

Funcons-beta: Linking *

The P_LanCompS Project

Linking.cbs | PLAIN | PRETTY

Linking

```
[ Datatype links
  Funcon initialise-linking
  Funcon link
  Funcon fresh-link
  Funcon fresh-initialised-link
  Alias fresh-init-link
  Funcon set-link
  Funcon follow-if-link ]
```

Meta-variables $T <: \text{values}$

Datatype `links ::= link(_ : variables)`

Funcon `initialise-linking($X : \Rightarrow T$) : $\Rightarrow T$`
 `\rightsquigarrow initialise-storing(X)`

`initialise-linking(X)` ensures that the entities used by the funcons for linking are properly initialised.

Funcon `fresh-link(T) : $\Rightarrow \text{links}$`
 `\rightsquigarrow link(allocate-variable(T))`

Funcon `fresh-initialised-link($T, V : T$) : $\Rightarrow \text{links}$`
 `\rightsquigarrow link(allocate-initialised-variable(T, V))`

Alias `fresh-init-link = fresh-initialised-link`

Funcon `set-link(_ : links, _ : T) : $\Rightarrow \text{null-type}$`

The value of a link can be set only once.

Rule `set-link(link($\text{Var} : \text{variables}$), $V : T$) \rightsquigarrow initialise-variable(Var, V)`

*Suggestions for improvement: plancomps@gmail.com.
Reports of issues: <https://github.com/plancomps/CBS-beta/issues>.

Funcon $\text{follow-link}(_ : \text{links}) : \Rightarrow \text{values}$

Rule $\text{follow-link}(\text{link}(Var : \text{variables})) \rightsquigarrow \text{assigned}(Var)$

Funcon $\text{follow-if-link}(_ : \text{values}) : \Rightarrow \text{values}$

If V is a link, $\text{follow-if-link}(V)$ computes the set value, and otherwise it evaluates to V .

Rule $\text{follow-if-link}(\text{link}(Var : \text{variables})) \rightsquigarrow \text{assigned}(Var)$

Rule $\text{follow-if-link}(V : \sim \text{links}) \rightsquigarrow V$