Type-Based Partial Buffering
for Stream Processing

Ryosuke Sato[1], Naoki Kobayashi[2], and Kohei Suenaga[3]

[1] School of Engineering, Tohoku University `ryosuke@kb.ecei.tohoku.ac.jp`
[2] Graduate School of Information Sciences, Tohoku University
`koba@kb.ecei.tohoku.ac.jp`
[3] Graduate School of Information Science and Technology, Tokyo University
`kohei@yl.is.u-tokyo.ac.jp`

Abstract. Stream-processing is important for efficient processing of tree-structured data such as XML. Suenaga et al. have proposed a type-based framework for automatically translating tree-processing programs into stream-processing ones. Their framework uses ordered linear types to check whether an input tree data is accessed once in the left-to-right, depth-first order, and to insert buffering primitives when that access restriction is violated. We refine their framework by enabling partial buffering of input tree data, so that only necessary part of data is copied from stream to main memory. We reduce the problem of inserting partial buffering primitives to a kind of usage analysis, and propose a type-based method for the analysis.