Journal of Universal Computer Science, vol. 16, no. 2 (2010), 218-220
submitted: 7/1/10, accepted: 15/1/10, appeared: 28/1/10 © *J.UCS*

Lisp: Research and Experience

J.UCS Special Issue

António Menezes Leitão
(Instituto Superior Técnico/Universidade Técnica de Lisboa
and
INESC-ID, Portugal
antonio.menezes.leitao@ist.utl.pt)

Lisp is now more than fifty years old. By all measures, this is an extraordinary feat. From the seminal work of John McCarthy, a Cambrian explosion of Lisp dialects emerged that explored (and continue to explore) the vastness of the programming language landscape. The Lisp community grew in an harsh environment but was capable of adapting that environment by inventing dedicated editors, special-purpose hardware, windowing systems and interactive programming environments. When AI's glacial age came, the Lisp community adapted itself in order to live and grow on commodity hardware. Nowadays, at a time where new programming languages require gigantic investments in libraries, compilers, and programming environments, Lisp continues to grow and evolve, generating new dialects or mingling itself with the newcomers. The Lisp family of languages is a driving force both in the programming language research field and in the software industry field. Lisp dialects have been used for the development of all kinds of software, from embedded systems to extremely large systems, and the Lisp DNA can be found in all modern programming languages.

This special issue of The Journal of Universal Computer Science is a proof of the vitality of Lisp. For this issue we invited high quality papers about novel research results, insights and lessons learned from theoretical topics, practical applications, and educational perspectives, all involving Lisp dialects, including Common Lisp, Scheme, Emacs Lisp, AutoLisp, ISLISP, Dylan, Clojure, and so on. Topics included, but were not limited to:

- Language design and implementation techniques
- Language integration, interoperation and deployment
- Language critique and future directions
- Reflection and meta-level architectures
- Educational approaches

- Software adaptation and evolution
- Configuration management
- Artificial intelligence
- Large and ultra-large-scale systems
- Development methodologies
- Development support and environments
- Persistent systems
- Scientific computing
- Parallel and distributed computing
- Data mining
- Semantic web
- Dynamic optimization
- Innovative applications
- Hardware and virtual machine support
- Domain-oriented programming
- Lisp pearls
- Experience reports and case studies

The range of papers received demonstrated the wide applicability of Lisp. Their topics included, among others, Unit Testing, Software Transactional Memory, Hygienic Macros, Design Patterns, and Compiler Extensions and Code Optimization.

All received papers were extensively peer reviewed and were selected on the basis of their quality and relevance to the theme of this special issue. The reviewers for this special issue were:

- Marco Antoniotti, Università degli Studi di Milano Bicocca, Italy
- Giuseppe Attardi, Università di Pisa , Italy
- Pascal Costanza, Vrije Universiteit Brussel, Belgium
- Marc Feeley, Université de Montréal, Canada

– Richard P. Gabriel, IBM Research, USA
– Ron Garret, Amalgamated Widgets Unlimited, USA
– António M. Leitão, Instituto Superior Técnico, Portugal
– Scott McKay, ITA Software, Inc., USA
– Peter Norvig, Google Inc., USA
– Julian Padget, University of Bath, UK
– Kent Pitman, PTC, USA
– Christian Queinnec, Université Pierre et Marie Curie, France
– Christophe Rhodes, Goldsmiths College, University of London, UK
– Robert Strandh, Université Bordeaux 1, France
– Didier Verna, EPITA Research and Development Laboratory, France
– JonL White, TheGingerIceCreamFactory of Palo Alto, USA
– Taiichi Yuasa, Kyoto University, Japan

It was my pleasure to work with the reviewers I want to personally thank all them for their careful and extensive reviews.

António Menezes Leitão
(Lisboa, January, 2010)