Systems Exhibition

* Abstract not received in time for publication

The PSG System: From Formal Language Definitions to Interactive Programming Environments

Rolf Bahlke, Gregor Snelting
Fachgebiet Praktische Informatik
Fachbereich Informatik
Technische Hochschule Darmstadt
Magdalenenstr. 11c, D-6100 Darmstadt

Overview

The PSG system developed at the University of Darmstadt generates interactive, language-specific programming environments from formal language definitions. All language-dependent parts of the environment are generated from an entirely nonprocedural specification of the language's syntax, context conditions and dynamic semantics. The generated environment consists of a language-specific hybrid editor, an interpreter and debugging system, and a library system.

The editor allows structure editing as well as text editing. Both modes are fully integrated and may be mixed freely. The user determines the granularity of incremental analysis: at one end of the spectrum there is pure text editing, while pure structure editing is the other extreme. When analysing textual input, PSG editors guarantee immediate detection of syntactic and static semantic errors. In structure mode, they even guarantee prevention of such errors. The editor will however not insist on immediate error correction; it tolerates incorrect or inconsistent programs. The user interface heavily utilizes raster graphics and mouse in order to achieve fast and ergonomic interaction.

PSG editors employ a novel algorithm for incremental semantic analysis, namely the concept of context relations. This concept is based on type inference rather than type checking. The context conditions of the language are described by inference rules. During editing, these rules are evaluated using a unification algorithm for order-sorted algebras. Change propagation is used to achieve fast incremental behaviour, and structure sharing avoids excessive memory requirements. The algorithm is language independent and guarantees immediate error detection in arbitrary incomplete program fragments. Error prevention is achieved by dynamically filtering all menus with respect to inferred context information.

The interactive interpreter is generated from a denotational semantics definition. It allows execution of incomplete program fragments. In order to generate an interpreter, semantic functions must be written in a functional language. The terms of this language are compiled into abstract machine code. This code is interpreted during fragment execution.

The debugger, which is currently available only as a prototype, is generated from an extension of the semantics definition. It offers additional features such as tracing, single-stepping, displaying variable values, setting conditional breakpoints etc. Interaction with the debugger is always on language level rather than on machine level.

The library is language-independent and stores programs as abstract trees. Usually, interaction with the library is invoked automatically by the editor resp. the interpreter. It is possible to import and export external text files into/from the environment.

PSG has been used to generate environments for Fortran 77, Lisp, Modula-2, Pascal, and the formal language definition language itself. PSG is also used to generate the programming environments of the German supercomputer SUPRENUM.

The PSG system is currently available on PCS Cadmus and SUN Workstations. Research institutions may obtain a copy for a nominal fee.

References

Bahlke, R., Snelting, G. : The PSG System: From Formal Language Definitions to Interactive Programming Environments. ACM TOPLAS, Vol. 8, No. 4 (October 1986), pp. 547-576.

Bahlke, R., Snelting, G.: Context-Sensitive Editing with PSG Environments. Proc. Advanced Programming Environments, Trondheim, June 1986, Springer Lecture Notes in Computer Science, Vol. 244, pp. 26-38.

Bahlke, R., Moritz, B., Snelting, G.: A Generator for Language-Specific Debugging Systems. Proc. SIGPLAN '87 Symposium on Interpreters and Interpretive Techniques, ACM Sigplan Notices, Vol. 22, No.7, July 1987, pp. 92-101.

Bahlke, R., Hunkel, M.: The User Interface of PSG Programming Environments. Proc. 2nd IFIP Conference on Human-Computer Interaction, September 1987, North Holland, pp. 311-315.