

State of the Lisp Family

Lily Carpenter

2016-07-14

Introduction

All about me

A very very brief history

Why care about lisp?

Common Lisp

Brief History

Implementations

Features

Problems

Projects

Real world uses

CLASP

ECL

Resources

Clojure

Brief History

Backends

Purpose

- Features
- Projects
- Resources

Racket

- Brief History
- Features
- Projects
- Resources

Guile Scheme

- Brief History
- Features
- Projects
- Resources

Chicken Scheme

- Purpose
- Features
- Projects
- Resources

Emacs Lisp

- Brief History

- Purpose

- Projects

- Resources

Picolisp

- General Notes

- Resources

Shen

- General Notes

- Resources

Arc

- General notes

- Resources

Honorable Mentions

- Lisp Flavored Erlang

- Hy

- Pixie

Kawa

Topic

Introduction

All about me

A very very brief history

Why care about lisp?

Common Lisp

Brief History

Implementations

Features

Problems

Projects

Real world uses

CLASP

ECL

Resources

Clojure

Brief History

Backends

- ▶ I do NOT use lisp at work. I write ruby (on rails) and javascript
- ▶ I have only been part of the lisp community for a couple years
- ▶ I am most experienced with Common Lisp, Emacs Lisp, and Clojure
- ▶ I have played with Guile briefly
- ▶ I have a blog at azrazalea.net and git repositories on gitlab
- ▶ This presentation is on gitlab at <https://gitlab.com/azrazalea/state-of-lisp-family>
- ▶ Lisp(NOT common lisp) was first specified in 1958
- ▶ Many many dialects of Lisp have appeared over the years. See wikipedia
- ▶ The general hallmark of a Lisp is its s-expression based syntax (equal '(Lisp) '(Lots of Insipid, Stupid Parentheses))

- ▶ Originally heavily used in academic circles and AI
- ▶ Now mostly limited to small communities (Yes Clojure is still small)
- ▶ First and foremost lisp is FUN
- ▶ Lack of syntax and s-expressions are very freeing once you get used to them (and have a good editor)
- ▶ A very smart community that can unfortunately sometimes be hard to get into
- ▶ Code as data is awesome

Topic

Introduction

All about me

A very very brief history

Why care about lisp?

Common Lisp

Brief History

Implementations

Features

Problems

Projects

Real world uses

CLASP

ECL

Resources

Clojure

Brief History

Backends

- ▶ Work started in 1981, draft published 1984, 2nd draft 1990, final standard 1994
- ▶ Standard was a compromise/design by committee between the authors of various lisp implementations
- ▶ Standard came out of Interlisp, Maclisp, and others
- ▶ The language itself has not changed since this standard was published
- ▶ Language improvements done as implementation specific extensions
- ▶ Many things can be implemented as macros/reader-macros and shipped as libraries
- ▶ Quicklisp (a library manager) released in October 2010
- ▶ Roswell lisp implementation manager and script framework released very recently (not sure on date)
- ▶ There are many different implementations of the CL standard
- ▶ Commercial: Allegro CL, LispWorks

- ▶ Open Source: ABCL, Clasp, Clozure CL, CLISP, CMUCL, ECL, MKCL, SBCL and more
- ▶ Popular free ones are SBCL and CCL(Clozure CL). Both fast and cross platform
- ▶ CLOS (Common Lisp Object System)
- ▶ Pretty much every standard data structure
- ▶ Optional tail call optimization
- ▶ Robust package (think namespaces) system
- ▶ Build manager (asdf)
- ▶ Library manager (quicklisp)
- ▶ Fast with the right implementation
- ▶ Old and crotchety (community and language)
- ▶ Sometimes large differences between implementations (usually patched over with a cross-implementation library)
- ▶ Some simple things baked into most modern languages are implementation specific (threads, garbage collection, FFI, Networking stuff, OS stuff)

- ▶ Pretty much everything new on my gitlab
- ▶ Mcclim
<https://github.com/robert-strandh/McCLIM> cross platform GUI/Windowing library
- ▶ Cluffer text editor buffer
<https://github.com/robert-strandh/Cluffer>
- ▶ Climacs emacs replacement <https://github.com/robert-strandh/Second-Climacs>
- ▶ Lots of game programming libraries at <https://github.com/lispgames>
- ▶ See <http://eudoxia.me/article/common-lisp-sotu-2015>
"State of the Common Lisp Ecosystem, 2015"
- ▶ Libraries for almost everything you'll want to do
- ▶ Used at grammarly <https://www.grammarly.com/http://tech.grammarly.com/blog/posts/Running-Lisp-in-Production.html>

- ▶ Open source Evernote alternative <https://turtl.it/>.
Server is in Common Lisp
- ▶ Commercial examples at <http://franz.com/success/>
and <http://www.lispworks.com/success-stories/index.html>
- ▶ There seems to actually be quite a bit of it, just not advertised and generally closed source.
- ▶ See <https://lispjobs.wordpress.com/>

1. Purpose

- ▶ "Seamless" integration with C++ using LLVM.
- ▶ Speed and power of existing C++ code combined with the rapid prototyping, incremental dev, and other common lisp advantages.

2. Projects

- ▶ Mostly academic use so far.
- ▶ Read creator's blog here:

<https://drmeister.wordpress.com/>

- ▶ Creator is implementing CANDO, a tool for biologists for molecular design
- ▶ I don't know of any production use cases yet, but it is pretty cool!

1. Purpose

- ▶ Supports many platforms (Linux, FreeBSD, NetBSD, OpenBSD, OS X, Solaris, Windows on Intel, Sparc, Alpha, PowerPC, and Arm)
- ▶ Extremely portable with small and fast binaries.
- ▶ Can be called like a C library with no FFI
- ▶ Can call C functions with no FFI

2. Projects

- ▶ ECL on Android with libSDL for game programming
<https://gitlab.com/dto/ecl-android-games-src>
- ▶ Various people working on general purpose projects. ECL is a full common lisp
- ▶ See <https://common-lisp.net/project/ecl/>

- ▶ Practical Common Lisp
<http://gigamonkeys.com/book/>
- ▶ Common Lisp Recipes (for after PCL)
<http://weitz.de/cl-recipes/>
- ▶ Land of Lisp (fun alternative to PCL [love the comics])
<http://landoflisp.com/>
- ▶ Common lisp hyperspec <http://www.lispworks.com/documentation/HyperSpec/Front/index.htm>
- ▶ Duckduckgo hyperspec search with !clhs
- ▶ Articulate Common Lisp <http://articulate-lisp.com>
- ▶ <http://lisp-lang.org/>
- ▶ #lisp and #lispgames on freenode IRC
- ▶ For the love of lisp, use Emacs + SLIME(or the newer sly) as your REPL even if not your editor

Topic

Introduction

All about me

A very very brief history

Why care about lisp?

Common Lisp

Brief History

Implementations

Features

Problems

Projects

Real world uses

CLASP

ECL

Resources

Clojure

Brief History

Backends

- ▶ Created by Rich Hickey
 - ▶ Original public release 2007-10-16
 - ▶ First stable release (1.0) 2009-05-04
 - ▶ Latest version 1.8
-
- ▶ Java, the original and most supported
 - ▶ Javascript, (clojurescript) official and run by David Nolen
 - ▶ Various others in various states of support
-
- ▶ See <http://clojure.org/about/rationale>
 - ▶ Basically wanted A lisp for functional programming symbiotic with Java and designed for concurrency.
-
- ▶ Immutability focused
 - ▶ Very good java/javascript interop
 - ▶ All the bells and whistles you'd expect with a modern language

- ▶ Functional programming "only" (I consider this an anti-feature personally)
- ▶ Can you think of it? Someone has probably done it in Clojure
- ▶ Heavily used for backend web services so far
- ▶ Climate Corporation (our location sponsor) is a heavy user for production
- ▶ Walmart, Puppet Labs, Thoughtworks are some big companies using Clojure
- ▶ Lot of the cool stuff is in Clojurescript land like Om and Reagent
- ▶ Clojure for the Brave and True
<http://www.braveclojure.com/>
- ▶ Cursive + IntelliJ IDE <https://cursive-ide.com/>
- ▶ Emacs + Cider IDE
<https://github.com/clojure-emacs/cider>

- ▶ Clojurescript info
<https://github.com/clojure/clojurescript/wiki>

- ▶ #clojure on freenode IRC

Topic

Introduction

All about me

A very very brief history

Why care about lisp?

Common Lisp

Brief History

Implementations

Features

Problems

Projects

Real world uses

CLASP

ECL

Resources

Clojure

Brief History

Backends

- ▶ Originally PLT Scheme
- ▶ First appeared in 1994
- ▶ Renamed Racket 2010-06-07
- ▶ Lots and lots of friendly libraries and documentation
- ▶ Ships with IDE Dr. Racket
- ▶ Lots of learning/teaching resources, especially for kids
- ▶ Designed to be very easy to get up and running and make simple programs
- ▶ General purpose, does not force you into a particular paradigm
- ▶ Scribble documentation language
- ▶ Naughty Dog uses Racket in Uncharted, The Last of Us, etc
- ▶ Racket controls a huge telescope in New Mexico
- ▶ Arc (see later slides) implemented in Racket

- ▶ John Carmack using Racket for Gear VR.
<https://groups.google.com/d/msg/racket-users/RF1h0o6l3Ls/8InN7uz-Mv4J>
- ▶ Watch the Racketcon videos or go to Racketcon for more information!
- ▶ Racketcon is right after the STL Strangeloop Conference!
- ▶ Cool game creating book <http://realmofracket.com/>
- ▶ Awesome official docs
<https://docs.racket-lang.org/>
- ▶ #racket on freenode IRC

Topic

Introduction

All about me

A very very brief history

Why care about lisp?

Common Lisp

Brief History

Implementations

Features

Problems

Projects

Real world uses

CLASP

ECL

Resources

Clojure

Brief History

Backends

- ▶ Began work in 1993
- ▶ Originally GEL or GNU Extension Language
- ▶ Designed as a spiritual and cleaner successor to Emacs lisp
- ▶ Development languished until Andy Wingo took over in 2009/2010
- ▶ Guile 2.0 in 2011 revitalized the language with many improvements
- ▶ Since 2.0 there have been many incremental improvements to the language
- ▶ Very embed-able, designed for a polyglot environment
- ▶ Full featured, lots of batteries included libraries
- ▶ Easy to use C API that goes both ways
- ▶ Support for writing in other languages that compile to Guile including ecmascript, emacs lisp, and WIP for lua
- ▶ General purpose

- ▶ Mostly GNU projects as it is the official GNU extension language
- ▶ Project in progress to replace Emacs Lisp with guile, but community is split
- ▶ GNU Guix & GuixSD (cool nix-like package manager and distribution)
- ▶ GnuCash
- ▶ gEDA
- ▶ GDB
- ▶ Artanis web framework (pretty new)
<http://web-artanis.com/>
- ▶ Sly game programming framework
<https://dthompson.us/pages/software/sly.html>
- ▶ Official tutorial <https://www.gnu.org/software/guile/docs/guile-tut/tutorial.html>
- ▶ Manual
<https://www.gnu.org/software/guile/manual/>

- ▶ List of resources

<https://www.gnu.org/software/guile/learn/>

- ▶ #guile on freenode IRC

Topic

Introduction

All about me

A very very brief history

Why care about lisp?

Common Lisp

Brief History

Implementations

Features

Problems

Projects

Real world uses

CLASP

ECL

Resources

Clojure

Brief History

Backends

- ▶ Practical and portable
- ▶ Wants to bring Scheme out of the academic world and into the industry
- ▶ Focus on being simple, fast, and easy to learn
- ▶ Compiles to standard C using the GNU toolchain
- ▶ Runs on x86, x86-64, ARM, MIPS, Sparc64, PowerPC, and more
- ▶ Well documented in the wiki and manual
- ▶ Plenty of libraries and a library manager
- ▶ Good FFI
- ▶ Tehila game engine
<https://wiki.call-cc.org/tehila>
- ▶ Wiki software qwiki
<https://wiki.call-cc.org/egg/qwiki>
- ▶ Really just see <https://wiki.call-cc.org/Software>

- ▶ Excellent official wiki <https://wiki.call-cc.org/>
- ▶ Official manual <http://wiki.call-cc.org/man/4/The%20User's%20Manual>

- ▶ #chicken on freenode IRC

Topic

Introduction

All about me

A very very brief history

Why care about lisp?

Common Lisp

Brief History

Implementations

Features

Problems

Projects

Real world uses

CLASP

ECL

Resources

Clojure

Brief History

Backends

- ▶ First appeared in 1985
- ▶ Based off Maclisp (a now dead lisp dialect)
- ▶ Has gradually gained more and more features over the years but no major revisions really
- ▶ Considered outdated compared to modern Scheme or Common Lisp
- ▶ Some in GNU want to replace with Guile
- ▶ Pretty much just for emacs
- ▶ Allows easier extensibility than C (which the rest of emacs is written in)
- ▶ Definitely NOT designed for general purpose programming
- ▶ Emacs of course
- ▶ Any of the hundreds (thousands?) of emacs packages
- ▶ Org mode (this presentation is Org Mode -> Latex + Beamer -> PDF)

- ▶ Web servers
 - ▶ Games
 - ▶ API glue
 - ▶ All kinds of fancy IDE features
-
- ▶ Emacs Lisp Intro (C-h i and look for 'Emacs Lisp Intro' in emacs) https://www.gnu.org/software/emacs/manual/html_node/eintr/index.html
 - ▶ Emacs Lisp Reference (C-h i and look for 'Elisp' in emacs) https://www.gnu.org/software/emacs/manual/html_node/elisp/index.html
 - ▶ Learn emacs lisp in the Wiki <https://www.emacswiki.org/emacs/LearnEmacsLisp>
 - ▶ The emacs wiki <https://www.emacswiki.org/emacs/LearnEmacsLisp>
 - ▶ #emacs in freenode IRC

Topic

Introduction

All about me

A very very brief history

Why care about lisp?

Common Lisp

Brief History

Implementations

Features

Problems

Projects

Real world uses

CLASP

ECL

Resources

Clojure

Brief History

Backends

- ▶ First appeared in 1988 for the Apple Macintosh
- ▶ Simplicity and minimalism
- ▶ Single internal data type (cell)
- ▶ Numbers, symbols, and lists are the ONLY built in data types
- ▶ Differs from other lisps in not having lambda, but does not require it due to the way its quote works
- ▶ Integrated database
- ▶ "Awesome" C/Java interop
- ▶ Official documentation
<http://picolisp.com/wiki/?Documentation>
- ▶ #picolisp on freenode IRC

Topic

Introduction

All about me

A very very brief history

Why care about lisp?

Common Lisp

Brief History

Implementations

Features

Problems

Projects

Real world uses

CLASP

ECL

Resources

Clojure

Brief History

Backends

- ▶ Originally called Qi
- ▶ Static types
- ▶ Optional laziness
- ▶ integrated prolog
- ▶ Macros
- ▶ Portability
- ▶ Runs on top of various languages including SBCL
Common Lisp, Clojure, Scheme, Ruby, Python, JVM,
Haskell, Javascript.
- ▶ Free learning resources are lacking
- ▶ Official wiki
<https://github.com/Shen-Language/wiki/wiki>
- ▶ Two official books described at
<http://shenlanguage.org/>
- ▶ #shen on freenode IRC

Topic

Introduction

All about me

A very very brief history

Why care about lisp?

Common Lisp

Brief History

Implementations

Features

Problems

Projects

Real world uses

CLASP

ECL

Resources

Clojure

Brief History

Backends

- ▶ Written by Paul Graham starting in 2001
- ▶ Written in Racket
- ▶ See essay <http://www.paulgraham.com/popular.html>
- ▶ Released in 2008
- ▶ Designed to be simple
- ▶ Seems to have a very small community
- ▶ Hackernews (news.ycombinator.com) is implemented in Arc
- ▶ Seems to only have <http://www.arclanguage.org/tut.txt>

Topic

Introduction

All about me

A very very brief history

Why care about lisp?

Common Lisp

Brief History

Implementations

Features

Problems

Projects

Real world uses

CLASP

ECL

Resources

Clojure

Brief History

Backends

- ▶ Written by Robert Virding
- ▶ Work begin in 2007
- ▶ Basically developed just because the author wanted to develop a language on top of erlang and likes lisp
- ▶ Author was one of the creators of Erlang
- ▶ Provides erlang with meta programming and a feature rich REPL
- ▶ <http://lfe.io/>
- ▶ Also called Hylang
- ▶ Written by Paul Tagliamonte
- ▶ Introduced at PyCon 2013
- ▶ Transparent Lisp front end to Python
- ▶ Extreme python interop, since it is basically python
- ▶ <http://docs.hylang.org/en/latest/#>
- ▶ Heavily inspired by Clojure

- ▶ Written by Timothy Baldrige
- ▶ Our own Chris Gore has contributed
- ▶ First appeared in 2015
- ▶ Implemented in RPython and uses PyPy Garbage Collector and tracing JIT
- ▶ Basically a clojure dialect with fast startup and native code
- ▶ Very young, good for small scripts/programs
- ▶ <http://pixielang.org/>
- ▶ Scheme on the JVM
- ▶ Many consider it to have better Java integration than Clojure or ABCL
- ▶ See <https://www.gnu.org/software/kawa/> and <http://lwn.net/Articles/623349/>