From typed λ-calculi to Cartesian closed categories

Last time we gave Lambek & Scott's definition of typed λ-calculus (minus their "natural numbers object"), involving

- types
- terms of a given type
- relations between terms

They show that for any typed λ-calculus, P, one can construct a CCC C_P where:

- objects of C_P are types of P
- morphisms $f: A \to B$ are (equivalence classes of) pairs $(x \in A, \varphi(x))$ where $\varphi(x)$ is a term of type B with x as its only free variable.
- equations between morphisms $(x \in A, \varphi(x))$ and $(y \in A, \psi(y))$ where $\varphi(x), \psi(y)$ are of the same type and

$$\varphi(x) \underset{\{x\}}{=} \psi(x).$$

Given morphisms

$$(x \in A, \varphi(x)) \qquad \qquad (\text{say } \varphi(x) \in B)$$

and

$$(y \in B, \psi(y)) \qquad \qquad (\text{say } \psi(y) \in C)$$

we define their composite to be

$$(x \in A, \psi(\varphi(x)))$$

Also, every object A has an identity morphism:

$$(x \in A, x)$$

Lambek & Scott show that C_p is a cartesian closed category, so we can think of a typed λ-calculus as a "presentation" of a CCC. In fact, they construct a category λCalc where the objects are typed λ-calculi & the morphisms are "translations", sending

- types to types
- terms to terms
- relations to relations

There's a category Cart where the objects are CCCs and the morphisms are cartesian closed functors

$$F : C \longrightarrow C'$$

$F : C \longrightarrow C'$ being <u>cartesian</u> means

$$F(1_C) \cong 1_{C'}$$

(where 1_C denotes 'the' terminal object of C)

and

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

where we don't need to specify an isomorphism in either case since terminal objects and products are defined by universal properties.

$F : C \longrightarrow C'$ being <u>closed</u> means

$$F(\hom(A,B)) \cong \hom(FA, FB)$$

(internal hom)

where again we needn't specify an iso since $\hom(A,B)$ is defined by a universal property:

$$\frac{X \times A \longrightarrow B}{X \longrightarrow \hom(A,B)}$$

Lambek and Scott show there's a functor:

$$C : \lambda\mathrm{Calc} \longrightarrow \mathrm{Cart}$$

which we've just described on objects. They show this is an equivalence of categories, by constructing an adjoint

$$L : \mathrm{Cart} \longrightarrow \lambda\mathrm{Calc}$$

where LC is the internal language of C, having all objects of C as types, terms built from variables of each type & morphisms in C, relations built from equations between morphisms.

Examples:

1) The typed λ-calculus for commutative rings. This has a type R, for our ring, and thus other types including $R \times R$, $\hom(R \times R, R)$, etc. We also have terms including

$$+ \in \hom(R \times R, R)$$
$$\cdot \in \hom(R \times R, R)$$
$$0 \in R$$
$$1 \in R$$

and relations including the usual comm. ring axioms. For example, to state the distributive law we include the relation

$$\cdot(x, +(y,z)) \underset{\{x,y,z\}}{=\!=} +(\cdot(x,y), \cdot(x,z))$$

Let's call this typed λ-calculus $\lambda\mathrm{Th}(\mathrm{CommRng})$.

What's a cartesian closed functor

$$F : C_{\lambda\mathrm{Th}(\mathrm{CommRng})} \longrightarrow \mathrm{Set} \qquad ?$$

A typical object in $C_{\lambda\mathrm{Th}(\mathrm{CommRng})}$ is

$$\hom(\hom(R \times R, 1), \hom(R,R))$$

i.e. any type in $\lambda\mathrm{Th}(\mathrm{CommRng})$. Some typical morphisms in $C_{\lambda\mathrm{Th}(\mathrm{CommRng})}$ are

$$\big(x \in \hom(R,R),\ y \in R \longmapsto x(x(y))\big)$$

(a morphism from $\hom(R,R)$ to $\hom(R,R)$ that "composes x with itself")

or

$$\big(x \in R,\ +(x, \cdot(x, \cdot(x,x)))\big)$$

To be continued...