# $\lambda_{\mathit{eff}}$

nymphium

April 7, 2019

## Syntax

$$
\begin{array}{rcl}
x & \in & \textit{Variables} \\
\mathit{eff} & \in & \textit{Effects} \\
\\
v & ::= & x \mid h \mid \lambda x.e \mid \mathit{eff} \\
e & ::= & v \mid e\ e \mid \texttt{let}\ x = e\ \texttt{in}\ e \\
 & \mid & \texttt{inst}\ () \mid \texttt{with}\ v\ \texttt{handle}\ e \\
 & \mid & \texttt{perform}\ e\ e \\
h & ::= & \texttt{handler}\ v\ \ (\texttt{val}\ x \to e)\,((x, k) \to e) \\
\\
F & ::= & e\ \square \mid \square\ v \mid \texttt{let}\ x = \square\ \texttt{in}\ e \\
 & \mid & \texttt{with}\ v\ \texttt{handle}\ \square \mid \texttt{perform}\ \square\ e \mid \texttt{perform}\ v\ \square \\
s & ::= & [] \mid F :: s
\end{array}
$$

Figure 1: the syntax of $\lambda_{\mathit{eff}}$

## Semantics

$$
\begin{array}{c}
\mathit{flatfn}\ [] = \lambda x.x \\
\mathit{flatfn}\ (F :: s) = \lambda x.\mathit{flatfn}\ s\ \ (F\,[x])
\end{array}
$$

Figure 2: utility function *flatfn*

$$
\langle F\,[e]\,;\, s;\, es\rangle \mapsto \langle e;\, F :: s;\, es\rangle \qquad (\textsc{Push})
$$

$$
\langle v;\, F :: s;\, es\rangle \mapsto \langle F\,[v]\,;\, s;\, es\rangle \qquad (\textsc{Pop})
$$

$$
\langle v;\, [];\, es\rangle \mapsto \langle v;\, [];\, es\rangle \qquad (\textsc{Result})
$$

$$
\langle \lambda x.e;\, (\square\ v) :: s;\, es\rangle \mapsto \langle e\,[x = v]\,;\, s;\, es\rangle \qquad (\textsc{Apply})
$$

$$
\langle \texttt{inst}\ ()\,;\, s;\, es\rangle \mapsto \langle \mathit{eff};\, s;\, es\rangle \qquad (\textsc{Instanciate})
$$

$$
\langle \texttt{perform}\ \mathit{eff}\, v;\, F :: s;\, es\rangle \mapsto \langle \texttt{perform}\ \mathit{eff}\, v;\, s;\, F :: es\rangle \qquad (\textsc{Rethrow})
$$

$$
\begin{array}{c}
\langle \texttt{perform}\ \mathit{eff}\, v;\, F :: s;\, es\rangle \mapsto \langle e_{\mathit{eff}}\,[x = v, k = \mathit{flatfn}\ es]\,;\, F :: s;\, []\rangle \\
\text{where}\quad F = \texttt{with}\ h\ \texttt{handle}\ \square \\
\qquad\ h = \texttt{handler}\ \mathit{eff}\ (\texttt{val}\ x \to e_v)\,((x, k) \to e_{\mathit{eff}})
\end{array}
\qquad (\textsc{Handle}_{\textsc{eff}})
$$

$$
\begin{array}{c}
\langle v;\, F :: s;\, es\rangle \mapsto \langle e_v[x = v];\, s;\, es\rangle \\
\text{where}\quad F = \texttt{with}\ h\ \texttt{handle}\ \square \\
\qquad\ h = \texttt{handler}\ \mathit{eff}\ (\texttt{val}\ x \to e_v)\,((x, k) \to e_{\mathit{eff}})
\end{array}
\qquad (\textsc{Handle}_{\textsc{v}})
$$

$$
\langle \texttt{perform}\ \mathit{eff}\, v;\, []\,;\, es\rangle \mapsto \mathit{abort} \qquad (\textsc{Leak})
$$

Figure 3: the semantics of $\lambda_{\mathit{eff}}$