From Retrospective Verification to Forward-Looking Development

K. Rustan M. Leino

Microsoft Research, Redmond, WA, USA
leino@microsoft.com

Abstract. One obstacle in applying program verification is coming up with specifications. That is, if you want to verify a program, you need to write down what it means for the program to be correct. But doesn't that seem terribly wrong? Why don't we see it as "one obstacle in program design is coming up with code"? That is, if you want to realize a specification, you need to write down how the machine is supposed do it. Phrased this way, we may want to change our efforts of verification into efforts of what is known as correct-by-construction or stepwise-refinement. But the choice is not so clear and there are plenty of obstacles on both sides. For example, many programs are developed from specifications, but the specifications are not in a form suitable for refinement tools. For other programs, the clearest specifications may be given by pseudo-code, but such specification may not be suitable for some verification tools. In this talk, I will discuss verification tools and refinement-based tools, considering how they may be combined.

M. Bobaru et al. (Eds.): NFM 2011, LNCS 6617, p. 1, 2011.