Model Driven Language Engineering

Stuart Kent

University of Kent at Canterbury, UK
Email: S.J.H.Kent@ukc.ac.uk

Abstract

There is currently much interest in model driven approaches to software engineering. The basic idea is to make business, architecture and design models first class artefacts that are not discarded once the code gets built, but rather can be exercised independently, be reconciled with each other and underlying implementations, and used to (partially) generate each other and implementations. This is particularly valuable when an application needs to be updated and changed, as up-to-date models will still be there to help understand and evolve it using the same abstractions originally used to build it. It is also valuable for applications that need to be delivered on multiple platforms and technologies, which may also change over time, as the models are generally independent of these, hence longer-lived.

This talk will argue that to support model driven software engineering, one needs to support model driven language engineering. The former is likely to lead to a larger number of more focused modelling languages, as opposed to a small number of monolithic, poorly integrated languages than we have at present. There will be more emphasis on domain specific languages. We will need to support the definitions of families of languages where overlap in language capabilities is recognized and repetition is avoided. Furthermore, more sophisticated modelling tools will be required, including tools to exercise models, for validation and verification, and tools for reconciliation and translation between models. Thus definitions should be platform independent, and it should be possible to automate (at least partially) the generation of language-specific modelling, reconciliation and translation tools. The talk will present some on-going work in this area, including a demonstration of language engineering tools that we are currently building, and a vision for the tools and languages we would like to build.