TT

Andrej Bauer Matija Pretnar Christopher A. Stone

December 4, 2018

1 Abstract syntax

Expression e	$::=$	X	Variable
	$\mid$	$\mathsf{fun}\ X \Rightarrow C$	Function
	$\mid$	h	Handler
	$\mid$	$\mathsf{cont}(\Gamma, \Delta, \mathcal{K})$	Continuation value
	$\mid$	$\langle B \rangle$	Galactic term
	$\mid$	$\langle T \rangle$	Galactic type
	$\mid$	$(e_1, \ldots, e_n)$	Tuple
	$\mid$	c	TT Constant
	$\mid$	$\mathsf{inj}_i\ e$	Coproduct
Computation C	$::=$	$\mathsf{val}\ e$	Pure expression
	$\mid$	$e_1\ e_2$	Application
	$\mid$	$\mathsf{let}\ X = C_1\ \mathsf{in}\ C_2$	let-binding
	$\mid$	$\mathsf{op}_i\ e$	Operation
	$\mid$	$\mathsf{with}\ e\ \mathsf{handle}\ C$	Handling
	$\mid$	$e_1[e_2]$	Invoke a continuation
	$\mid$	$e_1 :: e_2$	Type ascription
	$\mid$	$f(e_1, \ldots, e_n)$	Primitive operations
	$\mid$	$\mathsf{match}\ e\ \mathsf{with}\ (P_i \Rightarrow C_i)_{i=1}^n$	Pattern-match
	$\mid$	$\mathsf{check}\ T_1 \equiv T_2\ \mathsf{using}\ e\ \mathsf{in}\ C$	Run-time Equivalence Check
	$\mid$	$\mathsf{debruijn}\ n$	Build galactic term: variable
	$\mid$	$\lambda x : e \,.\, C$	Build galactic term: abstraction
	$\mid$	$\mathsf{app}(e_1, e_2)$	Build galactic term: application
Continuation $\mathcal{K}$	$::=$	$\diamond$	Hole
	$\mid$	$\mathsf{let}\ X = \mathcal{K}\ \mathsf{in}\ C_2$	let-binding
	$\mid$	$\mathsf{with}\ e\ \mathsf{handle}\ \mathcal{K}$	Handling
	$\mid$	$\lambda x : T \,.\, \mathcal{K}$	Abstraction
Handler h	$::=$	$\mathsf{handler}\ \mathsf{val}\ X \mapsto C_v \mid (\mathsf{op}_i(X, K) \mapsto C_n)_{i=1}^n$	
	$\mid$	$\mathsf{handler}\ \mathsf{val}\ X \mapsto C_v \mid (\mathsf{op}_i(X, K) \mapsto C_n)_{i=1}^n \mid \mathsf{finally}\ X_f \mapsto C_f$	
Pattern P	$::=$	$(X_1, \ldots, X_n)$	
	$\mid$	$\mathsf{inj}_i\ X$	
	$\mid$	c	

2 Operational semantics

Results:

$$\begin{array}{rcl} \text{Result } R & ::= & \mathsf{val}\, e \\ & | & \mathsf{op}_i(\Delta, e, \mathcal{K}) \end{array}$$

Judgments:

$\Gamma \vdash C \Downarrow R$	C evaluates to result R in context Γ
$\Gamma \vdash R\ \mathsf{ok}$	R is a valid result in context Γ
$\Gamma \vdash e\ \mathsf{ok}$	e is a valid expression in context Γ
$\Gamma \vdash C\ \mathsf{ok}$	C is a valid computation in context Γ
$\Gamma \vdash K : \Delta \to \mathsf{ok}$	K is a valid continuation in context Γ, with its hole inside additional binders Δ

Generic Computations

eval-val
$$\frac{}{\Gamma \vdash \mathsf{val}\, e \Downarrow \mathsf{val}\, e}$$

eval-app
$$\frac{\Gamma \vdash C[e/X] \Downarrow R}{\Gamma \vdash (\mathsf{fun}\ X \Rightarrow C)\, e \Downarrow R}$$

eval-let-val
$$\frac{\Gamma \vdash C_1 \Downarrow \mathsf{val}\, e \qquad \Gamma \vdash C_2[e/X] \Downarrow R}{\Gamma \vdash \mathsf{let}\ X = C_1\ \mathsf{in}\ C_2 \Downarrow R}$$

eval-let-op
$$\frac{\Gamma \vdash C_1 \Downarrow \mathsf{op}_i(\Delta, e, \mathcal{K})}{\Gamma \vdash \mathsf{let}\ X = C_1\ \mathsf{in}\ C_2 \Downarrow \mathsf{op}_i(\Delta, e, \mathsf{let}\ X = \mathcal{K}\ \mathsf{in}\ C_2)}$$

eval-kapp
$$\frac{\Gamma, \Delta \vdash \mathcal{K}[\diamond := e] \Downarrow R}{\Gamma, \Delta \vdash \mathsf{cont}(\Gamma, \Delta, \mathcal{K})[e] \Downarrow R}$$

eval-match-tuple
$$\frac{P_j = (X_1, \ldots, X_m) \qquad \Gamma \vdash C_j[e_1/X_1, \ldots, e_m/X_m] \Downarrow R}{\Gamma \vdash \mathsf{match}\ (e_1, \ldots, e_m)\ \mathsf{with}\ (P_i \Rightarrow C_i)_{i=1}^n \Downarrow R}$$

eval-match-inj
$$\frac{P_j = \mathsf{inj}_k\, X \qquad \Gamma \vdash C_j[e/X] \Downarrow R}{\Gamma \vdash \mathsf{match}\ \mathsf{inj}_k\, e\ \mathsf{with}\ (P_i \Rightarrow C_i)_{i=1}^n \Downarrow R}$$

eval-match-const
$$\frac{P_j = c \qquad \Gamma \vdash C_j \Downarrow R}{\Gamma \vdash \mathsf{match}\ c\ \mathsf{with}\ (P_i \Rightarrow C_i)_{i=1}^n \Downarrow R}$$

Operations and Handlers

eval-op

$$\frac{}{\Gamma \vdash \mathsf{op}_i\, e \ \Downarrow\ \mathsf{op}_i(\bullet, e, \diamond)}$$

eval-handle-val

$$\frac{\Gamma \vdash C \ \Downarrow\ \mathsf{val}\, e \qquad \Gamma \vdash C_v[e/X] \ \Downarrow\ R}{\Gamma \vdash \mathsf{with}\ \big(\mathsf{handler}\ \mathsf{val}\, X \mapsto C_v \mid (\mathsf{op}_i(X, K) \mapsto C_n)_{i=1}^{n}\big)\ \mathsf{handle}\ C \ \Downarrow\ R}$$

eval-handle-op-val

$$\frac{\begin{array}{c} h = \mathsf{handler}\ \mathsf{val}\, X \mapsto C_v \mid (\mathsf{op}_i(X, K) \mapsto C_n)_{i=1}^{n} \\ \Gamma \vdash C \ \Downarrow\ \mathsf{op}_i(\Delta, e, \mathcal{K}_1) \\ \Gamma, \Delta \vdash C_i[e/X, \mathsf{cont}(\Gamma, \Delta, \mathsf{with}\ h\ \mathsf{handle}\ \mathcal{K}_1)/K] \ \Downarrow\ \mathsf{val}\, e' \\ \Gamma \vdash e' \ \ \mathsf{ok} \end{array}}{\Gamma \vdash \mathsf{with}\ h\ \mathsf{handle}\ C \ \Downarrow\ \mathsf{val}\, e'}$$

eval-handle-finally

$$\frac{\begin{array}{c} h = \mathsf{handler}\ \mathsf{val}\, X \mapsto C_v \mid (\mathsf{op}_i(X, K) \mapsto C_n)_{i=1}^{n} \mid \mathsf{finally}\ X_f \mapsto C_f \\ h' = \mathsf{handler}\ \mathsf{val}\, X \mapsto C_v \mid (\mathsf{op}_i(X, K) \mapsto C_n)_{i=1}^{n} \\ \Gamma \vdash (\mathsf{let}\ X_f = \mathsf{with}\ h'\ \mathsf{handle}\ C\ \mathsf{in}\ C_f) \ \Downarrow\ R \end{array}}{\Gamma \vdash \mathsf{with}\ h\ \mathsf{handle}\ C \ \Downarrow\ R}$$

eval-handle-op-op

$$\frac{\begin{array}{c} h = \mathsf{handler}\ \mathsf{val}\, X \mapsto C_v \mid (\mathsf{op}_i(X, K) \mapsto C_n)_{i=1}^{n} \\ \Gamma \vdash C \ \Downarrow\ \mathsf{op}_i(\Delta, e, \mathcal{K}_1) \\ \Gamma, \Delta \vdash C_i[e/X, \mathsf{cont}(\Gamma, \Delta, \mathsf{with}\ h\ \mathsf{handle}\ \mathcal{K}_1)/K] \ \Downarrow\ \mathsf{op}_j(\Delta', e', \mathcal{K}_2) \end{array}}{\Gamma \vdash \mathsf{with}\ h\ \mathsf{handle}\ C \ \Downarrow\ \mathsf{op}_j((\Delta, \Delta'), e', \mathcal{K}_2)}$$

Built-In Functions

eval-prim
$$\frac{e = [\![f]\!](e_1, \ldots, e_n) \qquad [\Gamma \vdash e \;\; \mathsf{ok}]}{\Gamma \vdash f(e_1, \ldots, e_n) \;\Downarrow\; \mathsf{val}\, e}$$

eval-ascribe
$$\frac{\mathsf{type\_of}(\Gamma, B) = U \qquad \Gamma \vdash \begin{pmatrix} \mathsf{let}\ X_1 = \mathsf{equivTy}(\langle\, T\,\rangle, \langle\, U\,\rangle)\ \mathsf{in} \\ \mathsf{check}\ T \equiv U\ \mathsf{using}\ X_1\ \mathsf{in} \\ \mathsf{val}\,\langle\, B :: T\,\rangle \end{pmatrix} \Downarrow\; R}{\Gamma \vdash \langle\, B\,\rangle :: \langle\, T\,\rangle \;\Downarrow\; R}$$

eval-make-var
$$\frac{\Gamma = x_{m-1}{:}T_{m-1} \ldots, x_0{:}T_0 \qquad n < m}{\Gamma \vdash \mathsf{debruijn}\, n \;\Downarrow\; \mathsf{val}\,\langle\, x_n\,\rangle}$$

eval-make-lambda-val
$$\frac{\Gamma,\, x : T \vdash C \;\Downarrow\; \mathsf{val}\,\langle\, B\,\rangle \qquad \mathsf{type\_of}(\Gamma, B) = U}{\Gamma \vdash \lambda x : T \,.\, C \;\Downarrow\; \mathsf{val}\;\langle\, \lambda x{:}T.U \,.\, B\,\rangle}$$

eval-make-lambda-val-tuple
$$\frac{\Gamma,\, x : T \vdash C \;\Downarrow\; \mathsf{val}\,(\langle\, B_1\,\rangle, \ldots, \langle\, B_n\,\rangle) \qquad (U_i = \mathsf{type\_of}(\Gamma, B_i))_{i=1}^{n}}{\Gamma \vdash \lambda x : T \,.\, C \;\Downarrow\; \mathsf{val}\,(\langle\, \lambda x{:}T.U_1 \,.\, B_1\,\rangle, \ldots, \langle\, \lambda x{:}T.U_n \,.\, B_n\,\rangle)}$$

eval-make-lambda-op
$$\frac{\Gamma,\, x : T \vdash C \;\Downarrow\; \mathsf{op}_i(\Delta, e, \mathcal{K})}{\Gamma \vdash \lambda x : T \,.\, C \;\Downarrow\; \mathsf{op}_i((x{:}T, \Delta), e, \lambda x : T \,.\, \mathcal{K})}$$

eval-assert-type
$$\frac{\begin{array}{c} \vec{w} = (\langle\, B_1\,\rangle, \ldots, \langle\, B_n\,\rangle) \qquad (\mathsf{Id}_{U_i}(e'_i, e''_i) = \mathsf{type\_of}(\Gamma, B_i))_{i=1}^{n} \\ \Gamma\,;\, (e'_i \equiv e''_i)_{i=1}^{n} \vdash T_1 \approx T_2 \qquad \Gamma \vdash C \;\Downarrow\; R \end{array}}{\Gamma \vdash \mathsf{check}\ T_1 \equiv T_2\ \mathsf{using}\ \vec{w}\ \mathsf{in}\ C \;\Downarrow\; R}$$

eval-make-app
$$\frac{\Gamma \vdash \begin{pmatrix} \mathsf{type\_of}(\Gamma, B_1) = \prod_{(x:T_1)} T_2 \\ \mathsf{type\_of}(\Gamma, B_2) = T_3 \\ \mathsf{let}\ X = \mathsf{equivTy}(\langle\, T_1\,\rangle, \langle\, T_3\,\rangle)\ \mathsf{in} \\ \mathsf{check}\ T_1 \equiv T_3\ \mathsf{using}\ X\ \mathsf{in} \\ \langle\, \mathsf{equation}\ \vec{w}\ \mathsf{in}\ B_1 \mathbin{@^{x:T_1.T_2}} B_2\,\rangle \end{pmatrix} \Downarrow\; R}{\Gamma \vdash \mathsf{app}(\langle\, B_1\,\rangle, \langle\, B_2\,\rangle) \;\Downarrow\; R}$$

Type Equivalence Search

General Type equality

chk-tyeq-reflexivity

$$\frac{}{\Gamma \vdash \mathsf{equivTy}(\langle\, T \,\rangle, \langle\, T \,\rangle) \ \Downarrow\ \mathsf{val}\,()}$$

chk-tyeq-hnf

$$\frac{\Gamma \vdash T \rightsquigarrow^* T' \not\rightsquigarrow \qquad \Gamma \vdash U \rightsquigarrow^* U' \not\rightsquigarrow \qquad \Gamma \vdash \mathsf{equivWhnfTy}(\langle\, T' \,\rangle, \langle\, U' \,\rangle) \ \Downarrow\ R}{\Gamma \vdash \mathsf{equivTy}(\langle\, T \,\rangle, \langle\, U \,\rangle) \ \Downarrow\ R}$$

Equality of head-normal forms

chk-tyeq-path-reflexivity

$$\frac{}{\Gamma \vdash \mathsf{equivWhnfTy}(\langle\, T \,\rangle, \langle\, T \,\rangle) \ \Downarrow\ \mathsf{val}\,()}$$

chk-tyeq-el

$$\frac{\alpha = \beta \qquad \Gamma \vdash \mathsf{equivTerm}(e_1, \mathbb{U}_\alpha, e_2)\mathbb{U}_\alpha\mathbb{U}_\alpha \ \Downarrow\ R}{\Gamma \vdash \mathsf{equivWhnfTy}(\langle\, \mathsf{El}^\alpha\, e_1 \,\rangle, \langle\, \mathsf{El}^\beta\, e_2 \,\rangle) \ \Downarrow\ R}$$

chk-tyeq-prod

$$\frac{\Gamma \vdash \left(\begin{array}{l} \mathsf{let}\ X_1 = \mathsf{equivTy}(\langle\, T_1 \,\rangle, \langle\, U_1 \,\rangle)\ \mathsf{in} \\ \quad \mathsf{check}\ T_1 \equiv U_1\ \mathsf{using}\ X_1\ \mathsf{in} \\ \quad\quad \mathsf{let}\ X_2 = \lambda x : T_1\,.\,\mathsf{equivTy}(\langle\, T_2 \,\rangle, \langle\, U_2 \,\rangle)\ \mathsf{in} \\ \quad\quad\quad X_1 \mathbin{++} X_2 \end{array}\right) \Downarrow\ R}{\Gamma \vdash \mathsf{equivWhnfTy}(\langle\, \prod_{(x:T_1)} T_2 \,\rangle, \langle\, \prod_{(x:U_1)} U_2 \,\rangle) \ \Downarrow\ R}$$

chk-tyeq-paths

$$\frac{\Gamma \vdash \left(\begin{array}{l} \mathsf{let}\ X_1 = \mathsf{equivTy}(\langle\, T \,\rangle, \langle\, U \,\rangle)\ \mathsf{in} \\ \quad \mathsf{check}\ T \equiv U\ \mathsf{using}\ X_1\ \mathsf{in} \\ \quad\quad \mathsf{let}\ X_2 = \mathsf{equivTerm}(e_1, e_1', T)\ \mathsf{in} \\ \quad\quad\quad \mathsf{let}\ X_3 = \mathsf{equivTerm}(e_2, e_2', T)\ \mathsf{in} \\ \quad\quad\quad\quad X_1 \mathbin{++} X_2 \mathbin{++} X_3 \end{array}\right) \Downarrow\ R}{\Gamma \vdash \mathsf{equivWhnfTy}(\langle\, \mathsf{Paths}_T(e_1, e_2) \,\rangle, \langle\, \mathsf{Paths}_U(e_1', e_2') \,\rangle) \ \Downarrow\ R}$$

chk-tyeq-id

$$\frac{\Gamma \vdash \left(\begin{array}{l} \mathsf{let}\ X_1 = \mathsf{equivTy}(\langle\, T \,\rangle, \langle\, U \,\rangle)\ \mathsf{in} \\ \quad \mathsf{check}\ T \equiv U\ \mathsf{using}\ X_1\ \mathsf{in} \\ \quad\quad \mathsf{let}\ X_2 = \mathsf{equivTerm}(e_1, e_1', T)\ \mathsf{in} \\ \quad\quad\quad \mathsf{let}\ X_3 = \mathsf{equivTerm}(e_2, e_2', T)\ \mathsf{in} \\ \quad\quad\quad\quad X_1 \mathbin{++} X_2 \mathbin{++} X_3 \end{array}\right) \Downarrow\ R}{\Gamma \vdash \mathsf{equivWhnfTy}(\langle\, \mathsf{Id}_T(e_1, e_2) \,\rangle, \langle\, \mathsf{Id}_U(e_1', e_2') \,\rangle) \ \Downarrow\ R}$$

The reflexivity rule is not just an optimization, but also handles equivalence of base types and equivalence of universes.

2.1 Term equality

chk-eq-refl

$$\frac{}{\Gamma \vdash \mathsf{equivTerm}(e, e, T) \ \Downarrow\ \mathsf{val}\,()}$$

chk-eq-ext

$$\frac{\Gamma \vdash T \rightsquigarrow^{*} T' \not\rightsquigarrow \qquad \Gamma \vdash \mathsf{equivExtTerm}(e_1, e_2, \langle T' \rangle) \ \Downarrow\ R}{\Gamma \vdash \mathsf{equivTerm}(e_1, e_2, T) \ \Downarrow\ R}$$

Extensionality

chk-eq-ext-prod

$$\frac{\Gamma \vdash \lambda x\!:\!T\,.\left(\begin{array}{l}\mathsf{let}\ X_1 = \mathsf{app}(e_1, \langle x \rangle)\ \mathsf{in}\\ \quad \mathsf{let}\ X_2 = \mathsf{app}(e_2, \langle x \rangle)\ \mathsf{in}\\ \quad\quad \mathsf{equivTerm}(X_1, X_2, \langle U \rangle)\end{array}\right) \Downarrow\ R}{\Gamma \vdash \mathsf{equivExtTerm}(e_1, e_2, \langle \prod_{(x:T)} U \rangle) \ \Downarrow\ R}$$

chk-eq-ext-unit

$$\frac{}{\Gamma \vdash \mathsf{equivExtTerm}(e_1, e_2, \langle \mathsf{Unit} \rangle) \ \Downarrow\ \mathsf{val}\,()}$$

chk-eq-ext-K

$$\frac{}{\Gamma \vdash \mathsf{equivExtTerm}(e_1, e_2, \langle \mathsf{Id}_T(e_3, e_4) \rangle) \ \Downarrow\ \mathsf{val}\,()}$$

chk-eq-ext-whnf

$$\frac{\begin{array}{c}\Gamma \vdash e_1 \rightsquigarrow^{*} e_1' \not\rightsquigarrow \qquad \Gamma \vdash e_2 \rightsquigarrow^{*} e_2' \not\rightsquigarrow\\ \Gamma \vdash \mathsf{equivWhnf}(\langle e_1' \rangle, \langle e_2' \rangle) \ \Downarrow\ R\end{array}}{\Gamma \vdash \mathsf{equivExtTerm}(e_1, e_2, e_3) \ \Downarrow\ R}$$

In chk-eq-ext-whnf, we might want to check whether e_1' and e_2' are the same expressions before invoking the general comparison function.

Whnf equivalence

chk-eq-whnf-reflexivity

$$\frac{}{\Gamma \vdash \mathsf{equivWhnf}(\langle B \rangle, \langle B \rangle) \ \Downarrow\ \mathsf{val}\,()}$$

chk-eq-whnf-app

$$\frac{\Gamma \vdash \left(\begin{array}{l}\mathsf{let}\ X_1 = \mathsf{equivTy}(\langle T_1 \rangle, \langle U_1 \rangle)\ \mathsf{in}\\ \ \mathsf{check}\ T_1 \equiv T_2\ \mathsf{using}\ X_1\ \mathsf{in}\\ \quad \mathsf{let}\ X_2 = \lambda x\!:\!T_1\,.\ \begin{array}{l}\mathsf{let}\ X' = \mathsf{equivTy}(\langle T_2 \rangle, \langle U_2 \rangle)\ \mathsf{in}\\ \quad \mathsf{check}\ T_2 \equiv U_2\ \mathsf{using}\ X'\ \mathsf{in}\\ \quad \mathsf{val}\ X'\end{array} \quad \mathsf{in}\\ \quad\ \mathsf{let}\ X_3 = \mathsf{equivWhnf}(\langle B_1 \rangle, \langle B_1' \rangle)\ \mathsf{in}\\ \quad\quad \mathsf{let}\ X_4 = \mathsf{equivTerm}(B_2, B_2', T_1)\ \mathsf{in}\\ \quad\quad\ X_1 \mathbin{++} X_2 \mathbin{++} X_3 \mathbin{++} X_4\end{array}\right) \Downarrow\ R}{\Gamma \vdash \mathsf{equivWhnf}(\langle B_1 \mathbin{@^{x:T_1.T_2}} B_2 \rangle, \langle B_1' \mathbin{@^{x:U_1.U_2}} B_2' \rangle) \ \Downarrow\ R}$$

chk-eq-whnf-idpath

$$\frac{\Gamma \vdash \left(\begin{array}{l}\mathsf{let}\ X_1 = \mathsf{equivTy}(\langle\, T\,\rangle, \langle\, U\,\rangle)\ \mathsf{in}\\ \quad \mathsf{check}\ T \equiv U\ \mathsf{using}\ X_1\ \mathsf{in}\\ \quad\quad \mathsf{let}\ X_2 = \mathsf{equivTerm}(B_1, B_2, T)\ \mathsf{in}\\ \quad\quad\quad X_1 \mathbin{++} X_2\end{array}\right) \Downarrow\ R}{\Gamma \vdash \mathsf{equivWhnf}(\langle\, \mathsf{idpath}_T\ B_1\,\rangle, \langle\, \mathsf{idpath}_U\ B_2\,\rangle)\ \Downarrow\ R}$$

chk-eq-whnf-j

$$\frac{\Gamma \vdash \left(\begin{array}{l}\mathsf{let}\ X_1 = \mathsf{equivTy}(\langle\, T\,\rangle, \langle\, T'\,\rangle)\ \mathsf{in}\\ \mathsf{check}\ T \equiv T'\ \mathsf{using}\ X_1\ \mathsf{in}\\ \mathsf{let}\ X_2 = \lambda x : T \,.\, \lambda y : T \,.\, \lambda p : \mathsf{Paths}_T(x,y)\,.\ \mathsf{in}\\ \qquad\qquad \mathsf{let}\ X' = \mathsf{equivTy}(\langle\, U\,\rangle, \langle\, U'\,\rangle)\ \mathsf{in}\\ \qquad\qquad \mathsf{check}\ U \equiv U'\ \mathsf{using}\ X'\ \mathsf{in}\\ \qquad\qquad \mathsf{val}\, X'\\ \mathsf{let}\ X_3 = \lambda z : T \,.\, \mathsf{equivTerm}(B_1, B_1', P[z/x, z/y, (\mathsf{idpath}_T\ z)/p])\ \mathsf{in}\\ \mathsf{let}\ X_4 = \mathsf{equivTerm}(B_3, B_3', T)\ \mathsf{in}\\ \mathsf{let}\ X_5 = \mathsf{equivTerm}(B_4, B_4', T)\ \mathsf{in}\\ \mathsf{let}\ X_6 = \mathsf{equivWhnf}(\langle\, B_2\,\rangle, \langle\, B_2'\,\rangle)\ \mathsf{in}\\ X_1 \mathbin{++} X_2 \mathbin{++} X_3 \mathbin{++} X_4 \mathbin{++} X_5 \mathbin{++} X_6\end{array}\right) \Downarrow\ R}{\Gamma \vdash \mathsf{equivWhnf}(\langle\, \mathsf{J}_T([x\,y\,p\,.\,U], [z\,.\,B_1], B_2, B_3, B_4)\,\rangle, \langle\, \mathsf{J}_{T'}([x\,y\,p\,.\,U'], [z\,.\,B_1'], B_2', B_3', B_4')\,\rangle)\ \Downarrow\ R}$$

chk-eq-whnf-refl

$$\frac{\Gamma \vdash \left(\begin{array}{l}\mathsf{let}\ X_1 = \mathsf{equivTy}(\langle\, T\,\rangle, \langle\, U\,\rangle)\ \mathsf{in}\\ \quad \mathsf{check}\ T \equiv U\ \mathsf{using}\ X_1\ \mathsf{in}\\ \quad\quad \mathsf{let}\ X_2 = \mathsf{equivTerm}(B_1, B_2, T)\ \mathsf{in}\\ \quad\quad\quad X_1 \mathbin{++} X_2\end{array}\right) \Downarrow\ R}{\Gamma \vdash \mathsf{equivWhnf}(\langle\, \mathsf{refl}_T\ B_1\,\rangle, \langle\, \mathsf{refl}_U\ B_2\,\rangle)\ \Downarrow\ R}$$

Whnf equivalence of names

chk-eq-whnf-prod

$$\frac{\alpha = \alpha' \qquad \beta = \beta' \qquad \Gamma \vdash \left(\begin{array}{l} \mathsf{let}\ X_1 = \mathsf{equivTerm}(B_1, B_1', \mathbb{U}_\alpha)\ \mathsf{in} \\ \quad \mathsf{check}\ \mathsf{El}^\alpha\, B_1 \equiv \mathsf{El}^\alpha\, B_1'\ \mathsf{using}\ X_1\ \mathsf{in} \\ \quad\quad \mathsf{let}\ X_2 = \lambda x : \mathsf{El}^\alpha\, B_1\, .\, \mathsf{equivTerm}(e_2, B_2', \mathbb{U}_\beta)\ \mathsf{in} \end{array}\right) \Downarrow R}{\Gamma \vdash \mathsf{equivWhnf}(\langle\, \pi^{\alpha,\beta} x : B_1\, .\, B_2\, \rangle, \langle\, \pi^{\alpha',\beta'} x : B_1'\, .\, B_2'\, \rangle) \Downarrow R}$$

chk-eq-whnf-universe

$$\frac{\alpha = \beta}{\Gamma \vdash \mathsf{equivWhnf}(\langle\, u_\alpha\, \rangle, \langle\, u_\beta\, \rangle) \Downarrow \mathsf{val}\,()}$$

chk-eq-whnf-paths

$$\frac{\alpha = \alpha' \qquad \Gamma \vdash \left(\begin{array}{l} \mathsf{let}\ X_1 = \mathsf{equivTerm}(B_1, B_1', \mathbb{U}_\alpha)\ \mathsf{in} \\ \quad \mathsf{check}\ \mathsf{El}^\alpha\, B_1 \equiv \mathsf{El}^\alpha\, B_1'\ \mathsf{using}\ X_1\ \mathsf{in} \\ \quad\quad \mathsf{let}\ X_2 = \mathsf{equivTerm}(B_2, B_2', \mathsf{El}^\alpha\, B_1)\ \mathsf{in} \\ \quad\quad\quad \mathsf{let}\ X_3 = \mathsf{equivTerm}(B_3, B_3', \mathsf{El}^\alpha\, B_1)\ \mathsf{in} \\ \quad\quad\quad\quad X_1 \mathbin{++} X_2 \mathbin{++} X_3 \end{array}\right) \Downarrow R}{\Gamma \vdash \mathsf{equivWhnf}(\langle\, \mathsf{paths}^\alpha_{B_1}(B_2, B_3)\, \rangle, \langle\, \mathsf{paths}^{\alpha'}_{B_1'}(B_2', B_3')\, \rangle) \Downarrow R}$$

chk-eq-whnf-id

$$\frac{\alpha = \alpha' \qquad \Gamma \vdash \left(\begin{array}{l} \mathsf{let}\ X_1 = \mathsf{equivTerm}(B_1, B_1', \mathbb{U}_\alpha)\ \mathsf{in} \\ \quad \mathsf{check}\ \mathsf{El}^\alpha\, B_1 \equiv \mathsf{El}^\alpha\, B_1'\ \mathsf{using}\ X_1\ \mathsf{in} \\ \quad\quad \mathsf{let}\ X_2 = \mathsf{equivTerm}(B_2, B_2', \mathsf{El}^\alpha\, B_1)\ \mathsf{in} \\ \quad\quad\quad \mathsf{let}\ X_3 = \mathsf{equivTerm}(B_3, B_3', \mathsf{El}^\alpha\, B_1)\ \mathsf{in} \\ \quad\quad\quad\quad \mathsf{val}\,(X_1 \mathbin{++} X_2 \mathbin{++} X_3) \end{array}\right) \Downarrow R}{\Gamma \vdash \mathsf{equivWhnf}(\langle\, \mathsf{id}^\alpha_{B_1}(B_2, B_3)\, \rangle, \langle\, \mathsf{id}^{\alpha'}_{B_1'}(B_2', B_3')\, \rangle) \Downarrow R}$$

chk-eq-whnf-coerce

$$\frac{\alpha = \alpha' \qquad \beta = \beta' \qquad \Gamma \vdash \mathsf{equivTerm}(B_1, B_1', \mathbb{U}_\alpha) \Downarrow R}{\Gamma \vdash \mathsf{equivWhnf}(\langle\, \mathsf{coerce}^{\alpha \mapsto \beta}\, B_1\, \rangle, \langle\, \mathsf{coerce}^{\alpha' \mapsto \beta'}\, B_1'\, \rangle) \Downarrow R}$$

3 Well-Formedness

Expressions

ok-galactic-term
$$\frac{\Gamma \vdash B : T}{\Gamma \vdash \langle B \rangle \ \ \mathsf{ok}}$$

ok-galactic-type
$$\frac{\Gamma \vdash T \ \mathsf{type}}{\Gamma \vdash \langle T \rangle \ \ \mathsf{ok}}$$

ok-tt-var
$$\frac{}{\Gamma \vdash X \ \ \mathsf{ok}}$$

ok-cont
$$\frac{\Gamma \vdash \mathcal{K} \ : \ \Delta \to \mathsf{ok}}{\Gamma \vdash \mathsf{cont}(\Gamma, \Delta, \mathcal{K}) \ \ \mathsf{ok}}$$

ok-fun
$$\frac{\Gamma \vdash C \ \ \mathsf{ok}}{\Gamma \vdash \mathsf{fun}\ X \Rightarrow C \ \ \mathsf{ok}}$$

ok-const
$$\frac{}{\Gamma \vdash c \ \ \mathsf{ok}}$$

ok-tuple
$$\frac{\Gamma \vdash e_1 \ \ \mathsf{ok} \qquad \cdots \qquad \Gamma \vdash e_n \ \ \mathsf{ok}}{\Gamma \vdash (e_1, \ldots, e_n) \ \ \mathsf{ok}}$$

ok-inj
$$\frac{\Gamma \vdash e \ \ \mathsf{ok}}{\Gamma \vdash \mathsf{inj}_i\, e \ \ \mathsf{ok}}$$

ok-handler
$$\frac{\Gamma \vdash C_v \ \ \mathsf{ok} \qquad \Gamma \vdash C_1 \ \ \mathsf{ok} \qquad \cdots \qquad \Gamma \vdash C_n \ \ \mathsf{ok}}{\Gamma \vdash \mathsf{handler}\ \mathsf{val}\ X \mapsto C_v \mid (\mathsf{op}_i(X, K) \mapsto C_n)_{i=1}^{n} \ \ \mathsf{ok}}$$

Results

ok-result-val
$$\frac{\Gamma \vdash e \ \ \mathsf{ok}}{\Gamma \vdash \mathsf{val}\, e \ \mathsf{ok}}$$

ok-result-op
$$\frac{\Gamma, \Delta \vdash T \ \mathsf{type} \qquad \Gamma \vdash K \ : \ \Delta \to \mathsf{ok}}{\Gamma \vdash \mathsf{op}_i(\Delta, T, K) \ \mathsf{ok}}$$

Continuations

ok-hole
$$\frac{}{\Gamma \vdash \diamond \ : \ \bullet \to \mathsf{ok}}$$

ok-let-cont
$$\frac{\Gamma \vdash K_1 \ : \ \Delta \to \mathsf{ok} \qquad \Gamma \vdash C_2 \ \ \mathsf{ok}}{\Gamma \vdash (\mathsf{let}\ X = K_1\ \mathsf{in}\ C_2) \ : \ \Delta \to \mathsf{ok}}$$

ok-handle-cont
$$\frac{\Gamma \vdash e_1 \ \ \mathsf{ok} \qquad \Gamma \vdash K_2 \ : \ \Delta \to \mathsf{ok}}{\Gamma \vdash (\mathsf{with}\ e_1\ \mathsf{handle}\ K_2) \ : \ \Delta \to \mathsf{ok}}$$

ok-lam-cont
$$\frac{x, T_1 : \vdash K_2 \ : \ \Delta \to \mathsf{ok}}{\Gamma \vdash (\lambda x : T_1 \,.\, K_2) \ : \ ([, \Delta :]x T_1) \to \mathsf{ok}}$$

Computations

$$\frac{\Gamma \vdash e \ \ \mathsf{ok}}{\Gamma \vdash \mathsf{val}\, e \ \ \mathsf{ok}}\ \text{ok-val} \qquad \frac{\Gamma \vdash e_1 \ \ \mathsf{ok} \qquad \Gamma \vdash e_2 \ \ \mathsf{ok}}{\Gamma \vdash e_1\, e_2 \ \ \mathsf{ok}}\ \text{ok-app} \qquad \frac{\Gamma \vdash C_1 \ \ \mathsf{ok} \qquad \Gamma \vdash C_2 \ \ \mathsf{ok}}{\Gamma \vdash (\mathsf{let}\ X = C_1\ \mathsf{in}\ C_2) \ \ \mathsf{ok}}\ \text{ok-let}$$

$$\frac{\Gamma \vdash e \ \ \mathsf{ok}}{\Gamma \vdash \mathsf{op}_i\, e \ \ \mathsf{ok}}\ \text{ok-op} \qquad \frac{\Gamma \vdash e_1 \ \ \mathsf{ok} \qquad \Gamma \vdash C_2 \ \ \mathsf{ok}}{\Gamma \vdash (\mathsf{with}\ e_1\ \mathsf{handle}\ C_2) \ \ \mathsf{ok}}\ \text{ok-handle} \qquad \frac{\Gamma \vdash e_1 \ \ \mathsf{ok} \qquad \Gamma \vdash e_2 \ \ \mathsf{ok}}{\Gamma \vdash e_1[e_2] \ \ \mathsf{ok}}\ \text{ok-cont-app}$$

$$\frac{\Gamma \vdash e_1 \ \ \mathsf{ok} \qquad \Gamma \vdash e_2 \ \ \mathsf{ok}}{\Gamma \vdash e_1 :: e_2 \ \ \mathsf{ok}}\ \text{ok-ascribe} \qquad \frac{\Gamma \vdash e_1 \ \ \mathsf{ok} \qquad \cdots \Gamma \vdash e_n \ \ \mathsf{ok}}{\Gamma \vdash f(e_1, \ldots, e_n) \ \ \mathsf{ok}}\ \text{ok-prim}$$

$$\frac{\Gamma \vdash e \ \ \mathsf{ok} \qquad \Gamma \vdash C_1 \ \ \mathsf{ok} \qquad \cdots \qquad \Gamma \vdash C_n \ \ \mathsf{ok}}{\Gamma \vdash \mathsf{match}\ e\ \mathsf{with}\ (P_i \Rightarrow C_i)_{i=1}^{n} \ \ \mathsf{ok}}\ \text{ok-match} \qquad \frac{}{\Gamma \vdash (\mathsf{debruijn}\, n) \ \ \mathsf{ok}}\ \text{ok-make-var}$$

$$\frac{\Gamma \vdash e_1 \ \ \mathsf{ok} \qquad \Gamma \vdash e_2 \ \ \mathsf{ok}}{\Gamma \vdash \mathsf{app}(e_1, e_2) \ \ \mathsf{ok}}\ \text{ok-make-app} \qquad \frac{\Gamma \vdash e_1 \ \ \mathsf{ok} \qquad \Gamma \vdash C_2 \ \ \mathsf{ok}}{\Gamma \vdash (\lambda x : e_1 \,.\, C_2) \ \ \mathsf{ok}}\ \text{ok-make-lam1}$$

$$\frac{\Gamma \vdash e_1 \ \ \mathsf{ok} \qquad x, T_1 : \vdash C_2 \ \ \mathsf{ok}}{\Gamma \vdash (\lambda x : \langle\, T_1 \,\rangle \,.\, C_2) \ \ \mathsf{ok}}\ \text{ok-make-lam2} \qquad \frac{\Gamma \vdash e \ \ \mathsf{ok} \qquad \text{if } (\Gamma \vdash T_1 \equiv T_2) \text{ then } (\Gamma \vdash C \ \ \mathsf{ok})}{\Gamma \vdash \mathsf{check}\ T_1 \equiv T_2\ \mathsf{using}\ e\ \mathsf{in}\ C \ \ \mathsf{ok}}\ \text{eval-assert-type}$$

Lemma 1 (Substitution)

1. If $\Gamma \vdash e \ \ \mathsf{ok}$ and $\Gamma \vdash e' \ \ \mathsf{ok}$ then $\Gamma \vdash e[e'/X] \ \ \mathsf{ok}$.
2. If $\Gamma \vdash C \ \ \mathsf{ok}$ and $\Gamma \vdash e' \ \ \mathsf{ok}$ then $\Gamma \vdash C[e'/X] \ \ \mathsf{ok}$.
3. If $\Gamma \vdash \mathcal{K} \ : \ \Delta \to \mathsf{ok}$ and $\Gamma \vdash e' \ \ \mathsf{ok}$ then $\Gamma \vdash \mathcal{K}[e'/X] \ : \ \Delta \to \mathsf{ok}$.

Lemma 2 (Continuation Invocation) If $\Gamma \vdash \mathcal{K} \ : \ \Delta \to \mathsf{ok}$ and $\Gamma, \Delta \vdash e \ \ \mathsf{ok}$ then $\Gamma \vdash \mathcal{K}[\diamond := e] \ \ \mathsf{ok}$.

Lemma 3 (Preservation) If $\Gamma\ \mathsf{ctx}$ and $\Gamma \vdash C \ \ \mathsf{ok}$ and $\Gamma \vdash C \ \Downarrow\ R$ then $\Gamma \vdash R\ \mathsf{ok}$.