



ELSEVIER

The Journal of Logic Programming 43 (2000) 1–2

THE JOURNAL OF
LOGIC PROGRAMMING

www.elsevier.com/locate/jlpr

Guest editors' introduction

Special issue: Logic-based heterogeneous information systems

Raghu Ramakrishnan^a, V.S. Subrahmanian^{b,*}

^a *University of Wisconsin, Madison, USA*

^b *Department of Computer Science, University of Maryland, A.V. Williams Building, College Park, MD 20742, USA*

With the advent of the information superhighway, there is now a vast amount of information (data and software) accessible through the Internet, in addition to the many enterprise databases that are connected together on corporate intranets. Builders of complex software applications are thus faced with the need to integrate data residing in different sources, and accessible over some communications network. In recent years, several research projects have explored the use of logic-based frameworks for software integration and semantic data integration. In this special issue of the Journal of Logic Programming, we highlight the use of logic in integrating distributed repositories.

We begin with a paper by Aquilino et al., that reports on use of the MedLan mediator language to integrate geographical information systems. Specifically, the paper describes how MedLan was used to build a declarative analysis layer on top of a commercial geographic information system, Arc-Info. This analysis layer is then applied to the problem of finding the best habitat for wild boars in an area of Tuscany.

The second paper, by Bertino et al., goes one level deeper into the systems aspects of a logic-based framework for integration by describing a logical language that supports cooperative queries and change management across distributed repositories. The proposed language covers declarative querying based on deductive rules and update propagation across databases through an active extension of Datalog.

The third paper, by Duschka et al., studies an integration approach based on describing each source relation as a query over a set of “view relations”. The view relations are the relations available for querying by users, and queries are answered by automatically rewriting them to refer only to source relations. This paper advocates that the rewriting should not insist on equivalence to the query, only that the rewrit-

* Corresponding author. Tel.: +1-301-405-2711; fax: +1-301-405-8488.

E-mail address: vs@cs.umd.edu (V.S. Subrahmanian).

ing produce answers that belong to the query, and shows how to produce such “maximally contained” queries through rewriting, taking into account integrity constraints and query binding patterns; in general, this requires recursion.

The fourth paper is by Vassalos and Papakonstantinou, and it addresses two main issues: how to describe the class of queries that a source can answer, and how to take a given query and see if it can be expressed using queries supported by the source (where a source could itself be a mediator combining several other sources). A simple source description language called *p*-Datalog is studied first, and an efficient algorithm is presented for determining whether the source can answer a given query, and if so, how. Next, a more powerful description language called RQDL is motivated, and its relationship to Datalog with function symbols.

We are grateful to the Editor-in-Chief, Maurice Bruynooghe, for inviting us to edit this issue. We are thankful to the authors for contributing high-quality papers, and to our reviewers for their tireless work. We hope you will enjoy this special issue.