A solution to the PoplMark challenge in Isabelle/HOL

Stefan Berghofer
Technische Universität München

Important features

- Encoding of variables via de Bruijn indices
- Covers records
- Encoding of evaluation relation
  1. Using additional congruence rules
  2. Using evaluation contexts
- Animation of evaluation relation via ML code generator (only for version using additional congruence rules)

Syntax of System $F_{<:}$

Types

$$
\begin{array}{ll}
\textbf{datatype}\ \mathit{type} = & \\
\quad \mathit{TVar\ nat} & \\
\ |\ \mathit{Top} & \\
\ |\ \mathit{Fun\ type\ type} & (\textbf{infixr} \rightarrow \mathit{200}) \\
\ |\ \mathit{TyAll\ type\ type} & ((\mathit{3}\forall <:\text{-./ -})\ [\mathit{0},\ \mathit{10}]\ \mathit{10})
\end{array}
$$

Terms

$$
\begin{array}{ll}
\textbf{datatype}\ \mathit{trm} = & \\
\quad \mathit{Var\ nat} & \\
\ |\ \mathit{Abs\ type\ trm} & ((\mathit{3}\lambda\text{:-./ -})\ [\mathit{0},\ \mathit{10}]\ \mathit{10}) \\
\ |\ \mathit{TAbs\ type\ trm} & ((\mathit{3}\lambda<:\text{-./ -})\ [\mathit{0},\ \mathit{10}]\ \mathit{10}) \\
\ |\ \mathit{App\ trm\ trm} & (\textbf{infixl}\ \cdot\ \mathit{200}) \\
\ |\ \mathit{TApp\ trm\ type} & (\textbf{infixl}\ \cdot_{\tau}\ \mathit{200})
\end{array}
$$

Notation

$\Gamma \vdash S <: T$	Type S is subtye of T in context Γ
$\Gamma \vdash t : T$	Term t has type T in context Γ
$\Gamma \vdash_{wf}$	Context Γ is well-formed
$\Gamma \vdash_{wf} T$	Type T is well-formed in context Γ

Contexts

List of bindings for term and type variables

datatype $binding = VarB\ type \mid TVarB\ type$

types $env = binding\ list$

- Variable with index i corresponds i-th element of list (denoted by $\Gamma\langle i\rangle$)
- Types in Γ may refer to type variables "further to the right"
- New elements are appended to the left using $b \# \Gamma$
- Concatenation of contexts using $\Delta @ \Gamma$

Lifting and Substitution

$\uparrow_\tau\ n\ k\ T$	Increment free variables $\geq k$ in type T by n
$\uparrow\ n\ k\ t$	Increment free variables $\geq k$ in term t by n
$\uparrow_e\ n\ k\ \Gamma$	Increment free variables $\geq k$ in environment Γ by n
$T[k \mapsto_\tau S]_\tau$	Substitute type S for type variable with index k in type T
$t[k \mapsto_\tau S]$	Substitute type S for type variable with index k in term t
$t[k \mapsto s]$	Substitute term s for term variable with index k in term t
$\Gamma[k \mapsto_\tau T]_e$	Substitute type T for type variable with index k in environment Γ

Some equations

$$\uparrow n\ k\ (\mathit{Var}\ i) = (\mathit{if}\ i < k\ \mathit{then}\ \mathit{Var}\ i\ \mathit{else}\ \mathit{Var}\ (i + n))$$
$$\uparrow n\ k\ (\lambda{:}T.\ t) = (\lambda{:}\uparrow_\tau n\ k\ T.\ \uparrow n\ (k + \mathit{1})\ t)$$

$$(\mathit{Var}\ i)[k \mapsto s] = (\mathit{if}\ k < i\ \mathit{then}\ \mathit{Var}\ (i - \mathit{1})\ \mathit{else}\ \mathit{if}\ i = k\ \mathit{then}\ \uparrow k\ \mathit{0}\ s\ \mathit{else}\ \mathit{Var}\ i)$$
$$(\lambda{:}T.\ t)[k \mapsto s] = (\lambda{:}T[k \mapsto_\tau \mathit{Top}]_\tau.\ t[k{+}\mathit{1} \mapsto s])$$

$$[][k \mapsto_\tau T]_e = []$$
$$(B \# \Gamma)[k \mapsto_\tau T]_e = \mathit{mapB}\ (\lambda U.\ U[k + \|\Gamma\| \mapsto_\tau T]_\tau)\ B \# \Gamma[k \mapsto_\tau T]_e$$

Well-formedness of Types and Contexts

Intuition:

- A type is well-formed in a context, if all its free variables appear in the context.
- A context is well-formed, if all types only refer to type variables "further to the right"

$$\frac{\Gamma\langle i\rangle = \lfloor TVarB\ T\rfloor}{\Gamma \vdash_{wf} TVar\ i} \qquad \Gamma \vdash_{wf} Top$$

$$\frac{\Gamma \vdash_{wf} T \quad \Gamma \vdash_{wf} U}{\Gamma \vdash_{wf} T \rightarrow U} \qquad \frac{\Gamma \vdash_{wf} T \quad TVarB\ T \ \#\ \Gamma \vdash_{wf} U}{\Gamma \vdash_{wf} (\forall <: T.\ U)}$$

$$[] \vdash_{wf} \qquad \frac{\Gamma \vdash_{wf} type\text{-}ofB\ B \quad \Gamma \vdash_{wf}}{B\ \#\ \Gamma \vdash_{wf}}$$

Important property:

All terms and contexts involved in (sub)typing judgements are well-formed, i.e.

if $\Gamma \vdash S <: T$, then $\Gamma \vdash_{wf}$, $\Gamma \vdash_{wf} S$, $\Gamma \vdash_{wf} T$
if $\Gamma \vdash t : T$, then $\Gamma \vdash_{wf}$, $\Gamma \vdash_{wf} T$

Subtyping Relation

$$\frac{\Gamma \vdash_{wf} \quad \Gamma \vdash_{wf} S}{\Gamma \vdash S <: Top}$$

$$\frac{\Gamma \vdash_{wf} \quad \Gamma \vdash_{wf} TVar\ i}{\Gamma \vdash TVar\ i <: TVar\ i}$$

$$\frac{\Gamma\langle i \rangle = \lfloor TVarB\ U \rfloor \quad \Gamma \vdash \uparrow_\tau (Suc\ i)\ 0\ U <: T}{\Gamma \vdash TVar\ i <: T}$$

$$\frac{\Gamma \vdash T_1 <: S_1 \quad \Gamma \vdash S_2 <: T_2}{\Gamma \vdash S_1 \rightarrow S_2 <: T_1 \rightarrow T_2}$$

$$\frac{\Gamma \vdash T_1 <: S_1 \quad TVarB\ T_1 \# \Gamma \vdash S_2 <: T_2}{\Gamma \vdash (\forall <: S_1.\ S_2) <: (\forall <: T_1.\ T_2)}$$

Typing relation

$$\frac{\Gamma \vdash_{wf} \quad \Gamma\langle i\rangle = \lfloor \mathit{VarB}\ U \rfloor \quad T = \uparrow_\tau (\mathit{Suc}\ i)\ 0\ U}{\Gamma \vdash \mathit{Var}\ i : T}$$

$$\frac{\mathit{VarB}\ T_1 \# \Gamma \vdash t_2 : T_2}{\Gamma \vdash (\lambda{:}T_1.\ t_2) : T_1 \rightarrow T_2[0 \mapsto_\tau \mathit{Top}]_\tau}$$

$$\frac{\Gamma \vdash t_1 : T_{11} \rightarrow T_{12} \quad \Gamma \vdash t_2 : T_{11}}{\Gamma \vdash t_1 \cdot t_2 : T_{12}}$$

$$\frac{\mathit{TVarB}\ T_1 \# \Gamma \vdash t_2 : T_2}{\Gamma \vdash (\lambda{<:}T_1.\ t_2) : (\forall{<:}T_1.\ T_2)}$$

$$\frac{\Gamma \vdash t_1 : (\forall{<:}T_{11}.\ T_{12}) \quad \Gamma \vdash T_2 <: T_{11}}{\Gamma \vdash t_1 \cdot_\tau T_2 : T_{12}[0 \mapsto_\tau T_2]_\tau}$$

$$\frac{\Gamma \vdash t : S \quad \Gamma \vdash S <: T}{\Gamma \vdash t : T}$$

Typing relation – Issues

- In rule for variables, indices in type have to be incremented at lookup
- In rule for abstraction over term variables, indices in result type have to be decremented (by applying a dummy substitution), since the type cannot contain term variables.
  Alternative solution: Use different lookup function in typing rule for variables (as in Jerome Vouillon's solution)

Evaluation relation – using congruence rules

Values

$$(\lambda{:}T.\ t) \in \mathit{value}$$
$$(\lambda{<:}T.\ t) \in \mathit{value}$$

Evaluation rules

$$\frac{v_2 \in \mathit{value}}{(\lambda{:}T_{11}.\ t_{12}) \cdot v_2 \longmapsto t_{12}[\mathit{0} \mapsto v_2]}$$

$$(\lambda{<:}T_{11}.\ t_{12}) \cdot_\tau T_2 \longmapsto t_{12}[\mathit{0} \mapsto_\tau T_2]$$

Congruence rules

$$\frac{t \longmapsto t'}{t \cdot u \longmapsto t' \cdot u} \qquad \frac{v \in \mathit{value} \quad t \longmapsto t'}{v \cdot t \longmapsto v \cdot t'}$$

$$\frac{t \longmapsto t'}{t \cdot_\tau T \longmapsto t' \cdot_\tau T}$$

Evaluation relation – using contexts

Evaluation contexts

$$(\lambda t.\ t) \in \mathit{ctxt}$$

$$\frac{E \in \mathit{ctxt}}{(\lambda t.\ E\ t \cdot u) \in \mathit{ctxt}} \qquad \frac{v \in \mathit{value} \quad E \in \mathit{ctxt}}{(\lambda t.\ v \cdot E\ t) \in \mathit{ctxt}}$$

$$\frac{E \in \mathit{ctxt}}{(\lambda t.\ E\ t \cdot_\tau\ T) \in \mathit{ctxt}}$$

Evaluation rules

$$\frac{t \longmapsto t' \quad E \in \mathit{ctxt}}{E\ t \longmapsto E\ t'}$$

$$\frac{v_2 \in \mathit{value}}{(\lambda{:}T_{11}.\ t_{12}) \cdot v_2 \longmapsto t_{12}[\mathit{0} \mapsto v_2]}$$

$$(\lambda{<:}T_{11}.\ t_{12}) \cdot_\tau\ T_2 \longmapsto t_{12}[\mathit{0} \mapsto_\tau\ T_2]$$

Important properties

Weakening

$\Gamma \vdash t : T \Longrightarrow \Delta \ @\ \Gamma \vdash_{wf} \Longrightarrow \Delta \ @\ \Gamma \vdash \uparrow \|\Delta\|\ \mathit{0}\ t : \uparrow_\tau \|\Delta\|\ \mathit{0}\ T$

Substitution lemma

$\Delta \ @\ \mathit{VarB}\ U \ \#\ \Gamma \vdash t : T \Longrightarrow \Gamma \vdash u : U \Longrightarrow$
$\Delta[\mathit{0} \mapsto_\tau \mathit{Top}]_e \ @\ \Gamma \vdash t[\|\Delta\| \mapsto u] : T[\|\Delta\| \mapsto_\tau \mathit{Top}]_\tau$

$\Delta \ @\ \mathit{TVarB}\ Q \ \#\ \Gamma \vdash S <: T \Longrightarrow \Gamma \vdash P <: Q \Longrightarrow$
$\Delta[\mathit{0} \mapsto_\tau P]_e \ @\ \Gamma \vdash S[\|\Delta\| \mapsto_\tau P]_\tau <: T[\|\Delta\| \mapsto_\tau P]_\tau$

Type safety

$t \longmapsto t' \Longrightarrow \Gamma \vdash t : T \Longrightarrow \Gamma \vdash t' : T$ Preservation / Subject Reduction

$[] \vdash t : T \Longrightarrow t \in \mathit{value} \vee (\exists t'.\ t \longmapsto t')$ Progress

Properties of evaluation contexts

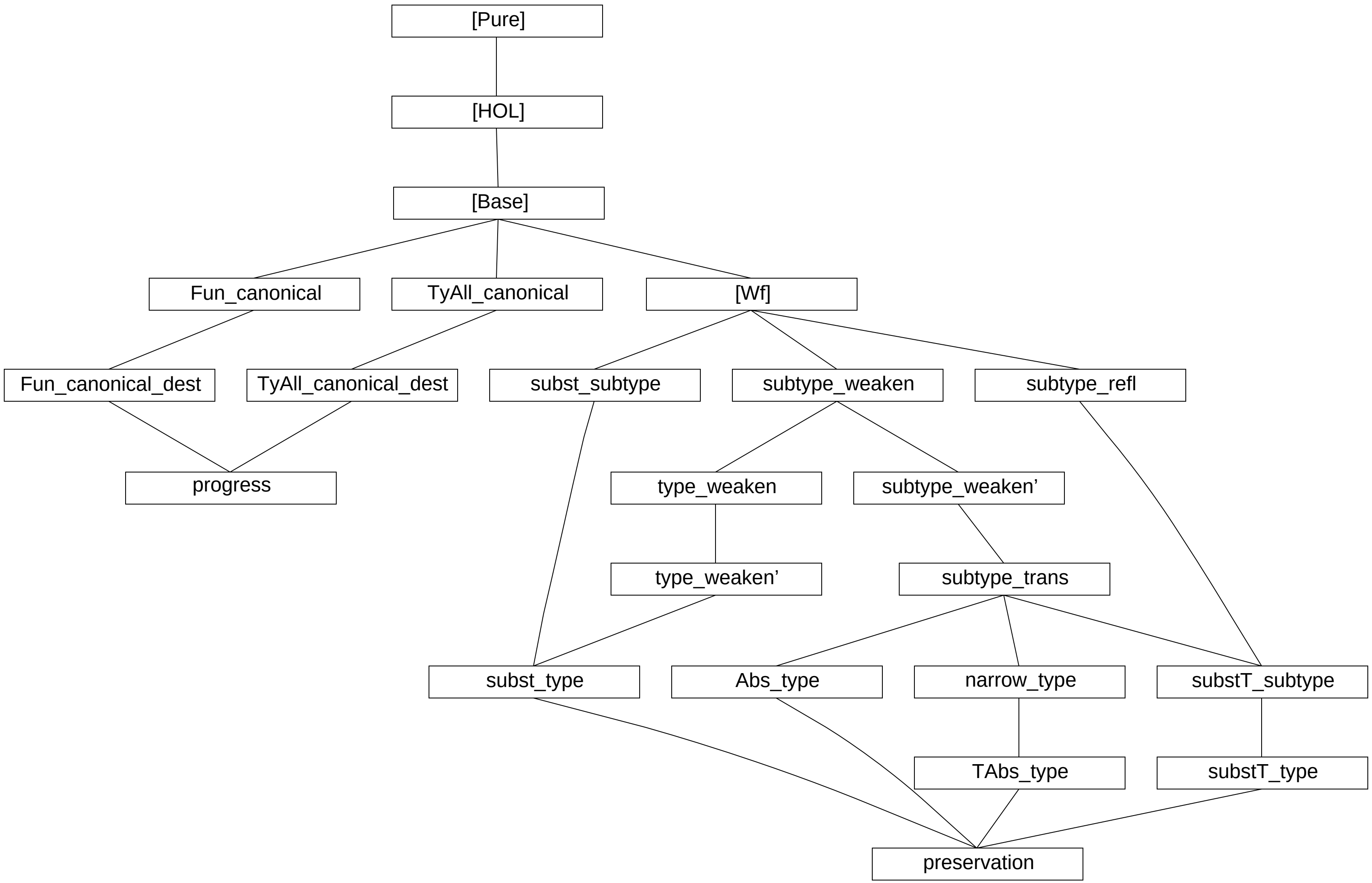
Decomposition

$[] \vdash t : T \Longrightarrow t \in \mathit{value} \vee (\exists E\ t_0\ t_0'.\ E \in \mathit{ctxt} \wedge t = E\ t_0 \wedge t_0 \longmapsto t_0')$

Typing

$\Gamma \vdash E\ t : T \Longrightarrow E \in \mathit{ctxt} \Longrightarrow (\bigwedge T_0.\ \Gamma \vdash t : T_0 \Longrightarrow \Gamma \vdash t' : T_0) \Longrightarrow \Gamma \vdash E\ t' : T$

Theorem dependencies



Executability

Idea

- Translate PROLOG-style inductive definitions to functional program (e.g. in ML) yielding sequence of possible outputs for a given input
- Requires mode analysis (see [Berghofer, Nipkow, TYPES 2000])
- Implementation using "backtracking monad"

  ```
  fun s :-> f = Seq.flat (Seq.map f s);
  ```

```
fun eval__1 inp =
  Seq.single inp :->
    (fn (App (Abs (T_1_1, t_1_2), v_2)) =>
      value__1 (v_2) :-> (fn () => Seq.single (subst t_1_2 0 v_2))
      | _ => Seq.empty) ++
  Seq.single inp :->
    (fn (TApp (TAbs (T_1_1, t_1_2), T_2)) => Seq.single (substT t_1_2 0 T_2)
      | _ => Seq.empty) ++
  Seq.single inp :->
    (fn (App (t, u)) => eval__1 (t) :-> (fn (t') => Seq.single (App (t', u)))
      | _ => Seq.empty) ++ ...
```

Records

- Records are modelled as association lists mapping field names to terms

  types
  $rcd = (name \times trm)\ list$
- Record types are modelled as association lists mapping field names to types

  types
  $rcdT = (name \times type)\ list$
- LET expressions can be treated like nested abstractions

$$LET\ \{l_1 = x_1, \ldots, l_n = x_n\} = \{l_1 = v_1, \ldots, l_n = v_n\}\ IN\ t$$
$$\approx$$
$$(\lambda x_1, \ldots, x_n.\ t) \cdot v_1 \cdot \cdots \cdot v_n$$

- Pattern typing judgement yields list of term variable bindings ($VarB$)
- Pattern matching judgement yields list of terms

More notation for records

Association list lookup

$[]\langle a\rangle_? = \bot$
$(x \mathbin{\#} xs)\langle a\rangle_? = (\mathit{if}\ \mathit{fst}\ x = a\ \mathit{then}\ \lfloor \mathit{snd}\ x \rfloor\ \mathit{else}\ xs\langle a\rangle_?)$

Uniqueness of keys in association lists

$\mathit{unique}\ [] = \mathit{True}$
$\mathit{unique}\ (x \mathbin{\#} xs) = (xs\langle \mathit{fst}\ x\rangle_? = \bot \wedge \mathit{unique}\ xs)$

New constructors for records

Types

datatype $type =$

$\quad \ldots$

$\quad | \; RcdT \; (name \times type) \; list$

Patterns

datatype $pat = PVar \; type \mid PRcd \; (name \times pat) \; list$

Terms

datatype $trm =$

$\quad \ldots$

$\quad | \; Rcd \; (name \times trm) \; list$

$\quad | \; Proj \; trm \; name \;\; ((\text{-..-}) \; [90, \, 91] \; 90)$

$\quad | \; LET \; pat \; trm \; trm \; ((LET \; (\text{-} =\!/ \; \text{-})/ \; IN \; (\text{-})) \; 10)$

Well-formedness and subtyping of record types

Well-formedness

$$\frac{\mathit{unique}\ \mathit{fs} \quad \forall\,(l,\ T)\in \mathit{set}\ \mathit{fs}.\ \Gamma \vdash_{\mathit{wf}}\ T}{\Gamma \vdash_{\mathit{wf}} \mathit{RcdT}\ \mathit{fs}}$$

Subtyping

$$\frac{\Gamma \vdash_{\mathit{wf}} \quad \Gamma \vdash_{\mathit{wf}} \mathit{RcdT}\ \mathit{fs} \quad \mathit{unique}\ \mathit{fs}' \quad \forall\,(l,\ T)\in \mathit{set}\ \mathit{fs}'.\ \exists\,(k,\ S)\in \mathit{set}\ \mathit{fs}.\ k = l \wedge \Gamma \vdash\ S <:\ T}{\Gamma \vdash\ \mathit{RcdT}\ \mathit{fs} <: \mathit{RcdT}\ \mathit{fs}'}$$

Additional typing rules for records

$$\frac{\Gamma \vdash t_1 : T_1 \quad \vdash p : T_1 \Rightarrow \Delta \quad \Delta \mathbin{@} \Gamma \vdash t_2 : T_2}{\Gamma \vdash (\mathit{LET}\ p = t_1\ \mathit{IN}\ t_2) : \downarrow_\tau \|\Delta\|\ \mathit{0}\ T_2}$$

$$\frac{\Gamma \vdash \mathit{fs}\ [:]\ \mathit{fTs}}{\Gamma \vdash \mathit{Rcd}\ \mathit{fs} : \mathit{RcdT}\ \mathit{fTs}} \qquad \frac{\Gamma \vdash t : \mathit{RcdT}\ \mathit{fTs} \quad \mathit{fTs}\langle l\rangle_? = \lfloor T \rfloor}{\Gamma \vdash t..l : T}$$

$$\frac{\Gamma \vdash_{wf}}{\Gamma \vdash []\ [:]\ []} \qquad \frac{\Gamma \vdash t : T \quad \Gamma \vdash \mathit{fs}\ [:]\ \mathit{fTs} \quad \mathit{fs}\langle l\rangle_? = \bot}{\Gamma \vdash (l,\ t)\ \#\ \mathit{fs}\ [:]\ (l,\ T)\ \#\ \mathit{fTs}}$$

Pattern typing

$$\vdash \mathit{PVar}\ T : T \Rightarrow [\mathit{VarB}\ T] \qquad \frac{\vdash \mathit{fps}\ [:]\ \mathit{fTs} \Rightarrow \Delta}{\vdash \mathit{PRcd}\ \mathit{fps} : \mathit{RcdT}\ \mathit{fTs} \Rightarrow \Delta}$$

$$\vdash []\ [:]\ [] \Rightarrow [] \qquad \frac{\vdash p : T \Rightarrow \Delta_1 \quad \vdash \mathit{fps}\ [:]\ \mathit{fTs} \Rightarrow \Delta_2 \quad \mathit{fps}\langle l\rangle_? = \bot}{\vdash (l,\ p)\ \#\ \mathit{fps}\ [:]\ (l,\ T)\ \#\ \mathit{fTs} \Rightarrow \uparrow_e \|\Delta_1\|\ \mathit{0}\ \Delta_2 \mathbin{@} \Delta_1}$$

Additional evaluation rules for records

$$\frac{v \in value \quad \vdash p \rhd v \Rightarrow ts}{(LET\ p = v\ IN\ t) \longmapsto t[0 \mapsto_s ts]} \qquad \frac{fs\langle l \rangle_? = \lfloor v \rfloor \quad v \in value}{Rcd\ fs..l \longmapsto v}$$

Contexts

$$\frac{E \in ctxt}{(\lambda t.\ E\ t..l) \in ctxt} \qquad \frac{E \in rctxt}{(\lambda t.\ Rcd\ (E\ t)) \in ctxt} \qquad \frac{E \in ctxt}{(\lambda t.\ LET\ p = E\ t\ IN\ u) \in ctxt}$$

$$\frac{E \in ctxt}{(\lambda t.\ (l,\ E\ t)\ \#\ fs) \in rctxt} \qquad \frac{v \in value \quad E \in rctxt}{(\lambda t.\ (l,\ v)\ \#\ E\ t) \in rctxt}$$

Matching

$$\vdash PVar\ T \rhd t \Rightarrow [t] \qquad \frac{\vdash fps\ [\rhd]\ fs \Rightarrow ts}{\vdash PRcd\ fps \rhd Rcd\ fs \Rightarrow ts}$$

$$\vdash []\ [\rhd]\ fs \Rightarrow [] \qquad \frac{fs\langle l \rangle_? = \lfloor t \rfloor \quad \vdash p \rhd t \Rightarrow ts \quad \vdash fps\ [\rhd]\ fs \Rightarrow us}{\vdash (l,\ p)\ \#\ fps\ [\rhd]\ fs \Rightarrow ts\ @\ us}$$

Additional theorems for records

Matched patterns preserve types

$$\vdash p : T_1 \Rightarrow \Delta \Longrightarrow$$
$$\Gamma_2 \vdash t_1 : T_1 \Longrightarrow$$
$$\Gamma_1 \ @\ \Delta \ @\ \Gamma_2 \vdash t_2 : T_2 \Longrightarrow$$
$$\vdash p \rhd t_1 \Rightarrow ts \Longrightarrow \downarrow_e \|\Delta\| \ \mathit{0} \ \Gamma_1 \ @\ \Gamma_2 \vdash t_2[\|\Gamma_1\| \mapsto_s ts] : \downarrow_\tau \|\Delta\| \ \|\Gamma_1\| \ T_2$$

$$\vdash fps \ [:] \ fTs \Rightarrow \Delta \Longrightarrow$$
$$\Gamma_2 \vdash fs \ [:] \ fTs \Longrightarrow$$
$$\Gamma_1 \ @\ \Delta \ @\ \Gamma_2 \vdash t_2 : T_2 \Longrightarrow$$
$$\vdash fps \ [\rhd] \ fs \Rightarrow ts \Longrightarrow \downarrow_e \|\Delta\| \ \mathit{0} \ \Gamma_1 \ @\ \Gamma_2 \vdash t_2[\|\Gamma_1\| \mapsto_s ts] : \downarrow_\tau \|\Delta\| \ \|\Gamma_1\| \ T_2$$

Well-typed pattern matching is defined

$$\vdash p : T \Rightarrow \Delta \Longrightarrow [] \vdash t : T \Longrightarrow t \in value \Longrightarrow \exists\, ts. \vdash p \rhd t \Rightarrow ts$$

$$\vdash fps \ [:] \ fTs \Rightarrow \Delta \Longrightarrow$$
$$[] \vdash fs \ [:] \ fTs \Longrightarrow \forall\, (l,\ t) \in set\ fs.\ t \in value \Longrightarrow \exists\, us. \vdash fps \ [\rhd] \ fs \Rightarrow us$$