

# Welcome Lispers!

黄 澗石 (Jianshi Huang)

November 27, 2011

- Common Lisp programmer in Mathematical Systems Inc.
  - We have a Common Lisp team
    - Almost all projects are written in CL (AllegroCL, SBCL, Lispworks)
    - Manager is a CL hacker
  - Projects covers
    - Machine Learning
    - High-speed network data acquisition
    - Text processing (twitter, blocking, clustering)
    - etc. (parallel utilities, GUI, ...)
  - Our team (3 ppl) finished #7 in ICFP 2009 programming contest
- Went to Japan 6.5 years ago
  - Ruby
  - Scheme (Fall in love with Lisp)
  - Common Lisp
  - etc. (Haskell, Prolog, ...)

# What is Lisp

- A family of programming languages
  - First appeared in 1958, designed by John McCarthy
  - Second oldest high-level programming language (second to Fortran)
  - Originally as mathematical notation for expressing computer programs
  - LISP for LISt Processing
- Ideas originated from Lisp
  - Tree data structure
  - Garbage collection
  - Dynamic typing
  - Code as data, data as code (s-expression)
  - etc.

# Lisp is evolved, and is still evolving

Currently most popular dialects are

- Emacs Lisp
- Common Lisp
- Scheme
- Clojure

# Lisp Myth

- Lisp is interpreted
  - Most Common Lisp and many scheme implementations compile to native code.
  - There are highly optimizing compilers (such as SBCL).
- Lisp is slow
  - <http://shootout.alioth.debian.org/u64/benchmark.php?test=all&lang:>
- Lisp is only for (classical) AI
  - <http://www.lispworks.com/success-stories/index.html>
- Lisp is OLD
  - Yes, but dialects like Common Lisp and Scheme are still evolving
  - R7RS next year?
  - Many language extensions in CL implementations

# Learning Lisp (Common Lisp)

- Learning programming methodologies
  - Structure and Interpretation of Computer Programs (Abelson and Sussman)
- Learning language features (CL)
  - Practical Common Lisp (Peter Seibel)
  - Land of Lisp (Conrad Barski)
  - Common Lisp the Language (2nd Edition) (Guy Steele)
- Learning from problem solving (CL)
  - Paradigms of Artificial Intelligence Programming (Peter Norvig)
- Advanced topics
  - On Lisp (Paul Graham)
  - Let Over Lambda (Doug Hoyte)
  - The Art of the Metaobject Protocol (Gregor Kiczales)
  - 
  - Read papers!

# Learning Lisp (Scheme)

(Somebody please leave comment)

- CLiki  
<http://www.cliki.net/index>
- comp.lang.lisp (google group)
- #lisp@freenode (irc)
- Lisp forum  
<http://www.lispforum.com/>
- Common Lisp Tips  
<http://lisptips.com/>
- And of course lisp-cn (google group)