Scala Inline

No Author Given

No Institute Given

Abstract. [1]

Keywords: Partial Evaluation, Macros

1 Translation

The syntax of our core language is the following:

Terms:

$$\begin{aligned} t &::= t\ t \mid \lambda x : \tau . t \mid x[\tau] \cdots [\tau] \mid \textbf{let}\, x : \sigma = t\, \textbf{in}\ \ t \mid \textbf{fix}\ \ t \mid x \\ v &::= \lambda x : \tau . t \mid c \\ c &::= \textbf{true} \mid \textbf{false} \mid 0 \mid 1 \mid \ldots \end{aligned}$$

Types:

$$\begin{aligned} \sigma &::= \tau \mid \forall X . \sigma \\ \tau &::= X \mid \tau \Rightarrow \tau \mid \iota \\ \iota &::= \textbf{Bool} \mid \textbf{Int} \end{aligned}$$

Extensions to the calculus like **if**, and arithmetic operations are straightforward so we use them freely in our examples.

References

1. Eugene Burmako and Martin Odersky. Scala Macros, a Technical Report. In *Third International Valentin Turchin Workshop on Metacomputation*, 2012.