Natural language applications played a key motivating role in the development of Prolog in the early 1970s by the A.I. group at Marseille, headed by Alain Colmerauer. Background ingredients for this development included Colmerauer’s prior work on Q-systems, which had some of the features of logic grammars, as well as discussions with Robert Kowalski on SL-resolution and the use of logic to represent grammars. The first application of interest to the Marseille group was a natural language understanding system. Colmerauer introduced the first logic grammar formalism, metamorphosis grammars, which are like type-0 phrase structure grammars, but with logic terms for grammar symbols, and with unification used for rewriting instead of simple replacement. Colmerauer showed how these can be translated to Horn clause systems. Since the introduction of metamorphosis grammars, logic programming and Prolog have been widely used in natural language computation, and it is quite fitting to devote a special issue of the Journal of Logic Programming to the subject.

The first paper of this issue, by Miguel Filgueiras, “Cooperating Rewrite Processes for Natural Language Analysis,” presents general methods for natural language analysis, including both syntactic and semantic analysis systems. The syntactic analyzer (which receives the main emphasis in the paper) is based on an intensive use of rewrite rules that produce functional structures in an essentially bottom-up fashion. The semantic representation language (based on previous joint work of the author with Antonio Porto) is a kind of typed logical form language, and the semantic interpreter uses a bottom-up rewrite process to build semantic representations from functional structures in a systematic way.

The second paper, by Lynette Hirschman, “Conjunction in Meta-Restriction Grammars,” presents a metagrammatical component for Restriction Grammars, a logic grammar implementation of Naomi Sager’s String Grammars. This component generates grammar rules specifying allowable coordination of specific types of nodes (by a method related to one of Sedogbo), and avoids the combinatorial explosion of the backtracking approach of earlier researchers. An advantage of this approach is that it involves compilation of rules rather than interpretation.

The third paper, by Patrick Saint-Dizier, “An Approach to Natural Language Semantics in Logic Programming,” considers an approach to the semantics of natural language sentences which involves the development of a small collection of
tools based on set theory which can be used to deal with such entities as determiners, negations, adjectives, and some adverbs. These tools are hypothesized to be used by people when they define the semantics of each lexical entity that occurs in a specific domain. The tools, however, are not tied to any specific domain, but are easily transported to various domains of discourse and also to various formalisms.

Michael C. McCord  
*IBM T. J. Watson Research Center*  
*Yorktown Heights, NY, USA*

Veronica Dahl  
*Department of Computing Science*  
*Simon Fraser University*  
*Burnaby, BC, Canada*

Harvey Abramson  
*Department of Computer Science*  
*University of British Columbia*  
*Vancouver, BC, Canada*