

# Lisp50

## The 50<sup>th</sup> Birthday of Lisp at OOPSLA 2008

Pascal Costanza  
Vrije Universiteit Brussel  
Programming Technology Lab  
Brussels, Belgium  
pascal.costanza@vub.ac.be

Richard P. Gabriel  
IBM Research  
Hawthorne, NY, USA  
rpg@dreamsongs.com  
rpg@us.ibm.com

Robert Hirschfeld  
Hasso-Plattner Institut  
Universität Potsdam  
Potsdam, Germany  
hirschfeld@hpi.uni-potsdam.de

Guy L. Steele, Jr.  
Sun Microsystems Laboratories  
Programming Language Research Group  
Burlington, MA, USA  
guy.steele@sun.com

### Abstract

In October 1958, John McCarthy published one in a series of reports about his then ongoing effort for designing a new programming language that would be especially suited for achieving artificial intelligence. That report was the first one to use the name LISP for this new programming language. 50 years later, Lisp is still in use. We would like to celebrate Lisp's 50<sup>th</sup> birthday. OOPSLA 2008 is an excellent venue for such a celebration, because object-oriented programming benefitted heavily from Lisp ideas and because OOPSLA 2008 takes place in October, exactly 50 years after the name Lisp has been used publicly for the first time. We will have talks by John McCarthy himself, and numerous other influential Lispers from the past five decades. We will also take a look at the next 50 years of Lisp.

**Categories and Subject Descriptors** D.3.0 [Software]: Programming Languages—General; K.2 [Computing Milieux]: History of Computing

**General Terms** Languages

**Keywords** Lisp, Common Lisp, Scheme

### 1. 50 Years of Lisp

In October 1958, John McCarthy published one in a series of reports about his then ongoing effort for designing a new programming language that would be especially suited for achieving artificial intelligence (5). That report was the first one to use the name LISP for this new programming language. 50 years later, Lisp is still in use. During the past five decades, it has been changed and turned, which led to dialects differing in many respects from the original design, but the central corner stones remained the same, making it one of the oldest programming language still in use today, second only to Fortran.

The final design for the first incarnation of Lisp was published in an issue of the Communications of the ACM in 1960 (6). That version of Lisp pioneered numerous languages features that are nowadays taken for granted. To name but a few: Lisp introduced a conditional expression, which was taken over in other languages as the ubiquitous `if-then-else` statement; it introduced recursion and first-class functions, the essential ingredients of functional, and many other, programming languages; it introduced reference semantics for variables, without which object-oriented programming would not exist; it introduced symbolic expressions, a generic and uniform representation for data (later reinvented as XML) and programs, the latter enabling program transformations from within an application; it introduced garbage collection for the very first time in programming languages; and it even already had a combination of metadata and dynamic dispatch in the form of symbols and property lists.

Lisp is one of the most influential programming languages in the history of computer science: Timothy Hart added macros to Lisp in the 1960's (7); Warren Teitelman invented an advice facility for Lisp in the 1960's as the very first precursor to aspect-oriented programming (10); Carl Hewitt used Lisp as a platform to develop logic programming (3) and the actor model (4); Alan Kay acknowledges the heavy influence of Lisp on Smalltalk, the first explicit object-oriented programming language (2); Brian Smith developed the concept of computational reflection using Lisp as a starting point (8); Paul Graham used Lisp to develop the first continuation-based web application (1); and even today Lisp is on the forefront for the upcoming Web 3.0.

This is the celebration of Lisp's 50<sup>th</sup> birthday. OOPSLA 2008 is an excellent venue for such a celebration, because object-oriented programming benefitted heavily from Lisp ideas and because OOPSLA 2008 takes place in October, exactly 50 years after the name Lisp was used publicly for the first time.

John McCarthy will talk about the early history of Lisp, returning to OOPSLA after his keynote talk at OOPSLA 2007. Guy Steele and Richard Gabriel will repeat their HOPL-II talk about the Evolution of Lisp from 1992, using a particularly unusual set of slides (9). Pascal Costanza will talk about the recent developments in the Lisp community, which has seen a surprising resurrection after its awakening after AI Winter. Other influential Lispsers will give presentations, covering important aspects in the development of Lisp during the past five decades. Finally, we will have an open panel discussion about the next 50 years of Lisp.

## References

- [1] Paul Graham, Five Questions about Language Design. *Dynamic Languages Wizards Series*, Panel on Language Design, Massachusetts Institute of Technology, Artificial Intelligence Laboratory, May 2001. Notes: <http://www.paulgraham.com/langdes.html>
- [2] Stuart Feldman, A Conversation with Alan Kay. *ACM Queue*, Volume 2, Issue 9, 2004/2005.
- [3] Carl Hewitt, PLANNER: A Language for Proving Theorems in Robots. *Proceedings of the First International Joint Conference on Artificial Intelligence (IJCAI)*, Washington DC, May 1969.
- [4] Carl Hewitt, Peter Bishop, Richard Steiger, A Universal Modular Actor Formalism for Artificial Intelligence. *Proceedings of the Third International Joint Conference on Artificial Intelligence (IJCAI)*, Stanford CA, August 1973.
- [5] John McCarthy, *Symbol Manipulating Language—Revisions of the Language*. Massachusetts Institute of Technology, Artificial Intelligence Laboratory, AI Memo No. 4, Cambridge, MA, October 1958.
- [6] John McCarthy, Recursive Functions of Symbolic Expressions and their Computation by Machine. *Communications of the ACM*, Volume 3, Issue 3, 1960.
- [7] Timothy P. Hart, *MACRO Definitions for LISP*. Massachusetts Institute of Technology, Artificial Intelligence Laboratory, AI Memo No. 57, Cambridge, MA, October 1963.
- [8] Brian C. Smith, Reflection and Semantics in LISP. *Proceedings of the 11<sup>th</sup> ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages (POPL)*, Salt Lake City, Utah, 1984.
- [9] Guy L. Steele, Jr. and Richard P. Gabriel, The Evolution of Lisp. *ACM Conference on the History of Programming Languages, II*, published in *ACM SIGPLAN Notices*, Volume 28, Number 3, Cambridge, MA, March 1993.
- [10] Warren Teitelman, *PILOT: A Step toward Man-Computer Symbiosis*. PhD Thesis, Massachusetts Institute of Technology, Project on Mathematics and Computation (MAC), Technical Report MAC-TR-32, Cambridge, MA, September 1966.