String Diagrams for Elementary Category Theory

4: Putting it all together

Dan Marsden
Based on joint work with Ralf Hinze

April 5, 2023

Definition (Free algebra functor)

In Lecture 2 we introduced the category of algebras for an endofunctor Σ, which equips a given category $\mathcal{C}$ with additional structure. There is a forgetful functor, the **underlying functor**,

$$\mathsf{U}^{\Sigma} : \Sigma\text{-}\mathbf{Alg}(\mathcal{C}) \to \mathcal{C} \qquad \begin{aligned} \mathsf{U}^{\Sigma}\,(A, a) &:= A, \\ \mathsf{U}^{\Sigma}\,(h) &:= h, \end{aligned}$$

Definition (Free algebra functor)

In Lecture 2 we introduced the category of algebras for an endofunctor Σ, which equips a given category $\mathcal{C}$ with additional structure. There is a forgetful functor, the **underlying functor**,

$$\mathsf{U}^{\Sigma} : \Sigma\text{-}\mathbf{Alg}(\mathcal{C}) \to \mathcal{C} \qquad \begin{aligned} \mathsf{U}^{\Sigma}\,(A, a) &:= A, \\ \mathsf{U}^{\Sigma}\,(h) &:= h, \end{aligned}$$

If it exists, the left adjoint to the forgetful functor sends an object to the **free algebra**:

$$\Sigma\text{-}\mathbf{Alg}(\mathcal{C}) \underset{\mathsf{U}^{\Sigma}}{\overset{\mathsf{Free}^{\Sigma}}{\rightleftarrows}} \mathcal{C} \quad (\bot) \qquad \begin{aligned} \mathsf{Free}^{\Sigma}\,A &=: (\Sigma^{*}\,A, \mathsf{in}\,A) \\ \mathsf{Free}^{\Sigma}\,f &=: \Sigma^{*}\,f \end{aligned} \, .$$

Free monads

Free algebra intuitions and terminology

We can think of the endofunctor Σ a signature of operations. For example if $\Sigma(X) = X \times X + 1$, then a Σ-algebra on A is a function of the form:

$$A \times A + 1 \to A$$

Free monads

Free algebra intuitions and terminology

We can think of the endofunctor Σ a signature of operations. For example if $\Sigma(X) = X \times X + 1$, then a Σ-algebra on A is a function of the form:

$$A \times A + 1 \to A$$

Which is equivalent to two functions:

$$A \times A \to A \quad \text{and} \quad 1 \to A$$

Free monads

Free algebra intuitions and terminology

We can think of the endofunctor Σ a signature of operations. For example if $\Sigma(X) = X \times X + 1$, then a Σ-algebra on A is a function of the form:

$$A \times A + 1 \to A$$

Which is equivalent to two functions:

$$A \times A \to A \quad \text{and} \quad 1 \to A$$

As a function of type $1 \to A$ encodes an element of A, this is equivalent to choosing a binary function $A \times A \to A$, which we shall denote $+$ an an element of A, which we shall denote 0.

Free monads

Free algebra intuitions and terminology

Example

For our example Σ, the following are algebras:

- The natural numbers, with $+$ interpreted by addition, and 0 by zero.
- The natural numbers, with $+$ interpreted by *multiplication*, and 0 by 33.
- The set of all strings of natural numbers, with $+$ concatenation, and 0 the empty string.
- The set of all strings of natural numbers, with $+$ concatenation, and 0 the string $[33]$.

Free monads

Free algebra intuitions and terminology

For our example Σ, the forgetful functor does have a left adjoint. $\Sigma^*\,A$ consists of all terms of the form:

$$0, a, (0 + a) + 0, a + a, \ldots$$

The free algebra structure map

$$\mathsf{in}\;A : \Sigma\,\Sigma^*\,A \to \Sigma^*\,A$$

picks out as constant element the term 0, and the binary operation is formal addition:

$$(s, t) \mapsto s + t$$

Free monads

Free algebra intuitions and terminology

Further structure:

- The unit of the adjunction, var $A : A \to \Sigma^* A$, turns an element of A into the term for the corresponding variable.
- The counit $\epsilon\,(A, a) : \mathsf{Free}^{\Sigma}\, A \to (A, a)$ evaluates a term using the operations of a given algebra.

This intuitive pattern repeats for different choices of Σ.

Free monads

Folds

We introduce shorthand for the evaluation map:

$$\mathsf{U}^{\Sigma}\,(\epsilon\,(A, a)).$$

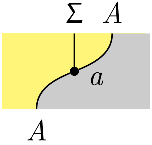
This is an arrow in the underlying category, which we shall denote $(\!|a|\!)$, and pronounce "**fold** a:

$$\frac{a : \Sigma\, A \to A}{(\!|a|\!) : \Sigma^{*}\, A \to A}.$$

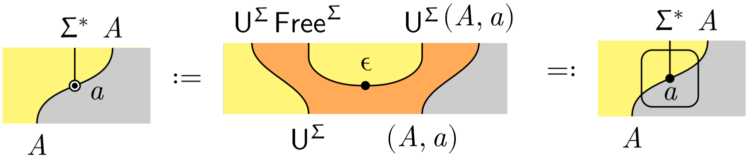
Free monads

Folds

For an algebra:



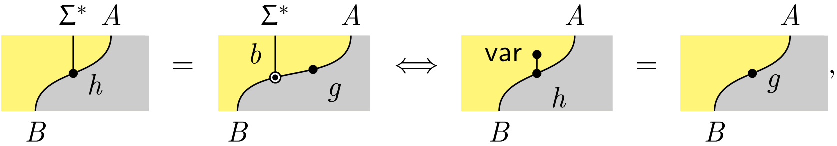
we can depict $(\!|a|\!)$ as follows:



Free monads

The free / forgetful adjunction

The universal property of the free / forgetful adjunction can be unpacked in terms of the base category as

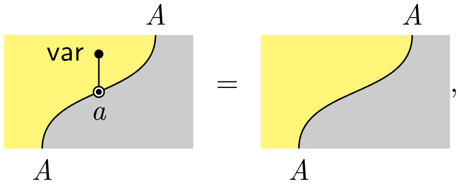


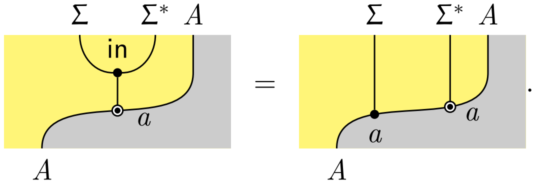
for all Σ-*homomorphisms* h : $\mathsf{Free}^{\Sigma}\ A \to (B, b)$ and all arrows $g : A \to B$.

Free monads

The free / forgetful adjunction

We have the following two **computation rules**:

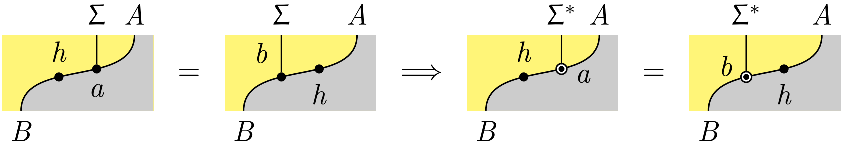




Free monads

The free / forgetful adjunction

Naturality of the counit gives rise to the *elevation rule*:



Free monads

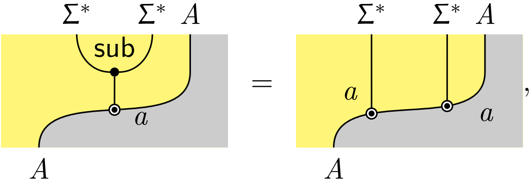
Algebras of endofunctors and monads

If we apply Huber's construction to the adjunction $\mathsf{Free}^{\Sigma} \dashv \mathsf{U}^{\Sigma}$, we obtain the so-called **free monad of a functor**: $(\Sigma^*, \mathsf{var}, \mathsf{sub})$ where $\mathsf{sub} = (\!|\mathsf{in}|\!)$.

Free monads

Algebras of endofunctors and monads

Substitution plays nicely with evaluation. Combining the second computational rule with elevation gives:



Combining this observation with the first computation rule, it tells us that for every Σ-algebra (A, a), $(\Sigma^* A, (\!|a|\!))$ is an Eilenberg–Moore algebra. That this mapping preserves homomorphisms follows from the elevation rule.

Free monads

Algebras of endofunctors and monads

- Fold yields a functor $\mathsf{Up} : \Sigma\text{-}\mathbf{Alg}(\mathcal{C}) \to \mathcal{C}^{\Sigma^*}$.

Free monads

Algebras of endofunctors and monads

- Fold yields a functor $\mathsf{Up} : \Sigma\text{-}\mathbf{Alg}(\mathcal{C}) \to \mathcal{C}^{\Sigma^*}$.
- It turns out that every Σ^* Eilenberg–Moore algebra arises this way.

Free monads

Algebras of endofunctors and monads

- Fold yields a functor $\mathsf{Up} : \Sigma\text{-}\mathbf{Alg}(\mathcal{C}) \to \mathcal{C}^{\Sigma^*}$.
- It turns out that every Σ^* Eilenberg–Moore algebra arises this way.
- Categorically, the category of Σ-algebras is isomorphic to the Eilenberg–Moore category of the free monad Σ^*:

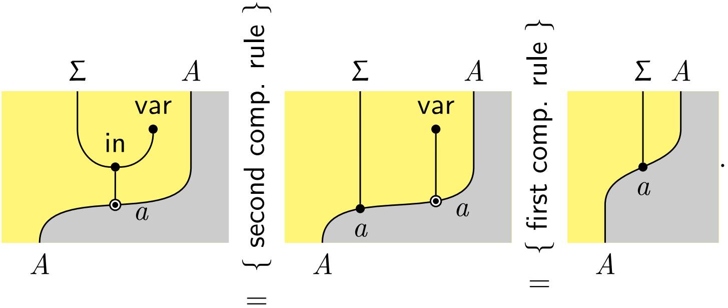
$$\mathsf{Up} : \Sigma\text{-}\mathbf{Alg}(\mathcal{C}) \cong \mathcal{C}^{\Sigma^*} : \mathsf{Dn}$$

To see this, we first define $\mathsf{Dn} : \mathcal{C}^{\Sigma^*} \to \Sigma\text{-}\mathbf{Alg}(\mathcal{C})$.

Free monads

Algebras of endofunctors and monads

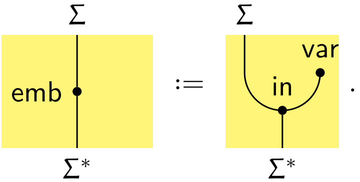
We can map Eilenberg–Moore to Σ-algebras algebras exploiting the computation rules, reversing Up in the process:



Free monads

Algebras of endofunctors and monads

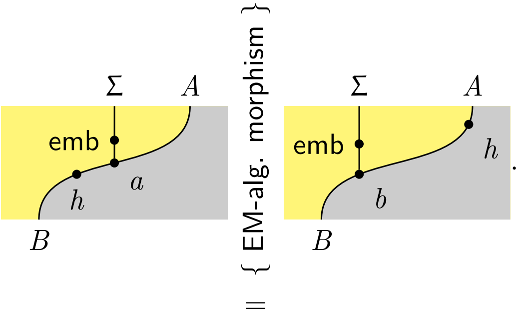
So precomposing with the following map takes Eilenberg–Moore to Σ-algebras:



Free monads

Algebras of endofunctors and monads

That precomposing with emb preserves homomorphisms is straightforward:



Free monads

Algebras of endofunctors and monads

We have two identity on morphisms functors:

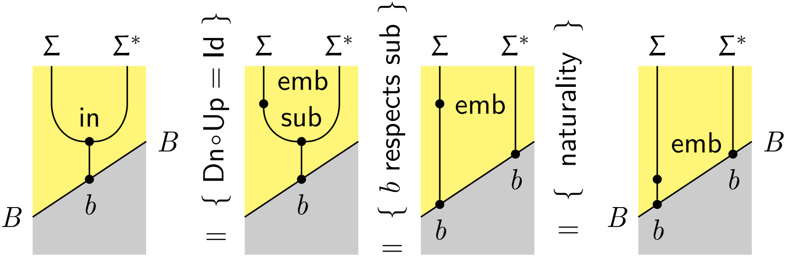
$$\mathsf{Up}\,(A, a : \Sigma\, A \to A) = (A, (\!|a|\!)) \qquad \mathsf{Up}\,h = h,$$
$$\mathsf{Dn}\,(B, b : \Sigma^*\, B \to B) = (B, b \cdot \mathsf{emb}\,B) \qquad \mathsf{Dn}\,h = h.$$

$\mathsf{Dn} \circ \mathsf{Up} = \mathsf{Id}$ by design. It remains to show $\mathsf{Up} \circ \mathsf{Dn} = \mathsf{Id}$. We need to prove that $(\!|b \cdot \mathsf{emb}\,B|\!) = b$.

Free monads

Algebras of endofunctors and monads

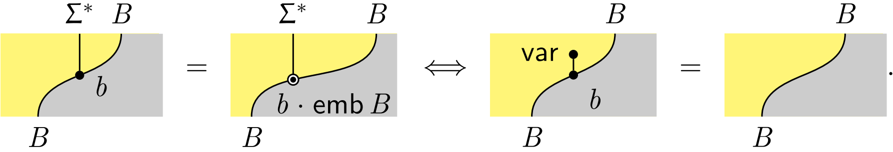
We would like to use the free algebra universal property. To do so, we must show that Eilenberg–Moore algebra $b : \Sigma^* B \to B$ is a Σ-homomorphism $(\Sigma^* B, \mathsf{in}\, B) \to (B, b \cdot \mathsf{emb}\, B)$:



Free monads

Algebras of endofunctors and monads

Therefore we can then appeal to the universal property:



Applying the first computation rule completes the proof of the isomorphism

$$\mathsf{Up} : \Sigma\text{-}\mathbf{Alg}(\mathcal{C}) \cong \mathcal{C}^{\Sigma^*} : \mathsf{Dn}.$$

The resumption monad

The challenge

The challenge

- Let $\mathsf{M} : \mathcal{C} \to \mathcal{C}$ be a monad and $\mathsf{F} : \mathcal{C} \to \mathcal{C}$ be an endofunctor; we aim to show that $\mathsf{M} \circ (\mathsf{F} \circ \mathsf{M})^*$ is a monad.

The challenge

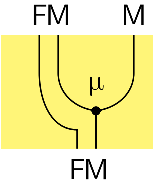
- Let $\mathsf{M} : \mathcal{C} \to \mathcal{C}$ be a monad and $\mathsf{F} : \mathcal{C} \to \mathcal{C}$ be an endofunctor; we aim to show that $\mathsf{M} \circ (\mathsf{F} \circ \mathsf{M})^*$ is a monad.
- In fact, we shall generalise, and show that given a right monad action $\alpha : \Sigma \circ \mathsf{M} \stackrel{.}{\to} \Sigma$, $\mathsf{M} \circ \Sigma^*$ is a monad.

The resumption monad

The challenge

The challenge

- Let $\mathsf{M} : \mathcal{C} \to \mathcal{C}$ be a monad and $\mathsf{F} : \mathcal{C} \to \mathcal{C}$ be an endofunctor; we aim to show that $\mathsf{M} \circ (\mathsf{F} \circ \mathsf{M})^*$ is a monad.
- In fact, we shall generalise, and show that given a right monad action $\alpha : \Sigma \circ \mathsf{M} \xrightarrow{\cdot} \Sigma$, $\mathsf{M} \circ \Sigma^*$ is a monad.
- We recover the original result with $\Sigma = \mathsf{F} \circ \mathsf{M}$ and the monad action:



The resumption monad

The plan

We have the following structure available to us:

$$\Sigma\text{-}\mathbf{Alg}(\mathcal{C}) \underset{\mathsf{U}^{\Sigma}}{\overset{\mathsf{Free}^{\Sigma}}{\rightleftarrows}} \mathcal{C} \circlearrowleft \mathsf{M}\,.$$

($\mathsf{Free}^{\Sigma} \dashv \mathsf{U}^{\Sigma}$)

The resumption monad

The plan

We have the following structure available to us:

$$\Sigma\text{-}\mathbf{Alg}(\mathcal{C}) \underset{\mathsf{U}^{\Sigma}}{\overset{\mathsf{Free}^{\Sigma}}{\underset{\longrightarrow}{\overset{\longleftarrow}{\bot}}}} \mathcal{C} \circlearrowleft \mathsf{M}\,.$$

- This looks close to the Huber situation, but M lives at the "wrong end".

The resumption monad

The plan

We have the following structure available to us:

$$\Sigma\text{-}\mathbf{Alg}(\mathcal{C}) \underset{\mathsf{U}^{\Sigma}}{\overset{\mathsf{Free}^{\Sigma}}{\underset{\to}{\overset{\leftarrow}{\bot}}}} \mathcal{C} \circlearrowright \mathsf{M}\,.$$

- This looks close to the Huber situation, but M lives at the "wrong end".
- If we have a monad $\overline{\mathsf{M}} : \Sigma\text{-}\mathbf{Alg}(\mathcal{C}) \to \Sigma\text{-}\mathbf{Alg}(\mathcal{C})$, we could apply Huber's construction, giving a monad $\mathsf{U}^{\Sigma} \circ \overline{\mathsf{M}} \circ \mathsf{Free}^{\Sigma}$.

The resumption monad

The plan

We have the following structure available to us:

$$\Sigma\text{-}\mathbf{Alg}(\mathcal{C}) \underset{\mathsf{U}^{\Sigma}}{\overset{\mathsf{Free}^{\Sigma}}{\rightleftarrows}} \mathcal{C} \circlearrowright \mathsf{M}\,.$$

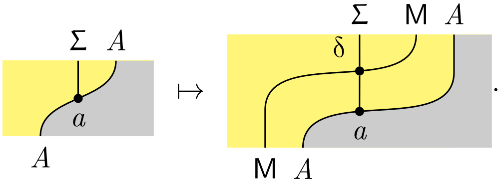
- This looks close to the Huber situation, but M lives at the "wrong end".
- If we have a monad $\overline{\mathsf{M}} : \Sigma\text{-}\mathbf{Alg}(\mathcal{C}) \to \Sigma\text{-}\mathbf{Alg}(\mathcal{C})$, we could apply Huber's construction, giving a monad $\mathsf{U}^{\Sigma} \circ \overline{\mathsf{M}} \circ \mathsf{Free}^{\Sigma}$.
- If additionally, $\mathsf{U}^{\Sigma} \circ \overline{\mathsf{M}} = \mathsf{M} \circ \mathsf{U}^{\Sigma}$, then we have:

$$\mathsf{U}^{\Sigma} \circ \overline{\mathsf{M}} \circ \mathsf{Free}^{\Sigma} = \mathsf{M} \circ \mathsf{U}^{\Sigma} \circ \mathsf{Free}^{\Sigma} = \mathsf{M} \circ \Sigma^{*}.$$

The resumption monad

How to build a suitable $\overline{\mathsf{M}}$

A natural transformation $\delta : \Sigma \circ \mathsf{M} \stackrel{\cdot}{\to} \mathsf{M} \circ \Sigma$ induces a functor $\overline{\mathsf{M}}$ with action:



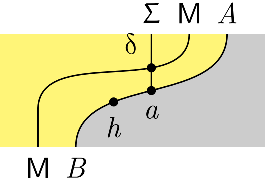
With:

$$\mathsf{U}^{\Sigma} \circ \overline{\mathsf{M}}\,(A, a) = \mathsf{U}^{\Sigma}\,(\mathsf{M}\,(A), \mathsf{M}\,(a) \cdot \delta_A) = \mathsf{M}\,(A) = \mathsf{M} \circ \mathsf{U}^{\Sigma}\,(A, a).$$

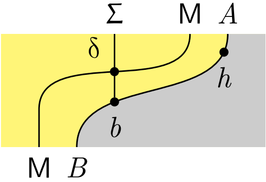
The resumption monad

How to build a suitable $\overline{\mathsf{M}}$

This operation preserves homomorphisms as:



$=\{h \text{ a } \Sigma\text{-mor.}\}$



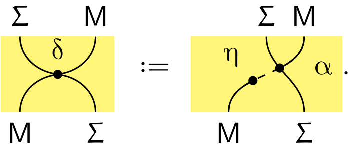
.

Preservation of identities and composition is then immediate as M does.

The resumption monad

Building a suitable δ

Given a right monad action $\alpha : \Sigma \circ M \dot{\to} \Sigma$, we can build a suitable δ as the composite:



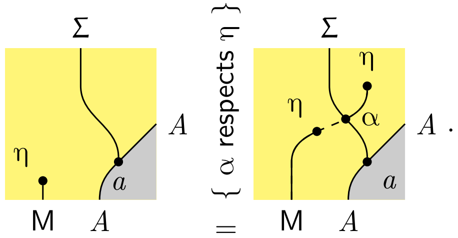
The resumption monad

Lifted unit and multiplication

To show that each component of $\overline{\eta} : \mathsf{Id} \stackrel{.}{\to} \overline{\mathsf{M}}$,

$$\overline{\eta}\,(A, a) : (A, a) \to (\mathsf{M}\,A, \mathsf{M}\,a \cdot \delta\,A),$$

is a homomorphism, we simply apply the unit action law:



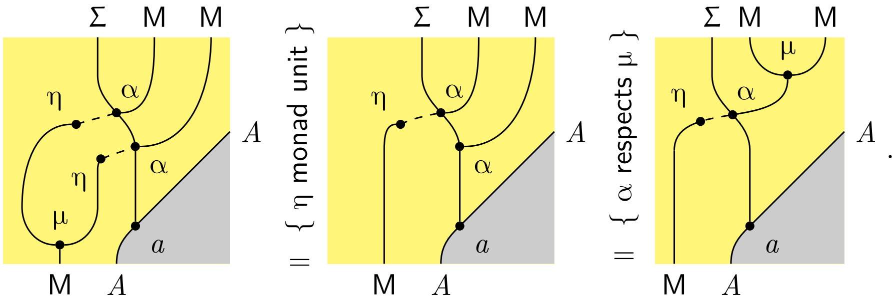
The resumption monad

Lifted unit and multiplication

Likewise, to establish that each component of $\overline{\mu} : \overline{\mathsf{M}} \circ \overline{\mathsf{M}} \stackrel{.}{\to} \overline{\mathsf{M}}$,

$$\overline{\mu}\,(A, a) : (\mathsf{M}\,(\mathsf{M}\,A), \mathsf{M}\,(\mathsf{M}\,a) \cdot \mathsf{M}\,(\delta\,A) \cdot \delta\,(\mathsf{M}\,A)) \to (\mathsf{M}\,A, \mathsf{M}\,a \cdot \delta\,A),$$

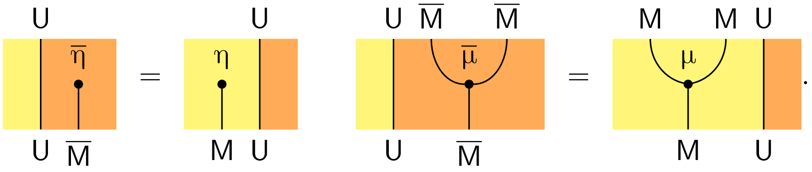
is a homomorphism, we reason



The resumption monad

Invoking Huber's construction

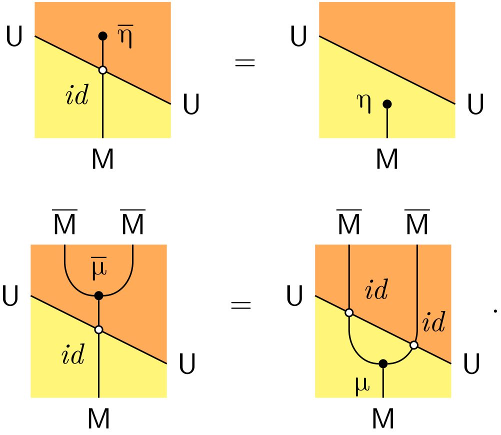
By construction, $\overline{\eta}$ and $\overline{\mu}$ satisfy the following equations:



The resumption monad

Invoking Huber's construction

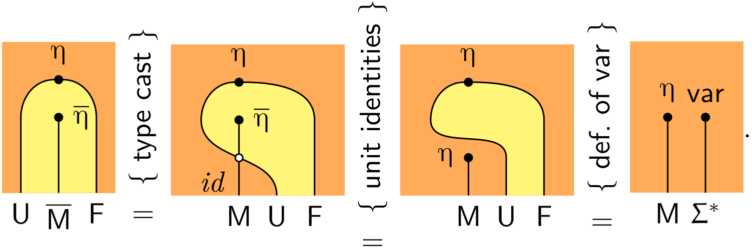
By adding explicit identity natural transformations, we get a more statisfactory rendition:



The resumption monad

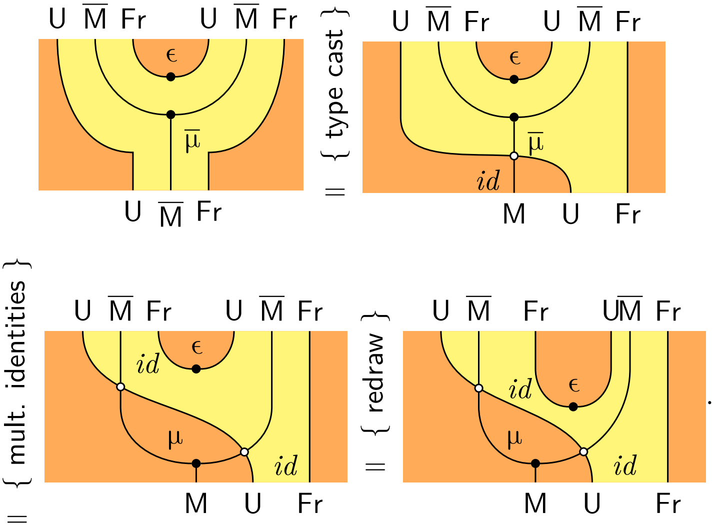
Invoking Huber's construction

We are now in a position to put the unit of the composite monad in concrete terms:



The resumption monad

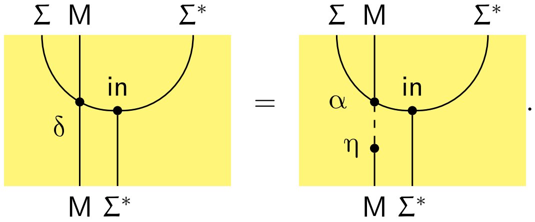
Invoking Huber's construction



The resumption monad

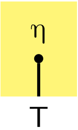
Explicit description

The counit can be described in terms of a fold for the following algebra:

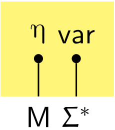


The resumption monad

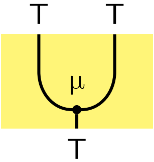
Explicit description



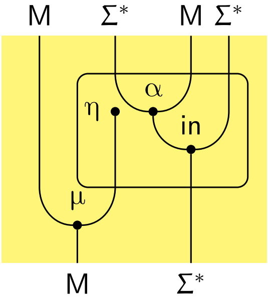
$=$



and



$=$



.

Further directions

Further elementary category theory with nice graphical perspectives, e.g.

- Universals and the Yoneda Lemma.
- Lots about distributive laws.
- Kan extensions and codensity monads.

Further directions

Other settings e.g.

- Monoidal categories, braided monoidal categories, symmetric monoidal categories...
- Double categories.
- Higher categories, and combinations of structures such as monoidal 2-categories.

Further directions

Applications e.g.

- Quantum theory / computation.
- Control theory.
- Linear algebra.
- Natural language semantics.
- Analog and digital electronics.

Further directions

Theory:

- PROs, PROPs, ...
- Coherence theorems.
- Expressivity, soundness and completeness results.

Further directions

Tools:

- Proof assistants.
- Diagramming tools.
- Diagramming libraries.