Containers

Thorsten Altenkirch

University of Nottingham

The gang

Joint work with

- Michael Abbott, Neil Ghani (Leicester)
- Conor McBride (Durham)

FOSSACS 03 Michael,Neil and me
Categories of Containers

TLCA 03 Michael,Neil, Conor and me
Derivatives of Containers

See my publications page for electronic copies:
`http://www.cs.nott.ac.uk/˜txa/publ/`

Health warning!

I am going to illustrate our results using **Sets**. However, they are applicable to a much wider range of categories.

What is a container?

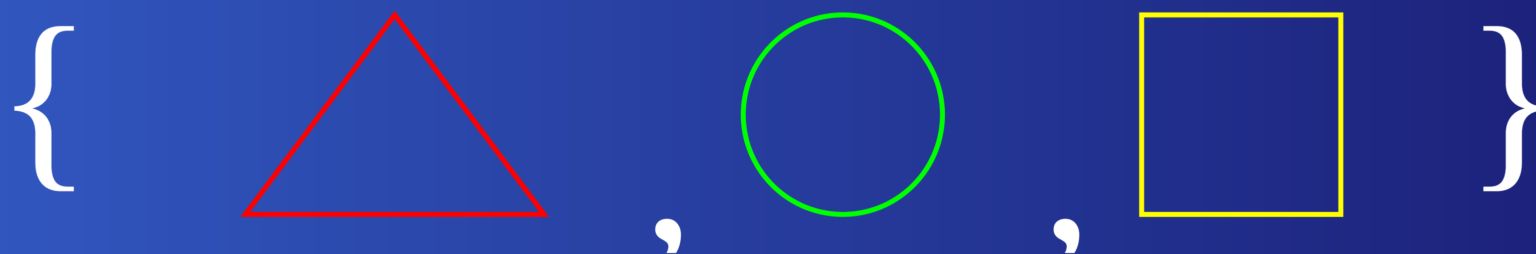
What is a container?

A container type $S \triangleright P$ is given by:

What is a container?

A container type $S \triangleright P$ is given by:

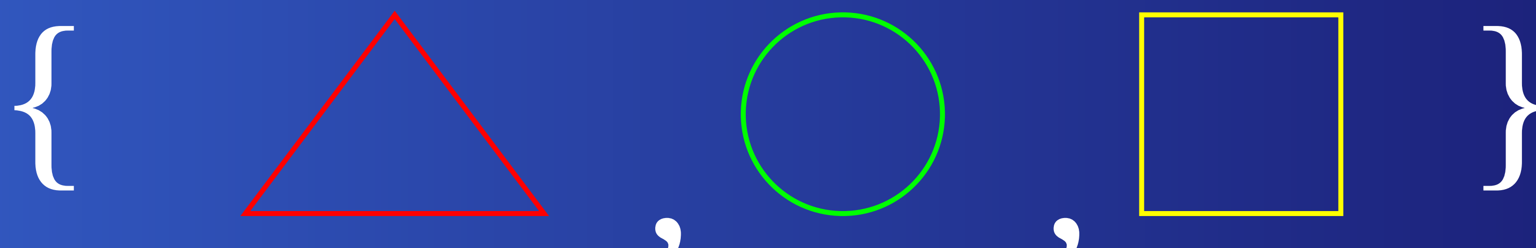
- A set S of shapes, e.g.



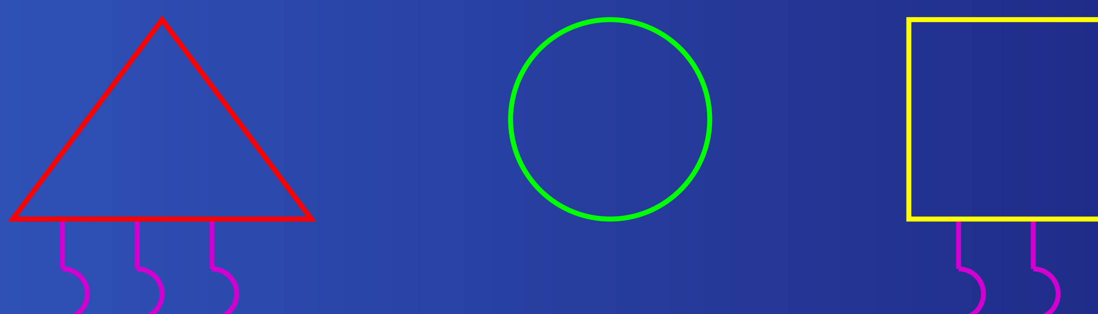
What is a container?

A container type $S \triangleright P$ is given by:

- A set S of shapes, e.g.



- For any shape $s \in S$ a set of positions $P(s)$,

e.g. 

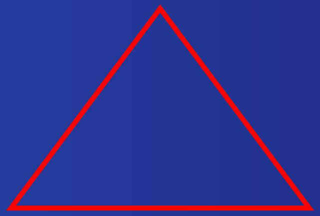
What to do with a container?

What to do with a container?

Given some payload X, e.g. $X = \mathbb{N}$ we can instantiate a container by

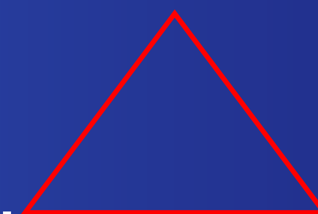
What to do with a container?

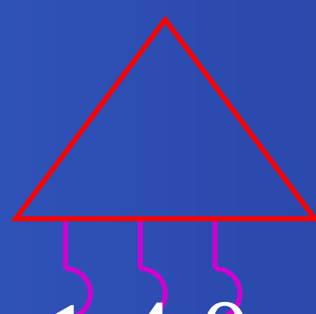
Given some payload X, e.g. $X = \mathbb{N}$ we can instantiate a container by

- Choosing a shape, e.g. 

What to do with a container?

Given some payload X, e.g. $X = \mathbb{N}$ we can instantiate a container by

- Choosing a shape, e.g. 
- Filling the positions with payload, e.g.



e.g. 1 4 0

Extension of a container type

Extension of a container type

The extension $T_{S \triangleright P}$ of a container is given by

$$T_{S \triangleright P}(X) = \Sigma s \in S.P(s) \to X$$

Extension of a container type

The extension $T_{S \triangleright P}$ of a container is given by

$$T_{S \triangleright P}(X) = \Sigma s \in S.P(s) \to X$$

where

$$\Sigma a \in A.B(a) = \{(a, b) \mid a \in A \wedge b \in B(a)\}$$

Extension of a container type

The extension $T_{S \triangleright P}$ of a container is given by

$$T_{S \triangleright P}(X) = \Sigma s \in S.P(s) \to X$$

where

$$\Sigma a \in A.B(a) = \{(a, b) \mid a \in A \wedge b \in B(a)\}$$

$$\Sigma a \in \{0, 1\}.B(a) = B(0) + B(1)$$

Containers are functors

$$T_{S \triangleright P}(X) = \Sigma s \in S.P(s) \to X$$

$$\begin{aligned} T_{S \triangleright P}(f \in X \to Y) &\in T_{S \triangleright P}(X) \to T_{S \triangleright P}(Y) \\ &= (s, h) \mapsto (s, f \circ h) \end{aligned}$$

Example of a container type: $\mathsf{List}(X)$

Example of a container type: $\mathsf{List}(X)$

Lists over X are given by

$$\begin{aligned} \mathrm{nil} &\in \mathsf{List}(X) \\ \mathrm{cons} &\in X \times \mathsf{List}(X) \to \mathsf{List}(X) \end{aligned}$$

Example of a container type: $\mathsf{List}(X)$

Lists over X are given by

$$\begin{aligned} \text{nil} &\in \mathsf{List}(X) \\ \text{cons} &\in X \times \mathsf{List}(X) \to \mathsf{List}(X) \end{aligned}$$

$$\mathsf{List}(X) = \mu Y.1 + X \times Y$$

Example of a container type: $\mathsf{List}(X)$

Lists over X are given by

$$\begin{aligned} \mathrm{nil} &\in \mathsf{List}(X) \\ \mathrm{cons} &\in X \times \mathsf{List}(X) \to \mathsf{List}(X) \end{aligned}$$

$$\mathsf{List}(X) = \mu Y.1 + X \times Y$$

$$\mathsf{List}(X) \simeq \Sigma n \in \mathbb{N}.\{i < n\} \to X$$

Example of a container type: $\mathsf{List}(X)$

Lists over X are given by

$$\begin{array}{rcl} \mathrm{nil} & \in & \mathsf{List}(X) \\ \mathrm{cons} & \in & X \times \mathsf{List}(X) \to \mathsf{List}(X) \end{array}$$

$$\mathsf{List}(X) = \mu Y.1 + X \times Y$$

$$\begin{array}{rcl} \mathsf{List}(X) & \simeq & \Sigma n \in \mathbb{N}.\{i < n\} \to X \\ & = & \Sigma n \in \mathbb{N}.n \to X \end{array}$$

Basic containers

Basic containers

$$I(X) = X$$

Basic containers

$$
\begin{aligned}
I(X) &= X \\
I(X) &\simeq \Sigma s \in 1.1 \to X
\end{aligned}
$$

Basic containers

$$
\begin{aligned}
I(X) &= X \\
I(X) &\simeq \Sigma s \in 1.1 \to X
\end{aligned}
$$

$$
K_C(X) = C
$$

Basic containers

$$
\begin{aligned}
I(X) &= X \\
I(X) &\simeq \Sigma s \in 1.1 \rightarrow X
\end{aligned}
$$

$$
\begin{aligned}
K_C(X) &= C \\
K_C(X) &\simeq \Sigma s \in C.0 \rightarrow X
\end{aligned}
$$

Coproducts of containers

Coproducts of containers

Given containers

$$\begin{aligned} F(X) &= \Sigma s \in S.P(s) \to X \\ G(X) &= \Sigma t \in T.Q(t) \to X \end{aligned}$$

Coproducts of containers

Given containers

$$\begin{aligned} F(X) &= \Sigma s \in S.P(s) \to X \\ G(X) &= \Sigma t \in T.Q(t) \to X \end{aligned}$$

$$F + G(X) = F(X) + G(X)$$

Coproducts of containers

Given containers

$$F(X) = \Sigma s \in S.P(s) \to X$$
$$G(X) = \Sigma t \in T.Q(t) \to X$$

$$F + G(X) = F(X) + G(X)$$

$$F + G(X) \simeq \Sigma u \in S + T.\left(\begin{cases} P(s) & \text{if } u = \mathrm{inl}(s) \\ Q(t) & \text{if } t = \mathrm{inr}(s) \end{cases}\right) \to X$$

Products of containers

Given containers

$$
\begin{aligned}
F(X) &= \Sigma s \in S.P(s) \to X \\
G(X) &= \Sigma t \in T.Q(t) \to X
\end{aligned}
$$

Products of containers

Given containers

$$\begin{aligned} F(X) &= \Sigma s \in S.P(s) \to X \\ G(X) &= \Sigma t \in T.Q(t) \to X \end{aligned}$$

$$F \times G(X) = F(X) \times G(X)$$

Products of containers

Given containers

$$F(X) = \Sigma s \in S.P(s) \to X$$
$$G(X) = \Sigma t \in T.Q(t) \to X$$

$$F \times G(X) = F(X) \times G(X)$$
$$F \times G(X) \simeq \Sigma(s,t) \in S \times T.(P(s) + Q(t)) \to X$$

(Least) Fixpoints of containers

(Least) Fixpoints of containers

Given a 2-ary container

$$H(X,Y) = \Sigma s \in S.P(s) \to X \times Q(s) \to Y$$

(Least) Fixpoints of containers

Given a 2-ary container

$$H(X, Y) = \Sigma s \in S.P(s) \to X \times Q(s) \to Y$$

$$\mu Y.H(X, Y) \;\simeq\; \Sigma t \in T.R(t) \to X$$

(Least) Fixpoints of containers

Given a 2-ary container

$$H(X,Y) = \Sigma s \in S.P(s) \to X \times Q(s) \to Y$$

$$\begin{aligned} \mu Y.H(X,Y) &\simeq \Sigma t \in T.R(t) \to X \\ T &= \mu Y.\Sigma s \in S.Q(s) \to Y \end{aligned}$$

(Least) Fixpoints of containers

Given a 2-ary container

$$H(X,Y) = \Sigma s \in S.P(s) \to X \times Q(s) \to Y$$

$$\begin{aligned} \mu Y.H(X,Y) &\simeq \Sigma t \in T.R(t) \to X \\ T &= \mu Y.\Sigma s \in S.Q(s) \to Y \\ &= W(S,Q) \end{aligned}$$

(Least) Fixpoints of containers

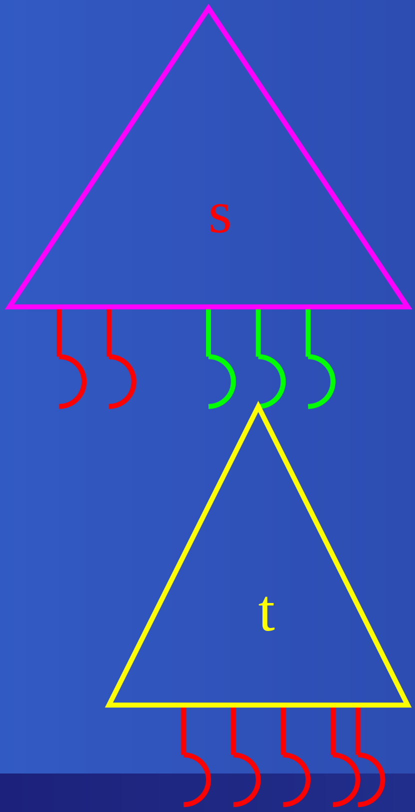
Given a 2-ary container

$$H(X,Y) = \Sigma s \in S.P(s) \to X \times Q(s) \to Y$$

$$\begin{aligned} \mu Y.H(X,Y) &\simeq \Sigma t \in T.R(t) \to X \\ T &= \mu Y.\Sigma s \in S.Q(s) \to Y \\ &= W(S,Q) \\ R(s,f) &\simeq P(s) + \Sigma q \in Q(s).R(f(q)) \end{aligned}$$

(Least) Fixpoints of containers

$$R(s,f) \;\simeq\; P(s) + \Sigma q \in Q(s).R(f(q))$$



Strictly positive types

Strictly positive types

$$(C \to F)(X) = C \to F(X)$$
$$\nu Y.H(X,Y)$$

are closed under containers.

Strictly positive types

$$(C \to F)(X) = C \to F(X)$$
$$\nu Y.H(X, Y)$$

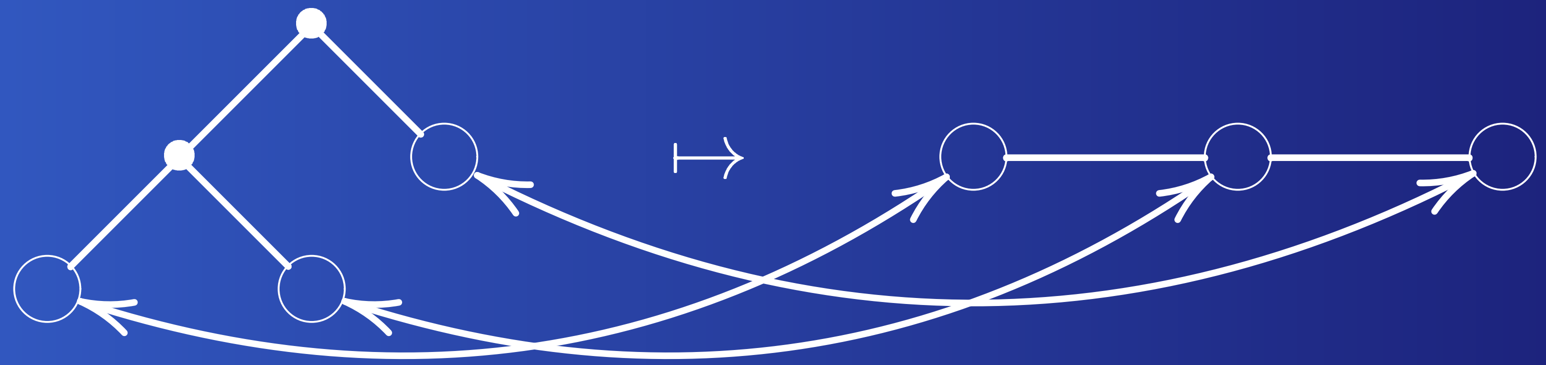
are closed under containers.

Theorem: All strictly positive type expressions give rise to containers.

Morphisms of containers: examples

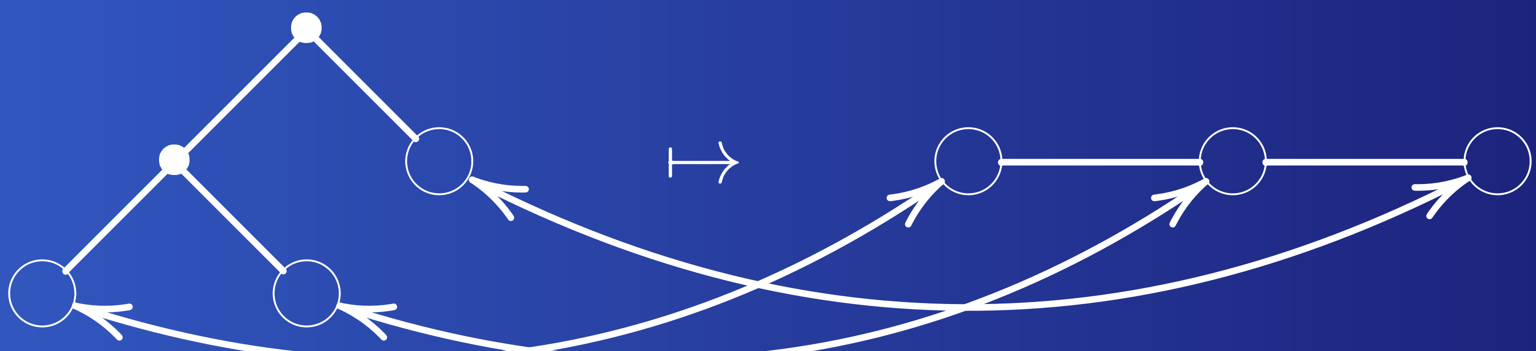
Morphisms of containers: examples

$$\mathsf{flatten} : \mathsf{Tree}\, X \to \mathsf{List}\, X$$



Morphisms of containers: examples

$$\text{flatten} : \text{Tree}\ X \to \text{List}\ X$$



$$\text{tail} : \text{List}\ X \to \text{List}\ X$$

Morphisms of containers

Morphisms of containers

Given containers

$$\begin{aligned} F(X) &= \Sigma s \in S.P(s) \to X \\ G(X) &= \Sigma t \in T.Q(t) \to X \end{aligned}$$

a morphism $f \triangleright u \in \mathcal{G}(F, G)$ is given by

Morphisms of containers

Given containers

$$\begin{aligned} F(X) &= \Sigma s \in S.P(s) \to X \\ G(X) &= \Sigma t \in T.Q(t) \to X \end{aligned}$$

a morphism $f \triangleright u \in \mathcal{G}(F, G)$ is given by

$$f \in S \to T$$

Morphisms of containers

Given containers

$$\begin{aligned} F(X) &= \Sigma s \in S.P(s) \to X \\ G(X) &= \Sigma t \in T.Q(t) \to X \end{aligned}$$

a morphism $f \triangleright u \in \mathcal{G}(F, G)$ is given by

$$\begin{aligned} f &\in S \to T \\ u &\in \Pi s \in S.Q(f(s)) \to P(s) \end{aligned}$$

Morphisms of containers

Given containers

$$\begin{aligned} F(X) &= \Sigma s \in S.P(s) \to X \\ G(X) &= \Sigma t \in T.Q(t) \to X \end{aligned}$$

a morphism $f \triangleright u \in \mathcal{G}(F, G)$ is given by

$$\begin{aligned} f &\in S \to T \\ u &\in \Pi s \in S.Q(f(s)) \to P(s) \\ T_{f \triangleright u}(X) &\in F(X) \to G(X) \end{aligned}$$

Morphisms of containers

Given containers

$$\begin{aligned} F(X) &= \Sigma s \in S.P(s) \to X \\ G(X) &= \Sigma t \in T.Q(t) \to X \end{aligned}$$

a morphism $f \triangleright u \in \mathcal{G}(F, G)$ is given by

$$\begin{aligned} f &\in S \to T \\ u &\in \Pi s \in S.Q(f(s)) \to P(s) \\ T_{f \triangleright u}(X) &\in F(X) \to G(X) \\ &= (s, h) \mapsto (f(s), h \circ u(s)) \end{aligned}$$

The representation theorem

The representation theorem

Theorem: The *extension functor* $T : \mathcal{G} \to (\mathbf{Set} \to \mathbf{Set})$ is full and faithful.

The representation theorem

Theorem: The *extension functor* $T : \mathcal{G} \to (\mathbf{Set} \to \mathbf{Set})$ is full and faithful.
Corollary: All natural transformations $\alpha_X \in \mathsf{List}(X) \to \mathsf{List}(X)$ are given by

- a function on the length $f \in \mathbb{N} \to \mathbb{N}$
- a *repotting function* $u \in \Pi n \in \mathbb{N}. f(n) \to n$.

Containers with holes

Containers with holes

What is a list with a hole?

Containers with holes

What is a list with a hole?

2 Lists !

Containers with holes

What is a list with a hole?
2 Lists !

$$\mathsf{List}'(X) \simeq \mathsf{List}(X) \times \mathsf{List}(X)$$

Containers with holes

What is a list with a hole?
2 Lists !

$$\mathsf{List}'(X) \simeq \mathsf{List}(X) \times \mathsf{List}(X)$$

Can we construct F' more systematically?

Making holes ($\times$)

Making holes ($\times$)

$$T(X) = F(X) \times G(X)$$

Making holes ($\times$)

$$T(X) = F(X) \times G(X)$$

$$T'(X) = F'(X) \times G(X) + F(X) \times G'(X)$$

Making holes (chain)

Making holes (chain)

$$T(X) = F(X, G(X))$$

Making holes (chain)

$$T(X) = F(X, G(X))$$

$$T'(X) = F_1'(X, G(X)) + F_2'(X, G(X)) \times G'(X)$$

Conor’s observation

Conor’s observation

The type of one-hole contexts of a datatype is its formal derivative.

Derivative of μ

Derivative of μ

$$F(X) = \mu Y.H(X, Y)$$

Derivative of μ

$$F(X) = \mu Y.H(X, Y)$$

$$F(X) \simeq H(X, F(X))$$

Derivative of μ

$$F(X) = \mu Y.H(X, Y)$$

$$F(X) \simeq H(X, F(X))$$

$$F'(X) \simeq H_1'(X, F(X)) + H_2'(X, F(X)) \times F'(X)$$

Derivative of μ

$$F(X) = \mu Y. H(X, Y)$$

$$F(X) \simeq H(X, F(X))$$

$$F'(X) \simeq H_1'(X, F(X)) + H_2'(X, F(X)) \times F'(X)$$

$$F'(X) \simeq \mu Y. H_1'(X, F(X)) + H_2'(X, F(X)) \times Y$$

Derivative of List

$$\begin{aligned} \mathsf{List}(X) &= \mu Y.H(X,Y) \\ H(X,Y) &= 1 + X \times Y \end{aligned}$$

Derivative of List

$$
\begin{aligned}
\mathsf{List}(X) &= \mu Y.H(X,Y) \\
H(X,Y) &= 1 + X \times Y \\
H'_1(X,Y) &= Y \\
H'_2(X,Y) &= X
\end{aligned}
$$

Derivative of List

$$
\begin{aligned}
\mathsf{List}(X) &= \mu Y.H(X,Y) \\
H(X,Y) &= 1 + X \times Y \\
H_1'(X,Y) &= Y \\
H_2'(X,Y) &= X \\
\mathsf{List}'(X) &= \mu Y.H_1'(X,\mathsf{List}(X)) + H_2'(X,\mathsf{List}(X)) \times Y
\end{aligned}
$$

Derivative of List

$$
\begin{aligned}
\mathsf{List}(X) &= \mu Y.H(X,Y) \\
H(X,Y) &= 1 + X \times Y \\
H_1'(X,Y) &= Y \\
H_2'(X,Y) &= X \\
\mathsf{List}'(X) &= \mu Y.H_1'(X,\mathsf{List}(X)) + H_2'(X,\mathsf{List}(X)) \times Y \\
&= \mu Y.\,\mathsf{List}(X) + X \times Y
\end{aligned}
$$

Derivative of List

$$
\begin{aligned}
\mathsf{List}(X) &= \mu Y.H(X,Y) \\
H(X,Y) &= 1 + X \times Y \\
H_1'(X,Y) &= Y \\
H_2'(X,Y) &= X \\
\mathsf{List}'(X) &= \mu Y.H_1'(X,\mathsf{List}(X)) + H_2'(X,\mathsf{List}(X)) \times Y \\
&= \mu Y.\,\mathsf{List}(X) + X \times Y \\
&\simeq \mathsf{List}(X) \times \mu Y.1 + X \times Y
\end{aligned}
$$

Derivative of List

$$
\begin{aligned}
\mathsf{List}(X) &= \mu Y.H(X,Y) \\
H(X,Y) &= 1 + X \times Y \\
H_1'(X,Y) &= Y \\
H_2'(X,Y) &= X \\
\mathsf{List}'(X) &= \mu Y.H_1'(X,\mathsf{List}(X)) + H_2'(X,\mathsf{List}(X)) \times Y \\
&= \mu Y.\,\mathsf{List}(X) + X \times Y \\
&\simeq \mathsf{List}(X) \times \mu Y.1 + X \times Y \\
&= \mathsf{List}(X) \times \mathsf{List}(X)
\end{aligned}
$$

Derivative of a container

Derivative of a container

$$F(X) = \Sigma s \in S.P(s) \to X$$

Derivative of a container

$$F(X) = \Sigma s \in S.P(s) \to X$$

$$F'(X) = \Sigma(s,p) \in (\Sigma s \in S.P(s)).(\Sigma q \in P(s).p \neq q) \to X$$

Tangent of a functor

Tangent of a functor

We consider the category $\widehat{\mathcal{G}}$ of **cartesian morphisms** of containers (each morphism between positions is an iso).

Tangent of a functor

We consider the category $\widehat{\mathcal{G}}$ of **cartesian morphisms** of containers (each morphism between positions is an iso).
We write $H \multimap -$ for the right adjoint of $- \times H$ in the category $\widehat{\mathcal{G}}$ (if it exists).

Tangent of a functor

We consider the category $\widehat{\mathcal{G}}$ of **cartesian morphisms** of containers (each morphism between positions is an iso).
We write $H \multimap -$ for the right adjoint of $- \times H$ in the category $\widehat{\mathcal{G}}$ (if it exists).

$$F' = I \multimap F$$

Tangent of a functor

We consider the category $\widehat{\mathcal{G}}$ of **cartesian morphisms** of containers (each morphism between positions is an iso).
We write $H \multimap -$ for the right adjoint of $- \times H$ in the category $\widehat{\mathcal{G}}$ (if it exists).

$$F' = I \multimap F$$

Theorem: Containers with a decidable equality on positions are differentiable and this class of containers is closed under derivaties.

What is e^x

Which container has the property $F'(X) = F(X)$?

What is e^x

Which container has the property $F'(X) = F(X)$?
To give the answer we have to generalize the notion of a container slightly . . .

What is e^x

Which container has the property
$F'(X) = F(X)$?
To give the answer we have to generalize the notion of a container slightly . . .
The answer can be found in our TLCA paper.

Generalizing Sets

Instead of **Sets** we consider
locally cartesian closed categories (LCCCs)
+ extensive coproducts
+ locally finitely presentable (LFP)

Generalizing Sets

Instead of **Sets** we consider
locally cartesian closed categories (LCCCs)
+ extensive coproducts
+ locally finitely presentable (LFP)

We should be able to replace **LFP** by something nicer . . .

Related work

- André Joyal
  Analytical functors and their derviatives.
- Paul Hoogendijk and Oege de Moor
  Containers in categories of relations.
- Bary Jay
  Shapely types.
- Peter Dybjer
  Inductive types as W-types.

What is this good for?

- Generic programming (e.g. the zipper)
- Reasoning about datatypes
- Reasoning about parametric programs
- Combinatorical representations of datatypes
- Optimizing functional programs

Forthcoming

Michael Abbott's PhD thesis
Categories of Containers