
 * COMPILER GENERATION FOR ATTRIBUTE GRAMMARS *
 * * * * *

Name of Project : Compiler Generation for Attribute Grammars

Organization : University of Montreal

Principle Investigator : G.V. Bochmann

Address : Département d'informatique
 Université de Montréal
 C.P. 6128, Montréal 101, Canada.

Description of Project :

Objectives : The comprehensive and effective definition of semantics for programming languages by means of attribute grammars. Theoretical studies, and the implementation of a compiler generating system for attribute grammars.
 Starting Date : September, 1972.

Estimated Completion Date : April, 1974.

Current Status : After some theoretical studies on the subject, and the inclusion of synthesized attributes in a compiler generator, the design of a compiler generator system for complete attribute grammars is in progress.

Computer Being Used : CDC Cyber 74, Xerox Sigma 6.

Language Involved : Pascal.

General Approach :

Attribute grammars [1,2,3] are context free grammars which have semantic attributes associated with the syntactic symbols. They allow a comprehensive and effective description of the semantics of languages by means of local transformations of the semantic information (attributes), executed by semantic routines which are associated with the productions of the grammar. The objective of the project is to use the formalism of attribute grammars for a compiler generator system and to allow a structured [4] description of the semantics of languages which is easy to read and efficiently and automatically implemented.

The theoretical aspects of the project can be divided into two parts :

- (a) Finding a good strategy for the evaluation of the semantic attributes of a given grammar. Investigating under what conditions the semantics can be evaluated in a single, or in several passes from left to right.
- (b) Finding methods for proving the equivalence between different descriptions of the syntax and semantics of languages. Such methods may be useful for simplifying grammars.

The practical aspect of the project is to implement a compiler generator system which accepts the definition of a language, for which a compiler is to be generated, in the form of a generalized attribute grammar. As a first step, the use of synthesized attributes can be combined with a compiler generator for bottom-up parsers [5]. Compilers for attribute grammars of type LL(k) [6] which allow a single left-to-right pass for the evaluation of the semantics can be

implemented by using recursive procedures. For more general grammars an intermediate parse tree has to be build explicitly.

We use the implementation language Pascal [7], which has already been found useful for similar work [5].

References :

1. D.E. Knuth, Semantics of context-free languages, Math. Systems Th., 2, 127 (1968). Correction : Math. Systems Th., 5, 95 (1971).
2. D.E. Knuth, Examples of formal semantics, in Lecture Notes in Mathematics No. 188, Springer-Verlag, Berlin (1971).
3. W.T. Wilner, Formal semantics definition using synthesized and inherited attributes, in Courant Computer Science Symposium 2, p. 25-39, Prentice Hall, 1972.
4. E.W. Dijkstra, Notes on structured programming, in Structured Programming by Dahl, Dijkstra, and Hoare, Acad. Press 1972.
5. O. Lecarme, Un générateur d'analyseurs syntaxiques, Working Document #27, Département d'Informatique, Université de Montréal. Further work is in progress.
6. D.E. Knuth, Top-down syntax analysis, Acta Informatica, 1, 79 (1971).
7. N. Wirth, The programming language Pascal, Acta Informatica, 1, 35 (1971).

Documentation :

Existing Documentation :

- (1) G.V. Bochmann, Une définition syntaxique et sémantique des langages pour un système d'écriture de compilateurs, Working Document #30, Département d'Informatique, Univ. de Montréal (1972).
 - (2) G.V. Bochmann, Semantics evaluated from left to right, Report, Département d'Informatique, Univ. de Montréal (April 1973).
- Potential Documentation :
- (1) Proving equivalences for attribute grammars (October, 1973).
 - (2) Compiler generator system for attribute grammars (December, 1973).
 - (3) Automatic simplification of grammars (April, 1974).

Source of Funding :

National Research Council of Canada.

 * TACTICS, AN INTEGRATED SYSTEM FOR STRUCTURED PROGRAMMING *
 * * * * *

Project Name: TACTICS, AN INTEGRATED SYSTEM FOR STRUCTURED PROGRAMMING

Organization: General Research Corporation, Program Validation Research Project

Contact: Nancy B. Brooks