Simplicity, Measuring, and Good Engineering

One Way to Build a World Class Automated Deduction System

Stephan Schulz[1,2]

[1] Institut für Informatik, Technische Universität München
[2] RISC-Linz, Johannes Kepler Universität Linz*
schulz@informatik.tu-muenchen.de

Abstract

Most published papers on implementation aspects of automated reasoning systems cover only a small set of new techniques. Overview papers are rare, and usually describe the fixed state of a system at a given point in the development process. Moreover, they often have to trade depth for generality. This is particularly true for system descriptions, which often are relegated to second-class status and allowed only a view pages at many major conferences.

In my talk, I will try to shed some lights into the practical aspects of building a complex high-performance theorem prover. I will give an overview on our equational theorem prover E [Sch02]. However, instead of giving a purely static view, I will describe the process that has resulted in a useful and resilient code base which has, up to now, survived at least three major changes without serious problems. I will also discuss some of the design decisions that later turned out to be wrong, and how they have either been fixed or still burden us.

Finally, I will describe some of the engineering tricks and tools we use to make sure that our code remains stable, mostly bug free, and, most of all, maintainable.

References

[Sch02] S. Schulz. E – A Brainiac Theorem Prover. *Journal of AI Communications*, 15(2/3):111–126, 2002.

This talk has been supported by the EU CALCULEMUS Human Potential Programme.

* Currently visiting at the University of Edinburgh.