Dummy title

Authors omitted for double-bind review.
Unspecified Institution.

Abstract

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent convallis orci arcu, eu mollis dolor. Aliquam eleifend suscipit lacinia. Maecenas quam mi, porta ut lacinia sed, convallis ac dui. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse potenti.

2012 ACM Subject Classification Dummy classification

Keywords and phrases Dummy keyword

Digital Object Identifier 10.4230/LIPIcs.CVIT.2016.23

1 Formal

$$
\begin{array}{rcll}
id & ::= & t \mid C & \\
T & ::= & \texttt{This}n.\, Cs & \\
CD & ::= & C\texttt{=}E & \text{class declaration} \\
CV & ::= & C\texttt{=}LV & \text{evaluated class declaration} \\
D & ::= & id\texttt{=}E & \text{declaration} \\
DL & ::= & id\texttt{=}L & \text{partially-evaluated-declaration} \\
DV & ::= & id\texttt{=}LV & \text{evaluated-declaration} \\
L & ::= & \texttt{interface}\,\{\,Tz;\ amtz\ ;\,\} \mid \{\,Tz;\ Ms\ ;\ K?\} & \text{literal} \\
LV & ::= & \texttt{interface}\,\{\,Tz;\ amtz\ ;\,\} \mid \{\,Tz;\ MVs\ ;\ K?\} & \text{literal value} \\
amt & ::= & T\ m(Txs) & \text{abstract method} \\
mt & ::= & T\ m(Txs)\ \ e? & \text{method} \\
Tx & ::= & T\ x & \text{paramater-declaration} \\
M & ::= & CD \mid mt & \text{member} \\
MV & ::= & CV \mid mt & \\
Mid & ::= & C \mid m & \text{member-id} \\
K & ::= & \texttt{constructor}(Txs) & \text{constructor} \\
e & ::= & x \mid e.m(es) \mid e.x \mid \texttt{new}\ T(es) & \text{expression} \\
E & ::= & L \mid t \mid E\ \texttt{<+}\ E \mid E(Cs\texttt{=}T) & \text{library-expression} \\
\mathcal{E}_V & ::= & \square \mid \mathcal{E}_V\ \texttt{<+}\ E \mid LV\ \texttt{<+}\ \mathcal{E}_V \mid \mathcal{E}_V(Cs\texttt{=}T) & \text{context of library-evaluation} \\
\mathcal{E}_v & ::= & \square \mid \mathcal{E}_v.m(es) \mid v.m(vs\ \mathcal{E}_v\ es) \mid \mathcal{E}_v.x \mid \texttt{new}\ T(vs\ \mathcal{E}_v\ es) & \\
v & ::= & \texttt{new}\ T(vs) & \\
p & ::= & DLs;\ DVs & \text{program} \\
S & ::= & Ds\ e & \text{source code}
\end{array}
$$

To look up the value of a type in the program we will use the notation $p(T)$, which is defined by the following, but only if the RHS denotes an LV:

42nd Conference on Very Important Topics (CVIT 2016).
Editors: John Q. Open and Joan R. Access; Article No. 23; pp. 23:1–23:3
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

$$(;\_, C\texttt{=}L, \_)(\texttt{This}0.C.Cs) \coloneqq L(Cs)$$
$$(id\texttt{=}L, \_)(\texttt{This}0.Cs) \coloneqq L(Cs)$$
$$(;\_, C\texttt{=}L, p)(\texttt{This}n+1.Cs) \coloneqq p(\texttt{This}n.Cs)$$

To get the relative value of a trait, we define $p[t]$:

$$(DLS;\_, t\texttt{=}LV, \_)[t] \coloneqq LV[\texttt{This}\#DLs]$$

To get a the value of a literal, in a way that can be understand from the current location ($\texttt{This}0$), we define:

$$p[T] \coloneqq p(T)[T]$$

And a few simple auxiliary definitions:

$$Ts \in p \coloneqq \forall T \in Ts \bullet p(T) \text{ is defined}$$
$$L(\emptyset) \coloneqq L$$
$$L(C\,.\,Cs) \coloneqq L(Cs) \text{ where } L = \texttt{interface?}\,\{\_;\ \_, C\texttt{=}L, \_;\ \_\}$$
$$L[C\texttt{=}E'] \coloneqq \texttt{interface?}\,\{Tz;\ MVs\ C\texttt{=}E'\ Ms;\ K?\}$$
$$\text{where } L = \texttt{interface?}\,\{Tz;\ MVs\ C\texttt{=}\_\ Ms;\ K?\}$$

– towel1:.. //Map: towel2:.. //Map: lib: T:towel1 f1 ... fn
MyProgram: T:towel2 Lib:lib[.T=This0.T] ... –

References