Jonathan Chan

May 7, 2022

$$
\begin{aligned}
i, j, m, n &::= \textsf{naturals} & s &::= \alpha \mid \widehat{s} \mid \omega & \Gamma &::= \cdot \mid \Gamma(x : \sigma) \mid \Gamma(x \coloneqq e) \mid \Gamma(\alpha) \\
f, g, w, x, y, z, X, c, \alpha, \beta &::= \textsf{variables} & \sigma &::= \forall \alpha.\, \sigma \mid \tau & \Delta &::= \cdot \mid \Delta(x : \sigma)
\end{aligned}
$$

$$
\begin{aligned}
e, a, b, p, A, B, P, \tau &::= x \mid U_i \mid \Pi x : \tau.\, \tau \mid \lambda x : \tau.\, e \mid \Lambda \alpha.\, e \mid e\; e \mid e\,[s] \mid \textsf{let}\; x : \sigma \coloneqq e\; \textsf{in}\; e \\
&\mid \textsf{fix}\; f : \sigma \coloneqq e \mid T \mid \textsf{zero} \mid \textsf{succ} \mid \textsf{sup}_{\mathbb{W} x : \tau.\, \tau} \\
&\mid \textsf{ifzero}\; e\; \textsf{return}\; \lambda x.\, e\; \textsf{then}\; e\; \textsf{else}\; \lambda x.\, e \mid \textsf{prj}_1\; e \mid \textsf{prj}_2\; e \\
&\mid \textsf{match}\; e\; \textsf{return}\; (y \ldots).x.P\; \textsf{with}\; (c\, x \ldots \Rightarrow e) \ldots \\
T &::= \mathbb{N} \mid \mathbb{W} x : \tau.\, \tau \\
U_i &::= \textsf{Prop}_i \mid \textsf{Type}_i
\end{aligned}
$$

Figure 1: Syntax

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

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

Figure 2: Syntactic sugar

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

Figure 3: Inductive definitions

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

Figure 4: Implicit judgements

$\boxed{\vdash \Sigma}$

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

$\boxed{\vdash \Gamma}$ *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 5: Well-formedness rules

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

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

$$\frac{}{\Gamma \vdash \mathsf{let}\ x : \sigma \coloneqq e'\ \mathsf{in}\ e \rhd_\zeta e[x \mapsto e']} \qquad \frac{(c'\, z' \ldots \Rightarrow e') \in (c\, z \ldots \Rightarrow e) \ldots}{\Gamma \vdash \mathsf{match}\ c'\, p \ldots\ a \ldots\ \mathsf{return}\ \_\ \mathsf{with}\ (c\, z \ldots \Rightarrow e) \ldots \rhd_\iota e'[z' \mapsto a] \ldots}$$

$$\frac{}{\Gamma \vdash \mathsf{ifzero}\ \mathsf{zero}\ [s]\ \mathsf{return}\ \lambda x.\, P\ \mathsf{then}\ e_z\ \mathsf{else}\ \lambda y.\, e_s \rhd_\iota e_z}$$

$$\frac{}{\Gamma \vdash \mathsf{ifzero}\ \mathsf{succ}\ [s]\ e\ \mathsf{return}\ \lambda x.\, P\ \mathsf{then}\ e_z\ \mathsf{else}\ \lambda y.\, e_s \rhd_\iota e_s[y \mapsto e]}$$

$$\frac{}{\Gamma \vdash \mathsf{prj}_1\ (\mathsf{sup}_{\mathsf{W} x : A. B}\ [s]\ a\ b) \rhd_\iota a} \qquad \frac{}{\Gamma \vdash \mathsf{prj}_2\ (\mathsf{sup}_{\mathsf{W} x : A. B}\ [s]\ a\ b) \rhd_\iota b}$$

$$\frac{\sigma \equiv \forall (\alpha \ldots).\, \Delta \to T\, [\alpha'] \to \tau \qquad \|\Delta\| = \|(e' \ldots)\|}{\Gamma \vdash (\mathsf{fix}\ f : \sigma \coloneqq e)\ [s] \ldots\ e' \ldots\ (c\, a \ldots) \rhd_\mu e[\alpha \mapsto s] \ldots [f \mapsto \mathsf{fix}\ f : \sigma \coloneqq e]\ e' \ldots\ (c\, a \ldots)}$$

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

Figure 6: Reduction rules

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

$$\textsc{Conv}\ \frac{\Gamma \vdash e \Uparrow \forall(\alpha \ldots).\, \tau' \qquad \Gamma(\alpha \ldots) \vdash \tau' \approx \tau}{\Gamma \vdash e \Downarrow \forall(\alpha \ldots).\, \tau} \qquad \textsc{Var}\ \frac{\vdash \Gamma \qquad (x : \sigma) \in \Gamma}{\Gamma \vdash x \Uparrow \sigma} \qquad \textsc{Type}\ \frac{\vdash \Gamma}{\Gamma \vdash U_i \Uparrow \mathsf{Type}_{i+1}}$$

$$\textsc{Pi}\ \frac{\Gamma \vdash \tau \Uparrow U_i \qquad \Gamma(x : \tau) \vdash \tau' \Uparrow U'_j}{\Gamma \vdash \Pi x : \tau.\, \tau' \Uparrow U'_{\max(i,j)}} \qquad \textsc{Lam}\ \frac{\Gamma \vdash \tau \Uparrow U_i \qquad \Gamma(x : \tau) \vdash e \Uparrow \tau'}{\Gamma \vdash \lambda x : \tau.\, e \Uparrow \Pi x : \tau.\, \tau'} \qquad \textsc{App}\ \frac{\Gamma \vdash e_1 \Uparrow \Pi x : \tau.\, \tau' \qquad \Gamma \vdash e_2 \Downarrow \tau}{\Gamma \vdash e_1\, e_2 \Uparrow \tau'[x \mapsto e_1]}$$

$$\textsc{Forall}\ \frac{\Gamma(\alpha) \vdash \sigma \Uparrow U_i}{\Gamma \vdash \forall \alpha.\, \sigma \Uparrow U_i} \qquad \textsc{SLam}\ \frac{\Gamma(\alpha) \vdash e \Uparrow \sigma}{\Gamma \vdash \Lambda \alpha.\, e \Uparrow \forall \alpha.\, \sigma} \qquad \textsc{SApp}\ \frac{\Gamma \vdash e \Uparrow \forall \alpha.\, \sigma \qquad \vdash s}{\Gamma \vdash e\, [s] \Uparrow \sigma[\alpha \mapsto s]} \qquad \textsc{Let}\ \frac{\begin{array}{c}\Gamma \vdash \sigma \Uparrow U_i \qquad \Gamma \vdash e_1 \Downarrow \sigma \\ \Gamma(x \coloneqq e_1)(x : \sigma) \vdash e_2 \Uparrow \sigma'\end{array}}{\Gamma \vdash \mathsf{let}\ x : \sigma \coloneqq e_1\ \mathsf{in}\ e_2 \Uparrow \sigma'[x \mapsto e_1]}$$

$$\textsc{Nat}\ \frac{\vdash \Gamma \qquad \vdash s}{\Gamma \vdash \mathbb{N}\, [s] \Uparrow \mathsf{Type}_0} \qquad \textsc{Zero}\ \frac{\vdash \Gamma \qquad \vdash s}{\Gamma \vdash \mathsf{zero}\, [s] \Uparrow \mathbb{N}\, [\hat{s}]} \qquad \textsc{Succ}\ \frac{\Gamma \vdash e \Downarrow \mathbb{N}\, [s]}{\Gamma \vdash \mathsf{succ}\, [s]\, e \Uparrow \mathbb{N}\, [\hat{s}]}$$

$$\textsc{W}\ \frac{\begin{array}{c}\vdash s \qquad \Gamma \vdash \tau \Uparrow U_i \\ \Gamma(x : \tau) \vdash \tau' \Uparrow U'_j\end{array}}{\Gamma \vdash \mathbb{W} x : \tau.\, \tau'\, [s] \Uparrow U_{\max(i,j)}} \qquad \textsc{Sup}\ \frac{\Gamma \vdash a \Downarrow \tau \qquad \Gamma(a : \tau) \vdash b \Downarrow \tau'[x \mapsto a] \to \mathbb{W} x : \tau.\, \tau'\, [s]}{\Gamma \vdash \mathsf{sup}_{\mathbb{W} x : \tau.\tau'}\, [s]\, a\, b \Uparrow \mathbb{W} x : \tau.\, \tau'\, [\hat{s}]}$$

$$\textsc{IfZero}\ \frac{\begin{array}{c}\Gamma \vdash e \Uparrow \mathbb{N}\, [\hat{s}] \qquad \Gamma(x : \mathbb{N}\, [\hat{s}]) \vdash P \Uparrow U_i \\ \Gamma \vdash e_z \Downarrow P[x \mapsto \mathsf{zero}\, [s]] \qquad \Gamma(y : \mathbb{N}\, [s]) \vdash e_s \Downarrow P[x \mapsto \mathsf{succ}\, [s]\, y]\end{array}}{\Gamma \vdash \mathsf{ifzero}\ e\ \mathsf{return}\ \lambda x.\, P\ \mathsf{then}\ e_z\ \mathsf{else}\ \lambda y.\, e_s \Uparrow P[x \mapsto e]}$$

$$\textsc{Prj}_1\ \frac{\Gamma \vdash e \Uparrow \mathbb{W} x : \tau.\, \tau'\, [\hat{s}]}{\Gamma \vdash \mathsf{prj}_1\ e \Uparrow \tau} \qquad \textsc{Prj}_2\ \frac{\Gamma \vdash e \Uparrow \mathbb{W} x : \tau.\, \tau'\, [\hat{s}]}{\Gamma \vdash \mathsf{prj}_2\ e \Uparrow \tau'[x \mapsto \mathsf{prj}_1\ e] \to \mathbb{W} x : \tau.\, \tau'\, [s]}$$

$$\textsc{Fix}\ \frac{\begin{array}{c}\Gamma \vdash \sigma \Uparrow U_i \qquad \sigma \equiv \forall(\alpha \ldots).\, \tau \\ \Gamma(\alpha) \ldots \vdash \tau \rhd^* \Delta \to (x : T\, [\alpha']) \to \tau' \\ \Gamma(\alpha) \ldots (f : \tau) \vdash e \Downarrow \tau[\alpha' \mapsto \alpha' + m] \qquad m \geq 1\end{array}}{\Gamma \vdash \mathsf{fix}\ f : \sigma \coloneqq e \Uparrow \sigma} \qquad \textsc{Fix-Unsized}\ \frac{\Gamma \vdash \tau \Uparrow U_i \qquad \Gamma(f : \tau) \vdash e \Downarrow \tau}{\Gamma \vdash \mathsf{fix}\ f : \tau \coloneqq e \Uparrow \tau}$$

$$\textsc{Ind}\ \frac{\vdash \Gamma \qquad (\mathsf{data}\ X\ \Delta : \tau\ \mathsf{where}\ \_) \in \Sigma}{\Gamma \vdash X \Uparrow \Delta \to \tau} \qquad \textsc{Constr}\ \frac{\vdash \Gamma \qquad (\mathsf{data}\ X\ \Delta : \_\ \mathsf{where}\ \Delta_c) \in \Sigma \qquad (c : \tau) \in \Delta_c}{\Gamma \vdash c \Uparrow \Delta \to \tau}$$

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

Figure 7: Typing rules

$$
\begin{aligned}
&\mathsf{data}\ \mathsf{Size} : \mathsf{Type}\ \mathsf{where}\\
&\quad \mathsf{base} : \mathsf{Size}\\
&\quad \mathsf{next} : \mathsf{Size} \to \mathsf{Size}
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{data}\ \mathsf{Eq}\ (A : \mathsf{Type})(a : A) : A \to \mathsf{Type}\ \mathsf{where}\\
&\quad \mathsf{refl} : \mathsf{Eq}\ A\ a\ a\\
&\quad \textit{with}\ a =_A a \equiv \mathsf{Eq}\ A\ a\ a
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{data}\ \mathsf{Nat} : \mathsf{Size} \to \mathsf{Type}\ \mathsf{where}\\
&\quad \mathsf{zero} : (\alpha : \mathsf{Size}) \to \mathsf{Nat}\ (\mathsf{next}\ \alpha)\\
&\quad \mathsf{succ} : (\alpha : \mathsf{Size}) \to \mathsf{Nat}\ \alpha \to \mathsf{Nat}\ (\mathsf{next}\ \alpha)
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{data}\ \mathsf{W}\ (A : \mathsf{Type})(B : A \to \mathsf{Type}) : \mathsf{Size} \to \mathsf{Type}\ \mathsf{where}\\
&\quad \mathsf{sup} : (\alpha : \mathsf{Size}) \to (a : A) \to (B\ a \to \mathsf{W}\ A\ B\ \alpha) \to \mathsf{W}\ A\ B\ (\mathsf{next}\ \alpha)\\
&\textit{with}\ \mathsf{W}(x : A).\, B \equiv \mathsf{W}\ A\ (\lambda x : A.\, B)\\
&\textit{and}\ \mathsf{sup}_{\mathsf{W}(x:A).B} \equiv \mathsf{sup}\ A\ (\lambda x : A.\, B)
\end{aligned}
$$

Figure 8: Common inductive definitions

```
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 shiftNat : (α : Size) → Nat α → Nat (next α) ≔
  fix shiftNat : (α : Size) → Nat α → Nat (next α) ≔
    λα : Size. λn : Nat α. match n return (β)._.Nat (next β) with
      (zero β ⇒ zero (next β))
      (succ β m ⇒ succ (next β) (shiftNat β m))

let shiftW : (A : Type) → (B : A → Type) → (α : Size) → W A B α → W A B (next α) ≔
  λA : Type. λB : A → Type.
    fix shiftWAB : (α : Size) → W A B α → W A B (next α) ≔
      λα : Size. λw : W A B α. match w return (β)._.W A B (next β) with
        (sup a b β ⇒ sup a (λba : B a. shiftWAB β (b ba)) (next β))
```

Figure 9: Preliminary unsized definitions

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

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

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

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

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

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

APP
$$\frac{\Gamma \vdash e_1 \Uparrow \Pi x : \tau.\,\tau' \rightsquigarrow e_1 \qquad \Gamma \vdash e_2 \Downarrow \tau \rightsquigarrow e_2}{\Gamma \vdash e_1\; e_2 \Uparrow \tau'[x \mapsto e_1] \rightsquigarrow e_1\; e_2}$$

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

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

SAPP
$$\frac{\Gamma \vdash e \Uparrow \forall\alpha.\,\sigma \rightsquigarrow e \qquad \vdash s \rightsquigarrow e'}{\Gamma \vdash e\,[s] \Uparrow \sigma[\alpha \mapsto s] \rightsquigarrow e\; e'}$$

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

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

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

FIX

$$\frac{\begin{array}{c} \Gamma \vdash \sigma \Uparrow U_i \rightsquigarrow \Pi\alpha : \mathsf{Size}.\ \ldots\ \tau \qquad \sigma \equiv \forall(\alpha \ldots).\,\tau \\ \Gamma(\alpha)\ldots \vdash \tau \rhd^{*} \Delta \to (x : T\,[\alpha']) \to \tau' \qquad \Gamma(\alpha : \mathsf{Size})\ldots \vdash \tau \rhd^{*} \Delta \to (x : T\ \alpha') \to \tau' \\ \Gamma(\alpha)\ldots(f : \tau) \vdash e \Downarrow \tau[\alpha' \mapsto \alpha' + m] \rightsquigarrow e \qquad m \geq 1 \qquad \beta\ \mathsf{fresh} \end{array}}{\Gamma \vdash \mathsf{fix}\ f : \sigma \coloneqq e \Uparrow \sigma \rightsquigarrow \begin{array}{l} \mathsf{fix}\ f : \Pi\alpha : \mathsf{Size}.\ \ldots\ \tau \coloneqq \lambda\alpha : \mathsf{Size}.\ \ldots \\ \quad \mathsf{match}\ \alpha'\ \mathsf{return}\ ().\alpha'.\tau\ \mathsf{with} \\ \quad\quad (\mathsf{base} \Rightarrow \lambda\Delta.\,\lambda x : T\ \mathsf{base}. \\ \quad\quad\quad (\mathsf{match}\ x\ \mathsf{return}\ (\beta\ \_\ \ldots).\_.(\mathsf{base} = \beta) \to \tau')\ \mathsf{with} \\ \quad\quad\quad\quad (c\ \beta\ \_\ \ldots \Rightarrow \mathsf{absurd}\ \tau'\ \beta)\ldots)\ (\mathsf{refl}\ \mathsf{Size}\ \mathsf{base})) \\ \quad\quad (\mathsf{next}\ \alpha' \Rightarrow e) \end{array}}$$

IFZERO

$$\frac{\begin{array}{c} \Gamma \vdash e \Uparrow \mathbb{N}\,[\hat{s}] \rightsquigarrow e \qquad \vdash s \rightsquigarrow e' \quad \Gamma(x : \mathbb{N}\,[\hat{s}]) \vdash P \Uparrow U_i \rightsquigarrow P \\ \Gamma \vdash e_z \Downarrow P[x \mapsto \mathsf{zero}\,[s]] \rightsquigarrow e_z \qquad \Gamma(y : \mathbb{N}\,[s]) \vdash e_s \Downarrow P[x \mapsto \mathsf{succ}\,[s]\ y] \rightsquigarrow e_s \qquad \beta, q\ \mathsf{fresh} \end{array}}{\begin{array}{l} \Gamma \vdash \mathsf{ifzero}\ e\ \mathsf{return}\ \lambda x.\,P\ \mathsf{then}\ e_z\ \mathsf{else}\ \lambda y.\,e_s \Uparrow P[x \mapsto e] \rightsquigarrow \\ \quad (\mathsf{match}\ e\ \mathsf{return}\ (\beta).x.(\beta = \mathsf{next}\ e') \to P\ \mathsf{with} \\ \quad\quad (\mathsf{zero}\ \beta \Rightarrow \lambda q : \mathsf{next}\ \beta = \mathsf{next}\ e'.\ e_z) \\ \quad\quad (\mathsf{succ}\ \beta\ y \Rightarrow \lambda q : \mathsf{next}\ \beta = \mathsf{next}\ e'. \\ \quad\quad\quad e_s[y \mapsto \mathsf{subst}\ \mathsf{Size}\ \mathsf{Nat}\ \beta\ e'\ (\mathsf{inj}\ \beta\ e'\ q)\ y])) \\ \quad (\mathsf{refl}\ \mathsf{Size}\ (\mathsf{next}\ e')) \end{array}}$$

MATCH

$$\frac{\begin{array}{c} (\mathsf{data}\ X\ \Delta_P : \Delta_I \to U_i\ \mathsf{where}\ \Delta_c) \in \Sigma \\ \Delta_P \equiv (w : \tau)\ldots \qquad \Delta_I \equiv (y : \_)\ldots \qquad \Delta_c \equiv (c : \forall\alpha.\,\Delta_a \to X\,[\hat{\alpha}]\ w \ldots\ a \ldots)\ldots \qquad \Delta_a \equiv (z : \_)\ldots \\ \Gamma \vdash e' \Uparrow X\,[\hat{s}]\ p \ldots\ a' \ldots \rightsquigarrow e' \qquad \{\Gamma \vdash p \Downarrow \tau \rightsquigarrow p\}\ldots \qquad \vdash s \rightsquigarrow e_s \\ \Gamma_P \equiv \Gamma\Delta_P(w \coloneqq p)\ldots \qquad \Gamma_P\Delta_I(x : X\,[\hat{s}]\ w \ldots\ y \ldots) \vdash P \Uparrow U'_j \rightsquigarrow P \\ X \mid U_i \succ U'_j \qquad \{\Gamma_P\Delta_a[\alpha \mapsto s] \vdash e \Downarrow P[y \mapsto a]\ldots[x \mapsto c\,[s]\ w \ldots\ z \ldots] \rightsquigarrow e\}\ldots \qquad \beta, \beta', q\ \mathsf{fresh} \end{array}}{\begin{array}{l} \Gamma \vdash \mathsf{match}\ e'\ \mathsf{return}\ (y \ldots).x.P\ \mathsf{with}\ (c\ z \ldots \Rightarrow e)\ldots \Uparrow P[y \mapsto a']\ldots[x \mapsto e'] \rightsquigarrow \\ (\mathsf{match}\ e'\ \mathsf{return}\ (\beta\ y \ldots).x.(\beta = \mathsf{next}\ e_s) \to P\ \mathsf{with} \\ \quad (c\ \beta\ z \ldots \Rightarrow \lambda q : \mathsf{next}\ \beta = \mathsf{next}\ e_s. \\ \quad\quad e'[z^{*} \mapsto \lambda\Delta.\,\mathsf{subst}\ \mathsf{Size}\ (\lambda\beta' : \mathsf{Size}.\,X\ p \ldots\ \beta'\ a^{*} \ldots)\ \beta\ e_s\ (\mathsf{inj}\ \beta\ e_s\ q)\ (z^{*}\ \Delta)])\ldots) \\ (\mathsf{refl}\ \mathsf{Size}\ (\mathsf{next}\ e_s)) \\ \quad\quad\quad \textit{where}\ (z^{*} : \tau^{*}) \subseteq \Delta_a, \tau^{*}[w \mapsto p]\ldots \equiv \Delta \to X\ p \ldots\ \beta\ a^{*} \ldots \end{array}}$$

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