Jonathan Chan

May 7, 2022

$$
\begin{aligned}
i, j, m, n &::= \textsf{naturals} & s &::= \alpha \mid \hat{s} \mid \omega & \Gamma &::= \cdot \mid \Gamma(x : \sigma) \mid \Gamma(x \coloneqq e) \mid \Gamma(\alpha) \\
f, g, w, x, y, z, X, c, \alpha, \beta &::= \textsf{variables} & \sigma &::= \forall \alpha.\, \sigma \mid \tau & \Delta &::= \cdot \mid \Delta(x : \sigma) \\
e, a, p, P, \tau &::= x \mid U_i \mid \Pi x : \tau.\, \tau \mid \lambda x : \tau.\, e \mid \Lambda \alpha.\, e \mid e\, e \mid e\, [s] \mid \textsf{let}\ x : \sigma \coloneqq e\ \textsf{in}\ e \\
&\mid \textsf{match}\ e\ \textsf{return}\ (y \ldots).x.P\ \textsf{with}\ (c\, x \ldots \Rightarrow e) \ldots \mid \textsf{fix}\ f : \sigma \coloneqq e \\
U_i &::= \textsf{Prop}_i \mid \textsf{Type}_i
\end{aligned}
$$

Figure 1: Syntax

$$
\begin{aligned}
\hat{\omega} &\equiv \omega & \forall(\alpha \ldots).\, \tau &\equiv \forall \alpha.\, \ldots \tau & \Delta \to \tau' &\equiv \Pi x : \tau.\, \ldots \tau' \\
s + 0 &\equiv s & (x : \tau_1) \to \tau_2 &\equiv \Pi x : \tau_1.\, \tau_2 & \lambda \Delta.\, e &\equiv \lambda x : \tau.\, \ldots e \\
s + n &\equiv \hat{s} + (n - 1) & \tau_1 \to \tau_2 &\equiv \Pi \_ : \tau_1.\, \tau_2 & \textit{when}\ \Delta &\equiv (x : \tau) \ldots
\end{aligned}
$$

$$(c, X) \in \Sigma \equiv (\textsf{data}\ X\ \_ : \_\ \textsf{where}\ \Delta_c) \in \Sigma\ \textit{and}\ (c : \_) \in \Delta_c$$

Figure 2: Syntactic sugar

$$
\begin{aligned}
\Sigma &::= \cdot \mid \Sigma(d) \\
d &::= \textsf{data}\ X\ \Delta_P : \Delta_I \to U_i\ \textsf{where}\ \Delta_c \\
\textit{where}\ X &\notin \textsf{FV}(\Delta_P \Delta_I) \\
\tau &\equiv \Delta \to X\, [\alpha]\, w \ldots\ a' \ldots\ \textit{and}\ X \notin \textsf{FV}(\Delta) \qquad \textit{or} \qquad X \notin \textsf{FV}(\tau) \\
\textit{when}\ \Delta_P &\equiv (w : \_) \ldots \\
\Delta_c &\equiv (c : \forall \alpha.\, \Delta_a \to X\, [\hat{\alpha}]\, w \ldots\ a \ldots) \ldots \\
\Delta_a &\equiv (x : \tau) \ldots
\end{aligned}
$$

Figure 3: Inductive definitions

$$
\begin{array}{l}
\mathsf{data}\ \mathsf{Nat} : \mathsf{Type}\ \mathsf{where} \\
\quad \mathsf{zero} : \forall\alpha.\, \mathsf{Nat}\,[\hat{\alpha}] \\
\quad \mathsf{succ} : \forall\alpha.\, \mathsf{Nat}\,[\alpha] \to \mathsf{Nat}\,[\hat{\alpha}]
\end{array}
$$

$$
\begin{array}{l}
\mathsf{data}\ \mathsf{Eq}\ (A : \mathsf{Type})(a : A) : A \to \mathsf{Type}\ \mathsf{where} \\
\quad \mathsf{refl} : \forall\alpha.\, \mathsf{Eq}\,[\hat{\alpha}]\ A\ a\ a \\
\textit{with}\ a =_A^s a \equiv \mathsf{Eq}\,[s]\ A\ a\ a
\end{array}
$$

$$
\begin{array}{l}
\mathsf{data}\ \mathsf{Pair}\ (A : \mathsf{Type})(B : A \to \mathsf{Type}) : \mathsf{Type}\ \mathsf{where} \\
\quad \mathsf{pair} : \forall\alpha.\, (a : A) \to B\ a \to \mathsf{Pair}\,[\hat{\alpha}]\ A\ B \\
\textit{with}\ (x : A \times B)^s \equiv \mathsf{Pair}\,[s]\ A\ (\lambda x : A.\, B) \\
\textit{and}\ (a, b)^s_{(x:A\times B)} \equiv \mathsf{pair}\,[s]\ A\ (\lambda x : A.\, B)\ a\ b
\end{array}
$$

$$
\begin{array}{l}
\mathsf{data}\ \mathsf{W}\ (A : \mathsf{Type})(B : A \to \mathsf{Type}) : \mathsf{Type}\ \mathsf{where} \\
\quad \mathsf{sup} : \forall\alpha.\, (a : A) \to (B\ a \to \mathsf{W}\,[\alpha]\ A\ B) \to \mathsf{W}\,[\hat{\alpha}]\ A\ B \\
\textit{with}\ \mathsf{W}(x : A).\, B\,[s] \equiv \mathsf{W}\,[s]\ A\ (\lambda x : A.\, B) \\
\textit{and}\ \mathsf{sup}_{\mathsf{W}(x:A).B}\,[s] \equiv \mathsf{sup}\,[s]\ A\ (\lambda x : A.\, B)
\end{array}
$$

Figure 4: Common inductive definitions

$\boxed{\Gamma \vdash e_1 \rhd^* e_2}$ $\boxed{\Gamma \vdash e_1 \approx e_2}$ $\boxed{X \mid U_i \succ U_j}$

Figure 5: Implicit judgements

$\boxed{\vdash \Sigma}$

$$\frac{}{\vdash \cdot}\ \Sigma\text{-NIL} \qquad \frac{\vdash \Sigma \qquad \cdot \vdash \Delta_P \Delta_I \to U_i \Uparrow \mathsf{Type}_j \qquad \{\Delta_P \vdash \sigma \Downarrow U_i\} \ldots}{\vdash \Sigma(\mathsf{data}\ X\ \Delta_P : \Delta_I \to U_i\ \mathsf{where}\ (c : \sigma) \ldots)}\ \Sigma\text{-CONS}$$

$\boxed{\vdash \Gamma}$ *with implicit* Σ

$$\frac{\vdash \Sigma}{\vdash \cdot}\ \Gamma\text{-NIL} \qquad \frac{\vdash \Gamma}{\vdash \Gamma(\alpha)}\ \Gamma\text{-CONS-SIZE} \qquad \frac{\vdash \Gamma \qquad \Gamma \vdash \tau \Uparrow U_i}{\vdash \Gamma(x : \tau)}\ \Gamma\text{-CONS-ASS} \qquad \frac{\vdash \Gamma \qquad \Gamma \vdash e \Uparrow \tau}{\vdash \Gamma(x \coloneqq e)}\ \Gamma\text{-CONS-DEF}$$

$\boxed{\vdash s}$ *with implicit* Γ

$$\frac{(\alpha) \in \Gamma}{\vdash \alpha}\ \text{SVAR} \qquad \frac{\vdash s}{\vdash \hat{s}}\ \text{SSUCC} \qquad \frac{}{\vdash \omega}\ \text{SINF}$$

Figure 6: Well-formedness rules

$\boxed{\Gamma \vdash e \rhd e}$ *with implicit* Σ

$$\frac{(x \coloneqq e) \in \Gamma}{\Gamma \vdash x \rhd_\delta e} \qquad \frac{}{\Gamma \vdash (\lambda x : \tau.\, e)\, e' \rhd_\beta e[x \mapsto e']} \qquad \frac{}{\Gamma \vdash (\Lambda \alpha.\, e)\, [s] \rhd_\beta e[\alpha \mapsto s]}$$

$$\frac{}{\Gamma \vdash \mathsf{let}\ x : \sigma \coloneqq e'\ \mathsf{in}\ e \rhd_\zeta e[x \mapsto e']}$$

$$\frac{(c'\, z' \ldots \Rightarrow e') \in (c\, z \ldots \Rightarrow e) \ldots}{\Gamma \vdash \mathsf{match}\ c'\, [s]\, p \ldots\ a \ldots\ \mathsf{return}\ \_\ \mathsf{with}\ (c\, z \ldots \Rightarrow e) \ldots \rhd_\iota e'[z' \mapsto a] \ldots}$$

$$\frac{\sigma \equiv \forall (\alpha \ldots).\, \Delta \to X\ \_ \ldots \to \tau \qquad (c, X) \in \Sigma \qquad \|\Delta\| = \|(e' \ldots)\|}{\Gamma \vdash (\mathsf{fix}\ f : \sigma \coloneqq e)\, [s] \ldots\ e' \ldots\ (c\, a \ldots) \rhd_\mu e[\alpha \mapsto s] \ldots [f \mapsto \mathsf{fix}\ f : \sigma \coloneqq e]\, e' \ldots\ (c\, a \ldots)}$$

Figure 7: Reduction rules

$\boxed{\Gamma \vdash e \Updownarrow \sigma}$ *with implicit* Σ

$$\textsc{Conv}\ \dfrac{\Gamma \vdash e \Uparrow \forall(\alpha\ldots).\,\tau' \qquad \Gamma(\alpha\ldots) \vdash \tau' \approx \tau}{\Gamma \vdash e \Downarrow \forall(\alpha\ldots).\,\tau}$$

$$\textsc{Var}\ \dfrac{\vdash \Gamma \qquad (x : \sigma) \in \Gamma}{\Gamma \vdash x \Uparrow \sigma}$$

$$\textsc{Type}\ \dfrac{\vdash \Gamma}{\Gamma \vdash U_i \Uparrow \mathsf{Type}_{i+1}}$$

$$\textsc{Pi}\ \dfrac{\Gamma \vdash \tau \Uparrow U_i \qquad \Gamma(x : \tau) \vdash \tau' \Uparrow U'_j}{\Gamma \vdash \Pi x : \tau.\,\tau' \Uparrow U'_{\max(i,j)}}$$

$$\textsc{Lam}\ \dfrac{\Gamma \vdash \tau \Uparrow U_i \qquad \Gamma(x : \tau) \vdash e \Uparrow \tau'}{\Gamma \vdash \lambda x : \tau.\, e \Uparrow \Pi x : \tau.\,\tau'}$$

$$\textsc{App}\ \dfrac{\Gamma \vdash e_1 \Uparrow \Pi x : \tau.\,\tau' \qquad \Gamma \vdash e_2 \Downarrow \tau}{\Gamma \vdash e_1\ e_2 \Uparrow \tau'[x \mapsto e_1]}$$

$$\textsc{Forall}\ \dfrac{\Gamma(\alpha) \vdash \sigma \Uparrow U_i}{\Gamma \vdash \forall \alpha.\,\sigma \Uparrow U_i}$$

$$\textsc{SLam}\ \dfrac{\Gamma(\alpha) \vdash e \Uparrow \sigma}{\Gamma \vdash \Lambda \alpha.\, e \Uparrow \forall \alpha.\,\sigma}$$

$$\textsc{SApp}\ \dfrac{\Gamma \vdash e \Uparrow \forall \alpha.\,\sigma \qquad \vdash s}{\Gamma \vdash e\,[s] \Uparrow \sigma[\alpha \mapsto s]}$$

$$\textsc{Let}\ \dfrac{\begin{array}{c}\Gamma \vdash \sigma \Uparrow U_i \qquad \Gamma \vdash e_1 \Downarrow \sigma \\ \Gamma(x \coloneqq e_1)(x : \sigma) \vdash e_2 \Uparrow \sigma'\end{array}}{\Gamma \vdash \mathsf{let}\ x : \sigma \coloneqq e_1\ \mathsf{in}\ e_2 \Uparrow \sigma'[x \mapsto e_1]}$$

$$\textsc{Ind}\ \dfrac{\begin{array}{c}\vdash \Gamma \qquad \vdash s \\ (\mathsf{data}\ X\ \Delta : \tau\ \mathsf{where}\ \_) \in \Sigma\end{array}}{\Gamma \vdash X\,[s] \Uparrow \Delta \to \tau}$$

$$\textsc{Constr}\ \dfrac{\begin{array}{c}\vdash \Gamma \qquad \vdash s \\ (\mathsf{data}\ X\ \Delta : \_\ \mathsf{where}\ \Delta_c) \in \Sigma \qquad (c : \forall \alpha.\,\tau) \in \Delta_c\end{array}}{\Gamma \vdash c\,[s] \Uparrow \Delta \to \tau[\alpha \mapsto s]}$$

$$\textsc{Fix}\ \dfrac{\begin{array}{c}\Gamma \vdash \sigma \Uparrow U_i \qquad \sigma \equiv \forall(\alpha\ldots).\,\tau \qquad \Gamma(\alpha)\ldots \vdash \tau \rhd^{*} \Delta \to (x : X\,[\alpha']\,\_\ldots) \to \tau' \\ (\mathsf{data}\ X\ \_ : \_\ \mathsf{where}\ \_) \in \Sigma \qquad \Gamma(\alpha)\ldots(f : \tau) \vdash e \Downarrow \tau[\alpha' \mapsto \alpha' + m] \qquad m \geq 1\end{array}}{\Gamma \vdash \mathsf{fix}\ f : \sigma \coloneqq e \Uparrow \sigma}$$

$$\textsc{Match}\ \dfrac{\begin{array}{c}(\mathsf{data}\ X\ \Delta_P : \Delta_I \to U_i\ \mathsf{where}\ \Delta_c) \in \Sigma \\ \Delta_P \equiv (w : \_)\ldots \qquad \Delta_I \equiv (y : \_)\ldots \qquad \Delta_c \equiv (c : \forall \alpha.\,\Delta_a \to X\,[\widehat{\alpha}]\,w\ldots\ a\ldots)\ldots \qquad \Delta_a \equiv (z : \_)\ldots \\ \Gamma \vdash e' \Uparrow X\,[\widehat{s}]\,p\ldots\ a'\ldots \qquad \Gamma_P \equiv \Gamma\Delta_P(w \coloneqq p)\ldots \qquad \Gamma_P\Delta_I(x : X\,[\widehat{s}]\,w\ldots\ y\ldots) \vdash P \Uparrow U'_j \\ X \mid U_i \succ U'_j \qquad \{\Gamma_P\Delta_a[\alpha \mapsto s] \vdash e \Downarrow P[y \mapsto a]\ldots[x \mapsto c\,[s]\,w\ldots\ z\ldots]\}\ldots\end{array}}{\Gamma \vdash \mathsf{match}\ e'\ \mathsf{return}\ (y\ldots).x.P\ \mathsf{with}\ (c\ z\ldots\ \Rightarrow e)\ldots \Uparrow P[y \mapsto a']\ldots[x \mapsto e']}$$

Figure 8: Typing rules

```
data Size : Type where
  base : Size
  next : Size → Size

data Eq (A : Type)(a : A) : A → Type where
  refl : Eq A a a
  with a =_A a ≡ Eq A a a

let subst : (A : Type) → (P : A → Type) → (x : A) → (y : A) → x = y → P x → P y :=
  λA : Type. λP : A → Type. λx : A. λy : A. λp : x = y.
    match p return (y)._.Πpx : P x. P y with
      (refl ⇒ λpx : P x. px)

let absurd : (A : Type) → (α : Size) → base = next α → A :=
  let discr : Size → Type :=
    λα : Size. match α return ()._.Type with
      (base ⇒ Size)
      (next _ ⇒ A) in
  λA : Type. λα : Size. λp : base = next α.
    subst Size discr base (next α) p base

let inj : (α : Size) → (β : Size) → next α = next β → α = β :=
  let pred : Size → Size :=
    λβ : Size. match β return ()._.Size with
      (base ⇒ base)
      (next β′ ⇒ β′) in
  λα : Size. λβ : Size. λp : next α = next β.
    match p return (β)._.(α = pred β) with
      (refl ⇒ refl Size α)

let shift_X : X p... α a... → X p... (next α) a... := _
```

Figure 9: Preliminary unsized definitions

$\boxed{\Gamma \vdash e \Updownarrow \sigma \rightsquigarrow \boxed{e}}$ *with implicit* Σ

$$\textsc{Conv}\quad \frac{\Gamma \vdash e \Uparrow \forall (\alpha \ldots).\, \tau' \rightsquigarrow \boxed{e} \qquad \Gamma(\alpha \ldots) \vdash \tau' \approx \tau}{\Gamma \vdash e \Downarrow \forall (\alpha \ldots).\, \tau \rightsquigarrow \boxed{e}}$$

$$\textsc{Var}\quad \frac{\vdash \Gamma \qquad (x : \sigma) \in \Gamma}{\Gamma \vdash x \Uparrow \sigma \rightsquigarrow x}$$

$$\textsc{Type}\quad \frac{\vdash \Gamma}{\Gamma \vdash U_i \Uparrow \mathsf{Type}_{i+1} \rightsquigarrow U_i}$$

$$\textsc{Pi}\quad \frac{\Gamma \vdash \tau \Uparrow U_i \rightsquigarrow \boxed{\tau} \qquad \Gamma(x : \tau) \vdash \tau' \Uparrow U'_j \rightsquigarrow \boxed{\tau'}}{\Gamma \vdash \Pi x : \tau.\, \tau' \Uparrow U'_{\max(i,j)} \rightsquigarrow \Pi x : \boxed{\tau}.\, \boxed{\tau'}}$$

$$\textsc{Lam}\quad \frac{\Gamma \vdash \tau \Uparrow U_i \rightsquigarrow \boxed{\tau} \qquad \Gamma(x : \tau) \vdash e \Uparrow \tau' \rightsquigarrow \boxed{\tau'}}{\Gamma \vdash \lambda x : \tau.\, e \Uparrow \Pi x : \tau.\, \tau' \rightsquigarrow \lambda x : \boxed{\tau}.\, \boxed{\tau'}}$$

$$\textsc{App}\quad \frac{\Gamma \vdash e_1 \Uparrow \Pi x : \tau.\, \tau' \rightsquigarrow \boxed{e_1} \qquad \Gamma \vdash e_2 \Downarrow \tau \rightsquigarrow \boxed{e_2}}{\Gamma \vdash e_1\ e_2 \Uparrow \tau'[x \mapsto e_1] \rightsquigarrow \boxed{e_1}\ \boxed{e_2}}$$

$$\textsc{Forall}\quad \frac{\Gamma(\alpha) \vdash \sigma \Uparrow U_i \rightsquigarrow \boxed{\tau}}{\Gamma \vdash \forall \alpha.\, \sigma \Uparrow U_i \rightsquigarrow \Pi \alpha : \mathsf{Size}.\, \boxed{\tau}}$$

$$\textsc{SLam}\quad \frac{\Gamma(\alpha) \vdash e \Uparrow \sigma \rightsquigarrow \boxed{e}}{\Gamma \vdash \Lambda \alpha.\, e \Uparrow \forall \alpha.\, \sigma \rightsquigarrow \lambda \alpha : \mathsf{Size}.\, \boxed{e}}$$

$$\textsc{SApp}\quad \frac{\Gamma \vdash e \Uparrow \forall \alpha.\, \sigma \rightsquigarrow \boxed{e} \qquad \vdash s \rightsquigarrow \boxed{e'}}{\Gamma \vdash e\,[s] \Uparrow \sigma[\alpha \mapsto s] \rightsquigarrow \boxed{e}\ \boxed{e'}}$$

$$\textsc{Let}\quad \frac{\Gamma \vdash \sigma \Uparrow U_i \rightsquigarrow \boxed{\tau} \qquad \Gamma \vdash e_1 \Downarrow \sigma \rightsquigarrow \boxed{e_1} \qquad \Gamma(x \coloneqq e_1)(x : \sigma) \vdash e_2 \Uparrow \sigma' \rightsquigarrow \boxed{e_2}}{\Gamma \vdash \mathsf{let}\ x : \sigma \coloneqq e_1\ \mathsf{in}\ e_2 \Uparrow \sigma'[x \mapsto e_1] \rightsquigarrow \mathsf{let}\ x : \boxed{\tau} \coloneqq \boxed{e_1}\ \mathsf{in}\ \boxed{e_2}}$$

$$\textsc{Ind}\quad \frac{\begin{array}{c} \vdash \Gamma \qquad \vdash s \rightsquigarrow \boxed{e} \\ (\mathsf{data}\ X\ \Delta : \tau\ \mathsf{where}\ \_) \in \Sigma \qquad \Delta \equiv (w : \_) \ldots \end{array}}{\Gamma \vdash X\,[s] \Uparrow \Delta \to \tau \rightsquigarrow \lambda \Delta.\, X\ w \ldots\ \boxed{e}}$$

$$\textsc{Constr}\quad \frac{\begin{array}{c} \vdash \Gamma \qquad \vdash s \rightsquigarrow \boxed{e} \\ (\mathsf{data}\ X\ \Delta : \_\ \mathsf{where}\ \Delta_c) \in \Sigma \qquad (c : \forall \alpha.\, \tau) \in \Delta_c \end{array}}{\Gamma \vdash c\,[s] \Uparrow \Delta \to \tau[\alpha \mapsto s] \rightsquigarrow c\ \boxed{e}}$$

Figure 10: Compilation from sized to unsized (1/2)

$\boxed{\Gamma \vdash e \Updownarrow \sigma \rightsquigarrow \boxed{e}}$ *with implicit* Σ

FIX

$$
\dfrac{\begin{array}{c}
\Gamma \vdash \sigma \Uparrow U_i \rightsquigarrow \boxed{\Pi\alpha : \mathsf{Size}.\ \ldots\tau} \qquad \sigma \equiv \forall(\alpha\ldots).\,\tau \\
\Gamma(\alpha)\ldots \vdash \tau \rhd^{*} \Delta \to (x : X\,[\alpha']\,\_\ldots) \to \tau' \qquad \Gamma(\alpha : \mathsf{Size})\ldots \vdash \boxed{\tau} \rhd^{*} \boxed{\Delta \to (x : X\,p\ldots\ \alpha'\,a\ldots) \to \tau'} \\
\Gamma(\alpha)\ldots(f : \tau) \vdash e \Downarrow \tau[\alpha' \mapsto \alpha' + m] \rightsquigarrow \boxed{e} \qquad m \geq 1 \qquad \beta\ \mathsf{fresh}
\end{array}}{\Gamma \vdash \mathsf{fix}\ f : \sigma \coloneqq e \Uparrow \sigma \rightsquigarrow
\begin{array}{l}
\mathsf{fix}\ f : \Pi\alpha : \mathsf{Size}.\ \ldots\tau \ \coloneqq \lambda\alpha : \mathsf{Size}.\ \ldots \\
\quad \mathsf{match}\ \alpha'\ \mathsf{return}\ ().\alpha'.\tau\ \mathsf{with} \\
\quad\quad (\mathsf{base} \Rightarrow \lambda\,\boxed{\Delta}\,.\,\lambda x : X\ \boxed{p}\ \ldots\ \mathsf{base}\ \boxed{a}\ \ldots. \\
\quad\quad\quad (\mathsf{match}\ x\ \mathsf{return}\ (\beta\ \_\ldots).\_.(\mathsf{base} = \beta) \to \boxed{\tau'}\,)\ \mathsf{with} \\
\quad\quad\quad\quad (c\ \beta\ \_\ldots \ \Rightarrow \mathsf{absurd}\ \boxed{\tau'}\ \beta)\ldots)\ (\mathsf{refl}\ \mathsf{Size}\ \mathsf{base})) \\
\quad\quad (\mathsf{next}\ \alpha' \Rightarrow \boxed{e}\,)
\end{array}}
$$

MATCH

$$
\dfrac{\begin{array}{c}
(\mathsf{data}\ X\ \Delta_P : \Delta_I \to U_i\ \mathsf{where}\ \Delta_c) \in \Sigma \\
\Delta_P \equiv (w : \tau)\ldots \qquad \Delta_I \equiv (y : \_)\ldots \qquad \Delta_c \equiv (c : \forall\alpha.\,\Delta_a \to X\,[\widehat{\alpha}]\,w\ldots\ a\ldots)\ldots \qquad \Delta_a \equiv (z : \_)\ldots \\
\Gamma \vdash e' \Uparrow X\,[\widehat{s}]\,p\ldots\ a'\ldots \rightsquigarrow \boxed{e'} \qquad \{\Gamma \vdash p \Downarrow \tau \rightsquigarrow \boxed{p}\}\ldots \qquad \vdash s \rightsquigarrow \boxed{e_s} \\
\Gamma_P \equiv \Gamma\Delta_P(w \coloneqq p)\ldots \qquad \Gamma_P\Delta_I(x : X\,[\widehat{s}]\,w\ldots\ y\ldots) \vdash P \Uparrow U_j' \rightsquigarrow \boxed{P} \\
X \mid U_i \succ U_j' \qquad \{\Gamma_P\Delta_a[\alpha \mapsto s] \vdash e \Downarrow P[y \mapsto a]\ldots[x \mapsto c\,[s]\,w\ldots\ z\ldots] \rightsquigarrow \boxed{e}\}\ldots \qquad \beta, \beta', q\ \mathsf{fresh}
\end{array}}{\Gamma \vdash \mathsf{match}\ e'\ \mathsf{return}\ (y\ldots).x.P\ \mathsf{with}\ (c\,z\ldots \ \Rightarrow e)\ldots \Uparrow P[y \mapsto a']\ldots[x \mapsto e'] \rightsquigarrow}
$$

$$
\begin{array}{l}
(\mathsf{match}\ \boxed{e'}\ \mathsf{return}\ (\beta\ y\ldots).x.(\beta = \mathsf{next}\ \boxed{e_s}\,) \to \boxed{P}\ \mathsf{with} \\
\quad (c\ \beta\ z\ldots \ \Rightarrow \lambda q : \mathsf{next}\ \beta = \mathsf{next}\ \boxed{e_s}\,. \\
\quad\quad \boxed{e'}[z^{*} \mapsto \lambda\Delta.\,\mathsf{subst}\ \mathsf{Size}\ (\lambda\beta' : \mathsf{Size}.\,X\ \boxed{p}\ \ldots\ \beta'\ a^{*}\ \ldots)\ \beta\ \boxed{e_s}\ (\mathsf{inj}\ \beta\ \boxed{e_s}\ q)\ (z^{*}\ \Delta)])\ldots) \\
(\mathsf{refl}\ \mathsf{Size}\ (\mathsf{next}\ \boxed{e_s}\,)) \\
\qquad\qquad \textit{where}\ (z^{*} : \tau^{*}) \subseteq \boxed{\Delta_a}\,, \tau^{*}[w \mapsto \boxed{p}\,]\ldots \equiv \Delta \to X\ \boxed{p}\ \ldots\ \beta\ a^{*}\ \ldots
\end{array}
$$

Figure 11: Compilation from sized to unsized (2/2)