

## Problem #1

*Originator: Jan Willem Klop*

*Date: April 1991*

*Summary: Which rewrite systems can be directly defined in lambda calculus?*

An important theme that is largely unexplored is definability (or implementability, or interpretability) of rewrite systems in rewrite systems. Which rewrite systems can be directly defined in lambda calculus? Here “directly defined” means that one has to find lambda terms representing the rewrite system operators, such that a rewrite step in the rewrite system translates to a reduction in lambda calculus. For example, Combinatory Logic is directly lambda definable. On the other hand, not every orthogonal rewrite system can be directly defined in lambda calculus. Are there universal rewrite systems, with respect to direct definability? (For alternative notions of definability, see [O’D85].)

### Remark

Some progress has been made in [BB92].

# Bibliography

- [BB92] Alessandro Berarducci and Corrado Böhm. A self-interpreter of lambda calculus having a normal form. Rapporto tecnico 16, Dip. di Matematica Pura ed Applicata, Universita di L'Aquila, October 1992.
- [O'D85] Michael J. O'Donnell. *Equational Logic as a Programming Language*. The MIT Press, Cambridge, MA, 1985.