This is the Project Title

Your Name

Master of Science
Computer Science
School of Informatics
University of Edinburgh
2020

Abstract

Formal development of Frank.

Chapter 1

Formalisation of Frank

(data types)	D
(value type variables)	X
(effect type variables)	E
(value types)	$A, B ::= D\,\overline{R}$
	$\mid\ \{C\} \mid X$
(computation types)	$C ::= \overline{T \to} G$
(argument types)	$T ::= \langle\Delta\rangle A$
(return types)	$G ::= [\Sigma]A$
(type binders)	$Z ::= X \mid [E]$
(type arguments)	$R ::= A \mid [\Sigma]$
(polytypes)	$P ::= \forall\overline{Z}.A$
(interfaces)	I
(term variables)	x, y, z, f
(instance variables)	s, a, b, c
(seeds)	$\sigma ::= \emptyset \mid E$
(abilities)	$\Sigma ::= \sigma \mid \Xi$
(extensions)	$\Xi ::= \iota \mid \Xi, I\,\overline{R}$
(adaptors)	$\Theta ::= \iota \mid \Theta, I(S \to S')$
(adjustments)	$\Delta ::= \Theta \mid \Xi$
(instance patterns)	$S ::= s \mid S\ a$
(kind environments)	$\Phi, \Psi ::= \cdot \mid \Phi, Z$
(type environments)	$\Gamma ::= \cdot \mid \Gamma, x : A \mid \Gamma, f : P$
(instance environments)	$\Omega ::= s : \Sigma \mid \Omega, a : I\,\overline{R}$

Figure 1.1: Types

(constructors)	k
(commands)	c
(uses)	$m ::= x \mid f\,\overline{R} \mid m\,\overline{n} \mid \uparrow(n : A)$
(constructions)	$n ::= \downarrow m \mid k\,\overline{n} \mid c\,\overline{R}\,\overline{n} \mid \{e\}$
	$\mid\ \textbf{let}\ f : P = n\ \textbf{in}\ n' \mid \textbf{letrec}\ \overline{f : P = e}\ \textbf{in}\ n$
	$\mid\ \langle\Theta\rangle\ n$
(computations)	$e ::= \overline{\overline{r} \mapsto n}$
(computation patterns)	$r ::= p \mid \langle c\,\overline{p} \to z\rangle \mid \langle x\rangle$
(value patterns)	$p ::= k\,\overline{p} \mid x$

Figure 1.2: Terms

$$\boxed{\Phi;\Gamma\,[\Sigma]\!\vdash m \Rightarrow A}$$

T-VAR
$$\frac{x : A \in \Gamma}{\Phi;\Gamma\,[\Sigma]\!\vdash x \Rightarrow A}$$

T-POLYVAR
$$\frac{\Phi \vdash \overline{R} \qquad f : \forall \overline{Z}.A \in \Gamma}{\Phi;\Gamma\,[\Sigma]\!\vdash f\,\overline{R} \Rightarrow A[\overline{R}/\overline{Z}]}$$

T-APP
$$\frac{\begin{array}{c}\Sigma' = \Sigma \qquad (\Sigma \vdash \Delta_i \dashv \Sigma'_i)_i \\ \Phi;\Gamma\,[\Sigma]\!\vdash m \Rightarrow \{\overline{\langle\Delta\rangle A \to}\ [\Sigma']B\} \qquad (\Phi;\Gamma\,[\Sigma'_i]\!\vdash n_i : A_i)_i\end{array}}{\Phi;\Gamma\,[\Sigma]\!\vdash m\,\overline{n} \Rightarrow B}$$

T-ASCRIBE
$$\frac{\Phi;\Gamma\,[\Sigma]\!\vdash n : A}{\Phi;\Gamma\,[\Sigma]\!\vdash \uparrow(n : A) \Rightarrow A}$$

$$\boxed{\Phi;\Gamma\,[\Sigma]\!\vdash n : A}$$

T-SWITCH
$$\frac{\Phi;\Gamma\,[\Sigma]\!\vdash m \Rightarrow A \qquad A = B}{\Phi;\Gamma\,[\Sigma]\!\vdash \downarrow m : B}$$

T-DATA
$$\frac{k\,\overline{A} \in D\,\overline{R} \qquad (\Phi;\Gamma\,[\Sigma]\!\vdash n_j : A_j)_j}{\Phi;\Gamma\,[\Sigma]\!\vdash k\,\overline{n} : D\,\overline{R}}$$

T-COMMAND
$$\frac{\Phi \vdash \overline{R} \qquad c : \forall \overline{Z}.\overline{A \to}\ B \in \Sigma \qquad (\Phi;\Gamma\,[\Sigma]\!\vdash n_j : A_j[\overline{R}/\overline{Z}])_j}{\Phi;\Gamma\,[\Sigma]\!\vdash c\,\overline{R}\,\overline{n} : B[\overline{R}/\overline{Z}]}$$

T-THUNK
$$\frac{\Phi;\Gamma \vdash e : C}{\Phi;\Gamma\,[\Sigma]\!\vdash \{e\} : \{C\}}$$

T-LET
$$\frac{\begin{array}{c}P = \forall \overline{Z}.A \\ \Phi,\overline{Z};\Gamma\,[\emptyset]\!\vdash n : A \qquad \Phi;\Gamma, f : P\,[\Sigma]\!\vdash n' : B\end{array}}{\Phi;\Gamma\,[\Sigma]\!\vdash \mathbf{let}\ f : P = n\ \mathbf{in}\ n' : B}$$

T-LETREC
$$\frac{\begin{array}{c}(P_i = \forall \overline{Z}_i.\{C_i\})_i \\ (\Phi,\overline{Z}_i;\Gamma,\overline{f : P} \vdash e_i : C)_i \qquad \Phi;\Gamma,\overline{f : P}\,[\Sigma]\!\vdash n : B\end{array}}{\Phi;\Gamma\,[\Sigma]\!\vdash \mathbf{letrec}\ \overline{f : P = e}\ \mathbf{in}\ n : B}$$

T-ADAPT
$$\frac{\Sigma \vdash \Theta \dashv \Sigma' \qquad \Phi;\Gamma\,[\Sigma']\!\vdash n : A}{\Phi;\Gamma\,[\Sigma]\!\vdash \langle\Theta\rangle\ n : A}$$

$$\boxed{\Phi;\Gamma \vdash e : C}$$

T-COMP
$$\frac{\begin{array}{c}(\Phi \vdash r_{i,j} : T_j \dashv\!\![\Sigma]\ \exists \Psi_{i,j}.\Gamma'_{i,j})_{i,j} \\ (\Phi,(\Psi_{i,j})_j;\Gamma,(\Gamma'_{i,j})_j\,[\Sigma]\!\vdash n_i : B)_i \qquad ((r_{i,j})_i \text{ covers } T_j)_j\end{array}}{\Phi;\Gamma \vdash ((r_{i,j})_j \mapsto n_i)_i : (T_j \to)_j\ [\Sigma]B}$$

Figure 1.3: Term Typing Rules

$$
\begin{array}{lrl}
\text{(uses)} & m ::= & \cdots \mid \lceil \mathcal{E}[c\ \overline{R}\ \overline{w}] \rceil \\
\text{(constructions)} & n ::= & \cdots \mid \lceil \mathcal{E}[c\ \overline{R}\ \overline{w}] \rceil \\
\text{(use values)} & u ::= & x \mid f\ \overline{R} \mid \uparrow(v : A) \\
\text{(non-use values)} & v ::= & k\ \overline{w} \mid \{e\} \\
\text{(construction values)} & w ::= & \downarrow u \mid v \\
\text{(normal forms)} & t ::= & w \mid \lceil \mathcal{E}[c\ \overline{R}\ \overline{w}] \rceil \\
\text{(evaluation frames)} & \mathcal{F} ::= & [\,]\ \overline{n} \mid u\ (\bar{t}, [\,], \overline{n}) \mid \uparrow([\,] : A) \\
 & \mid & \downarrow [\,] \mid k\ (\overline{w}, [\,], \overline{n}) \mid c\ \overline{R}\ (\overline{w}, [\,], \overline{n}) \\
 & \mid & \mathbf{let}\ f : P = [\,]\ \mathbf{in}\ n \mid \langle \Theta \rangle\ [\,] \\
\text{(evaluation contexts)} & \mathcal{E} ::= & [\,] \mid \mathcal{F}[\mathcal{E}]
\end{array}
$$

Figure 1.4: Runtime Syntax

$\boxed{\Phi; \Gamma\ [\Sigma]\!\!- m \Rightarrow A}$ $\boxed{\Phi; \Gamma\ [\Sigma]\!\!- n : A}$

$$
\frac{\neg(\mathcal{E} \text{ handles } c) \qquad \Phi; \Gamma\ [\Sigma]\!\!- \mathcal{E}[c\ \overline{R}\ \overline{w}] \Rightarrow A}{\Phi; \Gamma\ [\Sigma]\!\!- \lceil \mathcal{E}[c\ \overline{R}\ \overline{w}] \rceil \Rightarrow A}\ \textsc{T-Freeze-Use}
$$

$$
\frac{\neg(\mathcal{E} \text{ handles } c) \qquad \Phi; \Gamma\ [\Sigma]\!\!- \mathcal{E}[c\ \overline{R}\ \overline{w}] : A}{\Phi; \Gamma\ [\Sigma]\!\!- \lceil \mathcal{E}[c\ \overline{R}\ \overline{w}] \rceil : A}\ \textsc{T-Freeze-Cons}
$$

Figure 1.5: Frozen Commands

$\boxed{m \rightsquigarrow_{\mathrm{u}} m'}$ $\boxed{n \rightsquigarrow_{\mathrm{c}} n'}$ $\boxed{m \longrightarrow_{\mathrm{u}} m'}$ $\boxed{n \longrightarrow_{\mathrm{c}} n'}$

R-HANDLE
$$\frac{k = \min_i \{i \mid \exists\overline{\theta}.(r_{i,j} : \langle\Delta_j\rangle A_j \leftarrow t_j \dashv[\Sigma]\ \theta_j)_j\} \qquad (r_{k,j} : \langle\Delta_j\rangle A_j \leftarrow t_j \dashv[\Sigma]\ \theta_j)_j}{\uparrow(\{((r_{i,j})_j \to n_i)_i\} : \{\overline{\langle\Delta\rangle A \to}\ [\Sigma]B\})\ \overline{t} \rightsquigarrow_{\mathrm{u}} \uparrow((\overline{\theta}(n_k) : B)}$$

R-ASCRIBE-USE
$$\frac{}{\uparrow(\downarrow u : A) \rightsquigarrow_{\mathrm{u}} u}$$

R-ASCRIBE-CONS
$$\frac{}{\downarrow\uparrow(w : A) \rightsquigarrow_{\mathrm{c}} w}$$

R-LET
$$\frac{}{\textbf{let } f : P = w \textbf{ in } n \rightsquigarrow_{\mathrm{c}} n[\uparrow(w : P)/f]}$$

R-LETREC
$$\frac{\overline{e = \overline{\overline{r} \to n}}}{\textbf{letrec } \overline{f : P = e} \textbf{ in } n' \rightsquigarrow_{\mathrm{c}} n'[\overline{\uparrow(\{\overline{\overline{r} \to \textbf{letrec } \overline{f : P = e} \textbf{ in } n}\} : P)/f}]}$$

R-ADAPT
$$\frac{}{\langle\Theta\rangle\ w \rightsquigarrow_{\mathrm{c}} w}$$

R-FREEZE-COMM
$$\frac{}{c\ \overline{R}\ \overline{w} \rightsquigarrow_{\mathrm{c}} \lceil c\ \overline{R}\ \overline{w} \rceil}$$

R-FREEZE-FRAME-USE
$$\frac{\neg(\mathcal{F}[\mathcal{E}] \text{ handles } c)}{\mathcal{F}[\lceil\mathcal{E}[c\ \overline{R}\ \overline{w}]\rceil] \rightsquigarrow_{\mathrm{u}} \lceil\mathcal{F}[\mathcal{E}[c\ \overline{R}\ \overline{w}]]\rceil}$$

R-FREEZE-FRAME-CONS
$$\frac{\neg(\mathcal{F}[\mathcal{E}] \text{ handles } c)}{\mathcal{F}[\lceil\mathcal{E}[c\ \overline{R}\ \overline{w}]\rceil] \rightsquigarrow_{\mathrm{c}} \lceil\mathcal{F}[\mathcal{E}[c\ \overline{R}\ \overline{w}]]\rceil}$$

R-LIFT-UU
$$\frac{m \rightsquigarrow_{\mathrm{u}} m'}{\mathcal{E}[m] \longrightarrow_{\mathrm{u}} \mathcal{E}[m']}$$

R-LIFT-UC
$$\frac{m \rightsquigarrow_{\mathrm{u}} m'}{\mathcal{E}[m] \longrightarrow_{\mathrm{c}} \mathcal{E}[m']}$$

R-LIFT-CU
$$\frac{n \rightsquigarrow_{\mathrm{c}} n'}{\mathcal{E}[n] \longrightarrow_{\mathrm{u}} \mathcal{E}[n']}$$

R-LIFT-CC
$$\frac{n \rightsquigarrow_{\mathrm{c}} n'}{\mathcal{E}[n] \longrightarrow_{\mathrm{c}} \mathcal{E}[n']}$$

Figure 1.6: Operational Semantics

$\boxed{r : T \leftarrow t \dashv\!_{[\Sigma]}\ \theta}$

$$\textsc{B-Value}\quad \frac{\Sigma \vdash \Delta \dashv \Sigma' \qquad p : A \leftarrow w \dashv \theta}{p : \langle \Delta \rangle A \leftarrow w \dashv\!_{[\Sigma]}\ \theta}$$

$$\textsc{B-Request}\quad \frac{\begin{array}{c}\Sigma \vdash \Delta \dashv \Sigma' \qquad \mathcal{E}\ \mathsf{poisedfor}\ c \\ \Delta = \Theta \mid \Xi \qquad c : \forall \overline{Z}.\overline{B \rightarrow} B' \in \Xi \qquad (p_i : B_i \leftarrow w_i \dashv \theta_i)_i\end{array}}{\langle c\ \overline{p} \rightarrow z \rangle : \langle \Delta \rangle A \leftarrow \lceil \mathcal{E}[c\ \overline{R}\ \overline{w}] \rceil \dashv\!_{[\Sigma]}\ \overline{\theta}[\uparrow(\{x \mapsto \mathcal{E}[x]\} : \{B' \rightarrow [\Sigma']A\})/z]}$$

$$\textsc{B-CatchAll-Value}\quad \frac{\Sigma \vdash \Delta \dashv \Sigma'}{\langle x \rangle : \langle \Delta \rangle A \leftarrow w \dashv\!_{[\Sigma]}\ [\uparrow(\{w\} : \{[\Sigma']A\})/x]}$$

$$\textsc{B-CatchAll-Request}\quad \frac{\begin{array}{c}\Sigma \vdash \Delta \dashv \Sigma' \qquad \mathcal{E}\ \mathsf{poisedfor}\ c \\ \Delta = \Theta \mid \Xi \qquad c : \forall \overline{Z}.\overline{B \rightarrow} B' \in \Xi\end{array}}{\langle x \rangle : \langle \Delta \rangle A \leftarrow \lceil \mathcal{E}[c\ \overline{R}\ \overline{w}] \rceil \dashv\!_{[\Sigma]}\ [\uparrow(\{\lceil \mathcal{E}[c\ \overline{R}\ \overline{w}] \rceil\} : \{[\Sigma']A\})/x]}$$

$\boxed{p : A \leftarrow w \dashv \theta}$

$$\textsc{B-Var}\quad \frac{}{x : A \leftarrow w \dashv [\uparrow(w : A)/x]} \qquad \textsc{B-Data}\quad \frac{k\ \overline{A} \in D\ \overline{R} \qquad (p_i : A_i \leftarrow w_i \dashv \theta_i)_i}{k\ \overline{p} : D\ \overline{R} \leftarrow k\ \overline{w} \dashv \overline{\theta}}$$

Figure 1.7: Pattern Binding

Chapter 2

Arbitrary Thread Interruption

2.1 Relaxing Catches

$$\frac{\Sigma \vdash \Delta \dashv \Sigma'}{\langle x \rangle : \langle \Delta \rangle A \leftarrow \lceil \mathcal{E}[c\ \overline{R}\ \overline{w}] \rceil \dashv_{[\Sigma]} [\uparrow(\{\lceil \mathcal{E}[c\ \overline{R}\ \overline{w}] \rceil\}{:}\{[\Sigma'] A\})/x]} \ \textsc{B-CatchAll-Request-Loose}$$

Figure 2.1: Updated B-CATCHALL-REQUEST

2.2 Interrupting Arbitrary Terms

$$
\begin{array}{lrl}
\text{(uses)} & m ::= & \cdots \mid \lceil \mathcal{E}[c\,\overline{R}\,\overline{w}] \rceil \\
\text{(constructions)} & n ::= & \cdots \mid \lceil \mathcal{E}[c\,\overline{R}\,\overline{w}] \rceil \\
\text{(use values)} & u ::= & x \mid f\,\overline{R} \mid \uparrow(v : A) \\
\text{(non-use values)} & v ::= & k\,\overline{w} \mid \{e\} \\
\text{(construction values)} & w ::= & \downarrow u \mid v \\
\text{(normal forms)} & t ::= & w \mid \lceil \mathcal{E}[c\,\overline{R}\,\overline{w}] \rceil \mid !(m) \\
\text{(evaluation frames)} & \mathcal{F} ::= & [\,]\,\overline{n} \mid u\;(\overline{t},[\,],\overline{n}) \mid \uparrow([\,] : A) \\
 & \mid & \downarrow[\,] \mid k\;(\overline{w},[\,],\overline{n}) \mid c\,\overline{R}\;(\overline{w},[\,],\overline{n}) \\
 & \mid & \mathbf{let}\ f : P = [\,]\ \mathbf{in}\ n \mid \langle \Theta \rangle\ [\,] \\
\text{(evaluation contexts)} & \mathcal{E} ::= & [\,] \mid \mathcal{F}[\mathcal{E}]
\end{array}
$$

Figure 2.2: Runtime Syntax, Updated with Suspension of Uses

$$
\textsc{B-CatchAll-Interrupt}\quad
\frac{\Sigma \vdash \Delta \dashv \Sigma'}{\langle x \rangle : \langle \Delta \rangle A \leftarrow !(m) \dashv_{\Sigma} [\uparrow(\{m\}{:}\{[\Sigma']A\})/x]}
$$

Figure 2.3: Catching Interrupts rule.

2.3 Freezing

Another way of doing it is to just let any use become frozen, in the same way as commands become frozen once invoked.

TODO: Do we need the 'hoisting' rules for general suspended terms?

2.4 Yielding

Note that R-YIELD-EF relies on the predicate $\mathcal{F}[\mathcal{E}]$ allows c. This states that the Yield interface is in the ability of the term in the evaluation context.

For any frame apart from argument frames, $\mathcal{F}[\mathcal{E}]$ allows $c = \text{false}$. In this case, it is defined as follows;

$$
\uparrow(v : \{\overline{\langle \Delta \rangle A \to} [\Sigma]B\})\ (\overline{t},[\,],\overline{n})\ \text{allows}\ c = \begin{cases} \Xi\ \text{allows}\ c & \text{if } |\overline{n}| = 0 \ \text{ where } \Sigma = \sigma \mid \Xi \\ \text{false} & \text{otherwise} \end{cases}
$$

TODO: What happens for 0-ary constructors? I don't know if it's exactly how I think it is.

$$\frac{\textsc{R-Interrupt}}{m \rightsquigarrow_{\mathrm{u}} !(m)}$$

Figure 2.4: Use interruption rule

$\boxed{m \rightsquigarrow_{\mathrm{u}} m'}$ $\boxed{n \rightsquigarrow_{\mathrm{c}} n'}$ $\boxed{m \longrightarrow_{\mathrm{u}} m'}$ $\boxed{n \longrightarrow_{\mathrm{c}} n'}$

$$\textsc{R-Freeze-Use}\quad \frac{}{m \rightsquigarrow_{\mathrm{u}} \lceil m \rceil} \qquad \textsc{R-Freeze-Comm}\quad \frac{}{c\,\overline{R}\,\overline{w} \rightsquigarrow_{\mathrm{c}} \lceil c\,\overline{R}\,\overline{w} \rceil}$$

$$\textsc{R-Freeze-Frame-Use}\quad \frac{}{\mathcal{F}[\lceil m \rceil] \rightsquigarrow_{\mathrm{u}} \lceil \mathcal{F}[m] \rceil} \qquad \textsc{R-Freeze-Frame-Cons}\quad \frac{}{\mathcal{F}[\lceil m \rceil] \rightsquigarrow_{\mathrm{c}} \lceil \mathcal{F}[m] \rceil}$$

$$\textsc{R-Freeze-Frame-Use}\quad \frac{\neg(\mathcal{F}[\mathcal{E}] \text{ handles } c)}{\mathcal{F}[\lceil \mathcal{E}[c\,\overline{R}\,\overline{w}] \rceil] \rightsquigarrow_{\mathrm{u}} \lceil \mathcal{F}[\mathcal{E}[c\,\overline{R}\,\overline{w}]] \rceil} \qquad \textsc{R-Freeze-Frame-Cons}\quad \frac{\neg(\mathcal{F}[\mathcal{E}] \text{ handles } c)}{\mathcal{F}[\lceil \mathcal{E}[c\,\overline{R}\,\overline{w}] \rceil] \rightsquigarrow_{\mathrm{c}} \lceil \mathcal{F}[\mathcal{E}[c\,\overline{R}\,\overline{w}]] \rceil}$$

Figure 2.5: Updated Freezing

For an extension Ξ, the allows predicate is defined as

$$\begin{aligned} \iota \text{ allows } c &= \text{false} \\ (\Xi, I\,\overline{R}) \text{ allows } c &= \begin{cases} \text{true} & \text{if } c \in I \\ \Xi \text{ allows } c & \text{otherwise} \end{cases} \end{aligned}$$

Informally, $\mathcal{F}[\mathcal{E}]$ allows c is true when $\mathcal{F}[\mathcal{E}]$ is a handler with the command c as a member of an interface in its ability, *and* all of the arguments have been evaluated (this is what the $|\overline{n}| = 0$ constraint expresses).

We also make use of an auxilary combinator $\_;\_$. This is the traditional sequential composition operator, where both arguments are evaluated and hte result of the second one is returned. In the context of R-YIELD-EF this means we will perform the yield effect and then the use m, but discard the result from yield.

$$
\begin{array}{lrl}
\text{(uses)} & m ::= & \ldots \mid \lceil \mathcal{E}[c\,\overline{R}\,\overline{w}] \rceil \mid \boxed{\lceil m \rceil} \\
\text{(constructions)} & n ::= & \ldots \mid \lceil \mathcal{E}[c\,\overline{R}\,\overline{w}] \rceil \mid \boxed{\lceil m \rceil} \\
\text{(use values)} & u ::= & x \mid f\,\overline{R} \mid \uparrow(v : A) \\
\text{(non-use values)} & v ::= & k\,\overline{w} \mid \{e\} \\
\text{(construction values)} & w ::= & \downarrow u \mid v \\
\text{(normal forms)} & t ::= & w \mid \lceil \mathcal{E}[c\,\overline{R}\,\overline{w}] \rceil \mid \boxed{\lceil m \rceil} \\
\text{(evaluation frames)} & \mathcal{F} ::= & [\,]\,\overline{n} \mid u\,(\overline{t},[\,],\overline{n}) \mid \uparrow([\,] : A) \\
 & \mid & \downarrow[\,] \mid k\,(\overline{w},[\,],\overline{n}) \mid c\,\overline{R}\,(\overline{w},[\,],\overline{n}) \\
 & \mid & \mathbf{let}\ f : P = [\,]\ \mathbf{in}\ n \mid \langle \Theta \rangle\,[\,] \\
\text{(evaluation contexts)} & \mathcal{E} ::= & [\,] \mid \mathcal{F}[\mathcal{E}]
\end{array}
$$

Figure 2.6: Runtime Syntax, Updated with Freezing of Uses

B-CATCHALL-INTERRUPT

$$
\frac{\Sigma \vdash \Delta \dashv \Sigma'}{\langle x \rangle : \langle \Delta \rangle A \leftarrow \lceil m \rceil \dashv_{\Sigma} [\uparrow(\{m\} : \{[\Sigma']A\})/x]}
$$

Figure 2.7: Catching Frozen Terms rule.

2.5 Counting

In practise we count up through the amount of R-HANDLE rules we apply and only insert the yield when this count exceeds a threshold value t_y.

So we supplement the operational semantics with a *counter* c_y, so that our transitions are e.g. $m; c_y \rightsquigarrow_{\mathrm{u}} m'; c_y'$. We adopt the convention that, when the counter is not mentioned in a transition[1], the counter stays the same. Hence e.g. $m \rightsquigarrow_{\mathrm{u}} m'$ desugars to $m; c_y \rightsquigarrow_{\mathrm{u}} m'; c_y$.

So to get our counting semantics we just need to supplement Figure **?** with the updated rule in

[1]That is, the transition is of the form $m \rightsquigarrow_{\mathrm{u}} m'$.

$\boxed{m \rightsquigarrow_{\mathrm{u}} m'}$

R-YIELD

$$\frac{\Sigma = \sigma \mid \Xi \qquad \mathsf{Yield} \in \Xi}{\begin{array}{l}\uparrow(\{n\} : \{[\Sigma]A\}) \rightsquigarrow_{\mathrm{u}} \uparrow(\ \ \mathbf{let}\ f_n : \{\mathsf{Unit} \to [\Sigma]A\} = \{\_ \mapsto n\}\ \mathbf{in} \\ \qquad\qquad\qquad\qquad\quad \mathbf{let}\ y : \{[\Sigma]\mathsf{Unit}\} = \{\mathsf{yield!}\}\ \mathbf{in}\ \{f_n\ (\downarrow(y!))\} : \{[\Sigma]A\})\end{array}}$$

R-YIELD-EF

$$\frac{\mathcal{F}[\mathcal{E}]\ \text{allows Yield}}{\mathcal{F}[\mathcal{E}[m]] \rightsquigarrow_{\mathrm{u}} \mathcal{F}[\mathcal{E}[\mathsf{yield!}; m]]}$$

Figure 2.8: Inserting Yields

TODO: In R-Yield-EF we have to let terms inside eval CTXs be also uses. Does this not mean that we can then put any term in either one? Is that a problem?

$\boxed{m \rightsquigarrow_{\mathrm{u}} m'}$ $\boxed{n \rightsquigarrow_{\mathrm{c}} n'}$ $\boxed{m \longrightarrow_{\mathrm{u}} m'}$ $\boxed{n \longrightarrow_{\mathrm{c}} n'}$

R-HANDLE

$$\frac{k = \min_i \{i \mid \exists\overline{\theta}.(r_{i,j} : \langle\Delta_j\rangle A_j \leftarrow t_j \dashv[\Sigma]\ \theta_j)_j\} \qquad (r_{k,j} : \langle\Delta_j\rangle A_j \leftarrow t_j \dashv[\Sigma]\ \theta_j)_j}{\uparrow(\{((r_{i,j})_j \to n_i)_i\} : \{\overline{\langle\Delta\rangle A \to}\ [\Sigma]B\})\ \overline{t}; c_y \rightsquigarrow_{\mathrm{u}} \uparrow((\overline{\theta}(n_k) : B)); c_y + 1}$$

Figure 2.9: R-Handle with counting.

$\boxed{m \rightsquigarrow_{\mathrm{u}} m'}$

R-YIELD

$$\frac{c_y \geq t_y}{m; c_y \rightsquigarrow_{\mathrm{u}} \mathbf{let}\ \mathsf{yield} : [\mathsf{Yield}]\mathsf{Unit} = y\mathbf{in}\ m; 0}$$

Figure 2.10: Inserting Yields, when over counter

Chapter 3

Use-Only Reductions

Here we talk about how to change the operational semantics so that only uses may reduce.

$$
\begin{array}{lrl}
\text{(uses)} & m & ::= \cdots \mid \lceil \mathcal{E}[c\ \overline{R}\ \overline{w}] \rceil \\
\text{(constructions)} & n & ::= \ldots \\
\text{(use values)} & u & ::= x \mid f\ \overline{R} \mid \uparrow(v : A) \\
\text{(non-use values)} & v & ::= k\ \overline{w} \mid \{e\} \\
\text{(construction values)} & w & ::= \downarrow u \mid v \\
\text{(normal forms)} & t & ::= w \mid \lceil \mathcal{E}[c\ \overline{R}\ \overline{w}] \rceil \\
\text{(evaluation frames)} & \mathcal{F} & ::= [\,]\ \overline{n} \mid u\ (\overline{t}, [\,], \overline{n}) \mid \uparrow([\,] : A) \\
 & & \mid\ \ \downarrow[\,] \mid k\ (\overline{w}, [\,], \overline{n}) \mid c\ \overline{R}\ (\overline{w}, [\,], \overline{n}) \\
 & & \mid\ \ \mathbf{let}\ f : P = [\,]\ \mathbf{in}\ n \mid \langle \Theta \rangle\ [\,] \\
\text{(evaluation contexts)} & \mathcal{E} & ::= [\,] \mid \mathcal{F}[\mathcal{E}]
\end{array}
$$

Figure 3.1: Runtime Syntax for Use-Only Reductions

TODO: Remove command uses from constructiosn?

TODO: Change eval ctxs (i can't remember what to change to)?

$\boxed{m \rightsquigarrow_{\mathrm{u}} m'}$ $\boxed{m \longrightarrow_{\mathrm{u}} m'}$

$$\frac{k = \min_i \{ i \mid \exists \overline{\theta}.(r_{i,j} : \langle \Delta_j \rangle A_j \leftarrow t_j \dashv [\Sigma]\ \theta_j)_j \} \qquad (r_{k,j} : \langle \Delta_j \rangle A_j \leftarrow t_j \dashv [\Sigma]\ \theta_j)_j}{\uparrow(\{((r_{i,j})_j \to n_i)_i\} : \{\overline{\langle\Delta\rangle A \to} [\Sigma] B\})\ \overline{t} \rightsquigarrow_{\mathrm{u}} \uparrow((\overline{\theta}(n_k) : B)} \ \text{R-HANDLE}$$

$$\frac{}{\uparrow(\downarrow u : A) \rightsquigarrow_{\mathrm{u}} u} \ \text{R-ASCRIBE-USE}$$

$$\frac{}{\uparrow(\textbf{let } f : P = w \textbf{ in } n : A) \rightsquigarrow_{\mathrm{u}} \uparrow(n[\uparrow(w : P)/f] : A)} \ \text{R-LET}$$

$$\frac{\overline{e = \overline{r} \to n}}{\uparrow(\textbf{letrec } \overline{f : P = e} \textbf{ in } n' : A) \rightsquigarrow_{\mathrm{u}} \uparrow(n'[\overline{\uparrow(\{\overline{r} \to \textbf{letrec } \overline{f : P = e} \textbf{ in } n\} : P)/f}] : A)} \ \text{R-LETREC}$$

$$\frac{}{\uparrow(\langle\Theta\rangle\ w : A) \rightsquigarrow_{\mathrm{u}} \uparrow(w : A)} \ \text{R-ADAPT}$$

$$\frac{}{c\ \overline{R}\ \overline{w} \rightsquigarrow_{\mathrm{u}} \lceil c\ \overline{R}\ \overline{w} \rceil} \ \text{R-FREEZE-COMM}$$

$$\frac{\neg(\mathcal{F}[\mathcal{E}] \text{ handles } c)}{\mathcal{F}[\lceil \mathcal{E}[c\ \overline{R}\ \overline{w}] \rceil] \rightsquigarrow_{\mathrm{u}} \lceil \mathcal{F}[\mathcal{E}[c\ \overline{R}\ \overline{w}]] \rceil} \ \text{R-FREEZE-FRAME-USE}$$

$$\frac{m \rightsquigarrow_{\mathrm{u}} m'}{\mathcal{E}[m] \longrightarrow_{\mathrm{u}} \mathcal{E}[m']} \ \text{R-LIFT-UU}$$

Figure 3.2: Operational Semantics

TODO: Can R-Ascribe-Cons just get removed?

TODO: Can we just move commands to be in uses?

Appendix A

Remaining Formalisms

$\boxed{\Sigma \vdash \Delta \dashv \Sigma'}$

$$\textsc{A-Adj}\quad \frac{\Sigma \vdash \Theta \dashv \Sigma' \qquad \Sigma' \vdash \Xi \dashv \Sigma''}{\Sigma \vdash \Theta \mid \Xi \dashv \Sigma''}$$

$\boxed{\Sigma \vdash \Xi \dashv \Sigma'}$

$$\textsc{A-Ext-Id}\quad \frac{}{\Sigma \vdash \iota \dashv \Sigma} \qquad \textsc{A-Ext-Snoc}\quad \frac{\Sigma \vdash \Xi \dashv \Sigma'}{\Sigma \vdash \Xi, I\,\overline{R} \dashv \Sigma', I\,\overline{R}}$$

$\boxed{\Sigma \vdash \Theta \dashv \Sigma'}$

$$\textsc{A-Adapt-Id}\quad \frac{}{\Sigma \vdash \iota \dashv \Sigma} \qquad \textsc{A-Adapt-Snoc}\quad \frac{\Sigma \vdash \Theta \dashv \Sigma' \qquad \Sigma' \vdash I(S \to S') \dashv \Sigma''}{\Sigma \vdash \Theta, I(S \to S') \dashv \Sigma''}$$

$\boxed{\Sigma \vdash I(S \to S') \dashv \Sigma'}$

$$\textsc{A-Adapt-Com}\quad \frac{\Sigma \vdash S : I \dashv \Sigma'; \Omega \qquad \Omega \vdash S' : I \dashv \Xi \qquad \Sigma' \vdash \Xi \dashv \Sigma''}{\Sigma \vdash I(S \to S') \dashv \Sigma''}$$

$\boxed{\Sigma \vdash S : I \dashv \Sigma'; \Omega}$

$$\textsc{I-Pat-Id}\quad \frac{}{\Sigma \vdash s : I \dashv \Sigma; s : \Sigma} \qquad \textsc{I-Pat-Bind}\quad \frac{\Sigma \vdash S : I \dashv \Sigma'; \Omega}{\Sigma, I\,\overline{R} \vdash S\,a : I \dashv \Sigma'; \Omega, a : I\,\overline{R}}$$

$$\textsc{I-Pat-Skip}\quad \frac{\Sigma \vdash S\,a : I \dashv \Sigma'; \Omega \qquad I \neq I'}{\Sigma, I'\,\overline{R} \vdash S\,a : I \dashv \Sigma', I'\,\overline{R}; \Omega}$$

$\boxed{\Omega \vdash S : I \dashv \Xi}$

$$\textsc{I-Inst-Id}\quad \frac{s \in \mathsf{dom}(\Omega)}{\Omega \vdash s : I \dashv \iota} \qquad \textsc{I-Inst-Lkp}\quad \frac{a \in \mathsf{dom}(\Omega) \qquad \Omega \vdash S : I \dashv \Xi \qquad \Omega(a) = I\,\overline{R}}{\Omega \vdash S\,a : I \dashv \Xi, I\,\overline{R}}$$

Figure A.1: Action of an Adjustment on an Ability and Auxiliary Judgements

$$\mathcal{X} ::= A \mid C \mid T \mid G \mid Z \mid R \mid P \mid \sigma \mid \Sigma \mid \Xi \mid \Theta \mid \Delta \mid \Gamma \mid \exists\Psi.\Gamma \mid \Omega$$

$\boxed{\Phi \vdash \mathcal{X}}$

$$\frac{}{\Phi, X \vdash X}\ \textsc{WF-Val} \qquad \frac{}{\Phi, [E] \vdash E}\ \textsc{WF-Eff} \qquad \frac{\Phi, \overline{Z} \vdash A}{\Phi \vdash \forall \overline{Z}.A}\ \textsc{WF-Poly}$$

$$\frac{(\Phi \vdash R)_i}{\Phi \vdash D\,\overline{R}}\ \textsc{WF-Data} \qquad \frac{\Phi \vdash C}{\Phi \vdash \{C\}}\ \textsc{WF-Thunk} \qquad \frac{(\Phi \vdash T)_i \qquad \Phi \vdash G}{\Phi \vdash \overline{T \to} G}\ \textsc{WF-Comp} \qquad \frac{\Phi \vdash \Delta \qquad \Phi \vdash A}{\Phi \vdash \langle\Delta\rangle A}\ \textsc{WF-Arg}$$

$$\frac{\Phi \vdash \Sigma \qquad \Phi \vdash A}{\Phi \vdash [\Sigma]A}\ \textsc{WF-Ret} \qquad \frac{\Phi \vdash \Sigma}{\Phi \vdash [\Sigma]}\ \textsc{WF-Ability} \qquad \frac{}{\Phi \vdash \emptyset}\ \textsc{WF-Pure} \qquad \frac{}{\Phi \vdash \iota}\ \textsc{WF-Id} \qquad \frac{\Phi \vdash \Xi \qquad (\Phi \vdash R)_i}{\Phi \vdash \Xi, I\,\overline{R}}\ \textsc{WF-Ext}$$

$$\frac{\Phi \vdash \Theta}{\Phi \vdash \Theta, I\ (S \to S')}\ \textsc{WF-Adapt}$$

$$\frac{}{\Phi \vdash \cdot}\ \textsc{WF-Empty} \qquad \frac{\Phi \vdash \Gamma \qquad \Phi \vdash A}{\Phi \vdash \Gamma, x : A}\ \textsc{WF-Mono} \qquad \frac{\Phi \vdash \Gamma \qquad \Phi \vdash P}{\Phi \vdash \Gamma, f : P}\ \textsc{WF-Poly}$$

$$\frac{\Phi, \Psi \vdash \Gamma}{\Phi \vdash \exists\Psi.\Gamma}\ \textsc{WF-Existential} \qquad \frac{\Phi \vdash \Omega \qquad (\Phi \vdash R)_i}{\Phi \vdash \Omega, x : I\,\overline{R}}\ \textsc{WF-Interface}$$

Figure A.2: Well-Formedness Rules

$$\boxed{\Phi \vdash p : A \dashv \Gamma}$$

$$\textsc{P-Var}\quad \frac{}{\Phi \vdash x : A \dashv x : A} \qquad\qquad \textsc{P-Data}\quad \frac{k\,\overline{A} \in D\,\overline{R} \qquad (\Phi \vdash p_i : A_i \dashv \Gamma)_i}{\Phi \vdash k\,\overline{p} : D\,\overline{R} \dashv \overline{\Gamma}}$$

$$\boxed{\Phi \vdash r : T \dashv_{[\Sigma]} \exists\Psi.\Gamma}$$

$$\textsc{P-Value}\quad \frac{\Sigma \vdash \Delta \dashv \Sigma' \qquad \Phi \vdash p : A \dashv \Gamma}{\Phi \vdash p : \langle\Delta\rangle A \dashv_{[\Sigma]} \Gamma} \qquad\qquad \textsc{P-CatchAll}\quad \frac{\Sigma \vdash \Delta \dashv \Sigma'}{\Phi \vdash \langle x\rangle : \langle\Delta\rangle A \dashv_{[\Sigma]} x : \{[\Sigma']A\}}$$

$$\textsc{P-Command}\quad \frac{\Sigma \vdash \Delta \dashv \Sigma' \qquad \Delta = \Theta \mid \Xi \qquad c : \forall\overline{Z}.\overline{A \to} B \in \Xi \qquad (\Phi, \overline{Z} \vdash p_i : A_i \dashv \Gamma_i)_i}{\Phi \vdash \langle c\;\overline{p} \to z\rangle : \langle\Delta\rangle B' \dashv_{[\Sigma]} \exists\overline{Z}.\overline{\Gamma}, z : \{\langle\iota \mid \iota\rangle B \to [\Sigma']B'\}}$$

Figure A.3: Pattern Matching Typing Rules