

$$\begin{array}{ll}
\Pi & ::= \overline{\text{defmodtype } X \text{ do } \overline{P} \overline{D} \text{ end defmodule } Main \text{ do } \overline{B} \text{ end}} \\
N & ::= x \\
& \quad | X \\
S & ::= \$\text{behaviour } X \\
P & ::= \$\text{param } x \\
B & ::= \text{defmodule } X \text{ do } \overline{P} \overline{S} \overline{B} \text{ end} \\
& \quad | x = v \\
& \quad | \$\text{type } x = t \\
& \quad | \$\text{opaque } x = t \\
E & ::= v \\
& \quad | x \\
& \quad | E(\overline{E}, E) \\
& \quad | \% \{ \overline{\ell} = \overline{E} \} \\
& \quad | E.\ell \\
& \quad | (E \in t)?E : E \\
& \quad | \overline{X \ [x = t]}.x \\
& \quad | \overline{X \ [x = t]}.X \ [x = t] \\
v & ::= c \\
& \quad | \% \{ \overline{\ell} = v \} \\
& \quad | \$ \wedge \overline{t \rightarrow t \text{ fn } \overline{x} \rightarrow E} \\
& \quad | \$ \cap (\overline{N : T}) \rightarrow T \text{ fn } \overline{N} \rightarrow E \\
T & ::= t \\
& \quad | (\overline{N : T}) \rightarrow T \\
& \quad | \star \\
& \quad | \{ \overline{S}; \overline{D} \} \\
t & ::= \text{int} \\
& \quad | t \rightarrow t \\
& \quad | \% \{ \overline{f} \} \\
& \quad | t \vee t \\
& \quad | t \wedge t \\
& \quad | \neg t \\
& \quad | \alpha \\
& \quad | \mathbb{O} \\
& \quad | \overline{X \ [x = t]}.x \\
D & ::= \$\text{module } X : T \\
& \quad | \$\text{callback } x : \cap \overline{T} \\
& \quad | \$\text{opaque } x \\
& \quad | \$\text{type } x = t
\end{array}$$

Figure 1: Syntax of the surface language

$$\begin{array}{ll}
\tau & ::= t \\
& \quad | \star \\
& \quad | (\overline{N : \tau}) \rightarrow \tau \\
& \quad | \text{like } (\overline{X \ [x = t]}.X \ [x = t]) \\
& \quad | \cap \overline{\tau} \\
& \quad | \overline{X \ [x = t]}
\end{array}$$

Figure 2: Syntax of surface module types

$$\begin{array}{lll}
\$module\ X : T & \cap & \$module\ X : T' = \$module\ X : T \cap T' \\
\$callback\ X : \bigcap \overline{T} & \cap & \$callback\ X : \bigcap \overline{T'} = \$callback\ X : \bigcap \overline{T} \overline{T'} \\
\$type\ x = t & \cap & \$type\ x = t' = \$type\ x = t \wedge t' \\
\$opaque\ x & \cap & \$opaque\ x = \$opaque\ x \\
\$opaque\ x & \cap & \$type\ x = t = \$type\ x = t \\
\$type\ x = t & \cap & \$opaque\ x = \$type\ x = t \\
D & \cap & D' = \epsilon
\end{array}$$

Figure 3: Component-wise intersection

$$\begin{array}{lll}
\text{ElEnv-EMPTY} & \text{ElEnv-EXPR} & \text{ElEnv-TYPE} \\
\frac{}{\Sigma; \Gamma \vdash \epsilon} & \frac{\Sigma; \Gamma \vdash t : \star}{\Sigma, \Gamma \vdash x : t, \Gamma} & \frac{\Sigma; \Gamma \vdash t : \star}{\Sigma; \Gamma \vdash x = t, \Gamma} \\
\text{ModEnv-ModuleType} & \text{ModEnv-EMPTY} & \text{ModEnv-Module} \\
\frac{\Sigma; \Gamma, \overline{x} : \star \vdash \{\overline{D}\}}{\Sigma; \Gamma \vdash X = \overline{x} \mapsto \overline{D}, \Sigma} & \frac{}{\Sigma; \Gamma \vdash \epsilon} & \frac{\Sigma; \Gamma, \overline{x} : \star \vdash \{\overline{B}\}}{\Sigma; \Gamma \vdash X = \overline{x} \mapsto \overline{B}, \Sigma}
\end{array}$$

Figure 4: Formation rules for environments

$$\begin{array}{ll}
\text{EqPath-EMPTY} & \text{EqPath-ADD} \\
\frac{}{\Sigma; \Gamma \vdash \epsilon \cong \epsilon} & \frac{\Sigma; \Gamma \vdash P_1 \cong P_2 \quad \forall i. \Sigma, \Gamma \vdash t_i \cong t'_i}{\Sigma; \Gamma \vdash P_1.X \left[\overline{x_i = t_i} \right] \cong P_2.X \left[\overline{x_i = t'_i} \right]}
\end{array}$$

Figure 5: Rules for path equivalence

Figure 6: Typing rules for declarations

$$\begin{array}{c}
\text{BIND-DEFMODULE} \\
\frac{\Sigma; \Gamma, \overline{P : \star} \vdash \overline{B} : \overline{D} \quad \Sigma; \Gamma, X : (\overline{P : \star}) \rightarrow \{\overline{S_i; \overline{D}}\} \vdash \overline{B_0} : \overline{D_0} \quad \forall i. \Sigma(S_i) = \overline{x_i : \star} \rightarrow \overline{D_i} \quad \forall i \neq j. (\overline{x_i}, \text{dom}(D_i) \# \overline{x_j}, \text{dom}(D_j)) \quad \{\overline{D}\} \preccurlyeq \{\overline{x_i : [= _]\overline{D_i}}\}}{\Sigma; \Gamma \vdash (\text{defmodule } X \text{ do } \overline{P S_i B} \text{ end}) \overline{B_0} : (X : (\overline{P : \star}) \rightarrow \{\overline{S_i; \overline{D}}\}) \overline{D_0}} \\
\\
\text{BIND-TYPE} \\
\frac{\Sigma; \Gamma \vdash t : \star \quad \Sigma; \Gamma, x : [= t] \vdash \overline{B} : \overline{D}}{\Sigma; \Gamma \vdash (\text{\$type } x = t) \overline{B} : (x : [= t]) \overline{D}} \\
\\
\text{BIND-OPAQUE} \\
\frac{\Sigma; \Gamma \vdash t : \star \quad \Sigma; \Gamma, x : [= t] \vdash \overline{B} : \overline{D}}{\Sigma; \Gamma \vdash (\text{\$opaque } x = t) \overline{B} : (x : \star) \overline{D}} \\
\\
\text{BIND-EMPTY} \quad \text{BIND-VALUE} \\
\frac{}{\Sigma; \Gamma \vdash \epsilon : \epsilon} \quad \frac{\Sigma; \Gamma \vdash v : \cap \overline{T} \quad \Sigma; \Gamma, x : \cap \overline{T} \vdash \overline{B} : \overline{D}}{\Sigma; \Gamma \vdash (x = v) \overline{B} : (x : \cap \overline{T}) \overline{D}}
\end{array}$$

Figure 7: Typing rules for bindings

$$\begin{array}{c}
\text{PATH-EMPTY} \\
\frac{}{\Sigma; \Gamma \vdash \epsilon : \{\epsilon; \Gamma\}} \\
\\
\text{PATH-SUBMODULE} \\
\frac{}{}
\end{array}$$

Figure 8: Well-formedness rules for paths

$$\begin{array}{c}
\text{TYPE-VARIABLE} \\
\frac{\Sigma; \Gamma \vdash P : \{\bar{S}; \bar{D}(x : \cap T) \bar{D}'\}}{\Sigma; \Gamma \vdash P.x : \cap T} \\
\\
\text{TYPE-SUBSUMPTION} \\
\frac{\Sigma; \Gamma \vdash T \quad \Sigma \Gamma \vdash E : \cap \bar{T}' \quad \Sigma; \Gamma \vdash \cap \bar{T}' \preccurlyeq T}{\Sigma; \Gamma \vdash E : T} \\
\\
\text{TYPE-BIGFUNCTION} \\
\frac{\Sigma; \Gamma \vdash \cap \bar{N} : \bar{T} \rightarrow T' \Sigma; \Gamma, \bar{N} : \bar{T} \vdash E : T'}{\Sigma; \Gamma \vdash \$ \cap \forall \bar{\alpha} (\bar{N} : \bar{T}) \rightarrow T' \text{fn } \bar{N} \rightarrow E : \cap \bar{N} : \bar{T} \rightarrow T'} \\
\\
\text{TYPE-MODULE} \\
\frac{\Sigma; \Gamma \vdash P : \{\bar{S}_0; \bar{D}_0(X : \bar{y} : \star \rightarrow \{\bar{S}; \bar{D}\}) \bar{D}_1\} \quad \Sigma; \Gamma \vdash \bar{t} \quad \bar{x} \simeq \bar{y}}{\Sigma; \Gamma \vdash P.X [x = \bar{t}] : \{\bar{S}; \bar{D}\}}
\end{array}$$

Figure 9: Typing rules for the surface language

$$\begin{array}{c}
\text{SUB-STARREFL} \qquad \qquad \text{SUB-ELIXIR} \qquad \qquad \text{SUB-INTERSECTION} \\
\frac{}{\star \preccurlyeq \star} \qquad \frac{t \preccurlyeq t'}{\Sigma; \Gamma \vdash t \preccurlyeq t'} \qquad \frac{\exists i \in I, T_i \preccurlyeq T}{\Sigma; \Gamma \vdash \cap_I \bar{T}_i \preccurlyeq T} \\
\\
\text{SUB-MODULELEFT} \\
\frac{\Sigma; \Gamma \vdash P : \{\bar{S}; \bar{D}(x : [= t]) \bar{D}'\} \quad \Sigma; \Gamma \vdash t \preccurlyeq T}{\Sigma; \Gamma \vdash P.x \preccurlyeq T} \\
\\
\text{SUB-MODULERIGHT} \\
\frac{\Sigma; \Gamma \vdash P : \{\bar{S}; \bar{D}(x : [= t]) \bar{D}'\} \quad \Sigma; \Gamma \vdash \cap \bar{T} \preccurlyeq t}{\Sigma; \Gamma \vdash \cap \bar{T} \preccurlyeq P.x} \\
\\
\text{SUB-OPAQUE} \\
\frac{\Sigma; \Gamma \vdash P \cong P' \quad \Sigma; \Gamma \vdash P : \{\bar{S}; \bar{D}(x : \star) \bar{D}'\}}{\Sigma; \Gamma \vdash P.x \preccurlyeq P'.x} \\
\\
\text{SUB-BIGFUNCTION} \\
\frac{\forall i. \Sigma; \Gamma, x_1 : R_1, \dots, x_{i-1} : R_{i-1} \vdash T_i \preccurlyeq R_i \quad \Sigma; \Gamma, \bar{X}_i : R_i \vdash T' \preccurlyeq R'}{\Sigma; \Gamma \vdash (X_i : T_i) \rightarrow T' \preccurlyeq (X_i : R_i) \rightarrow R'}
\end{array}$$

Figure 10: Subtyping rules $\Sigma; \Gamma \vdash \cap \bar{T} \preccurlyeq T$

$$\begin{array}{rclcl}
\text{\$module } X : T & \cup & \text{\$module } X : T' & = & \text{\$module } X : T \cup T' \\
\text{\$callback } X : \bigcap \overline{T} & \cup & \text{\$callback } X : \bigcap \overline{T'} & = & ? \\
\text{\$type } x = t & \cup & \text{\$type } x = t' & = & \text{\$type } x = t \cup t' \\
\text{\$opaque } x & \cup & \text{\$opaque } x & = & \text{\$opaque } x \\
\text{\$opaque } x & \cup & \text{\$type } x = t & = & \text{\$opaque } x \\
\text{\$type } x = t & \cup & \text{\$opaque } x & = & \text{\$opaque } x \\
D & \cup & D' & = & \epsilon
\end{array}$$

Figure 11: Component-wise union