Research Statement for IFIP WG 2.11

Methods and tools for developing domain-specific languages

Krzysztof Czarnecki, May 13, 2004

My research interests lie in the context of methods and tools for systematic development of domain-specific languages (DSLs).

Despite the advantages and the growing popularity of the DSL approach, there is still relatively little systematic guidance and no comprehensive method to develop DSLs. A promising direction is to consider domain analysis and system-family approach as a starting point for DSLs. In particular, feature modeling can be used to scope and design the abstractions to be supported by a DSL. Feature modeling is a key approach to capturing and managing the common and variable features of systems in a system family. In the early stages of service family development, feature models provide a basis for scoping the family. Later, feature models play a central role in the development of a software architecture, which has to realize the variation points specified in the feature models. Finally, feature models can be used to drive the configuration of software.

My recent work includes the concept of *staged configuration* using feature models and the formalization of *cardinality-based feature modeling* and their staged configuration. My current research activities focus on providing expressive and efficient means to capture constraints between features and mechanisms to support interactive configuration. A test bed for this approach is an Eclipse plug-in for feature modeling and feature-based configuration, which is currently under development. My research is driven by concrete case studies. Currently, I'm performing domain analysis of the field of e-commerce and web-applications.

In another project, I plan to investigate the different aspects of DSL development, including domain scoping, design of abstractions (esp. their supported variabilities and composability), design of abstract and concrete syntax, and definition of their semantics. I hope to arrive at concrete guidelines and development steps, which will eventually lead to a coherent method for DSL development. I plan to investigate the evolution of feature models into DSLs with graph-like structures, devise guidelines for feature modeling and deriving DSLs with different structures from feature models, and propose metrics for assessing feature models and DSL (such as degree of orthogonality and supported variability). Another topic of interest is the design of concrete syntax (graphical, textual, wizard-based, etc.) to fit a problem domain. Currently I'm exploring the need for DSLs in the context of e-commerce and web-applications.

Finally, I'm interested in tool support for DSL implementation -- in particular, tools for DSLs with graphical concrete syntax. In this context I explore the relationship between generative software development and model-driven development. Aspects of interest include support for model transformations, specification of concrete syntax for graphical languages, composition of DSLs, consistency maintenance between models, source-level debugging, and the use of languages related to UML.