Programming with handlers in Links[1]

A brief introduction @ St. Andrews

Daniel Hillerström
s1467124@sms.ed.ac.uk

School of Informatics
The University of Edinburgh

August 4, 2015

Based on work by Plotkin and Pretnar [1] and Kammar et al. [2].

[1] Excerpt from my dissertation "Handlers for Algebraic Effects in Links"

A compelling programming model

Handlers for algebraic effects provide a compelling alternative to monads as a basis for effectful programming.

- **Key idea:** Separate effect signatures from their implementation.
- **Consequence:** High-degree of modularity.

Effects and handlers

Algebraic effect

An effect[a] is a collection of abstract operations, e.g. $\{Op_i : a_i \to b_i\}$

[a] We are assuming the free algebra, i.e. the equationless theory

Abstract computation

An abstract computation is composed from abstract operations.

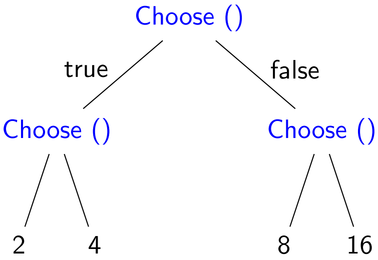
Handler

A handler is an interpreter. It instantiates an abstract computation with a concrete implementation.

Effects as computation trees [3]

Operation $\text{Choose} : () \to \texttt{Bool}$.

Picture the CPS term $\text{Choose}_{()}(\text{Choose}_{()}(2, 4), \text{Choose}_{()}(8, 16))$, e.g.

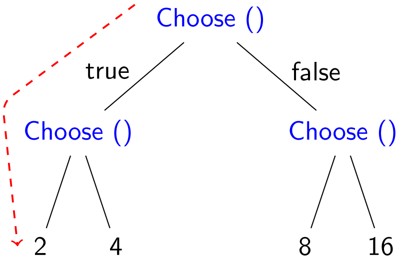


How should we interpret this computation?

Effects as computation trees [3]

Operation $\mathsf{Choose} : () \to \texttt{Bool}$.

Picture the CPS term $\mathsf{Choose}_{()}(\mathsf{Choose}_{()}(2, 4), \mathsf{Choose}_{()}(8, 16))$, e.g.

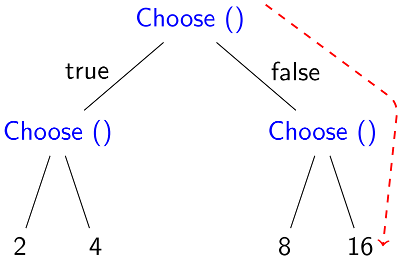


$\Rightarrow 2 : \texttt{Int}$ (Strictly positive)

Effects as computation trees [3]

Operation $\mathsf{Choose} : () \to \texttt{Bool}$.

Picture the CPS term $\mathsf{Choose}_{()}(\mathsf{Choose}_{()}(2,4), \mathsf{Choose}_{()}(8,16))$, e.g.

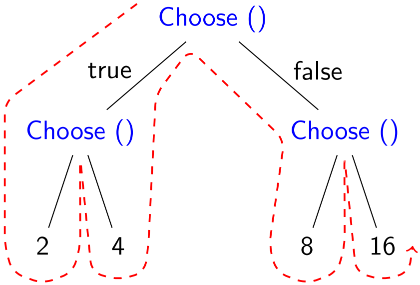


$\Rightarrow 16 : \texttt{Int}$ (depressingly pessimistic)

Effects as computation trees [3]

Operation Choose : $() \to$ `Bool`.

Picture the CPS term $\mathrm{Choose}_{()}(\mathrm{Choose}_{()}(2, 4), \mathrm{Choose}_{()}(8, 16))$, e.g.

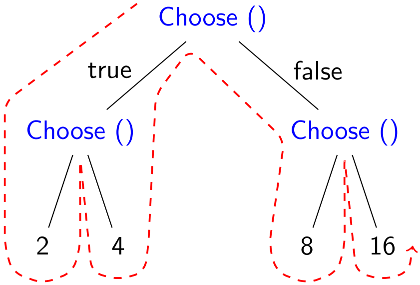


How about this?

Effects as computation trees [3]

Operation Choose : () $\to$ `Bool`.

Picture the CPS term $\mathsf{Choose}_{()}(\mathsf{Choose}_{()}(2,4), \mathsf{Choose}_{()}(8,16))$, e.g.



$\Rightarrow [2,4,8,16] : [\mathtt{Int}]$

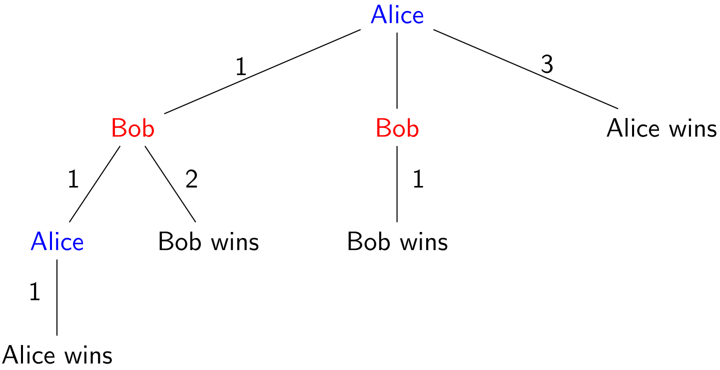
A game with sticks

Set-up

- Two players: Alice and Bob; Alice always starts.
- One heap of *n* sticks.
- Turn-based. Each player take between 1-3 sticks.
- The one, who takes the last stick, wins.

We'll demonstrate how to encode strategic behaviour, compute game data, and cheat using handlers.

Game tree generated by `mtGen` with $n = 3$



References

Gordon D. Plotkin and Matija Pretnar.
Handling algebraic effects.
Logical Methods in Computer Science, 9(4), 2013.

Ohad Kammar, Sam Lindley, and Nicolas Oury.
Handlers in action.
In *ICFP'13*, pages 145–158, 2013.

Sam Lindley.
Algebraic effects and effect handlers for idioms and arrows.
In *Proceedings of the 10th ACM SIGPLAN workshop on Generic programming, WGP 2014, Gothenburg, Sweden, August 31, 2014*, pages 47–58, 2014.