\omega_X = r \circ \omega^X$$

$$= \Lambda(\omega \circ ev) \circ \Lambda(T(ev \circ st))$$

$$= \Lambda(\omega \circ ev \circ (\Lambda(T(ev) \circ st)) \otimes id))$$

$$= \Lambda(\omega \circ T(ev) \circ st)$$
First we show that $\omega_X : T(\Omega^X) \to \Omega^X$ is a $T$-algebra. Namely

$$\omega_X \circ \eta = \Lambda(\omega \circ T(ev) \circ st) \circ \eta$$
$$= \Lambda(\omega \circ T(ev) \circ st \circ (\eta \otimes id))$$
$$= \Lambda(\omega \circ T(ev) \circ \eta)$$
$$= \Lambda(\omega \circ \eta \circ ev)$$
$$= \Lambda(ev)$$
$$= id_{\Omega^X}$$

Moreover we have

$$\omega_X \circ \mu = \Lambda(\omega \circ T(ev) \circ st) \circ \mu$$
$$= \Lambda(\omega \circ T(ev) \circ st \circ (\mu \otimes id))$$
$$= \Lambda(\omega \circ T(ev) \circ \mu \circ T(st) \circ st)$$
$$= \Lambda(\omega \circ T(st) \circ T(ev) \circ st)$$
$$= \Lambda(\omega \circ T(st) \circ T(ev) \circ (\omega_X \otimes id)) \circ st$$
$$= \Lambda(\omega \circ (T(ev) \circ st \circ (\omega_X \otimes id)) \circ st)$$
$$= \omega_X \circ T(\omega_X)$$

And for $f : X \to Y$ in $\mathbb{C}$: $\omega_f$ is a $T$-algebra morphism.

$$\omega_f \circ \omega_Y = \Lambda(ev \circ (id \otimes f)) \circ \Lambda(\omega \circ T(ev) \circ st)$$
$$= \Lambda(ev \circ (id \otimes f) \circ (\Lambda(\omega \circ T(ev) \circ st) \otimes id))$$
$$= \Lambda(ev \circ (\Lambda(\omega \circ T(ev) \circ st) \otimes id) \circ (id \otimes f))$$
$$= \Lambda(\omega \circ T(ev) \circ st \circ (id(id \otimes f)))$$
$$= \Lambda(\omega \circ T(id \otimes f) \circ st)$$
$$= \Lambda(\omega \circ T(id \otimes f) \circ st)$$
$$= \Lambda(\omega \circ T(id \otimes f) \circ st)$$
$$= \omega_X \circ T(\omega_f)$$

It remains to prove that $\omega_{(-)}$ preserves identities and composition. But

$$\omega_{id} = \Lambda(ev \circ (id \otimes id)) = \Lambda(ev) = id$$

and if $f : Z \to Y$ and $g : Y \to X$ in $\mathbb{C}$ then

$$\omega_f \circ \omega_g = \Lambda(ev \circ (id \otimes f)) \circ \Lambda(ev \circ (id \otimes g))$$
$$= \Lambda(ev \circ (id \otimes f) \circ (\Lambda(ev \circ (id \otimes g) \otimes id))$$
$$= \Lambda(ev \circ (\Lambda(ev \circ (id \otimes g) \otimes id) \circ (id \otimes f))$$
$$= \Lambda(ev \circ (id \otimes g) \circ (id \otimes f))$$
$$= \Lambda(ev \circ (id \otimes (g \circ f)))$$
$$= \omega_{gf}$$

Thus we conclude that $\omega_{(-)}$ is a functor. $\square$
**Definition 3.3.** Let $\mathcal{C}$ be symmetric monoidal closed category with equalizers, $\mathcal{EM}(T)$ the category of algebras of a strong monad $T$ on $\mathcal{C}$ and $\omega : T(\Omega) \to \Omega$ a fixed algebra.

Define

$$H_\omega : \mathcal{EM}(T) \to \mathcal{C}^{\text{op}}$$

For a $T$-algebra $\alpha : T(A) \to A$ let $H_\omega(\alpha)$ be the equalizer of arrows $\omega_\alpha$ and $\Lambda(\omega \circ T(ev) \circ st)$, as in the following diagram:

$$H_\omega(\alpha) \xrightarrow{e_\alpha} \Omega^A \xrightarrow{\omega_\alpha} \Omega T(A)$$

And for a $T$-algebra morphism $f : (A, \alpha) \to (B, \beta)$ let $H_\omega(f)$ be the unique arrow $h : H_\omega(\beta) \to H_\omega(\alpha)$ such that $e_\alpha \circ h = \Omega^f \circ e_\beta$ (which has to be proven to exist) as in the following diagram.

$$H_\omega(\alpha) \xrightarrow{e_\alpha} \Omega^A \xrightarrow{\omega_\alpha} \Omega T(A)$$

$$H_\omega(\beta) \xrightarrow{e_\beta} \Omega^B \xrightarrow{\omega_\beta} \Omega T(B)$$

**Lemma 3.4.** The just defined $H_\omega : \mathcal{EM}(T) \to \mathcal{C}^{\text{op}}$ is a functor.

**Proof.** First note that $H_\omega(f)$ is well-defined if

$$\omega_\alpha \circ \Omega^f \circ e_\beta = \Lambda(\omega \circ T(ev) \circ st) \circ \Omega^f \circ e_\beta$$

And because $e_\beta$ is the equaliser of the arrows on the bottom of the rectangle in the diagram above, the equation is valid if both these parallel rectangles commute. Now the first rectangle commutes for $f$ is a $T$-algebra morphism i.e.

$$\omega_\alpha \circ \Omega^f = \Omega^\alpha \circ \Omega^f = \Omega^{f \circ \alpha} = \Omega^{\beta \circ T(f)} = \Omega^T(f) \circ \omega_\beta$$

The second rectangle will take some more effort, but

$$\Lambda(\omega \circ T(ev) \circ st') \circ \Omega^f$$

$$= \Lambda(\omega \circ T(ev) \circ st' \circ (\Omega^f \otimes id_{T(A)}))$$

$$= \Lambda(\omega \circ T(ev) \circ st' \circ (\Omega^f \otimes T(id_A)))$$

$$= \Lambda(\omega \circ T(ev) \circ T(\Omega^f \otimes id) \circ st')$$

$$= \Lambda(\omega \circ T(ev \circ (id \otimes f)) \circ st')$$

$$= \Lambda(\omega \circ T(ev \circ st' \circ (id \otimes T(f)))$$

$$= \Lambda(ev \circ (\Lambda(\omega \circ T(ev) \circ st') \otimes id) \circ (id \otimes T(f)))$$

$$= \Lambda(ev \circ (id \otimes T(f)) \circ (\Lambda(\omega \circ T(ev) \circ st') \otimes id))$$

$$= \Lambda(ev \circ (id \otimes T(f)) \circ \Lambda(\omega \circ T(ev) \circ st')$$

$$= \Omega^T(f) \circ \Lambda(\omega \circ T(ev) \circ st')$$
Hence $H_\omega(f)$ defines an arrow in $C^{op}$. It remains to check that $H_\omega$ preserves identities and composition. But $H_\omega(id_\alpha): H_\omega(\alpha) \to H_\omega(\alpha)$ is the unique arrow such that

$$e_\alpha \circ H_\omega(id_\alpha) = \Omega^{id} \circ e_\alpha = id \circ e_\alpha = e_\alpha$$

Hence $H_\omega(id_\alpha) = id_{H_\omega(\alpha)}$. And for $T$-algebra morphisms $f: (A, \alpha) \to (B, \beta)$ and $g: (B, \beta) \to (C, \gamma)$, $H_\omega(g \circ f)$ is the unique arrow such that

$$e_\alpha \circ H_\omega(g \circ f) = \Omega^{gf} \circ e_\alpha = \Omega^f \circ \Omega^g \circ e_\gamma = \Omega^f \circ e_\beta \circ H_\omega(g) = e_\alpha \circ H_\omega(f) \circ H_\omega(g)$$

Hence $H_\omega(g \circ f) = H_\omega(f) \circ H_\omega(g)$. Thus we conclude that $H_\omega$ is a functor. \qed

### 3.2 The adjunction

We will now relate the just defined functors $\omega(-)$ and $H_\omega$. We will show that they are adjoints, sometimes depicted as

$$\begin{array}{c}
H_\omega(\beta) \longrightarrow Y \\
\beta \longrightarrow \omega Y
\end{array}$$

where in this case $H_\omega$ is left adjoint to $\omega(-)$. To prove this we will need the following natural transformation.

**Definition 3.5.** Let $C$ be a symmetric monoidal closed category with equalizers. For a $T$-algebra $\beta: T(B) \to B$ and $Y$ an object in $C$ define

$$\phi_{\beta,Y}: C^{op}(H_\omega(\beta), Y) \to \mathcal{EM}(T)(\beta, \omega Y)$$

in the following way

$$\phi_{\beta,Y}(f) = \Lambda(B \otimes Y \overset{\gamma}{\longrightarrow} Y \otimes B \overset{f \otimes id}{\longrightarrow} H_\omega(\beta) \otimes B \overset{\epsilon_\beta \otimes id}{\longrightarrow} \Omega B \otimes B \overset{ev}{\longrightarrow} \Omega)$$

or in short

$$\phi_{\beta,Y}(f) = \Lambda(\epsilon_\beta \circ f \circ \gamma): B \to \Omega^Y$$

**Proposition 3.6.** $\phi_{\beta,Y}: C^{op}(H_\omega(\beta), Y) \to \mathcal{EM}(T)(\beta, \omega Y)$ defines a natural transformation.

**Proof.** First we will show that for a map $f: Y \to H_\omega(\beta)$ in $C$, with $(B, \beta)$ a $T$-algebra and $Y$ an object in $C$: $\phi(f)$ defines a $T$-algebra morphism i.e. $\phi(f) \circ \beta = \omega Y \circ T(\phi(f))$. The first equation follows from $\epsilon_\beta$ being the equalizer of the other maps involved.

$$\omega_\beta \circ e_\beta = \Lambda(\omega \circ T(ev) \circ st') \circ e_\beta$$

Hence

$$\omega_\beta \circ e_\beta \circ f = \Lambda(\omega \circ T(ev) \circ st') \circ e_\beta \circ f$$

$$\Lambda(\omega \circ (id \otimes \beta)) \circ e_\beta \circ f = \Lambda(\omega \circ T(ev) \circ st') \circ e_\beta \circ f$$

$$\Lambda(\omega \circ (id \otimes \beta)) \circ ((e_\beta \circ f) \otimes id) = \Lambda(\omega \circ T(ev) \circ st' \circ ((e_\beta \circ f) \otimes id))$$

$$ev \circ (id \otimes \beta) \circ ((e_\beta \circ f) \otimes id) = \omega \circ T(ev) \circ st' \circ ((e_\beta \circ f) \otimes id)$$

$$ev \circ (id \otimes \beta) \circ ((e_\beta \circ f) \otimes id) \circ \gamma = \omega \circ T(ev) \circ st' \circ ((e_\beta \circ f) \otimes id) \circ \gamma$$
So

\[ \Lambda(\epsilon v \circ (id \otimes \beta) \circ ((e_\beta \circ f) \otimes id) \circ \gamma) = \Lambda(\omega \circ T(\epsilon v) \circ st' \circ ((e_\beta \circ f) \otimes id) \circ \gamma) \]  

(3.1)

Now we will prove that \( \phi(f) \) is a \( T \)-algebra morphism.

\[
\phi(f) \circ \beta = \Lambda(\epsilon v \circ ((e_\beta \circ f) \otimes id) \circ \gamma) \circ \beta \\
= \Lambda(\epsilon v \circ ((e_\beta \circ f) \otimes id) \circ \gamma \circ (\beta \otimes id)) \\
= \Lambda(\epsilon v \circ (id \otimes \beta) \circ ((e_\beta \circ f) \otimes id) \circ \gamma) \\
= \Lambda(\omega \circ T(\epsilon v) \circ st' \circ ((e_\beta \circ f) \otimes id) \circ \gamma) \\
= \Lambda(\omega \circ T(\epsilon v) \circ st' \circ ((e_\beta \circ f) \otimes T(id)) \circ \gamma) \\
= \Lambda(\omega \circ T(\epsilon v) \circ ((e_\beta \circ f) \otimes id) \circ T(\gamma) \circ st) \\
= \Lambda(\omega \circ T(\epsilon v) \circ ((e_\beta \circ f) \otimes id) \circ T(\gamma) \circ st) \\
= \Lambda(\omega \circ T(\epsilon v) \circ (\omega f) \otimes id) \circ \gamma \circ st) \\
= \Lambda(\omega \circ T(\epsilon v) \circ (\omega f) \otimes id) \circ \gamma \circ st) \\
= \Lambda(\omega \circ T(\epsilon v) \circ (\omega f) \circ id) \circ \gamma \circ st) \\
= \Lambda(\omega \circ T(\epsilon v) \circ (\omega f) \circ id) \circ \gamma \circ st) \\
= \Lambda(\omega \circ T(\epsilon v) \circ (\omega f) \circ id) \circ \gamma \circ st) \\
= \Lambda(\omega \circ T(\epsilon v) \circ (\omega f) \circ id) \circ \gamma \circ st) \\
= \lambda \circ T(\phi(f))
\]

(3.1)

Now we will prove that \( \phi \) is natural in both arguments. For this end let \( f : (B, \beta) \to (C, \gamma) \) be a \( T \)-algebra morphism. Then the following diagram commutes.

\[
\begin{array}{ccc}
\gamma & \xrightarrow{C^{op}(H_\omega(\gamma), Y) \phi_{\gamma,Y}} & \mathcal{EM}(T)(\gamma, \omega_Y) \\
f & \downarrow & \downarrow \\
\beta & \xrightarrow{C^{op}(H_\omega(\beta), Y) \phi_{\beta,Y}} & \mathcal{EM}(T)(\beta, \omega_Y)
\end{array}
\]

Because if \( h : H_\omega(\gamma) \to Y \) in \( C^{op} \)

\[
\mathcal{EM}(T)(f, \omega_Y) \circ \phi_{\gamma,Y}(h) = \phi_{\gamma,Y}(h) \circ f \\
= \Lambda(\epsilon v \circ ((e_\gamma \circ h) \otimes id) \circ \gamma \circ (f \otimes id)) \\
= \Lambda(\epsilon v \circ (id \otimes f) \circ ((e_\gamma \circ h) \otimes id) \circ \gamma) \\
= \Lambda(\epsilon v \circ (\Omega^f \otimes id) \circ ((e_\gamma \circ h) \otimes id) \circ \gamma) \\
= \Lambda(\epsilon v \circ ((\Omega^f \circ e_\gamma \circ h) \otimes id) \circ \gamma) \\
= \Lambda(\epsilon v \circ (\Omega^f \circ e_\gamma \circ h) \otimes id) \circ \gamma \\
= \phi_{\beta,Y}(H_\omega(f) \circ h) \\
= \phi_{\beta,Y} \circ C^{op}(H_\omega(f), Y)(h)
\]

(definition \( H_\omega \))

Now let \( g : X \to Y \) in \( C \) and \( (B, \beta) \) a \( T \)-algebra, then we have the following diagram.

\[
\begin{array}{ccc}
Y & \xrightarrow{C^{op}(H_\omega(\beta), Y) \phi_{\beta,Y}} & \mathcal{EM}(T)(\beta, \omega_Y) \\
g & \downarrow & \downarrow \\
X & \xrightarrow{C^{op}(H_\omega(\beta), X) \phi_{\beta,X}} & \mathcal{EM}(T)(\beta, \omega_X)
\end{array}
\]

22
Proof.

We claim that the natural transformation \( \phi \) is a bijection, proving the adjunction. To this end we define the following:

\[
\begin{align*}
  \mathcal{E}\mathcal{M}(T)(\beta, \omega_f) \circ \phi_{\beta,Y}(h) &= \omega_f \circ \phi_{\beta,Y}(h) \\
  &= \omega_f \circ \Lambda(ev \circ ((e_\beta \circ h) \otimes id) \circ \gamma) \\
  &= \Lambda(ev \circ (id \otimes f) \circ (\Lambda(ev \circ ((e_\beta \circ h) \otimes id) \circ \gamma) \otimes id) \\
  &= \Lambda(ev \circ (id \otimes f) \circ \gamma \circ (id \otimes f)) \\
  &= \Lambda(ev \circ ((e_\beta \circ h) \otimes id) \circ \gamma \circ (id \otimes f)) \\
  &= \Lambda(ev \circ (e_\beta \circ h \circ f) \circ \gamma) \\
  &= \phi_{\beta,X}(h \circ f) \\
  &= \phi_{\beta,X} \circ \mathbb{C}^{op}(H_\omega(\beta), f)(h)
\end{align*}
\]

Thus we conclude that \( \phi \) is a natural transformation.

\( \square \)

Now we have everything in place to proof the main result of this section.

**Theorem 3.7.** The functor \( H_\omega \) is the left adjoint of \( \omega_{(-)} \) i.e. \( H_\omega \dashv \omega_{(-)} \).

**Proof.** We claim that the natural transformation \( \phi \) is a bijection, proving the adjunction. To this end we define the following:

\[
F : \mathcal{E}\mathcal{M}(T)(\beta, \omega_Y) \rightarrow \mathbb{C}(Y, \Omega^B) \\
g \mapsto \Lambda(ev \circ (g \otimes id) \circ \gamma)
\]

Or is short

\[
F(g) = \Lambda(\overline{g} \circ \gamma)
\]

Now for a \( T \)-algebra morphism \( g : (B, \beta) \rightarrow \omega_Y \) the following equation holds

\[
\begin{align*}
  \omega_\beta \circ F(g) &= \Lambda(ev \circ (id \otimes \beta) \circ (F(g) \otimes id)) \\
  &= \Lambda(ev \circ (F(g) \otimes id) \circ (id \otimes \beta)) \\
  &= \Lambda(ev \circ (g \otimes id) \circ \gamma \circ (id \otimes \beta)) \\
  &= \Lambda(ev \circ ((g \circ \beta) \otimes id) \circ \gamma)
\end{align*}
\]

\( (T\text{-algebra morphism}) \)

\[
\begin{align*}
  &= \Lambda(ev \circ ((\omega_Y \circ T(g)) \otimes id) \circ \gamma) \\
  &= \Lambda(ev \circ (\omega_Y \otimes id) \circ (T(g) \otimes id) \circ \gamma) \\
  &= \Lambda(\omega \circ T(ev) \circ st \circ (T(g) \otimes id) \circ \gamma) \\
  &= \Lambda(\omega \circ T(ev) \circ T(g \otimes id) \circ st \circ \gamma) \\
  &= \Lambda(\omega \circ T(ev \circ (g \otimes id) \circ \gamma) \circ st \circ \gamma) \\
  &= \Lambda(\omega \circ T(ev \circ (g \otimes id) \circ \gamma) \circ st') \\
  &= \Lambda(\omega \circ T(ev \circ (F(g) \otimes id)) \circ st') \\
  &= \Lambda(\omega \circ T(ev) \circ st' \circ (F(g) \otimes T(id_B))) \\
  &= \Lambda(\omega \circ T(ev) \circ st' \circ (F(g) \otimes id_{T(B)})) \\
  &= \Lambda(\omega \circ T(ev) \circ st' \circ F(g))
\end{align*}
\]

Which is depicted in the following diagram:

\[
\begin{align*}
  H_\omega(\beta) &\xrightarrow{e_\beta} \Omega^B \\
  \psi(g) &\xrightarrow{F(g)} \Omega^{T(B)} \\
  \omega_\beta &\xrightarrow{\Lambda(\omega \circ T(ev) \circ st')} \Omega^{T(B)}
\end{align*}
\]
For $e_\beta$ is the equalizer of the two arrows involved, other than $F(g)$, we can define

$$\psi_{\beta,Y} : \mathcal{EM}(T)(\beta, \omega_Y) \to \mathbb{C}^{op}(\mathcal{H}_\omega(\beta), Y)$$

as follows: For an arrow $g : (B, \beta) \to \omega_Y$ define $\psi(g)$ as the unique arrow $h : Y \to \mathcal{H}_\omega(\beta)$ in $\mathbb{C}$ such that $e_\beta \circ h = F(g)$.

Now we claim that $\psi_{\beta,Y}$ is the inverse of $\phi_{\beta,Y}$, for every $(B, \beta)$ and $Y$. For let $f : Y \to \mathcal{H}_\omega(\beta)$ in $\mathbb{C}$ then $\psi \circ \phi(f)$ is the unique arrow such that $e_\beta \circ \psi \circ \phi(f) = F(g) = \Lambda(\ev \cdot (g \otimes \id) \circ \gamma)$. Hence

$$\phi \circ \psi(g) = \Lambda(\ev \cdot ((e_\beta \circ \psi(g)) \otimes \id) \circ \gamma) = \Lambda(\ev \cdot (F(g) \otimes \id) \circ \gamma) = \Lambda(\ev \cdot (g \otimes \id) \circ \gamma \circ \gamma) = g$$

Thus we see that $\psi_{\beta,Y}$ is indeed the inverse of $\phi_{\beta,Y}$. And therefore $\phi$ is a natural isomorphism, proving the adjunction.

### 3.3 The adjunction for $Sets$

We will now show that this adjunction is indeed a generalisation of the adjunction $\text{Hom}(-, \omega) \dashv \text{Hom}(-, \Omega)$, from our motivating article [3], namely for the case that our category $\mathbb{C}$ is $Sets$. Moreover we give a nicer description of the unit and counit in this case.

Suppose $\mathbb{C} = Sets$. Then for $g \in \Omega^X$ and $x \in X$ we have

$$\omega_X(T(g))(x) = \Lambda(\omega \circ T(ev) \circ st)(T(g))(x) = \omega \circ T(ev) \circ st(T(g), x) = \omega \circ T(ev)(T(g, x)) = \omega \circ T(g(x))$$

It is not made explicit in the article [3] but this is exactly the way that $\text{Hom}(X, \Omega)$ gets its $T$-algebra structure. Hence $\omega_X = \text{Hom}(X, \Omega)$. Moreover if $f : Y \to X$ is a function and $y \in Y$ then

$$\omega_f(g)(y) = \Lambda(\ev \cdot (\id \otimes f))(g)(y) = \ev \cdot (\id \otimes f)(g, y) = \ev(g, f(y)) = g \circ f(y)$$

24
Hence $\omega_f = \text{Hom}(f, \Omega)$. Now for the functor $H_\omega$: If $(A, \alpha)$ is a $T$-algebra, then

$$H_\omega(\alpha) = \{ f : A \rightarrow \Omega \mid \omega_\alpha(f) = \Lambda(\omega \circ T(ev) \circ st')(f) \}$$

Thus for $y = T(x) \in T(A)$ we have:

$$f \in H_\omega(\alpha) \iff f \circ \alpha(y) = \omega_\alpha(f)(y)$$
$$= \Lambda(\omega \circ T(ev) \circ st')(f)(y)$$
$$= \omega \circ T(ev) \circ st'(f, T(x))$$
$$= \omega \circ T(ev)(T(f, x))$$
$$= \omega \circ T(f(x))$$
$$= \omega \circ T(f)(y)$$

Hence $H_\omega(\alpha) = \text{Hom}(\alpha, \omega)$. Moreover if $f : (A, \alpha) \rightarrow (B, \beta)$ is a $T$-algebra morphism then $H_\omega(f)$ is the unique arrow such that $e_\alpha \circ H_\omega(f) = \Omega f \circ e_\beta$. But the $e$’s in the case of $\text{Sets}$ are just inclusions, hence $H_\omega(f) = \Omega f = \text{Hom}(f, \Omega)$. Thus our adjunction is indeed a generalization of $\text{Hom}(-, \omega) \dashv \text{Hom}(-, \Omega)$.

The next simple computations show that in the case that $\mathcal{C} = \text{Sets}$ the unit and the counit of the adjunction $H_\omega = \text{Hom}(-, \omega) \dashv \text{Hom}(-, \Omega) = \omega(-)$ are just evaluation of the reversed components.

We have $\eta_\alpha : (A, \alpha) \rightarrow \text{Hom} \left( \text{Hom}(\alpha, \omega), \Omega \right)$ in $\mathcal{E}M(T)$ where for $f \in \text{Hom}(\alpha, \omega)$ and $x \in A$:

$$\eta_\alpha(x)(f) = \phi(id)(x)(f)$$
$$= \Lambda(\phi \circ ((e_\alpha \circ id) \otimes id) \circ \gamma)(x)(f)$$
$$= ev \circ (e_\alpha \otimes id) \circ \gamma(x, f)$$
$$= ev(f, x)$$
$$= f(x)$$

The counit $\epsilon_B : B \rightarrow \text{Hom} \left( \text{Hom}(B, \Omega), \omega \right)$, defined as $\psi(id)$, is the unique arrow such that

$$\epsilon_B = e_\beta \circ \epsilon_B = F(id) = \Lambda(ev \circ (id \otimes id) \circ \gamma) = \Lambda(ev \circ \gamma)$$

hence for $f \in \text{Hom}(B, \Omega)$ and $b \in B$ we have

$$\epsilon_B(b)(f) = \Lambda(ev \circ \gamma)(b)(f) = ev \circ \gamma(b, f) = ev(f, b) = f(b)$$

Therefore the unit and counit are just evaluation of the reversed components in the case that $\mathcal{C} = \text{Sets}$.  

25
4. Relating program logic

In the following section we try to give a construction for acquiring the left hand side of the triangle using the basic adjunction from the previous section. We will obtain a new category and a new adjunction. At the end we will complete our interpretation with the \textit{Pred} functor, also placed at the left hand side of the triangle.

4.1 Beck’s theorem

The following theorem is the dual of Beck’s tripleability theorem, stating sufficient properties for a functor to be monadic. The theorem and its proof in this form are the dual of theorem 4.2, from MacLane and Moerdijk [11], p.179. First we give an overview of the situation in the following diagram, where \( F \dashv G \) and \( S = FG \) hence defining a comonad on \( A \). All the arrows commute by the properties of the comparison functor, except for the dotted arrow that is obtained from the theorem.

\[ \xymatrix{ \mathcal{E} \mathcal{M}(S) \ar[r]^{K} \ar[d]_{U^S} & B \ar[d]^{F^S} \ar[dl]_{K} \\
A & B \ar[l]^{G} } \]

\textbf{Theorem 4.1.} Let \( G : A \rightarrow B \) be a functor with left adjoint \( F \). Let \( \mathcal{E} \mathcal{M}(S) \) be the category of coalgebras for the comonad \( S = FG \) on \( A \), with forgetful functor \( U^S : \mathcal{E} \mathcal{M}(S) \rightarrow A \) which has a right adjoint \( F^S \).

\begin{itemize}
  \item [(i)] If \( B \) has equalizers of reflexive pairs then the comparison functor \( K : B \rightarrow \mathcal{E} \mathcal{M}(S) \) has a right adjoint 
  \[ L : \mathcal{E} \mathcal{M}(S) \rightarrow B \]
  For an \( S \)-coalgebra \( \alpha : A \rightarrow S(A) \), let \( L(A, \alpha) \) be the equalizer of arrows \( G(\alpha) \)
\end{itemize}
and $\eta_{G(A)}$, as in the following diagram:

$$
\begin{array}{ccc}
L(A, \alpha) & \xrightarrow{e_\alpha} & G(A) \\
& & \xrightarrow{G(\alpha)} \\
& & \eta_{G(A)} \\
\end{array}
\quad
\begin{array}{ccc}
& & GFG(A) \\
& & \eta_{G(A)} \\
\end{array}
$$

And for a $S$-coalgebra morphism $f : (A, \alpha) \to (B, \beta)$ let $L(f) : L(A, \alpha) \to L(B, \beta)$ be the unique arrow such that $G(f) \circ e_\alpha = e_\beta \circ L(f)$ (which has to be proven to exist) as in the following diagram

$$
\begin{array}{ccc}
L(A, \alpha) & \xrightarrow{e_\alpha} & G(A) \\
& & \xrightarrow{G(\alpha)} \\
& & \eta_{G(A)} \\
\downarrow & & \downarrow & & \downarrow \\
L(f) & & G(f) & & GFG(f) \\
& & \eta_{G(A)} & & \eta_{G(B)} \\
\end{array}
\quad
\begin{array}{ccc}
& & GFG(A) \\
& & \eta_{G(A)} \\
\end{array}
$$

(ii) if, in addition, $F$ preserves these equalizers, then the counit of the adjunction $L \dashv K$ is an isomorphism.

(iii) if, in addition to (i) and (ii), $F$ reflects isomorphisms, then the unit of the adjunction $L \dashv K$ is an isomorphism. Consequently, $\mathcal{A}$ is equivalent to $E\mathcal{M}(S)$.

Note that in this situation the functor $K : \mathcal{A} \to E\mathcal{M}(S)$ is called comonadic, the dual form of monadic.

Proof. (i) First we will show that $L$ is a well-defined functor. Let $(A, \alpha)$ be an $S$-algebra then

$$
G(\alpha) \circ G(\epsilon_A) = G(\alpha \circ \epsilon_A) = id_{GFG(A)} = \eta_{G(A)} \circ G(\epsilon_A)
$$

hence $G(\alpha), \eta_{G(A)}$ is a reflexive pair. Thus $L(A, \alpha)$ exists.

For $S$-algebra morphism $f : (A, \alpha) \to (B, \beta)$ we have:

$$
G(\beta) \circ G(f) \circ e_\alpha = G(\beta \circ f) \circ e_\alpha = G(S(f) \circ \alpha) \circ e_\alpha = GFG(f) \circ G(\alpha) \circ e_\alpha = GFG(f) \circ \eta_{G(A)} \circ e_\alpha = \eta_{G(B)} \circ G(f) \circ e_\alpha
$$

Hence $L(f)$ is well-defined. Also $L(id_{(A, \alpha)})$ is the unique arrow such that

$$
e_\beta \circ L(id_{(A, \alpha)}) = G(id_{(A, \alpha)}) \circ e_\alpha = e_\alpha = e_\alpha \circ id_{L(A, \alpha)}
$$

So $L(id_{(A, \alpha)}) = id_{L(A, \alpha)}$. 

27
And for $S$-algebra morphisms $f : (A, \alpha) \rightarrow (B, \beta)$ and $g : (B, \beta) \rightarrow (C, \gamma)$ we have that $L(g \circ f)$ is the unique arrow such that $G(g \circ f) \circ e_\alpha = e_\alpha \circ L(g \circ f)$. But

$$G(g \circ f) \circ e_\alpha = G(g) \circ G(f) \circ e_\alpha = G(g) \circ e_\beta \circ L(f) = e_\gamma \circ L(g) \circ L(f)$$

Hence $L(g \circ f) = L(g) \circ L(f)$.

To evade any confusion we state that the comparison functor $K : \mathcal{B} \rightarrow \mathcal{E}\mathcal{M}(S)$ is given by $K(X) = (F(X), F\eta_X)$ and $K(f) = F(f)$.

Define:

$$\psi_{B,A} : \mathcal{B}(B, L(A, \alpha)) \rightarrow \mathcal{E}\mathcal{M}(S)(K(B), (A, \alpha))$$

give by

$$\psi_{B,A}(h) = \epsilon_A \circ F(e_\alpha \circ h)$$

Then $\psi(h)$ is an $S$-algebra morphism:

$$\alpha \circ \psi(h) = K \circ \epsilon_A \circ F(e_\alpha \circ h) = \epsilon_{FG(A)} \circ FG(\alpha) \circ F(e_\alpha \circ h) = \epsilon_{FG(A)} \circ F(\eta_G(A) \circ e_\alpha \circ h) = F(e_\alpha \circ h) = F(G(\epsilon_B \circ \eta_G(A) \circ e_\alpha \circ h) = F(G(\epsilon_G(A)) \circ GF(e_\alpha \circ h) \circ \eta_B) = FG(\psi(h)) \circ F(\eta_B) = S(\psi(h)) \circ F\eta_B$$

Now we will show that $\psi$ is natural in both components. Consider the following diagram:

$$\begin{array}{ccc}
B' & \xrightarrow{\mathcal{B}(B, L(A, \alpha))} & \mathcal{E}\mathcal{M}(S)(K(B), (A, \alpha)) \\
| f \downarrow & \mathcal{E}\mathcal{M}(S)(K(f), (A, \alpha)) & \mathcal{E}\mathcal{M}(S)(K(B'), (A, \alpha)) \\
B & \xrightarrow{\mathcal{B}(B', L(A, \alpha))} & \mathcal{E}\mathcal{M}(S)(K(B'), (A, \alpha))
\end{array}$$

where $f : B' \rightarrow B$ is an arrow in $\mathcal{B}$. Now let $a : B \rightarrow L(A, \alpha)$ in $\mathcal{B}$, then

$$\psi_{B',A} \circ \mathcal{B}(f, L(A, \alpha))(a) = \psi_{B',A}(a \circ f) = \epsilon_A \circ F(e_\alpha \circ a \circ f) = \psi_{B,A}(a) \circ F(f) = \mathcal{E}\mathcal{M}(S)(K(f), (A, \alpha)) \circ \psi_{B,A}(a)$$

and let $g : (A, \alpha) \rightarrow (A', \alpha')$ in $\mathcal{E}\mathcal{M}(S)$ then:

$$\psi_{B,A'} \circ \mathcal{B}(B, L(g))(a) = \psi_{B,A'}(L(g) \circ a) = \epsilon_{A'} \circ F(e_{\alpha'} \circ L(g) \circ a) = \epsilon_{A'} \circ F(G(g) \circ e_\alpha \circ a) = g \circ e_A \circ F(e_\alpha \circ a) = g \circ \psi_{B,A}(a) = \mathcal{E}\mathcal{M}(S)(K(B), g \circ \psi)A, B(a)$$
Therefore $\psi$ is a natural transformation. We claim that $\psi$ is an isomorphism with inverse $\phi: \mathcal{EM}(S)(K(B), (A, \alpha)) \to \mathcal{B}(B, L(A, \alpha))$ constructed as follows:

For an arrow $f : K(B) \to (A, \alpha)$ consider the map $G(f) \circ \eta_B : B \to G(A)$. Then

$$G(\alpha) \circ G(f) \circ \eta_B = G(\alpha \circ f) \circ \eta_B = G(S(f) \circ F(\eta_B)) \circ \eta_B = GFG(f) \circ \eta_{GF(B)} \circ \eta_B = \eta_{G(A)} \circ G(f) \circ \eta_B$$

hence we can define $\phi(f)$ to be the unique arrow such that $e_{\alpha} \circ \phi(f) = G(f) \circ \eta_B$.

Now

$$\psi \circ \phi(f) = \epsilon_A \circ F(e_{\alpha} \circ \phi(f)) = \epsilon_A \circ F(G(f) \circ \eta_B) = f \circ \epsilon_{F(B)} \circ F(\eta_B) = f$$

And $\phi \circ \psi(g) : B \to L(A, \alpha)$ is the unique arrow such that $e_{\alpha} \circ \phi \circ \psi(g) = G(\psi(g)) \circ \eta_B$. But

$$G(\psi(g)) \circ \eta_B = G(\epsilon_A \circ F(e_{\alpha} \circ g)) \circ \eta_B = G(\epsilon_A) \circ \eta_{G(C)} \circ e_{\alpha} \circ g = e_{\alpha} \circ g$$

hence $\phi \circ \psi(g) = g$. Thus follows the adjunction $K \dashv L$.

(ii) Consider the $S$-coalgebra $(A, \alpha)$. Then $F$ preserves the equalizer from the definition of $L(A, \alpha)$ yielding the equalizer

$$FL(A, \alpha) \xrightarrow{F(e_{\alpha})} FG(A) \xrightarrow{FG(\alpha)} FGFG(A)$$

But also $\alpha : FG(A) \to A$ canonically is the (split)equalizers of $S(\alpha) = FG(\alpha)$ and $F(\eta_{G(A)}) = \delta_A$. Suppose there exists an arrow $h : Z \to FG(A)$ such that $S(\alpha) \circ h = \delta_A \circ h$ then we have $\epsilon_A \circ h : Z \to A$ such that $\alpha \circ \epsilon \circ h = h$.

Hence both $(A, \alpha)$ and $(FL(A, \alpha), F(e_{\alpha}))$ are equalizers of the same diagram. So an arrow between them has to be an isomorphism. We claim that this is $\kappa_A$, the counit if the adjunction $K \dashv L$. For

$$\kappa_A = \psi(id_{L(A, \alpha)}) = \epsilon_A \circ F(e_{\alpha} \circ id) = \epsilon_A \circ F(e_{\alpha})$$

hence $\alpha \circ \kappa = \alpha \circ \epsilon_A \circ F(e_{\alpha}) = F(e_{\alpha})$.

(iii) Suppose $B \in \mathcal{B}$ and consider the unit of the adjunction $K \dashv L$, the unique arrow $\lambda_B = \phi(id_{K(B)})$ such that $\epsilon_{F(\eta_B)} \circ \lambda_B = G(id) \circ \eta_B = \eta_B$ as depicted below

$$LK(B) \xrightarrow{\epsilon_{F(\eta_B)}} GF(B) \xrightarrow{G(\eta_B)} GFGF(B)$$
Now $F$ preserves this equalizers as assumed so $(FLK(B), F(\varepsilon_{F(\eta_B)}))$ is the equalizer of $FGF(\eta_B)$ and $F(\eta_{GF(B)})$ in $A$. But also $K(B) = (F(B), F(\eta_B))$ is canonically the split equalizers of $S(F(\eta_B)) = FGF(\eta_B)$ and $\delta_{F(B)} = F(\eta_{GF(B)})$. Hence $F(\lambda_B)$ is an isomorphism. Moreover $F$ reflects isomorphisms thus $\lambda_B$ is an isomorphism.

We now want to apply this theorem to our situation where we have

\[
A = \mathcal{C}\text{op} \xrightarrow{G=\omega_{(-)}} \mathcal{E}\mathcal{M}(T) = \mathcal{B}
\]

Therefore we need for the category $\mathcal{E}\mathcal{M}(T)$ to have equalizers. The following well known result will help us, here we use the formulation from Borceux [10].

**Proposition 4.2.** Let $T$ be a monad on a category $\mathcal{C}$, then the forgetful functor $U : \mathcal{E}\mathcal{M}(T) \to \mathcal{C}$ creates any limit that exists in $\mathcal{C}$. This means that if $D : I \to \mathcal{E}\mathcal{M}(T)$ is a diagram such that $U \circ D$ has a limit in $\mathcal{C}$, then $D$ has a limit in $\mathcal{E}\mathcal{M}(T)$ and $U$ preserves this limit.

**Proof.** First note that $U$ will preserve every limit for it is a right adjoint. Now let us denote $D(C) = (UD(C), \xi_C : TUD(C) \to UD(C))$ and let $(p_C : L \to UD(C))_{C \in I}$ be a limit of $UD$.

Then $(\xi_C \circ T(p_C) : T(L) \to UD(C))_{C \in I}$ is a cone on $UD$ for if we have $f : C \to B$ in $I$ then

\[
UD(f) \circ \xi_C \circ T(p_C) = \xi_C \circ TUD(f) \circ T(p_C) = \xi_C \circ T(p_B)
\]

Hence there exists a unique $\phi : T(L) \to L$ such that $p_C \circ \phi = \xi_C \circ T(p_C)$. Now we claim that $p_C : (L, \phi) \to (UD(C), \xi_C)$ is the limit of $D$.

We have that $(L, \phi)$ is a $T$-algebra for

\[
p_C \circ \phi \circ \eta = \xi_C \circ T(p_C) \circ \eta = \xi_C \circ \eta \circ p_C = p_C
\]

and

\[
p_C \circ \phi \circ \mu = \xi_C \circ T(p_C) \circ \mu = \xi_C \circ \mu \circ T^2(p_C) = \xi_C \circ T(\xi_C) \circ T^2(p_C) = \xi_C \circ T(p_C \circ \phi) = p_C \circ \phi \circ T(\phi)
\]

Hence by definition of a limit we get $\phi \circ \eta = id$ and $\phi \circ \mu = \phi \circ T(\phi)$. Also $p_C$ is a $T$-algebra morphism for $\xi_C \circ T(p_C) = p_C \circ \phi$ by definition of $\phi$.

Moreover $(p_C : (L, \phi) \to D(C))_{C \in I}$ forms a cone on $D$ for we have a limiting cone $p$ and $U : \mathcal{E}\mathcal{M}(T) \to \mathcal{C}$ is faithful.
If we now suppose that $(q_C : (M, \psi) \rightarrow (UD(C), \xi_C))_{C \in I}$ is a cone on $D$, then we have that $p_C$ is the limit of $UD$ so there exists a unique $m : M \rightarrow L$ such that $p_C \circ m = q_C$. Now $m$ is actually a $T$-algebra morphism $m : (M, \psi) \rightarrow (L, \phi)$, for

$$p_C \circ m \circ \psi = q_C \circ \psi = \xi_C \circ T(q_C) = \xi_C \circ T(p_C) \circ T(m) = p_C \circ \phi \circ T(m)$$

So by definition of a limit we have $m \circ \psi = \phi \circ T(m)$ concluding the proof. \qed

The requirement of $EM(T)$ having equalizers, in order to let our construction succeed, justifies in a sense the choice of $T$ being a monad on a symmetric monoidal closed category with equalizers. We will work out what the previous proposition means in this case.

**Example 4.3.** We have the index category $I = (\bullet \Rightarrow \bullet)$ so in $EM(T)$ we have the following commuting diagram where $(L, p)$ is the equalizer in $C$

$$\begin{array}{ccc}
T(C) & \xrightarrow{T(f)} & T(B) \\
\downarrow^{T(g)} & & \downarrow^{\beta} \\
L \xrightarrow{p} C & \xrightarrow{f} & B
\end{array}$$

Then $(L, \phi)$ is a $T$-algebra where $\phi$ is given by the unique arrow in $C$ such that $\alpha \circ T(p) = p \circ \phi$, as in

$$\begin{array}{ccc}
T(L) & \xrightarrow{\alpha \circ T(p)} & T(B) \\
\downarrow^{1} & & \downarrow^{\beta} \\
L \xrightarrow{p} C & \xrightarrow{f} & B
\end{array}$$

Thus we conclude that the adjunction $K \dashv L$ exists in our case. Note that for every fixed algebra $\omega$ we have different functors $F$ and $G$, hence different functors $K$ and $L$. Thus from now on we will refer to these functors as $K_\omega$ and $L_\omega$. We are left with the following situation, the picture deliberately suggests that we have shrunk the problem.
4.2 The $\text{Pred}$ functor

To finally fully interpret the triangle we need a functor $\text{Pred} : \mathcal{K}(T) \to \mathcal{E}(S)$ making the triangle commute in both directions. But first we will spend some time on the codomain of this functor.

Consider the functor $\mathcal{K}_\omega : \mathcal{E}(T) \to \mathcal{E}(S)$. On arrows this is just the functor $F$ which is obviously a contravariant functor. Nevertheless the codomain of $\mathcal{K}_\omega$ is not of the form $(-)^{\text{op}}$. This could cause some confusion that we want to clear up. The core of the problem is that we have that $S$ is a comonad on $\mathbb{C}^{\text{op}}$, hence two co’s in some sense. We want to make this explicit by writing

$$\mathcal{E}(S) = \mathcal{E}(S^{\text{op}})^{\text{op}}$$

where $S^{\text{op}} = F^{\text{op}} \circ G^{\text{op}} : \mathbb{C} \to \mathbb{C}$ is a monad on $\mathbb{C}$.

Once this possible confusion is acknowledged we can switch between these two notations if the context would ask for this. Surely $S$ and $S^{\text{op}}$ are essentially the same functor but to emphasise the contravariance of the functors we mostly will write $\mathcal{E}(S^{\text{op}})^{\text{op}}$. Thus from now on the triangle has the following form

We still need a functor $\text{Pred} : \mathcal{K}(T) \to \mathcal{E}(S^{\text{op}})^{\text{op}}$ making the triangle commute in both directions. An obvious choice would be to take

$$\text{Pred} = \mathcal{K}_\omega \circ \mathcal{K}$$

making the triangle commute trivially in one direction. We will now give sufficient conditions for this choice of $\text{Pred}$ to give a correct interpretation.

**Lemma 4.4.** Let $\eta$ be the unit of the adjunction $\mathcal{K}_\omega \dashv \mathcal{L}_\omega$. If for every free $T$-algebra we have that $\eta_{(T(X), \mu_X)}$ is an isomorphism, then $\text{Pred} = \mathcal{K}_\omega \circ \mathcal{K}$ makes the triangle commute in both directions.

**Proof.** We begin by stating that $\mathcal{K}(X) = (T(X), \mu_X)$ hence the isomorphisms $\eta_{\mathcal{K}(X)} : \mathcal{K}(X) \to L_\omega \circ \mathcal{K}_\omega \circ \mathcal{K}(X)$ justify the first equation of

$$\mathcal{K} \cong L_\omega \circ \mathcal{K}_\omega \circ \mathcal{K} = L_\omega \circ \text{Pred}$$

concluding the proof. \qed

Note that Beck’s theorem 4.1 (ii) and (iii) give a handle on this in the case of an equivalence.
When we indeed have an appropriate \textit{Pred}-functor i.e. making the triangle commute in both directions, it can be interpreted as the \textit{weakest precondition} functor. An idea from Dijkstra as a reformulation of Hoare logic of a program. We will shortly address these notions.

Hoare logic centers around Hoare triples and forms a formal system to reason about correctness of programs. A \textit{Hoare triple} is of the form

\[ \{P\} C \{R\} \]

where \(P\) and \(R\) are assertions and \(C\) a program. It is interpreted as: 'When precondition \(P\) is met, executing command \(C\) will ensure postcondition \(R\).' The system then provides rules to reason with these triples.

Dijkstra in [12] reformulated this into \textit{predicate transformer semantics} that doesn't provide rules but builds valid deductions in Hoare logic. Given a statement \(S\) the \textit{weakest precondition} of \(S\) is a function mapping any postcondition \(R\) to a precondition \(wp(S, R)\), which is the weakest precondition \(P\) ensuring \(\{P\} S \{R\}\). More formally, \(\{P\} S \{R\}\) is provable in Hoare logic if and only if the formula \(\forall x \ [P \to wp(S, R)]\) holds.

This is exactly the interpretation we have for the left hand side of the triangle. The \textit{Pred} functor takes a computation i.e. a map in \(\mathcal{K}\ell(T)\) and produces a map in the category of predicate transformers, interpreted as the weakest precondition of this computation.
5. Non-deterministic computations

In this section we consider the powerset monad $P$. And we want to show that for a certain choice of the algebra $\omega$ we get the categories and functors as in the example in our motivating article [3].

5.1 The desired situation

We will just state that this is the situation we need to obtain:

\[
\begin{array}{ccc}
(Cl_{\land})^{op} & \cong & EM(P) \cong Cl_{\lor} \\
\sim & \Leftrightarrow & \sim \\
Pred & \perp & K \\
& \perp & K \uparrow \uparrow \\
& \perp & \downarrow \downarrow \\
\end{array}
\]

The functors are not explicitly stated in the article as we will do here. The following lemma is from Davey and Priestley [9] but is adjusted to our categorical point of view.

**Lemma 5.1.** Let $\phi : P \to Q$ be a map in Pos then $\phi$ is join-preserving if and only if $\phi \dashv \phi^\sharp : Q \to P$ given by

\[
\phi^\sharp(q) = \bigvee \phi^{-1}(\downarrow q)
\]

for $q \in Q$.

**Proof.** $\Leftarrow$ Trivial, for left-adjoints preserve colimits (joins).

$\Rightarrow$ Suppose $\phi : P \to Q$ is join-preserving. Then $\phi^\sharp(q) = \bigvee \phi^{-1}(\downarrow q) \in P$. And

\[
p \leq q \iff \downarrow p \subseteq \downarrow q \\
\iff \phi^{-1}(\downarrow p) \subseteq \phi^{-1}(\downarrow q) \\
\iff \bigvee \phi^{-1}(\downarrow p) \subseteq \bigvee \phi^{-1}(\downarrow q) \\
\iff \phi^\sharp(p) \leq \phi^\sharp(q)
\]
So $\phi^\sharp: Q \to P$ is order-preserving. Moreover for $p \in P$ and $q \in Q$

$\phi(p) \leq q \iff p \in \phi^{-1}(\downarrow q) \iff p \leq \bigvee \phi^{-1}(\downarrow q) = \phi^\sharp(q)$

The naturality conditions are trivial in this case hence $\phi \dashv \phi^\sharp$.

**Lemma 5.2.** $(-)^\sharp : Cl \to (Cl)^{\text{op}}$ defines a functor by assigning

$$X^\sharp = X$$

**Proof.** Note that the codomain is $(Cl)^{\text{op}}$ for right-adjoints preserve limits (meets). We need to check the following:

Let $q \in Q$ and $f : R \to P$, $g : P \to Q$ be in $\text{Pos}$, then first note that $\phi^\sharp(q)$ is the (necessarily unique) element $s \in P$ such that $\downarrow s = \phi^{-1}(\downarrow q)$. Hence

$$id^\sharp(q) = \bigvee id^{-1}(\downarrow q) = \bigvee \downarrow q = q$$

and

$$(g \circ f)^\sharp(q) = \bigvee (g \circ f)^{-1}(\downarrow q) = \bigvee \downarrow u = u$$

with $u \in R$ the unique object such that $(g \circ f)^{-1}(\downarrow q) = \downarrow u$. Also

$$f^\sharp \circ g^\sharp(q) = f^\sharp(\bigvee g^{-1}(\downarrow q)) = f^\sharp(\bigvee \downarrow t) = f^\sharp(t)$$

with $t \in P$ the unique object such that $g^{-1}(\downarrow q) = \downarrow t$. But then

$$f^{-1}(\downarrow t) = f^{-1} \circ g^{-1}(\downarrow q) = (g \circ f)^{-1}(\downarrow q) = \downarrow u$$

Hence $f^\sharp \circ g^\sharp(q) = \bigvee f^{-1}(\downarrow t) = \bigvee \downarrow u = u = (g \circ f)^\sharp(q)$, making $(-)^\sharp$ a functor.

Note that lemma 5.1 and the the statements after it dualize: A map $f : Q \to P$ is meet-preserving iff $f^\flat \dashv f$ where $f^\flat(p) = \bigwedge f^{-1}(\uparrow p)$. Moreover the operation can be extended to a functor $(-)^\flat : (Cl)^{\text{op}} \to Cl$.

**Proposition 5.3.** $(-)^\sharp$ is an isomorphism with invers $(-)^\flat$.

**Proof.** Note that on objects $(-)^\sharp$ is obviously an isomorphism. Let $f : P \to Q$ be a join-preserving map and $p \in P$. Then

$$(f^\sharp)^\flat(p) = \bigwedge (f^\sharp)^{-1}(\uparrow p) = \bigwedge \{ q \in Q \mid f^{-1}(\downarrow q) \geq p \}$$

Now for every subset $A \subseteq Q$ we have $f \circ f^{-1}(A) \subseteq A$ hence

$$q = \bigvee f^{-1}(\downarrow q) \geq f(\bigvee f^{-1}(\downarrow q)) = f(f^{-1}(\downarrow q)) \geq f(p)$$

35
Therefore \( f(p) \in \downarrow q \), so \( p \in f^{-1}(\downarrow q) \). Then \( \bigvee f^{-1}(\downarrow q) \geq p \), hence \( f(p) \leq (f^\sharp)^\flat(p) \). Also \( f(p) \in \downarrow f(p) \) hence \( \bigvee f^{-1}(\downarrow f(p)) \geq p \) and so \( (f^\sharp)^\flat(p) \leq f(p) \). Proving \((f^\sharp)^\flat = f\).

A dual argument states that \((g^\flat)^\sharp = g\) for every meet-preserving map \( g \). Hence we conclude that \((-)^\sharp : Cl \to (Cl)^{op}\) is an isomorphism with inverse \((-)^\flat\). \(\square\)

### 5.2 Constructing the category

We return to our part of the work. In order to come to these categories and functors we will fix the algebra as the free algebra on the singleton set \( 1 = \{0\} \).

Note that \( P(1) = \{\emptyset, 1\} \cong 2 \) so \( \omega = \mu_1 : P(2) \to 2 \). At this point it is unclear how to systematically choose this algebra to come to relevant examples.

Before we begin the actual work, describing the category \( E.M(S^{op})^{op} \), we need two isomorphisms. The first one is a simple one but needs to be made precise. It considers the contravariant powerset functor \( P : Sets^{op} \to Sets \). Note that we write \( G = Hom(-, 2) \) and \( F = Cl \circ (-, 2) \), for the functor \( G \) and \( F \) from the previous section in this particular example.

**Definition 5.4.** For a set \( X \) and a subset \( A \subseteq X \) we define:

\[
a_X : G(X) \to P(X), h \mapsto h^{-1}(1)
\]

and

\[
a_X^{-1}(A) = \chi_A
\]

where

\[
\chi_A : X \to 2 : x \mapsto \begin{cases} 
1 & \text{if } x \in A \\
0 & \text{otherwise}
\end{cases}
\]

is the indicator function of \( A \).

These maps are obviously well-defined and we claim that \( a : G = P \) is a natural isomorphism with inverse \( a^{-1} \). Let \( f : Y \to X \) and \( h : X \to 2 \) be functions, then

\[
a_Y \circ G(f)(h) = a_Y(h \circ f)
= (h \circ f)^{-1}(1)
= f^{-1} \circ h^{-1}(1)
= \mathcal{P}(f)(h^{-1}(1))
= \mathcal{P}(f) \circ a_X(h)
\]

And for \( y \in X \) we have

\[
1 = a_X^{-1} \circ a_X(h)(y) = a_X^{-1}(h^{-1}(1))(y) = \chi_{h^{-1}(1)}(y)
\]

iff \( y \in h^{-1}(1) \) iff \( h(y) = 1 \). Hence \( a_X^{-1} \circ a_X(h) = h \).
Also
\[ a_X \circ a_X^{-1}(A) = a_X(\chi_A) = \chi_A^{-1}(1) = A \]

hence \( a \) is a natural isomorphism.

The second transformation is from the adjunction associated with the (covariant) powerset monad \( (P, \eta_P, \mu_P) \).

**Definition 5.5.** Let \( X \) be a set. Define \( \phi \) to be the natural isomorphism given by
\[
\phi_X : F(P(X)) = Cl_P(P(X), 2) \to Hom(X, U(2)) = G(X)
\]

**Proposition 5.6.** Let \( T = P \) be our monad and \( \omega = \mu_1 : P(2) \to 2 \) be our fixed algebra then the monad \( S^{op} \) is isomorphic to the powerset monad i.e.
\[
(S^{op}, \epsilon, \delta) \cong (P, \{-\}, \bigcup)
\]

Note that \( S^{op} \) does the same as \( S \) both on objects and on arrows. We therefore only denote the monad with \( S^{op} \), the rest with just \( S \), even so for the functors \( F \) and \( G \). In order to shrink the calculations in the following proof we examine an isomorphism, constructed from the \( a_X \) and \( \phi_X \) as above, a bit further: Define
\[
\xi_X = F(a_X) \circ \phi_X^{-1} \circ a_X^{-1} : P(X) \to S(X) = FG(X)
\]

then for \( A \subseteq X \) and \( h \in G(X) \) we have
\[
\xi_X(A)(h) = F(a_X) \circ \phi_X^{-1}(\chi_A)(h) = F(a_X)(\epsilon_X \circ P(\chi_A))(h) = \epsilon_X \circ P(\chi_A) \circ a_X(h) = \bigvee \chi_A(h^{-1}(1)) \quad (1)
\]

hence
\[
\xi_X(A)(h) = 1 \iff \exists b \in h^{-1}(1) \ [\chi_A(b) = 1] \\
\quad \iff \exists b \in A \ [h(b) = 1] \quad (2)
\]

The inverse of \( \xi \) is given by the inverses of its components. Hence for a \( h \in G(X) \) we have:
\[
\xi_X^{-1}(h) = a_X \circ \phi_X \circ F(a_X^{-1})(h) = (U_P(h \circ a_X^{-1}) \circ \eta_X^{-1})^{-1}(1)
\]

Therefore
\[
x \in \xi_X^{-1}(h) \iff h(\chi_{\{x\}}) = U_P(h \circ a_X^{-1}) \circ \eta_X(x) = 1 \quad (3)
\]

We are now ready to begin the proof.
Proof. We will use the natural isomorphism $\xi$ to prove that $S^{op} \cong \mathcal{P}$, the (covariant) powerset functor. Let $f : X \to Y$ be a function. Then we need the following to commute.

$$
\begin{array}{ccc}
X & \xrightarrow{\mathcal{P}(X)} & S(X) \\
\downarrow f & & \downarrow S(f) \\
Y & \xrightarrow{\mathcal{P}(Y)} & S(Y)
\end{array}
$$

Or in other words $\xi_Y^{-1} \circ S(f) \circ \xi_X = \mathcal{P}(f)$. Now for $A \subseteq X$ we have

$$
\xi_Y^{-1} \circ S(f) \circ \xi_X(A) = \xi_Y^{-1} \circ FG(f)(\epsilon_Y^P \circ \mathcal{P}(\chi_A) \circ a_Y) = \xi_Y^{-1}(\epsilon_Y^P \circ \mathcal{P}(\chi_A) \circ a_X \circ G(f))
$$

Thus

$$
x \in \xi_Y^{-1} \circ S(f) \circ \xi_X(A) \iff \epsilon_Y^P \circ \mathcal{P}(\chi_A) \circ a_X \circ G(f)(\chi_{\{x\}}) = 1 \quad (3)
$$

$$
\iff \chi_{\mathcal{P}(\chi_A)}((\chi_{\{x\}} \circ f)^{-1}(1)) = 1 \quad (1')
$$

$$
\iff \exists b \in (\chi_{\{x\}} \circ f)^{-1}(1) \left[\chi_A(b) = 1\right]
$$

$$
\iff \exists b \in A \left[\chi_{\{x\}}(b) = 1\right]
$$

$$
\iff \exists b \in A \left[f(b) = x\right]
$$

$$
\iff x \in f(A) = \mathcal{P}(f)(A)
$$

We conclude that $S^{op} \cong \mathcal{P}$. To conclude the result we need to check that we have a map of monads i.e. the natural isomorphism behaves well with respect to unit and multiplication of both monads. First for the unit we like for the following diagram to commute.

$$
\begin{array}{ccc}
X & \xrightarrow{\mathcal{P}(X)} & S(X) \\
\downarrow \{-\}_X & & \downarrow \epsilon_X \\
\mathcal{P}(X) & \xrightarrow{\xi_X} & S(X)
\end{array}
$$

Or equivalently $\{-\}_X = \xi_X^{-1} \circ \epsilon_X$. Now for $y \in X$ we have

$$
x \in \xi_X^{-1} \circ \epsilon_X(y) \iff 1 = \epsilon_X(y)(\chi_{\{x\}}) = \chi_{\{x\}}(y) \quad (3)
$$

$$
\iff y = x
$$

Therefore we conclude that $\xi_X^{-1} \circ \epsilon_X(y) = \{y\}$. If we continue with the multiplication we wish for the next diagram to commute.

$$
\begin{array}{ccc}
\mathcal{P}^2(X) & \xrightarrow{\xi_{\mathcal{P}(X)}} & S(\mathcal{P}(X)) \\
\downarrow \cup_X & & \downarrow S(\xi_X) \\
\mathcal{P}(X) & \xrightarrow{\xi_X} & S(X)
\end{array}
$$

$$
\begin{array}{ccc}
\mathcal{P}(X) & \xrightarrow{\mathcal{P}(\xi_X)} & S(\mathcal{P}(X)) \\
\downarrow \cup_X & & \downarrow \delta_X \\
\mathcal{P}(X) & \xrightarrow{\xi_X} & S(X)
\end{array}
$$
Or equivalently that \( \bigcup X = \xi^{-1}_X \circ \delta_X \circ S(\xi_X) \circ \xi_{\mathcal{P}(X)} \) where we will refer to the latter one as \( \delta'_X \). For \( A \in \mathcal{P}^2(X) \) we have

\[
\delta'_X(A) = \xi^{-1}_X \circ \delta_X \circ S(\xi_X) \circ \xi_{\mathcal{P}(X)}(A) = \xi^{-1}_X \circ \delta_X \circ FG(\xi_X) \circ F(\alpha_{\mathcal{P}(X)}) \circ \phi_{\mathcal{P}(X)}^{-1} \circ a_{\mathcal{P}(X)}^{-1}(A)
\]

Thus we conclude that

\[
\xi^{-1}_X \circ \delta_X \circ \phi_{\mathcal{P}(X)}^{-1} \circ a_{\mathcal{P}(X)}^{-1}(A)
\]

\( X \) is given by

\[
\chi A(\eta_{\mathcal{G}(X)}) \circ \xi_X^{-1}(1)
\]

Hence

\[
x \in \delta'_X(A) \iff \epsilon_{\mathcal{P}(X)} \circ \mathcal{P}(\chi_A) \circ a_{\mathcal{P}(X)} \circ G(\xi_X) \circ \eta_{\mathcal{G}(X)}(\chi(x)) = 1
\]

\[
\exists B \in \eta_{\mathcal{G}(X)}(\chi(x)) \circ \xi_X^{-1}(1) \ [\chi_A(B) = 1]
\]

\[
\exists B \in A \ [\eta_{\mathcal{G}(X)}(\chi(x)) \circ \xi_X(B) = 1]
\]

\[
\exists B \in A \ [\xi_X(B) = 1]
\]

\[
\exists B \in A \ [x = b \in B]
\]

\[
x \in \bigcup X A
\]

Thus we conclude that \( \xi^{-1}_X \circ \delta_X \circ S(\xi_X) \circ \xi_{\mathcal{P}(X)} = \bigcup X \), which proves the isomorphism of monads.

The following result is a direct consequence of the previous proposition and example 2.1.

**Corollary 5.7.** For the comonad \( S \) we have

\[
\mathcal{EM}(S) = \mathcal{EM}(S^{op})^{op} \cong (\mathcal{Cl}_V)^{op}
\]

This is indeed not quite the category that we aimed for. We need just one more isomorphism. But at this time it would seem artificial and we therefore wait for the justification in the next subsection.

### 5.3 Constructing the functors

We now turn our attention to the functor \( K_\omega : \mathcal{EM}(T) \to \mathcal{EM}(S) \), so in our case

\[
K_{\mu_1} : \mathcal{Cl}_V \to (\mathcal{Cl}_A)^{op}
\]

which on objects is given by

\[
K_{\mu_1}(X) = (F(X), F(\eta_X))
\]

where \( F(X) = \mathcal{C}_V(X, 2) \), maybe hinting the next known isomorphism:

\[
X^\partial \cong \mathcal{Cl}_V(X, 2)
\]

that we will spell out now. The notation \((-)^\partial\) comes from order theory, and can be extended to a functor on a category \( A \) if the objects of this category can be seen as preorders/categories.

39
**Definition 5.8.** Let \( A \) be a category of categories i.e. with categories as objects and functors as morphisms, then we define a *dualising* functor

\[
(-)^\partial : A \rightarrow A
\]
given by

\[
A^\partial = A^{\text{op}} \quad \text{and} \quad f^\partial = f
\]

Note that this is not to confuse with \((-)^{\text{op}}\) where we turn around the arrows and do nothing with the objects. Whereas here we turn around the objects and do nothing on the arrows. Moreover \((-)^\partial\) is obviously an isomorphism which is its own inverse.

**Lemma 5.9.** Let \( X \) be a complete lattice. Then

\[
X^\partial \cong (\text{Cl}_\vee(X, 2), \text{Cl}_\vee(\eta_X, 2)) = K_{\mu_1}(X)
\]
as complete lattices, given by the isomorphism

\[
p_X : X^\partial \rightarrow K_{\mu_1}(X) ; a \mapsto \chi_A'
\]
where for a subset \( A \subseteq X \):

\[
\chi_A' : X \rightarrow 2 ; x \mapsto \begin{cases} 0 \quad \text{if } x \in A \\ 1 \quad \text{otherwise} \end{cases}
\]

the 'dual' version of the indicator function of \( A \). And the inverse is given by

\[
p_X^{-1}(h) = \bigvee h^{-1}(0)
\]

**Proof.** We begin by stating that \( p_X \) is well-defined i.e. for \( a \) in \( X \) we have \( p_X(a) \) is join-preserving. So let \( B \subseteq X \) be a subset then

\[
p_X(a)(\bigvee B) = \chi_A'((\bigvee B)) = 0 \iff \bigvee B \leq a
\]

\[
\iff \forall b \in B \quad [b \leq a]
\]

\[
\iff \forall b \in B \quad [\chi_A'(b) = 0]
\]

\[
\iff \bigvee_{b \in B} p_X(a)(b) = \bigvee_{b \in B} \chi_A'(b) = 0
\]

In order to continue we need the following identity: Let \( h \in \text{Cl}_\vee(Y, 2) \) then

\[
h^{-1}(0) = \downarrow \bigvee h^{-1}(0)
\]

For

\[
b \in \downarrow \bigvee h^{-1}(0) \iff b \leq \bigvee h^{-1}(0)
\]

\[
\Rightarrow h(b) \leq h(\bigvee h^{-1}(0)) = \bigvee hh^{-1}(0) = \bigvee 0 = 0
\]

\[
\Rightarrow h(b) = 0
\]

\[
\Rightarrow b \in h^{-1}(0)
\]

\[
\Rightarrow b \leq \bigvee h^{-1}(0)
\]
Now $p_X^{-1}$ is a well-defined map in $Cl\mathcal{V}$. Let $A \subseteq F(X)$ then

$$p_X^{-1}(\bigvee A) = \bigvee (\bigvee A)^{-1}(0)$$

$$= \bigvee_{f\in A} f^{-1}(0)$$

$$= \bigwedge_{f\in A} \bigvee f^{-1}(0)$$

$$= \bigwedge_{f\in A} \bigvee f^{-1}(0)$$

$$= \bigwedge_{f\in A} p_X^{-1}(f)$$

$$= \bigvee \mathcal{P}(p_X^{-1})(A)$$

where the (⋆) is a well-known fact from order theory: Let $A = \{A_i | i \in I\}$ and $j \in I$, we have

$$\bigcap_{i \in I} A_i \subseteq A_j \implies \bigcup_{i \in I} A_i \subseteq \downarrow A_j$$

$$\implies \bigvee_{i \in I} A_i \subseteq A_j$$

Hence $\bigvee_{i \in I} A_i \subseteq \bigwedge_{i \in I} \bigvee A_i$. For the other relation we start again with $\bigcap_{i \in I} A_i \subseteq A_j$ so $\bigvee_{i \in I} A_i \leq \bigvee A_j$ for all $j \in I$. Thus $\bigvee_{i \in I} A_i \leq \bigwedge_{i \in I} \bigvee A_i$, concluding the result.

Moreover we have that $p_X^{-1}$ is the inverse of $p_X$. Let $h : X \to 2$ be join-preserving and $y, a \in X$

$$p_X \circ p_X^{-1}(h)(y) = p_X(\bigvee h^{-1}(0))(y) = \chi'_{\bigvee h^{-1}(0)}(y) = 0$$

$$\iff y \in \downarrow \bigvee h^{-1}(0) = h^{-1}(0)$$

$$\iff h(y) = 0$$

And

$$p_X^{-1} \circ p_X(a) = p_X^{-1}(\chi'_{\downarrow a}) = \bigvee (\chi'_{\downarrow a})^{-1}(0) = \bigvee \downarrow a = a$$

Hence $p_X : X^\partial \to K_{\mu_1}(X)$ is an isomorphism.

We have now reasons to believe that $\mathcal{E}\mathcal{M}(S) \cong (Cl\mathcal{V})^{op}$ is not yet in the right form for we have $K_{\mu_1}(X) \cong (X, \bigvee) = (X, \bigwedge)$. We have already seen that $(-)^\partial$ is an isomorphism so we have the following.

**Corollary 5.10.** By the isomorphism $(-)^\partial$ we have

$$\mathcal{E}\mathcal{M}(S) \cong (Cl\mathcal{A})^{op}$$

This is finally the form we aimed for, yielding a triangle of the following form

$$\begin{array}{ccc}
\mathcal{E}\mathcal{M}(S) \cong (Cl\mathcal{A})^{op} & \xrightarrow{L_{\omega} \circ (-)^\partial} & Cl\mathcal{V} \cong \mathcal{E}\mathcal{M}(T) \\
\downarrow \mathcal{T} & & \downarrow \mathcal{T} \\
\mathcal{K} & \xrightarrow{(-)^\partial \circ K_{\omega}} & \mathcal{K}_{\ell}(T)
\end{array}$$
The final objective of this section will be to describe the interesting $\text{Pred}$ functor in a nicer way. The following proposition will help us.

**Proposition 5.11.** For the powerset monad $\mathcal{P}$ and fixed algebra $\mu_1 : \mathcal{P}(2) \to 2$ we have

$$K_{\mu_1} \cong (-)^{\partial} \circ (-)^\sharp : \text{Cl}_\mathcal{V} \to (\text{Cl}_\mathcal{A})^{\text{op}} \to (\text{Cl}_\mathcal{V})^{\text{op}}$$

by the natural isomorphism induced by the maps

$$p_X : X^{\text{op}} \to K_{\mu_1}(X)$$

from lemma 5.9.

**Proof.** We already saw that each $p_X$ is a well-defined isomorphism. Hence we only need to check the naturality. Let $f : (X, \alpha) \to (Y, \beta)$ be a $\mathcal{P}$-algebra morphism then

$$p_X \circ K_{\mu_1}(f)(h) = p_X \circ F(f)(h) = p_X(h \circ f) = \bigvee(h \circ f)^{-1}(0) = \bigvee f^{-1}(h^{-1}(0)) = \bigvee f^{-1}(\bigvee h^{-1}(0)) = f^\sharp(\bigvee h^{-1}(0)) = (f^\partial)^{\partial} \circ p_Y(h)$$

Concluding the proof. $\square$

**Corollary 5.12.**

$\text{Pred} \cong (-)^{\partial} \circ K_{\mu_1} \circ K \cong (-)^\sharp \circ K$

Hence for a Kleisli map $f : X \to \mathcal{P}(Y)$ and $A \subseteq Y$ this yields a map

$$\text{Pred}(f) : \mathcal{P}(Y) \to \mathcal{P}(X)$$

given by

$$\text{Pred}(f)(A) = K(f)^\sharp(A) = \bigcup K(f)^{-1}(\downarrow A) = \bigcup(\mu_Y \circ \mathcal{P}(f))^{-1}(\downarrow A) = \bigcup\{B \in \mathcal{P} \mid \mu_Y \circ \mathcal{P}(f) \subseteq A\} = \bigcup\{B \in \mathcal{P} \mid \bigcup_{b \in B} f(b) \subseteq A\} = \{x \in X \mid f(x) \subseteq A\}$$

where the last equality needs some more thought. But if $f(X) \subseteq A$ the surely $\{x\} \in \mathcal{P}(X)$ and $\bigcup_{b \in \{x\}} f(b) = f(x) \subseteq A$. Conversely if the exists a $B \in \mathcal{P}(X)$ such that $x \in B$ and $\bigcup_{b \in B} f(b) \subseteq A$ then $f(x) \subseteq \bigcup_{b \in B} f(b) \subseteq A$.

We have described categories and functors in this example and thereby obtained a full interpretation of the triangle. Hopefully more examples will follow.
Conclusions and Future work

We had the following three questions we wanted to answer, all new contributions to the subject:

1. What is the most general category on which the monad can be defined?
2. How can the triangle be constructed only knowing the monad and the fixed algebra?
3. How do the vertices of the triangle relate to each other?

Note that we have a lot more background than in the introduction. So we can adapt the questions without changing the content. Let us now see how this thesis answered these questions:

1. We generalised the category from $\text{Sets}$ to a symmetric monoidal closed category with equalizers. The first part is intensively needed throughout the thesis. The equalizers are needed in section 4, trying to answer the second question.

2. Section 4 hands us a way of constructing a category and functors for which in one of our examples, section 5, these are the desired ones. This indeed doesn’t answer the question but could turn out to be so in future work.

3. This question is strongly related to the previous and so is the answer. Beck’s theorem and our own work give a handle on things, but further examples will illumine all their consequences.

The bulk of the work was put into section 4, finding a way into the heart of the problem. As every mathematician will agree, this does not always show in the final text. Although I have not come to explore the depth of the consequences of this idea, I hope this can be done in the future by working out more examples handed by our motivating article. This is where I have to stop.
Bibliography


