

History of LISP

This web site is a pilot project of the [Computer History Museum](http://www.computerhistory.org/) [<http://www.computerhistory.org/>]'s Software expertise in the collection, preservation, and presentation of historic software. The specific goal was to locate the LISP I/1.5 system, and, as time permits, its many follow-ons. LISP was one of the earliest high-level programming languages for symbolic expressions, and dynamic type-checking. The project is being carried out by [Paul McJones](http://www.mcjones.org/dustydecks/) [<http://www.mcjones.org/dustydecks/>].

Contents

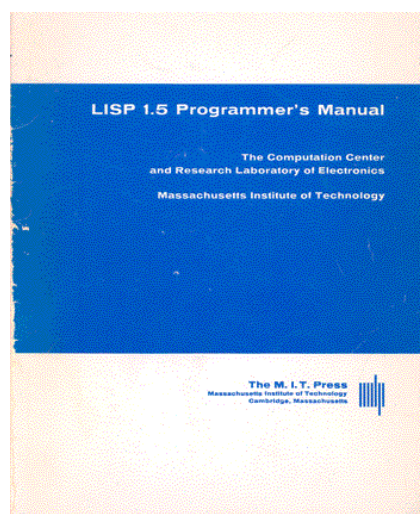
- [Acknowledgements \[#Acknowledgements_\]](#)
- [LISP I and LISP 1.5 for IBM 704, 709, 7090 \[#LISP_I_and_LISP_1.5_for_IBM_704, 709, 7090_\]](#)
- [LISP 1.5 for CTSS \[#LISP_1.5_for_CTSS_\]](#)
- [LISP 1.5 for Univac M-460 \[#LISP_1.5_for_Univac_M-460_\]](#)
- [LISP 1.5 for AN/FSQ-32/V \[#LISP_1.5_for_AN/FSQ-32/V_\]](#)
- [LISP 1.5 at Stanford \[#LISP_1.5_at_Stanford_\]](#)
- [SHARE LISP 1.5 \[#SHARE_LISP_1.5_\]](#)
- [LISP 2 \[#LISP_2_\]](#)
- [LISP 1.5 for Univac 1108 \[#LISP_1.5_for_Univac_1108_\]](#)
- [Basic PDP-1 Lisp \[#Basic_PDP-1_Lisp_\]](#)
- [LISP 1.5/1.6/MACLISP for PDP-6/10 \[#LISP_1.5/1.6/MACLISP_for_PDP-6/10_\]](#)
- [BBN LISP \[#BBN_LISP_\]](#)
- [INTERLISP \[#INTERLISP_\]](#)
- [Stanford LISP 1.6 \[#Stanford_LISP_1.6_\]](#)
- [IBM System/360 Lisp \[#IBM_System/360_Lisp_\]](#)
- [Multics Lisp \[#Multics_Lisp_\]](#)
- [Zetalisp for Lisp Machines \[#Zetalisp_for_Lisp_Machines_\]](#)
- [Scheme \[#Scheme_\]](#)
- [New Implementation of Lisp \(NIL\) \[#New_Implementation_of_Lisp_\(NIL\)_\]](#)
- [S-1 Lisp \[#S-1_Lisp_\]](#)
- [SpiceLisp \[#SpiceLisp_\]](#)
- [Standard LISP, Portable Standard Lisp \(PSL\) \[#Standard_LISP, Portable Standard Lisp \(PSL\)_\]](#)
- [Franz Lisp \[#Franz_Lisp_\]](#)
- [Common Lisp \[#Common_Lisp_\]](#)
- [Bibliographies \[#Bibliographies_\]](#)
- [Papers about LISP history \[#Papers_about_LISP_history_\]](#)
- [Published biographies and interviews \[#Published_biographies_and_interviews_\]](#)
- [Correspondence and memoranda \[#Correspondence_and_memoranda_\]](#)
- [Photographs \[#Photographs_\]](#)
- [Audio/video \[#Audio/video_\]](#)
- [Related resources \[#Related_resources_\]](#)

Acknowledgements

- Paul W. Abrahams, Jeff Barnett, Alan Bawden, Fred Blair, Robert Brayton, Daniel G. Bobrow, Phil Budne, I. Robert R. Fenichel, Richard Greenblatt, Jack Harper, Timothy P. Hart, Zane H. Healy, Al Kossow, Alan Kotkin, Marvin L. Minsky, Eric Norman, Kent M. Pitman, Stephen R. Russell, Robert A. Saunders, Olin Sibert, Her-

LISP I and LISP 1.5 for IBM 704, 709, 7090





From the Preface to the LISP 1.5 Programmer's Manual [McCarthy et al. 1962]:

"The overall design of the LISP Programming System is the work of John McCarthy and is based on his paper 'The LISP Machine' which was published in *Communications of the ACM*, April 1960.

This manual was written by Michael I. Levin.

The interpreter was programmed by Stephen R. Russell and Daniel J. Edwards.

The print and read programs were written by John McCarthy, Klim Maling, Daniel J. Edwards, and Paul V.

The garbage collector and arithmetic features were written by Daniel J. Edwards.

The compiler and assembler were written by Timothy P. Hart and Michael I. Levin.

An earlier compiler was written by Robert Brayton.

The 'LISP I Programmer's Manual,' March 1, 1960, was written by Phyllis A. Fox.

Additional programs and suggestions were contributed by the following members of the Artificial Intelligence Laboratory: Bertram Raphael, Louis Hodes, David M.R. Park, David C. Luckham, Daniel G. Bobrow, James R. Slagle, and

Source code

System

- LISP system assembly listing. "FIELD TEST ASSEMBLY OF LISP 1.5 SEPTEMBER 1961", labeled "Bonnie's Birthday Assembly". PDF (16MB) [[LISP1.5-Bonnie-sBirthdayAssembly.pdf](#)]
- See also LISP 1.5 for CTSS, below.
- LISP system assembly listing. Steve Russell may have in his archives "a listing of some version of 704/709/710". The Computer History Museum, but it will be some time before he is able to search the depths of his archives. (Personal communication with Steve Russell, 1998)

Applications

The only LISP 1.5 application source code encountered to date has been in various publications.

- Berkeley and Bobrow, editors. *The Programming Language Lisp : Its Operation and Applications*. Inform Systems Series, Volume 1. 1966. PDF [[III_LispBook_Apr66.pdf](#)]
- Daniel G. Bobrow. The LISP Program for METEOR. Pages 249-259. PDF [[III_LispBook_Apr66.pdf#page=249](#)]
- See also: Daniel G. Bobrow. METEOR: A LISP Interpreter for String Transformations. Pages 161-191. 1966. PDF [[III_LispBook_Apr66.pdf#page=161](#)]
- See also: Daniel G. Bobrow. METEOR: A LISP Interpreter for String Transformations. Memo 51, Artificial Intelligence Laboratory, MIT. 1966. PDF at MIT [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-051.pdf>]

Earlier version of the two METEOR articles in Berkeley and Bobrow.

- Malcolm Pivar and Elaine Gord. The LISP Programs for Inductive Inference on Sequences. Pages 260-271. 1966. PDF [[III_LispBook_Apr66.pdf#page=260](#)]
- See also: Malcolm Pivar and Mark Finkelstein. Automation, Using LISP, of Inductive Inference on Sequences. 1966. PDF [[III_LispBook_Apr66.pdf#page=271](#)]
- William Henneman. The LISP Program for the A-Language. Pages 318-325. PDF [[III_LispBook_Apr66.pdf#page=318](#)]

- See also: An Auxiliary Language for More Natural Expression – the A-Language. Pages 239-248. [PD](#)

- Lewis Mark Norton. ADEPT – A Heuristic Program for Proving Theorems of Group Theory. Ph.D. Thesis, MIT-LCS-TR-033, 178 pages. [PDF at MIT \[http://www.lcs.mit.edu/publications/pubs/pdf/MIT-LCS-TR-033.pdf\]](http://www.lcs.mit.edu/publications/pubs/pdf/MIT-LCS-TR-033.pdf)

Appendix III lists the source for ADEPT.

- W.A. Martin. "Symbolic Mathematical Laboratory. Ph.D. Thesis, MIT-LCS-TR-036, Project MAC, Massachusetts Institute of Technology. Includes source listing.
- Joel Moses. Symbolic Integration. Ph.D. Thesis, MIT-LCS-TR-047, Project MAC, Massachusetts Institute of Technology. [\[http://www.lcs.mit.edu/publications/pubs/pdf/MIT-LCS-TR-047.pdf\]](http://www.lcs.mit.edu/publications/pubs/pdf/MIT-LCS-TR-047.pdf)

Appendix F lists the source for SIN (symbolic integration) and SOLDIER (solutions of ordinary differential equations).

- Marvin Minsky, editor. *Semantic Information Processing*. MIT Press, 1968.
 - Daniel G. Bobrow. Natural Language Input for a Computer Problem-Solving System. Pages 146-226. Appendix 6.1 is a listing of a basic natural language parser.
 - Fischer Black. A Deductive Question Answering System. Pages 354-402. Appendix 6.1 is a listing of a basic natural language parser.

Documentation

- S. Russell. Writing and Debugging Programs. Memo 6, Artificial Intelligence Project, RLE and MIT Computation Center. [\[ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-006.pdf\]](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-006.pdf)
- J. McCarthy. Notes on the Compiler. Memo 6, Artificial Intelligence Project, RLE and MIT Computation Center. [\[ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-007.pdf\]](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-007.pdf)
- J. McCarthy. Recursive Functions of Symbolic Expressions and Their Computation by Machine. Memo 8, Artificial Intelligence Project, RLE and MIT Computation Center. 101 pages. [PDF at MIT \[ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-008.pdf\]](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-008.pdf)
- S.R. Russell. Explanation of Big "P" as of March 20, 1959. Memo 9, Artificial Intelligence Project, RLE and MIT Computation Center. [\[ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-009.pdf\]](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-009.pdf)
- K. Maling. The LISP Differentiation Demonstration Program. Memo 10, Artificial Intelligence Project, RLE and MIT Computation Center. [\[ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-010.pdf\]](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-010.pdf)
- J. McCarthy. Recursive Functions of Symbolic Expressions and Their Computation by Machine. Memo 11, Artificial Intelligence Project, RLE and MIT Computation Center. 101 pages. [PDF at MIT \[ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-008.pdf\]](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-008.pdf)

"This memorandum is a continuation of Memo 8"

- John McCarthy. Programs in LISP. Memo 12, Artificial Intelligence Project, RLE and MIT Computation Center. [\[ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-012.pdf\]](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-012.pdf)

"This memo depends only on the RLE QPR [Quarterly Project Report?] No. 53 discussion of LISP. This contains sequences of Fortran-like statements, e.g. `y=cons[ff[subst[A;y;z]];(A,B)]`"

- K. Maling. The Maling-Silver Read Program. Memo 13, Symbol Manipulation Language, Artificial Intelligence Project, RLE and MIT Computation Center. "1-20-59". [PDF at MIT \[ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-013.pdf\]](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-013.pdf)
- J. McCarthy. The Wang Algorithm for the Propositional Calculus. Memo 14, Symbol Manipulation Language, Artificial Intelligence Project, RLE and MIT Computation Center. [PDF at MIT \[ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-014.pdf\]](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-014.pdf)
- Daniel J. Edwards. LISP II Garbage Collector. Memo 19, Artificial Intelligence Project, RLE and MIT Computation Center. [\[ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-019.pdf\]](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-019.pdf)
- Phyllis Fox. LISP I Programmer's Manual. Artificial Intelligence Group, Computation Center and Research Laboratory. [Cited in McCarthy 1981 [\[http://www-formal.stanford.edu/jmc/history/lisp/lisp.html\]](http://www-formal.stanford.edu/jmc/history/lisp/lisp.html)]
- Paul W. Abrahams. Character-handling Facilities in the LISP System. Memo 22A[?], Artificial Intelligence Project, RLE and MIT Computation Center. [\[ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-022.pdf\]](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-022.pdf)
- Robert Brayton. Trace-Printing for Compiled Programs. Memo 23, Artificial Intelligence Project, RLE and MIT Computation Center. [\[ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-023.pdf\]](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-023.pdf)
- Michael Levin. Arithmetic in LISP 1.5. Memo 24, Artificial Intelligence Project, RLE and MIT Computation Center. [\[ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-024.pdf\]](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-024.pdf)

"As of present, the following parts of LISP 1.5 are working. This is an excerpt from the forth coming LIS

- Anonymous. Error Stops. Memo 25, Artificial Intelligence Project, RLE and MIT Computation Center, May [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-025.pdf]
- Michael Levin. LISP 1.5 Programmer's Manual [Errata]. Memo 28, Artificial Intelligence Project, RLE and MIT [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-028.pdf]

"The manual is intended to apply to a version of LISP 1.5 called 'LISP 1.5 Export A' which has not yet be

- John McCarthy. A Basis for a Mathematical Theory of Computation. Memo 31, Artificial Intelligence Project, MIT, 39, Massachusetts, January 1962, 33 pages. PDF at MIT [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-031.pdf]

"This paper is a corrected version of the paper of the same title given at the Western Joint Computer Conference. The mathematical logic and computation has been added."

- John McCarthy. On Efficient Ways of Evaluating Certain Recursive Functions. Artificial Intelligence Memo at MIT [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-032.pdf]
- John McCarthy. A New Eval Function. Artificial Intelligence Memo No. 34, Massachusetts Institute of Technology [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-034.pdf]
- Anonymous. LAP (LISP Assembly Program). Memo 35, Artificial Intelligence Project, RLE and MIT Computation Center [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-035.pdf]
- T. Hart and M. Levin. The New Compiler. Memo 39, Artificial Intelligence Project, RLE and MIT Computation Center [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-039.pdf]

"This memo introduces the brand new LISP 1.5 Compiler designed and programmed by Tim Hart and Michael Levin. It compiled itself by being executed interpretively."

- John McCarthy, Paul W. Abrahams, Daniel J. Edwards, Timothy P. Hart, Michael I. Levin. *Lisp 1.5 Program* [http://mhss.nease.net/lisp1.5/mccarthy.html] MIT Press [http://mitpress.mit.edu/catalog/item/default.asp?category=3&collection=1&isbn=0262081565&pubdate=1978]
- Timothy P. Hart. MACRO Definitions for LISP. Memo 57, Artificial Intelligence Project, RLE and MIT Computation Center [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-057.pdf]
- M.L. Minsky. A LISP Garbage Collector Algorithm Using Serial Secondary Storage. Memo 58 (Revised), Artificial Intelligence Project, MIT, Massachusetts Institute of Technology, December 27, 1963, 4 pages (out of order). PDF at MIT [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-058.pdf]

"Paper to be presented at the First International LISP Conference, Mexico City, Mexico, December 30 – January 2, 1964."

[Was this algorithm being contemplated for 7090 LISP on CTSS?]

- Daniel J. Edwards. Secondary Storage in LISP. Memo 63, Artificial Intelligence Project and Memorandum 63, MIT, 1963, 4 pages. PDF at MIT [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-063.pdf]

"Paper to be presented at the First International LISP Conference, Mexico City, Mexico, December 30 – January 2, 1964."

- Timothy P. Hart and Michael I. Levin. LISP Exercises. Memo 64, Artificial Intelligence Project and Memorandum 64, MIT, 1964, 27 pages. PDF at MIT [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-064.pdf]

A version of this appears in Berkeley and Bobrow.

Applications

- Anthony Valiant Phillips. A Question-Answering Routine. Memo 16, Artificial Intelligence Project, RLE and MIT Computation Center [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-016.pdf]
- John McCarthy. Programs with Common Sense. Memo 17, Artificial Intelligence Project, RLE and MIT Computation Center [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-017.pdf]

"This paper was presented at a Symposium on The Mechanization of Thought Processes, which was held at the National Physical Laboratory, 24th-27th November 1958. The papers and the discussions were published by H.M.S.O. in the Proceedings of the Symposium, with the permission of the author and of the Secretary, National Physical Laboratory."

- Louis Hodes. Some Results from a Pattern Recognition Program using LISP. Memo 18, Artificial Intelligence [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-016.pdf]
- John McCarthy. Puzzle Solving Program in LISP. Memo 20, Artificial Intelligence Project, RLE and MIT Co [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-020.pdf]
- Paul W. Abrahams. The Proofchecker. Memo 21, Artificial Intelligence Project, RLE and MIT Computation [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-021.pdf]
- Michael Levin. SIMPLIFY. Memo 27, Artificial Intelligence Project, RLE and MIT Computation Center, no [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-027.pdf]
- D.J. Edwards and T.P. Hart. The Alpha-Beta Heuristic. Memo 30 (revised), Artificial Intelligence Project, R [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-030.pdf]

"(This is a revised version of The Tree Prune (TP) Algorithm, AI Memo 30, December 4, 1961.)"

- T.G. Evans. A Heuristic program to solve Geometry Analogy Problems. Memo 46, Artificial Intelligence Pro [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-046.pdf]
- Bertram Raphael. Computer Representation of Semantic Information. Memo 49, Artificial Intelligence Proj [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-049.pdf]

LISP 1.5 for CTSS

LISP 1.5 initially ran on the "bare machine" (with its own monitor called Overlord for handling tapes, reading a *Programmer's Manual*). LISP was ported to the Compatible Time-Sharing System (CTSS) by Joel Moses and F Edwards, Tim Hart, and William Martin, with debugging assistance from Lewis Norton and Warren Teitelman

- F.J. Corbató, M.M. Daggett, R.C. Daley, R.J Creasy, J.D. Hellwig, R.H. Orenstein, and L.K. Korn. *The Comp* at bitsavers.org [http://www.bitsavers.org/pdf/mit/ctss/CTSS_ProgrammersGuide.pdf]

Source code

- LISP system assembly listing. CTSS LISP 1.5, 7 February 1966. [PDF to be available as soon as an 11"x17" sc
- LISP system assembly listing. CTSS LISP 1.5. Belonged to Joel Moses; he notes "It had handwritten annotat Unfortunately Joel believes his listing was lost during one of his many interoffice moves. (Personal commu

Documentation

- Richard A. Robnett. Suggestions for LISP Time-Sharing System. Memo 50, Artificial Intelligence Project, R [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-050.pdf]
- William Martin and Timothy Hart. Revised User's Version: Time Sharing LISP. Memo 67 (Revised), Artifici Massachusetts Institute of Technology, April 15, 1964, 8 pages. PDF at MIT [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-067.pdf]
- T. Hart. CTSS LISP NOTICE – Supplement to A.I. Memo No. 67. Memo 74, Artificial Intelligence Project, M December 7, 1964, 4 pages. PDF at MIT [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-074.pdf]
- Robert R. Fenichel and Joel Moses. A New Version of CTSS LISP. Artificial Intelligence Memo 93, Project M 2, 1966, 13 pages, PDF at MIT [ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-093.pdf]

"Summary: A new version of CTSS LISP is now available. The new system provides additional data stor; error comments, and several routines have been improved. Much irrelevant code and many bugs have a

FAP source decks and BCD listings are available. The decks are organized so as to ease the job of assembl

Without reassembling, the user can create a private LISP system in which the data storage space has be word space, and free storage.

It is assumed that the reader is familiar with the old version of LISP as described in AI Memos 67 and 7.

Applications

- Warren Teitelman. EDIT and BREAK functions for LISP. Artificial Intelligence Memo 84, Memorandum M PDF at MIT [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-084.pdf>]

"This memo describes some LISP functions which have been found to be extremely useful in easing the final debugged code. They are part of a much larger system currently being developed but may be used as a break function than that in the current CTSS version of LISP, which includes facilities for breaking on a feature. The Edit package combines many of the features of the CTSS 'ed' with a knowledge of the structure to leave a LISP system to edit a function, and therefore one may edit even when track quote is exhausted containing all of the latest definitions of the functions in that file, even where some of them may currently be in use."

- William A. Martin. Syntax and Display of Mathematical Expressions. Memo 85, Artificial Intelligence Project Technology, July 29, 1965, 22 pages. PDF at MIT [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-085.pdf>]

Describes a picture compiler written in LISP running on 7090 CTSS LISP that communicated with a display terminal.

- Warren Teitelman. FLIP – A Format List Processor. Artificial Intelligence Memo 87, Memorandum MAC-M at MIT [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-087.pdf>]

"This memo describes a notation and a programming language for expressing, from within a LISP system, the structure of a list."

- Adolfo Guzmán and Harold McIntosh. A Program Feature for CONVERT. Memo 95, Artificial Intelligence Project Technology, April 1966, 18 pages. PDF at MIT [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-095.pdf>]

"A program feature has been constructed for CONVERT, closely modeled after the similar facility found in the SAINT program."

- Adolfo Guzmán. Polybrick: Adventures in the Domain of Parallelepipeds : A world without perspective. Art 41 pages. PDF at MIT [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-096.pdf>]
- Joel Moses. Symbolic Integration. AI Memo 97, Artificial Intelligence Project, MAC-M-310, Project MAC, MIT [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-097.pdf>]

"A program has been written which is capable of integrating all but two of the problems solved by Slagle's algorithmic program and it has achieved running times two to three orders of magnitude faster than SAINT."

- Joel Moses. Symbolic Integration II. AI Memo 97a, Artificial Intelligence Project, MAC-M-327, Project MAC, MIT [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-097a.pdf>]

"The current program can integrate all the problems which were solved by SAINT and also the two problems which SAINT could not handle."

- Adolfo Guzmán and Harold V. McIntosh. CONVERT. Memo 99, Artificial Intelligence Project, Memorandum 43 pages. PDF at MIT [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-099.pdf>]

"A pattern-driven symbolic manipulation language and its preprocessor (an interpreter) are presented."

Presented at an ACM Symposium on Symbolic and Algebraic Manipulation, held March 29-31, 1966, in Washington, D.C.

To be published in Communications of the ACM, August 1966."

- Harold V. McIntosh and Adolfo Guzmán. A Miscellany of CONVERT Programming. Artificial Intelligence Project Technology, April 1967, 41 pages. PDF at MIT [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-130.pdf>]

"A collection of examples to illustrate CONVERT."

- Carl Hewitt. PLANNER: A Language for Proving Theorems. Artificial Intelligence Memo 137, Project MAC, MIT [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-137.pdf>]

Describes SCHEMATISE, "a proposal for a program that proves very elementary theorems through the most succinctly described as a cross between SNOBOL and CONVERT", and PLANNER, "a powerful tree searching theorem proving procedures".

LISP 1.5 for Univac M-460

The Univac M-460 was a military version of Univac 490. The LISP system was written by Timothy P. Hart and Robert A. Saunders. The system was bootstrapped from Lisp 1.5 on the IBM 7090 using a cross-compiler and a small amount of machine code.

- Timothy P. Hart and Thomas G. Evans. Notes on Implementing LISP for the M-460 Computer. March 1964 [[III_LispBook_Apr66.pdf#page=201](#)]

'This article describes the process used to implement LISP 1.5 on the Univac M 460 (an early military version of Univac 490, which, incidentally, there are seven) from either the left or right half of the word addressed by the same

Includes LISP source code of interpreter.

LISP 1.5 for AN/FSQ-32/V

The AN/FSQ-32/V was a military computer with 48-bit words; only one was built and it belonged to System Development Corporation. It was developed by Robert Saunders, with assistance from Tim Hart, Dan Edwards, Mike Levin, John McCarthy, and Steve Russell.

"The Q-32 implementation was the first LISP that did not run on an interpreter: all code was compiled before execution. The idea was to have an idea why other than for the fun and instruction of doing it. [As stated in Berkeley and Bobrow], the LISP was written by Robert A. Saunders as machine instructions on magnetic tape. Some fundamental code (e.g. CONS, and later the garbage collector) was written by Robert A. Saunders, personal communications, May 6, 2005.]

- Robert A. Saunders. The LISP System for the Q-32 Computer. March 1964, pages 220-238 in Berkeley and Bobrow, 1964.

This paper says "Assistance was also given by D. Edwards and M. Levin of the Massachusetts Institute of Technology. Computer time on the Stanford 7090 and PDP-1 was used in conjunction with Stanford's contract with the Defense Advanced Research Projects Agency for the development of a language for artificial intelligence." However Saunders now says "The Stanford PDP-1 played no role whatever in the Lisp port to the Q-32."

709/90/94 system, using punched cards." Russell concurs. [Personal communications to Paul McJones, 2005.]

- Robert A. Saunders. The LISP Listing for the Q-32 Compiler, and Some Samples. March 1964, pages 290-300.
- S.L. Kameny. Input-Output File and Library Functions: the Q-32 LISP 1.5 Mod. 2.5 System, TM-2337/102.
- S.L. Kameny. LISP 1.5 Reference Manual for Q-32, TM-2337/101/00, 1965 August 9.
 - United States National Bureau of Standards Collection of Computer Literature, NBS# 6523060. (Box 3) [http://www.cbi.umn.edu/collections/inv/cbi00032_3.html]
 - Also cited by Blair 1971.
- S.L. Kameny, G.P. DeFlorio, and A.H. Vorhaus. General-Purpose Display System, SP-1688, September 23, 1964.
 - United States National Bureau of Standards Collection of Computer Literature, NBS# 6421245. (Box 2) [http://www.cbi.umn.edu/collections/inv/cbi00032_3.html]
- S.L. Kameny and C. Weissman. Q-32 LISP 1.5 MOD. 2.6 SYSTEM: Operating System, Input-Output, File, and Library Functions. March 1964.
 - United States National Bureau of Standards Collection of Computer Literature, NBS# 6623332. (Box 3) [http://www.cbi.umn.edu/collections/inv/cbi00032_3.html]

LISP 1.5 at Stanford

There was LISP 1.5 work at Stanford on the IBM 7090 or 7094. Steele & Gabriel say "The PDP-1 Lisp at Stanford was written by Robert A. Saunders. Steve Russell says:

"I sure don't remember anything about that. ...

While I was at Stanford, I did some work on 7090 Lisp, but worked primarily on getting the PDP-1 and PDP-6 Lisp.

Since the PDP-1 system was used as a multi-user system with only 4k PDP-1 words per user, and very non-s to Paul McJones, May 15, 2005]

- Horace Enea. Clock Function for LISP 1.5. AIM-4, 2 pages, August 1963.
- S.R. Russell. Improvements in LISP Debugging, Stanford University Artificial Intelligence Project, AIM-10,
- Dean E. Wooldridge. The New LISP System (LISP 1.55). Stanford University Artificial Intelligence Project, ,
- Jan Hext. An expression input routine for LISP. Memo No. 18, Stanford Artificial Intelligence Project, July

SHARE LISP 1.5

- Appendix I [<http://mhss.nease.net/lisp1.5/node45.html>] of *Lisp 1.5 Programmer's Manual* describes a SH
- "The Artificial Intelligence Project at Stanford University has produced a version of LISP 1.5 to be distr and is available from Stanford. The system should be available from SHARE by the end of March 1965.

LISP 2

Documentation

Most of these items come from Stoyan [<http://www8.informatik.uni-erlangen.de/html/lisp/histlit.html>]. Com [<http://hopl.murdoch.edu.au/showlanguage.prx?exp=264&language=LISP 2>].

- J. McCarthy. Storage Conventions in LISP 2, Stanford University Artificial Intelligence Project, AIM-8 (Pre
 - R.W. Mitchell. LISP 2 Specifications Proposal, Stanford Artificial Intelligence Laboratory AIM-21, Stanford [<http://www-formal.stanford.edu/jmc/history/lisp/node7.html>], Stoyan [<http://www8.informatik.uni-erl>
 - Daniel G. Bobrow. The COMIT Feature in LISP 2. A I Memo 76, MAC Memo 219, LISP II Project Memo 2, I [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-076.pdf>]
-
- P.W. Abrahams, J.A. Barnett, E. Book, D. Firth, L. Hawkinson, S.L. Kameny, M.I. Levin, R.A. Saunders, C. ' Joint Computer Conference, 1966.
 - Paul W. Abrahams. The Implementation of LISP 2. ??? 1967.
 - Paul Abrahams. Conventions for writing LISP 2 syntax Equations. SBC Memorandum 1967.
 - Paul W. Abrahams. LISP 2 Debugging. Tech Memo, 1967.
 - Paul W. Abrahams. LISP 2 Specifications, Systems Development Corporation Technical report TM-3417/2C
 - Paul Abrahams, Donna Firth. LISP 2 Language Specifications. Tech Memo, 1967. [Same as previous item?]
 - Anonymous. LISP 2 Specifications Conference, Stanford University, Stanford, 1963.
 - D. Anschultz. LISP 2 Compiler Register Counter and Code Generator Specifications. Tech Memo 1967.
 - D. Anschultz. LISP 2 Compiler Machine Link Specifications. Tech Memo 1967.
 - J.A. Barnett. SIM. An S-Expression Pattern-Matching Function. Tech Memo 1966.
 - Jeff Barnett. LISP 2 Compiler Specifications. Tech Memo 1967.
 - Jeff Barnett. LISP 2 Compiler Context Resolver Language and Processor Specifications. Tech Memo 1967.
 - Erwin Book. The LISP 2 Syntax Translator. Tech Memo 1966.
 - D. Crandell. LISP 2 Assembly Program (LAP) Specification. Tech Memo 1967.
 - Donna Firth. LISP 2 Input/Output Specifications. Tech Memo 1967.
 - Donna Firth and S.L. Kameny. Syntax of LISP 2 Tokens. Tech Memo 1966.
 - L. Hawkinson. LISP 2 Core Image Generator (CIG) Specifications. Tech Memo 1967.
 - L. Hawkinson. LISP 2 Internal Storage Conventions. TM-3147/550/00, 1967 April.
 - United States National Bureau of Standards Collection of Computer Literature, NBS# 70400404. (Box 1 [http://www.cbi.umn.edu/collections/inv/cbi00032_3.html])
 - L. Hawkinson. Technical Progress Report LISP 2 (Period Ending July 6, 1966). Information International I
 - M.V. Howard. Operating Instructions for the LISP 2 Supervisor in the LISP 2 Core Image. Tech Memo 1966
 - S.L. Kameny. LISP 2 Source Language Syntax. Specifications for Syntax Translator. Tech Memo 1966.
 - S.L. Kameny and L. Hawkinson. LISP II Internal Storage Conventions. Tech Memo 1965.
 - S.L. Kameny and L. Hawkinson. LISP II Intermediate Language. Tech Memo 1965.

- Stanley L. Kameny, Lowell Hawkinson, Clark Weissman, Jeffrey A. Barnett, Robert A. Saunders, Erwin Boc
- Robert E. Long. LISP 2 Storage Management Paging of Binary Programs. Tech Memo 1967.
- R. A. Saunders, J.A. Barnett, Donna Firth. The LISP 2 Compiler. Tech Memo 1966.
- P. Stygar. LISP 2 Garbage Collector Specifications. Tech Memo 1967.
- P. Stygar. LISP 2 Storage Management : The "Growing Pain" Problem. Tech Memo 1967.
- TM-3163 The LISP 2 Programming Language and System. SDC 1966.

LISP 1.5 for Univac 1108

"In the late 1960's, *Eric Norman* of the University of Wisconsin - Madison developed a LISP 1.5 dialect for 1

The LISP system developed by Norman consisted of approximately 5,000 lines of Univac assembly language. Several interesting applications were developed under or ported to Univac LISP including: *Planner* [<http://www.cs.umbc.edu/courses/undergraduate/CMSC331/fall00/notes/intro1p.pdf>] (an early planning LISP); *Fuzzy* (a system that worked with *Fuzzy Logic* [<http://www-2.cs.cmu.edu/Groups/AI/html/faqs/ai/>], package, a *Math Library*, a source *Editor*, and a *Debugger*. The original source code for all of the above iter discussed in some detail." [Jack Harper, <http://www.frobenius.com/univac.htm>] [<http://www.frobenius.co>]

"At the time, the Computing Center here at Wisconsin had decided to purchase a Univac 1108. It couldn't b an 1108 up in Minneapolis. Just about all the interpreter was developed that way. That is, I submitted a d arrived on campus, LISP was already running on it." [Eric Norman, personal communication to Paul McJoi

Source code and documentation

- Jack Harper. Univac 1100 LISP Documentation and Source Browser. <http://www.frobenius.com/source.ht>
 - Eric Norman. LISP Primer. <http://www.frobenius.com/primer.htm> [<http://www.frobenius.com/prime>
 - Eric Norman and Rick LeFaivre. LISP Reference Manual. <http://www.frobenius.com/reference.htm> [<ht>
 - Eric Norman. Interpreter source code (ASM). Date??? <http://www.frobenius.com/interpreter.htm> [<htt>
 - Eric Norman. Compiler source code (LISP). Date??? <http://www.frobenius.com/compiler.htm> [<http://v>
 - Rick LeFaivre. Description of 1108 MICRO-PLANNER. October 16, 1973. <http://www.frobenius.com/pl>
 - ????. MICRO-PLANNER source code (LISP). <http://www.frobenius.com/planner.htm> [<http://www.frob>
 - Jack Harper. First-Order Predicate Calculus Theorem Prover source code (LISP). Written for University 1974. <http://www.frobenius.com/theorem.htm> [<http://www.frobenius.com/theorem.htm>]
 - Rick LeFaivre. Fuzzy Logic source code (LISP). Date??? <http://www.frobenius.com/fuzzy.htm> [<http://v>
 - Eric Norman and Rick LeFaivre. Editor source code (LISP). Date??? <http://www.frobenius.com/editor>.
 - Eric Norman and Rick LeFaivre. Debugger source code (LISP). Date??? <http://www.frobenius.com/deb>
 - Eric Norman and Rick LeFaivre. PrettyPrint source code (LISP). Date??? <http://www.frobenius.com/pi>
 - Eric Norman and Rick LeFaivre. Arithmetic source code (LISP). Date??? <http://www.frobenius.com/ar>

Basic PDP-1 LISP

L. Peter Deutsch developed PDP-1 LISP at M.I.T.:

"I wrote PDP-1 Lisp because I had a strong mathematical bent, I'd become intrigued with the Lisp language Programmer's Manual at MIT, and I wanted to have an interactive Lisp implementation to play with rather mind that I was in high school at the time -- 1960-1964.) I'd ingratiated myself with the folks at the TX-0 (a Peter Deutsch, quoted in [lisp_doc.txt](#) [<http://simh.trailing-edge.com/kits/lispswre.zip>] accompanying Bob

- Basic PDP-1 LISP is described in detail, including full PDP-1 assembly language source code, in this article:
 - L. Peter Deutsch and Edmund C. Berkeley. The LISP Implementation for the PDP-1 Computer. March 1 [[III_LispBook_Apr66.pdf#page=336](#)]
 - A machine-readable version has been recreated from the book, and made to run on Bob Supnik's SIMH [<http://simh.trailing-edge.com/software.html>].
 - Another online simulator written in Java -- see <http://lcs.www.media.mit.edu/groups/el/projects/pdp>

LISP 1.5/1.6/MACLISP for PDP-6/10

In AIM-190, John L. White wrote:

"MACLISP is a descendent of the first operational program ever written for the Digital Equipment Corporation and debugged in the spring of 1964 by the staff at DEC and members of the Tech Model Railroad Club – an supplied a continuing stream of prospective programmers to the Artificial Intelligence Group. This unlikely assembling and punching the program on paper tape. At about that time a working PDP/6 was delivered to

Ideas from several other implementations influenced the initial design, notably the CTSS version on the IBL dispense with the a-list in the implementation, a major factor in the space economy and running speed of M adjunct to the system – compilation is done 'off-line', and the resulting LAP code loaded into the system wh

Source code

- Alan Bawden. Partial online archive of AI filesystem of ITS: The Incompatible Time Sharing System. <http://ftp.its.os.org/its/ai/lisp.tgz> [<ftp://ftp.its.os.org/its/ai/lisp.tgz>] says:

```
Directories with MacLISP relevance are:
LISP; -- FASL files
LSPDMP; -- Dump files
L; -- MIDAS sources for main lisp file, and for others
COMLAP; -- COMpiler (with FASLAP) and LAP assembler
LSPSRC; -- Various auxillary out-of-core files written in lisp
NILCOM; -- More LSPSRC. Mostly lisp macros, etc.
LSPMAI; -- Bug mail and other notices.
LISP20; -- Stuff for 20X MacLisp
```

- Phil Budne. Source files and text documents. <ftp://ftp.ultimate.com/pdp10/macisp/> [<ftp://ftp.ultimate.com/pdp10/macisp/>]
 - These items were obtained from a TOPS-20 system at XKL.com:
 - MACLISP LISP sources including COMPLR.LSP [lisp.tar.gz](ftp://ftp.ultimate.com/pdp10/macisp/lisp.tar.gz) [<ftp://ftp.ultimate.com/pdp10/macisp/lisp.tar.gz>]
 - MACLISP .DOC and .MID files including LISP-NEWS.DOC from April 28, 1981. [lisp2.tar.gz](ftp://ftp.ultimate.com/pdp10/macisp/lisp2.tar.gz) [<ftp://ftp.ultimate.com/pdp10/macisp/lisp2.tar.gz>]
 - These items were obtained from tape images of Chemistry Department disk packs from the Stevens Tech
 - MIDAS sources for building MACLISP version 804 and MIDAS. Circa June 1979. [macisp804.tar.gz](ftp://ftp.ultimate.com/pdp10/macisp/macisp804.tar.gz)
 - MACLISP version 861 (no additional MIDAS files) and MacLisp Reference Manual, March 3, 1979. [macisp861.tar.gz](ftp://ftp.ultimate.com/pdp10/macisp/macisp861.tar.gz)

Documentation

- William A. Martin. PDP-6 LISP Input-Output for the Dataphone. Memo 79, Artificial Intelligence Project, MIT, June 1, 1965, 6 pages. [PDF at MIT](http://publications.ai.mit.edu/ai-publications/pdf/AIM-079.pdf) [<http://publications.ai.mit.edu/ai-publications/pdf/AIM-079.pdf>]

"Abstract: A version of LISP 1.5 for the PDP-6 Computer has been extended to include IO through the d Project MAC time sharing and LISP programs running on the PDP-6. The method of handling input-output and paper tape reader. Three useful LISP functions are presented as examples of dataphone programming

- William A. Martin. PDP-6 LISP Input-Output for the Display. Memo 80, Artificial Intelligence Project, MIT, Mer 28, 1965, 22 pages. [PDF at MIT](http://publications.ai.mit.edu/ai-publications/pdf/AIM-080.pdf) [<http://publications.ai.mit.edu/ai-publications/pdf/AIM-080.pdf>]

"Abstract: An intermediate level language for display programming has been embedded in LISP 1.5. The facilities for both generation and analysis of display information. ..."

- Peter Samson. PDP-6 LISP. Artificial Intelligence Memo 98, Project MAC Memo MAC-M-313, Massachusetts Institute of Technology, 1965. [\[http://publications.ai.mit.edu/ai-publications/pdf/AIM-098.pdf\]](http://publications.ai.mit.edu/ai-publications/pdf/AIM-098.pdf)

"This is a mosaic description of PDP-6 LISP, intended for readers familiar with the LISP 1.5 Programme features (e.g., the display) are experimental and subject to change; in such respects this should not be re

- Anonymous. PDP-6 LISP (LISP 1.6) Revised. A.I. Memo No. 116A, Project MAC, April 1967, 23 pages, (corrected) [<http://publications.ai.mit.edu/ai-publications/pdf/AIM-116a.pdf>]

"This is a mosaic description of PDP-6 LISP, intended for readers familiar with the LISP 1.5 Programme features, such as the display, are subject to change. Thus, consult a PDP-6 systems programmer for any

the system tape."

Steel and Sussman note: "The report does not bear the author's name, but Jeffrey P. Golden [Golden 19'

- Roland Silver. Incorporating MIDAS Routines into PDP6 LISP. Systems Program Memo, Artificial Intelligence pages. PDF at MIT [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-127.pdf>]

"Some PDP6 LISP users have felt a need for a way to incorporate MIDAS subroutines into LISP. LISP h microtape."

- Roland Silver. LISP Linkage Feature: Incorporating MIDAS Routines into PDP6 LISP. Systems Program M Technology, October 1967, 5 pages (corrected edition of Memo 127, March 1967). PDF at MIT [<ftp://publici>]
- John L. White. Additions to LAP. Artificial Intelligence Memo 132, Program Memo, Project MAC, Massachi [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-132.pdf>]
- John L. White. Matrix Inversion in LISP. Artificial Intelligence Memo 136, PDP-6 Program Memo, Project I [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-136.pdf>]

"Very soon there will appear on the vision library tape a file named @IAS which is a collection of compil

- Jon L. White. PDP-6 LAP. Artificial Intelligence Project Memo 152, Project MAC, Massachusetts Institute o [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-152.pdf>]
- John White. Time-Sharing LISP for the PDP-6. Artificial Intelligence Memo 157, Project MAC, Massachuse [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-157.pdf>]

"This Memo. written in the style and convention of A.I. memo. No. 116A, may be considered an addendi

- Donald Eastlake III. ITS 1.4 Reference Manual. Artificial Intelligence Memo No. 161, MAC-M-377, Massach [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-161.pdf>]

ITS (Incompatible Timesharing System) was the operating system on which MacLisp was developed.

- D. Eastlake, R. Greenblatt, J. Holloway, T. Knight, S. Nelson. ITS 1.5 Reference Manual. Artificial Intelligen 180 pages. PDF at MIT [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-161A.pdf>] Reconstructed te
- Thomas O. Binford. Display Functions in LISP. Artificial Intelligence Memo No. 182, Project MAC, Massacl [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-182.pdf>]
- John L. White. An Interim LISP User's Guide. Artificial Intelligence Memo No. 190, Project MAC, Massachi [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-190.pdf>]

"The substance of this memo is to initiate the na?ve LISP user into the intricacies of the system at the P comprehensive document will be issued, consisting of an elementary introduction to LISP, a self-primer comprehensive guide will then replace A.I. memos numbers 116A, 152, 157, the LISP Progress Report, tl

The Notation section says "'MACLISP' refers to the PDP/6 implementation of the programming languag

- Joel Moses. The Function of FUNCTION in LISP or Why the FUNARG Problem Should be Called the Envir Technology, June 1970, 15 pages. PDF at MIT [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-199>]
- Jeffrey P. Golden. A User's Guide to the A. I. Group LISCOM Lisp Compiler: Interim Report. Memo No. 210 December 1970. PDF at MIT CSAIL [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-210.pdf>]
- David A. Moon. MacLISP Reference Manual, Revision 0. Project MAC, Massachusetts Institute of Technolo
- Anonymous. MACLISP Reference Manual, March 6, 1976. LISP INFO file from MIT TECO EMACS Version [http://pdp-10.trailing-edge.com/mit_emacs_170_teco_1220/01/info/lisp.info.html]

"This file is a fast edit of the MACLISP Reference Manual Dated March 6, 1976, so expect stuff to be out Sources."

From the same release: MACLISP-COMMANDS INFO file. Online at pdp-10.trailing-edge.com [http://pdp-10.trailing-edge.com/mit_emacs_170_teco_1220/01/info/macisp-commands.info.html]

- Guy Lewis Steele, Jr. Data Representations in PDP-10 MacLISP. AI Memo 420, Artificial Intelligence Labor

MIT [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-420.pdf>]

- Guy Lewis Steele, Jr. Fast Arithmetic in MacLISP. AI Memo 421, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-421.pdf>]

"This paper was presented at the MACSYMA Users Conference, Berkeley, California, July 1977."

- MacLISP information from ITS AI:INFO at www.avanthur.com [http://www.avanthur.com/~healyzh/_info_/lisp.news] from February 17, 1982.
- LISP NEWS [http://www.avanthur.com/~healyzh/_info_/lisp.news] from February 17, 1982.
- Anonymous. MacLisp Reference Manual, March 3, 1979. Online at www.avanthur.com [<http://www.avanthur.com>]
- K. M. Pitman. The Revised MacLisp Manual, M.I.T. Laboratory for Computer Science Technical Report MIT/LCS/TR-100

"Abstract [from www.multicians.org bibliography [<http://www.multicians.org/biblio.html>]]: MACLISP (for Computer Science) and the MIT Artificial Intelligence Laboratory for use in artificial intelligence research. Important dialects (for example Lisp Machine Lisp and NIL) have evolved from MacLisp. David Moon's Moonual) provided in-depth coverage of a number of areas of the MacLisp world. Some parts of that document (Moonual's I/O system); other parts are no longer accurate due to changes that have occurred in the language of Lisp, but is not intended as tutorial. It is intended primarily as a reference manual; particularly, it comes from the MacLisp community. It has been borrowed directly from the Moonual, but there has been a shift in emphasis. While the Moonual was in the way of examples and style notes. Also, since Moon had worked on the Multics implementation, the MacLisp. While it is hoped that Multics users will still find the information contained herein to be useful those were the implementations most familiar to the author."

Applications

- Adolfo Guzmán. Decomposition of a Visual Scene into Bodies. Artificial Intelligence Memo 139, Project MAC, MIT, 1967, 47 pages. PDF at MIT [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-139.pdf>]

"A shorter but improved version of this memorandum has been published: 6. Guzmán, A. Decomposition of a Visual Scene into Bodies. *Computer Conference*. Thompson Book Co., Washington, D.C."

- Stephen Smoliar. EUTERPE-LISP: A LISP System with Music Output. Artificial Intelligence Memo 141, Project MAC, MIT, 1967, 47 pages. PDF at MIT [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-141.pdf>]
- Richard Bogen, Jeffrey Golden, Michael Genesereth, and Alexander Doohovskoy. MACSYMA Reference Manual. MIT, Massachusetts Institute of Technology, Version 9, Second Printing, December 1977, v+308+x pages. PDF at [bitsavers.org](http://www.bitsavers.org/pdf/mit/macsyma/MACSYMA_RefMan_V9_Dec77.pdf) [http://www.bitsavers.org/pdf/mit/macsyma/MACSYMA_RefMan_V9_Dec77.pdf]

BBN LISP

LISP for PDP-1 and SDS 940 were written by Daniel Bobrow and D. L. Murphy at Bolt Beranek and Newman, Inc. The PDP-1 version was "rewritten" starting from L. Peter Deutsch's Basic PDP-1 LISP.

LISP for PDP-10 was written by D.L. Murphy and Alice Hartley at BBN.

Source

- Dan Murphy had an extensive archive of old sources for BBN LISP (and many other things including TENE) and was an overzealous sysadmin in 1987. He still has a set of files in LISP and PDP-10 assembly language. (Personal communication with the author)
- Section 4 [BBN-LISP-System_Feb1966.pdf#page=49] of Bobrow et al. 1966 below gives the source code for the PDP-10 version.

Documentation

- Daniel G. Bobrow, D. Lucille Darley, Daniel L. Murphy, Cynthia Solomon, and Warren Teitelman. The BBN LISP System. AF19(628)-5065, BBN Report 1346, Bolt Beranek and Newman Inc., Cambridge, Massachusetts, February 1966

This report describes the LISP system for a PDP-1 with 8392 18-bit words of 5 microsecond core memory. It describes the internal structure including paging and binding techniques. Section III defines all the system primitives.

implemented in LISP.

- W. Teitelman, D. G. Bobrow, A. K. Hartley, and D. L. Murphy. *BBN-LISP: TENEX Reference Manual*. Bolt August 1972). PDF at bitsaver.org [http://www.bitsavers.org/pdf/bbn/tenex/TenexLispRef_Aug72.pdf]

Papers

- Daniel G. Bobrow and Daniel L. Murphy. Structure of a LISP system using two-level storage. *Communications of the ACM* [http://doi.acm.org/10.1145/363162.363185]

"Abstract: In an ideal list-processing system there would be enough core memory to contain all the data been used to build a LISP system utilizing a drum for its principal storage medium, with a surprisingly careful segmentation of system programs, allocation of virtual memory to allow address arithmetic for the lists. A scheme for binding variables is described which is good in this environment and allows for complex declarations."

Describes BBN LISP on a Digital Equipment Corporation PDP-1 with a core memory of 16K 18-bit word milliseconds average access time).

- Daniel G. Bobrow and Daniel L. Murphy. A note on the efficiency of a LISP computation in a paged machine. *ACM DL* [http://doi.acm.org/10.1145/363567.363581]

"Abstract: The problem of the use of two levels of storage for programs is explored in the context of a LISP on a drum. Details of timing are given for one particular problem."

Discusses BBN LISP on the SDS 940, which had 16K of 24-bit core memory and 1M words of drum memory.

- Dan Murphy. TENEX and TOPS-20 Papers. Online at www.opost.com/dlm/tenex [http://www.opost.com/dlm/tenex]

TENEX and TOPS-20 were popular operating systems for running LISP. See especially [Origins and Development](#) for extensive background on the PDP-6, PDP-10, LISP, and more.

INTERLISP



PDP-10, D-machine: Xerox PARC. References at HOPL [<http://hopl.murdoch.edu.au/showlanguage.prx?exp=>

Documentation

- Warren Teitelman et al. *INTERLISP Reference Manual*. Xerox Palo Alto Research Center, Palo Alto, California. 1974. PDF at bitsaver.org [http://www.bitsavers.org/pdf/xerox/interlisp/1974_InterlispRefMan.pdf]
- Warren Teitelman et al. *INTERLISP Reference Manual*. Xerox Palo Alto Research Center, Palo Alto, California. 1974. Scanned. Personal communication, 7 March 2005.]

- J. Strother Moore II. The INTERLISP Virtual Machine Specification. Technical Report CSL 76-5, Xerox Pal

"In order to implement the INTERLISP System (as described in *The INTERLISP Reference Manual* by implement the INTERLISP Virtual Machine, since Virtual Machine compatible source code for the rest

- [Warren Teitelman et al.] INTERLISP Documentation for TOPS-20 operating system. Chapter 1 missing. D decus:2lisp.tty.2 through decus:a3lisp.tty.2. [Online at pdp-10.trailing-edge.com](http://pdp-10.trailing-edge.com) [<http://pdp-10.trailing-edge.com>]

Interlisp-VAX

- [References at HOPL](http://hopl.murdoch.edu.au/showlanguage.prx?exp=6817&language=Interlisp-VAX) [<http://hopl.murdoch.edu.au/showlanguage.prx?exp=6817&language=Interlisp-VAX>]

Papers

- L. Peter Deutsch. A Lisp machine with very compact programs. *Proceedings of the Third International Joint Conference on Artificial Intelligence*, 1977, pages 101-105.
- L. Peter Deutsch and Daniel G. Bobrow. An efficient, incremental, automatic garbage collector. *Communications of the ACM* [http://doi.acm.org/10.1145/360336.360345]
- L. Peter Deutsch. Experience with a microprogrammed Interlisp system. *Proceedings of the 11th annual workshop on LISP and functional programming*, Stanford University, 1980, pages 243-247. [ACM DL](http://portal.acm.org/citation.cfm?id=804321) [http://portal.acm.org/citation.cfm?id=804321]
- Richard R. Burton, L. M. Masinter, Daniel G. Bobrow, Willie Sue Haugeland, Ronald M. Kaplan, and B. A. Smith. *on LISP and functional programming*, Stanford University, 1980, pages 243-247. [ACM DL](http://portal.acm.org/citation.cfm?id=804321) [http://portal.acm.org/citation.cfm?id=804321]
- Larry M. Masinter and L. Peter Deutsch. Local optimization in a compiler for stack-based Lisp machines. *Proceedings of the 11th annual workshop on LISP and functional programming*, Stanford University, pages 223-230. [ACM DL](http://portal.acm.org/citation.cfm?id=802810) [http://portal.acm.org/citation.cfm?id=802810]
- Larry M. Masinter. Interlisp-VAX: A Report. Report No. SUN-CS-81-879, Department of Computer Science, Stanford University, 1981. [\[http://historical.ncstrl.org/litesite-data/stan/CS-TR-81-879.pdf\]](http://historical.ncstrl.org/litesite-data/stan/CS-TR-81-879.pdf)
- Raymond L. Bates, David Dyer, and Johannes A.G.M. Koomen. Implementation of Interlisp on the VAX. *Proceedings of the 11th annual workshop on LISP and functional programming*, Pittsburgh, Pennsylvania, United States, 1982, pages 81-87. [ACM DL](http://portal.acm.org/citation.cfm?id=802810) [http://portal.acm.org/citation.cfm?id=802810]

Stanford LISP 1.6

Source

- Lynn H. Quan and Whitfield Diffie et al. LISP 1.6. DECUS Library Program 10-59, May 1, 1973. [Online at pdp-10.trailing-edge.com](http://pdp-10.trailing-edge.com/decuslib10-01/index.html) [<http://pdp-10.trailing-edge.com/decuslib10-01/index.html>]

Although the DECUS Library Catalog lists this version as authored and submitted by Quan and Diffie, a LISP 1.6 (Utah version) 1-May-73".

Abstract from DECUS Library Catalog: "The interactive LISP 1.6 system has been developed for the PDP-10. The user is familiar with either some other LISP system or the LISP 1.5 Primer by Clark Weissman. This LISP 1.5 systems. In addition, there are several new features such as an arbitrary precision integer package, control the size of memory spaces, a standard relocating loader assembly language or compiled program produces machine code. Compiled functions are approximately ten times as fast and also take less memory. Structures are described; then the functions for operating on them. The appendices present more details on auxiliary packages."

1	25(7)	18-Jul-80	59.inf	[http://pdp-10.trailing-edge.com/decuslib10-01/01/43,50073/59.inf.h
1	128(36)	17-Dec-73	alvine.lsp	[http://pdp-10.trailing-edge.com/decuslib10-01/01/43,50073/alvi
48	6144(36)	17-Dec-73	alvine.mac	[http://pdp-10.trailing-edge.com/decuslib10-01/01/43,50073/alvi
141	18048(36)	17-Dec-73	complr.	[http://pdp-10.trailing-edge.com/decuslib10-01/01/43,50073/complr.
7	896(36)	17-Dec-73	grin.	[http://pdp-10.trailing-edge.com/decuslib10-01/01/43,50073/grin..htm
21	2688(36)	17-Dec-73	lap.	[http://pdp-10.trailing-edge.com/decuslib10-01/01/43,50073/lap..html]
11	1408(36)	17-Dec-73	lisp.lsp	[http://pdp-10.trailing-edge.com/decuslib10-01/01/43,50073/lisp.l
150	19154(36)	17-Dec-73	lisp.mac	[http://pdp-10.trailing-edge.com/decuslib10-01/01/43,50073/lisp.m
130	16531(36)	17-Dec-73	loader.mac	[http://pdp-10.trailing-edge.com/decuslib10-01/01/43,50073/load
12	1536(36)	17-Dec-73	smile.	[http://pdp-10.trailing-edge.com/decuslib10-01/01/43,50073/smile..h
1	128(36)	17-Dec-73	symmak.mac	[http://pdp-10.trailing-edge.com/decuslib10-01/01/43,50073/symm
10	1280(36)	17-Dec-73	trace.	[http://pdp-10.trailing-edge.com/decuslib10-01/01/43,50073/trace..h

- Carl Engelman. MATHLAB. DECUS Library Program 10-142, November 7, 1969. [Online at pdp-10.trailing-](http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/142.inf)

We include this item because it includes (parts of) a Stanford LISP 1.6 system and because it is interesti

Abstract from DECUS Library Catalog: "MATHLAB is an on-line system providing machine aid for the performing, automatically and symbolically, such common procedures as simplification, substitution, d inverse Laplace transforms, the solution of linear differential equations with constant coefficients, the s supplies fairly elaborate bookkeeping facilities appropriate to its on-line operation."

1	25(7)	29-Apr-75	142.inf	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/142.inf]
1	128(36)	21-Nov-69	alvine.lsp	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/alvin]
31	3968(36)	21-Nov-69	alvine.mac	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/alvin]
3	302(36)	6-Apr-71	bhand1.lap	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/bhand]
1	128(36)	21-Nov-69	bignum.lsp	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/bignu]
29	3670(36)	20-Apr-71	bignum.mac	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/bignu]
14	1781(36)	7-Apr-71	bpre1.l	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/bpre1.l]
40	5037(36)	7-Apr-71	brats1.l	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/brats1]
40	5072(36)	6-Apr-71	bsimp1.l	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/bsimp1]
18	2203(36)	14-Oct-70	celt6.l	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/celt6.l]
138	17664(36)	12-Nov-70	cmp47x.	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/cmp47x..]
18	2304(36)	21-Nov-69	debug.	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/debug..ht]
14	1789(36)	22-Jul-71	dif8.l	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/dif8.l.ht]
41	5176(36)	14-Jun-71	disp9.l	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/disp9.l]
33	4201(36)	11-Mar-71	fact6.l	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/fact6.l]
5	640(36)	21-Nov-69	grin.	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/grin..html]
23	2879(36)	7-Oct-70	ilt2.l	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/ilt2.l.ht]
21	2597(36)	30-Sep-70	int2.l	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/int2.l.ht]
16	2048(36)	20-Apr-71	lap.	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/lap..html]
11	1337(36)	19-Jan-72	lde6.l	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/lde6.l.ht]
5	640(36)	21-Nov-69	lisp.lsp	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/lisp.ls]
123	15743(36)	19-Mar-71	lisp.mac	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/lisp.ma]
3	384(36)	21-Nov-69	lisp.txt	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/lisp.tx]
129	16512(36)	21-Nov-69	loader.mac	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/load]
1	128(36)	5-Aug-70	macros.l2	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/macros]
44	5619(36)	14-Jan-72	mat11.l	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/mat11.l]
449	57380(36)	19-Jan-72	mathla.sav	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/mathl]
58	7415(36)	23-Jul-71	mlab37.l	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/mlab37]
9	1152(36)	21-Nov-69	smile.	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/smile..ht]
9	1117(36)	16-Mar-71	sol6.l	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/sol6.l.ht]
1	128(36)	21-Nov-69	symmak.mac	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/symma]
15	1873(36)	20-Apr-71	top9.l	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/top9.l.ht]
6	768(36)	29-Apr-71	trace.	[http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/trace..ht]

Includes MATHLAB.

- University of California, Irvine. UCI-LISP. DECUS Library Program 10-210, March 27, 1975. [Online at pdp](http://pdp-10.trailing-edge.com/decuslib10-02/01/43,50216/210.inf)

Abstract from DECUS Library Catalog: "This is U.C.I.'s version of LISP. The DECUS write-up is compos "DOC" file on the tape, in upper case. It is also available in much more readable upper and lower case di University of California, Irvine."

1	25(7)	8-Oct-75	210.inf	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/210.inf
2	690(7)	26-Sep-73	arith.lsp	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/arith
14	8740(7)	27-Mar-75	arith.lst	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/arith
4	1950(7)	24-Mar-75	arith.mac	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/arith
2	1225(7)	24-Mar-75	arith.rel	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/arith
17	10735(7)	1-Feb-73	break.com	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/break
84	53305(7)	6-Sep-74	break.doc	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/break
50	31760(7)	26-Sep-73	break.lap	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/break
1	255(7)	25-Sep-73	break.lsp	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/break
12	7475(7)	15-Oct-72	cbugs.	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/cbugs..ht
10	6310(7)	6-Sep-74	change.doc	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/chang
1	635(7)	8-Sep-73	compat.lsp	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/compa
120	76230(7)	10-Sep-73	complr.com	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/compl
305	194755(7)	1-Sep-78	complr.lap	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/compl
108	68750(7)	16-Nov-72	complr.org	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/compl
159	20337(36)	27-Mar-75	complr.sav	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/compl
51	32575(7)	26-Sep-73	edit.com	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/edit.co
259	165745(7)	21-Mar-79	edit.doc	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/edit.do
174	111185(7)	26-Sep-73	edit.lap	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/edit.la
3	1435(7)	26-Sep-73	edit.lsp	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/edit.ls
2	900(7)	26-Sep-73	editst.lap	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/edits
23	14150(7)	19-Sep-73	errorx.com	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/error
70	44280(7)	26-Sep-73	errorx.lap	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/error
2	825(7)	13-Jun-73	errorx.lsp	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/error
28	17700(7)	6-Sep-74	index.doc	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/index
37	23345(7)	6-Sep-74	intro.doc	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/intro
21	12945(7)	25-Sep-73	lap.	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/lap..html
21	12945(7)	27-Mar-75	lapv.	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/lapv..html
28	3514(36)	26-Mar-75	lisp.lod	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/lisp.lo
57	7186(36)	27-Mar-75	lisp.low	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/lisp.lo
120	15360(36)	27-Mar-75	lisp.shr	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/lisp.sh
5	552(36)	27-Mar-75	lisp.sym	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/lisp.sy
641	409955(7)	27-Mar-75	loader.lst	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/load
282	179950(7)	24-Mar-75	loader.mac	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/load
38	4851(36)	24-Mar-75	loader.rel	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/load
98	62350(7)	6-Sep-74	lspdoc.doc	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/lspdo
8	4525(7)	27-Mar-75	read.me	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/read.me
3	1890(7)	27-Mar-75	remob.lsp	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/remob
3	1890(7)	27-Mar-75	remobv.lsp	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/remob
14	8580(7)	25-Oct-73	scan.lst	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/scan.ls
3	1800(7)	23-Jan-73	scan.mac	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/scan.ma
2	990(7)	23-Jan-73	scan.rel	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/scan.re
4	2195(7)	27-Mar-75	symmak.lst	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/symma
1	435(7)	24-Mar-75	symmak.mac	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/symma
1	58(36)	24-Mar-75	symmak.rel	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/symma
20	12385(7)	5-Nov-73	sys1.com	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/sys1.co
57	36065(7)	5-Nov-73	sys1.lap	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/sys1.la
3	1480(7)	25-Sep-73	sys2.lsp	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/sys2.ls
3	1480(7)	27-Mar-75	sys2v.lsp	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/sys2v
15	9215(7)	6-Sep-74	tabcon.doc	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/tabco
1	30(7)	27-Mar-75	ucilsp.cmd	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/ucils
932	596135(7)	27-Mar-75	ucilsp.lst	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/ucils
254	162480(7)	27-Mar-75	ucilsp.mac	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/ucils
134	17087(36)	27-Mar-75	ucilsp.rel	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/ucils
45	28385(7)	6-Sep-74	update.doc	http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/updat

- [Is a version of this included in the SAILDART archive that Bruce Baumgart is rumored to be making availa
<http://www.baumgart.org/>] and http://www.nomodes.com/pub_manual.html [<http://www.nomodes.cor>

Documentation

- L. Quam. LISP Display Functions. Operating Note 7, Artificial Intelligence Laboratory, Stanford University,
- Lynn H. Quam and Whitfield Diffie. Stanford LISP 1.6 Manual. Operating Note 28.7, Stanford Artificial Int

The SAIL Operating Notes Guide

<http://www.google.com/search?q=cache:poMJyVOHHOMJ:z.baumgart.org/saildart/prog/CSR/html>
includes this entry:

SAILON-28.7, Lynn H. Quam and Whitfield Diffie, "Stanford LISP 1.6 Manual", September 1969, (DSK compiler, the editor ALVINE, and other aspects of this venerated list processing system.

- Robert J. Bobrow, Richard R. Burton, Jeffrey M. Jacobs, Daryle Lewis. UCI LISP Manual. Technical Report Irvine, 1973, 175 pages. Online at pdp-10.trailing-edge.com:

- Title page; Table of Contents [<http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/tabcon.doc>]
- Introduction [<http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/intro.doc.html>]
- Debugging Facilities [<http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/break.doc.html>]
- The LISP Editor [<http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/edit.doc.html>]
- Extended Interpretation of LISP Forms [<http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/>]
- Extensions to the Standard Input/Output Functions, New Predicates, New Numeric Functions, Contigu [<http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/update.doc.html>]
- Index [<http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/index.doc.html>]
- Changes to the UCILISP System 9/73 [<http://pdp-10.trailing-edge.com/decuslib10-04/01/43,50322/cl>]
- William J. Earl. UCILISP Random Notes. March 27, 1975. [<http://pdp-10.trailing-edge.com/decuslib10>]
- R. Lefavre. Rutgers/UCI LISP Manual. Online manual, RS:RUTLSP.MAN, Computer Science Department,
- James R. Meehan. *The New UCI-LISP Manual*. With contributions by Lynn H. Quam, Whitfield Diffie, Rot LeFavre. Lawrence Erlbaum Associates, Publishers, Hillsdale, New Jersey, 1979, xix+366 pages.

Applications

- H. Enea and D. Wooldridge. Algebraic Simplification. Memo AIM-5, Stanford Artificial Intelligence Laboratory
- D. Wooldridge. An Algebraic Simplify Program in LISP. Memo AIM-11, Stanford Artificial Intelligence Laboratory
- Anthony Hearn. Computation of Algebraic Properties of Elementary Particle Reactions Using a Digital Computer. 573-577. *ACM DL* [<http://doi.acm.org/10.1145/365758.365766>]

This paper describes work performed in LISP on an IBM 7090. The help of Stephen Russell and Dean W. members of the Stanford Artificial Intelligence Project".

- Anthony Hearn. REDUCE User's Manual. Report No. ITP-292, Institute of Theoretical Physics, Department of Physics, Stanford University
- A. Hearn. REDUCE Users' Manual. Memo AIM-50, Stanford Artificial Intelligence Laboratory, February 1968 [<http://historical.ncstrl.org/litesite-data/stan/CS-TR-74-409.pdf>] -- is this really AIM-57?]
- Anthony Hearn. REDUCE, a User-oriented Interactive System for Algebraic Simplification. Technical Report 74-409, Stanford University
- Anthony Hearn. REDUCE - A User Oriented Interactive System for Algebraic Simplification. In *Interactive Systems*, edited by J. Reinfelds, Academic Press, New York, 1968).

IBM System/360 Lisp

- Fred W. Blair, James H. Griesmer, Joseph Harry, and Mark Pivovonsky. Design and Development Document for the IBM System/360 Lisp Compiler. IBM Research, Yorktown Heights, New York, revised June 24, 1971, 137 pages. [PDF \[Blair-LISP360.pdf%20\(10.7MB\)\]](#)

Reference manual for implementation of LISP running on OS/360, TSS/360, and CP/CMS.

- Fred W. Blair. Structure of the Lisp Compiler. IBM Research, Yorktown Heights, no date (approximately 1971)

"The first Lisp compiler was written by Robert Brayton with the assistance of David Park, in SAP for the IBM System/360. Brayton left MIT. During that interval of time a Lisp compiler written in Lisp was implemented by Klim. That Brayton's being written in assembly language, would obviously be faster. Difficulties in maintenance led Hart and Michael Levin wrote a compiler in Lisp which was distributed with the 704 Lisp 1.5 system. That of that compiler."

- Richard Ryniker and Mark Pivovonsky. CMS-LISP I/O Supervisory Functions. IBM Research, Yorktown Heights
- F. W. Blair. The definition of LISP 1.8 + 0.3i. Internal report, IBM Thomas J. Watson Research Center, Yorktown Heights, New York, 1978. [Cited in [Padgett and Fitch 1985](#) [<http://doi.acm.org/10.1145/318593.318656>].]

Multics Lisp

- Bernard Greenberg, with contributions from Daves Moon and Reed and Carl Hoffman. Multics MacLisp. See [<http://www.multicians.org/features.html>]

"LISP was one of the first LISP implementations on virtual memory [<http://www.multicians.org/mgv.h>]

compiler, which is extremely unusual) by Dave Reed [<http://www.multicians.org/multicians.html#Reed> libraries. It was not compatible with any other well-known Lisp. There was no Multics software written

Version II Lisp was known as 'Multics MacLisp' ... The need for it arose from the MIT Mathlab group's . space constraints of the PDP-10 systems on which it was developed. The large virtual memory of Multic MacLisp was developed to support Macsyma.

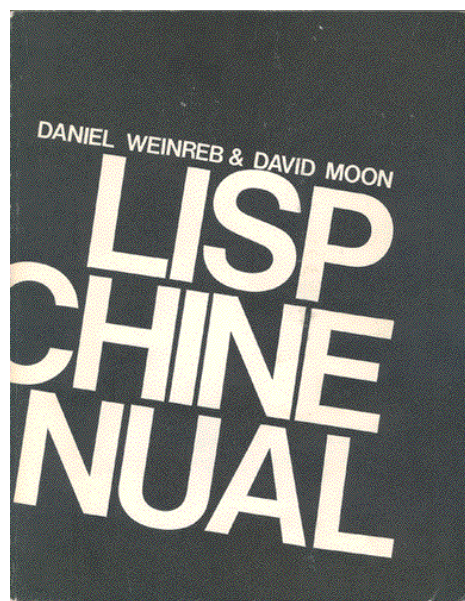
Multics MacLisp was designed to be compatible with the large, mature, and heavily used 'MACLISP' dialect between 1970 and 1973. Reed, then an undergraduate at MIT in the Multics group, started the project by most performance-critical sections, most notably the evaluator, were rewritten in a tour-de-force of ALI [<http://www.multicians.org/multicians.html#Moon>].

...

... Greenberg chose Multics Lisp as the implementation and extension vehicles for Multics Emacs [<http://www.gnu.org/emacs/multics/>], from all over the Multics community and to Multics from all over the Lisp community. Multics Emacs is offering. Multics Emacs' (see separate section) highly successful use of Lisp as (inter alia) an extension language (GNU)."

- Jerome C. Yochelson. Multics LISP. Unpublished, Massachusetts Institute of Technology, Cambridge, Mass.
- Robert R. Fenichel and Jerome C. Yochelson. A LISP garbage-collector for virtual-memory computer system 611-612. *ACM DL* [<http://doi.acm.org/10.1145/363269.363280>]
- Bernard S. Greenberg. The Multics MACLISP Compiler : The Basic Hackery -- a Tutorial. Unpublished, Dec 1972.
- Alan Bawden, Glenn S. Burke, and Carl W. Hoffman. MacLisp Extensions. Project MAC Memorandum MAC-20, MIT, 1971.

Zetalisp for Lisp Machines (CONS, CADR, Lisp Machines Inc., Symbolics)



CONS and CADR: see <http://www.ugcs.caltech.edu/~weel/lispm.shtml>

- Richard Greenblatt. The LISP Machine. Working Paper 79, M.I.T. A.I. Laboratory, November 1974. [PDF at](#)
- Tom Knight. CONS. Working Paper 88, M.I.T. A.I. Laboratory, November 1974. [PDF at bitsavers.org](#) [<http://bitsavers.org/pdf/mit/ai/cons/cons.pdf>]
- The Lisp Machine Group (Alan Bawden, Richard Greenblatt, Jack Holloway, Thomas Knight, David Moon, Intelligence Laboratory, Massachusetts Institute of Technology, August 1977, 29 pages. [PDF at MIT](#) [<ftp://ftp.prunesquallor.com/pub/lisp-machine/memo444.htm>]
- Daniel Weinreb and David Moon. LISP Machine Manual, Preliminary Version. MIT Artificial Intelligence Laboratory, 1974.
- Daniel Weinreb and David Moon. Lisp Machine Manual, Third Edition. MIT Artificial Intelligence Laboratory, 1981. (31MB) [http://www.bitsavers.org/pdf/mit/cadr/chinual_3rdEd_Mar81.pdf]
- Daniel Weinreb and David Moon. Lisp Machine Manual, Fourth Edition. MIT Artificial Intelligence Laboratory, 1984.

[http://www.bitsavers.org/pdf/mit/cadr/chinual_4thEd_Jul81.pdf]

- Daniel Weinreb, David Moon, and Richard M. Stallman. Lisp Machine Manual. Fifth Edition, System Version 1983, 692 pages. [online at bitsavers.org](http://www.bitsavers.org/pdf/mit/cadr/chinual_5thEd_Jan83/) [http://www.bitsavers.org/pdf/mit/cadr/chinual_5thEd_Jan83/]
- Richard Stallman, Daniel Weinreb, and David Moon. Lisp Machine Window System Manual. Edition 1.1, System Version 1983, 261 pages. [online at bitsavers.org](http://www.bitsavers.org/pdf/mit/cadr/LI) [<http://www.bitsavers.org/pdf/mit/cadr/LI>]
- Richard M. Stallman, Daniel Weinreb, and David Moon. Lisp Machine Manual. Sixth Edition, System Version 1984, 859 pages. [online at bitsavers.org](http://www.bitsavers.org/pdf/mit/cadr/chinual_5thEd_Jan83/) [http://www.bitsavers.org/pdf/mit/cadr/chinual_5thEd_Jan83/]
- Thomas F. Knight, Jr., David A. Moon, Jack Holloway, and Guy L. Steele, Jr. CADR. AI Memo 528, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, October 1977, 23 pages. [PDF at MIT](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-528.pdf) [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-528.pdf>] [HTML at ~prunesquallor](http://prunesquallor.com/cadr/) [[ht](http://prunesquallor.com/cadr/)]

Abstract: The CADR machine, a revised version of the CONS machine, is a general-purpose, 32-bit microcomputer system being developed by the Laboratory as a high-performance, economical implementation of hardware and low-level software.

- Anonymous. ZetaLisp Reference Manual, Symbolics, Cambridge, MA, 1984.

Emulators

- Brad Parker. MIT CADR LISP Machine emulator. [<http://www.heeltoe.com/retro/cadr/>] [[#refhttp://www](http://www.heeltoe.com/retro/cadr/)]
- Explorer III Project. Texas Instruments Explorer II Lisp machine emulator. Site includes some software. [[h](http://www.ti.com/explorer/)]

Scheme

- Gerald Jay Sussman and Guy Lewis Steele, Jr. Scheme: An Interpreter for Extended Lambda Calculus. AI Memo No. 359, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, December 1975, 43 pages. [PDF at MIT](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-359.pdf) [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-359.pdf>]
- Guy Lewis Steele, Jr. and Gerald Jay Sussman. Lambda: The Ultimate Imperative. AI Memo No. 353, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, December 1976, 40 pages. [PDF at MIT](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-353.pdf) [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-353.pdf>]
- Guy Lewis Steele, Jr. Lambda: The Ultimate Declarative. AI Memo No. 379, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, October 1977, 23 pages. [PDF at MIT](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-379.pdf) [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-379.pdf>]
- Guy Lewis Steele, Jr. Debunking the 'Expensive Procedure Call' Myth, or, Procedure Call Implementations in Scheme. AI Memo No. 379, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, October 1977, 23 pages. [PDF at MIT](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-379.pdf) [[ftp://](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-379.pdf)]

A version of this report was published in *Proceedings of the 1977 ACM annual conference*, pages 153-166. [<http://portal.acm.org/citation.cfm?id=810196&coll=portal&dl=ACM>]

- Guy Lewis Steele, Jr. and Gerald Jay Sussman. The Revised Report on SCHEME: A Dialect of LISP. AI Memo No. 404, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, January 1978, 35 pages. [PDF at MIT](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-404.pdf) [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-404.pdf>]
- Guy Lewis Steele, Jr. and Gerald Jay Sussman. The Art of the Interpreter of, the Modularity Complex (Parts I and II). AI Memo No. 405, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, May 1978, 75 pages. [PDF at MIT](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-405.pdf) [[ftp://publications.ai.mit.edu/ai-p](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-405.pdf)]
- Guy Lewis Steele, Jr. RABBIT: A Compiler for SCHEME. Technical Report 474, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, October 1977, 23 pages. [PDF at MIT](ftp://publications.ai.mit.edu/ai-publications/pdf/AITR-474.pdf) [<ftp://publications.ai.mit.edu/ai-publications/pdf/AITR-474.pdf>]
- Guy Lewis Steele, Jr. and Gerald Jay Sussman. Design of LISP-based Processors, or SCHEME: A Dielectric. AI Memo No. 514, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, March 1979, 75 pages. [PDF at MIT](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-514.pdf) [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-514.pdf>]

A version of this report was published as: Design of a Lisp-based Microprocessor. *Communications of the ACM*, 22(12):153-166, December 1979. [<http://doi.acm.org/10.1145/359024.359031>]

- Guy Lewis Steele, Jr. Compiler Optimization Based on Viewing LAMBDA as RENAME + GOTO. *AI: An MIT Quarterly Journal*, 1(1):1-12, Spring 1980.
- William Clinger, editor. The Revised Revised Report on Scheme or An UnCommon Lisp. AI Memo No. 848, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, March 1985, 76 pages. [PDF at MIT](ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-848.pdf) [<ftp://publications.ai.mit.edu/ai-publications/pdf/AIM-848.pdf>]

Also published as: Technical Report 174, Computer Science Department, Indiana University, June 1985

- Jonathan Rees and William Clinger, editors. Revised(3) Report on the Algorithmic Language Scheme. *SIGPLAN Notices*, 20(12):1-12, December 1985. [<http://doi.acm.org/10.1145/15042.15043>]
- William Clinger and Jonathan Rees, editors. Revised(4) Report on the Algorithmic Language Scheme. In *AI Memo No. 848*, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, March 1985, 76 pages.

- [\[http://doi.acm.org/10.1145/382130.382133\]](http://doi.acm.org/10.1145/382130.382133) <http://www.cs.indiana.edu/scheme-repository/doc.standard>
- IEEE. IEEE Standard for the Scheme Programming Language. IEEE Standard 1178-1990, ISBN 1-55937-12
- Richard Kelsey, William Clinger, and Jonathan Rees, editors. Revised(5) Report on the Algorithmic Language August 1998, pages 7-105. Online at [www.brics.dk](http://www.brics.dk/~hosc/11-1/168705.html) [<http://www.brics.dk/~hosc/11-1/168705.html>] Other files [<http://www.schemers.org/Documents/Standards/R5RS/>]
- See also: Bibliography of Scheme-related Research. <http://library.readscheme.org/> [<http://library.readscheme.org/>]

Implementations

- <http://www.schemers.org/Documents/FAQ/#implementations> [<http://www.schemers.org/Documents/FAQ/#implementations>]

T

T was based on Scheme and was started at Yale University in 1981 by Jonathan A. Rees and Norman I. Adams IV

Documentation

- Jonathan A. Rees, Norman I Adams IV, and James R. Meehan. The T manual, fourth edition. Yale University [http://repository.readscheme.org/ftp/papers/t_manual.pdf]

Papers

- Jonathan A. Rees and Norman I. Adams IV. T: A Dialect of Lisp or, LAMBDA: The ultimate software tool. *P* pages 114-122. ACM DL [<http://portal.acm.org/citation.cfm?id=802142>]
- D.A. Krantz, R. Kelsey, J.A. Rees, P. Hudak, J. Philbin and N.I. Adams. ORBIT: An optimizing compiler for *SIGPLAN Notices* Volume 21, Number 7, pages 219-223. ACM DL [<http://doi.acm.org/10.1145/12276.1333>]
- D. Kranz. ORBIT: An Optimizing Compiler for Scheme. Technical Report YALEU/DCS/RR-632, Yale University
- Richard Kelsey and Paul Hudak. Realistic Compilation by Program Transformation : Detailed Summary. *Pr Languages*, 1989, pages 281-292. ACM DL [<http://doi.acm.org/10.1145/75277.75302>]
- Olin Shivers. History of T, December 2001. <http://www.paulgraham.com/thist.html> [<http://www.paulgraham.com/thist.html>]

Originally appeared as: Olin Shivers. T. Posting to LL1-Discuss@ai.mit.edu mailing list, December 11, 2001 [<http://people.csail.mit.edu/gregs/ll1-discuss-archive-html/msg00650.html>]

- Jonathan Rees. The T Project. Website, December 2004. <http://mumble.net/~jar/tproject> [<http://mumble.net/~jar/tproject>]
- Taylor Campbell and Brian Mastenbrook. T Revival Project. <http://www.bloodandcoffee.net/campbell/t/t.1>

New Implementation of Lisp (NIL)

Documentation

- G. Burke. Introduction to NIL. Laboratory for Computer Science, Massachusetts Institute of Technology, MIT
- G.S. Burke, G.J. Carrette, and C.R. Eliot. NIL Notes for Release 0.259, Laboratory for Computer Science, MIT
- G.S. Burke, G.J. Carrette, and C. R. Eliot. NIL Reference Manual. Report MIT/LCS/TR-311, Laboratory for Computer Science, MIT

Papers

- Jon L. White. Nil: A perspective. *Proceedings of 1979 Macsyma Users' Conference*, Washington, D.C., June 1979
- Massachusetts Institute of Technology, Cambridge, Massachusetts, 1983.
- NIL is discussed in this article/response pair:
 - Olin Shivers. History of T. 2001. <http://www.paulgraham.com/thist.html> [<http://www.paulgraham.com/thist.html>]
 - Dan Weinreb: Dan Weinreb on NIL. <http://www.paulgraham.com/weinreb.html> [<http://www.paulgraham.com/weinreb.html>]

S-1 Lisp

For S-1 Mark IIA supercomputer.

- Rodney A. Brooks, Richard P. Gabriel, and Guy L. Steele Jr. S-1 Common Lisp Implementation. Proceeding 1982, pages 108?113. ACM DL [<http://portal.acm.org/citation.cfm?id=802141>]

SpiceLisp

Documentation

- G.L. Steele, Jr. Spice Lisp Reference Manual. Technical Report, Carnegie.Mellon University (interim Spice l documentation), 1981. [Cited in David S. Touretzky. A comparison of reduction in APL with polyarity in LIS DL [<http://portal.acm.org/citation.cfm?id=801225>]]
- Further references in HOPL [<http://hopl.murdoch.edu.au/showlanguage.prx?exp=3526&language=Spice I>]

Standard LISP, Portable Standard Lisp (PSL)

Standard LISP arose from a desire to port REDUCE (see Stanford LISP 1.6 Applications, above and Application

Source code

- E. Karabudak, G. Üçoluk, and T. Yilmaz. Portable Utah Standard LISP, public release 6.1, June 1993. Online [<http://www.ceng.metu.edu.tr/~ucoluk/research/lisp/generalinfo.html>]

Documentation

- A.C. Hearn. Standard LISP. Memo AIM-90, Stanford Artificial Intelligence Project, Stanford University, Ma
- Anthony C. Hearn. Standard LISP. *SIGPLAN Notices*, Volume 4, Number 9, September 1969, pages 25-45.
- J.B. Marti, A.C. Hearn, M.L. Griss, and C. Griss. Standard LISP Report. University of Utah Symbolic Comp
- I.B. Frick. Manual for Standard LISP on the DECSYSTEM 10 and 20. Technical Report TR-2, Utah Symbol
- J.B. Marti, A.C. Hearn, M.L. Griss, and C. Griss. Standard LISP Report. *SIGPLAN Notices*, Volume 14, No. 1

The 1979 report gives the date for Hearn's original Standard LISP article in *SIGPLAN Notices* as Volum in 1969 (although it's missing Number 9).

- M.L. Griss and B. Morrison. The Portable Standard LISP Users Manual. Technical Report TR-10, Utah Sym March 1981. (PSL version 3, 6 May 1982)

"Abstract: This manual describes the primitive data structures, facilities and functions present in the Pc functions of interest to a PSL programmer. Except for a small number of hand-coded routines for I/O a machine-oriented mode of PSL, called SYSLISP, to perform word, byte, and efficient integer and string Compiler, and currently runs on the DEC-20, VAX, and MC68000."

- J.B. Marti, A.C. Hearn, M.L. Griss, and C. Griss. Standard LISP Report. Date? Online at www.reduce-algeb

Papers

- M.L. Griss. BIL: A Portable Implementation Language for LISP-Like Systems. Operating Note No. 36, Sym
- J. Fitch and J. Marti. A Small Standard LISP Interpreter. Symbolic Computation Group, University of Utah
- J. Fitch and J. Marti. SLISP - A Standard LISP Implementation in SDL. Operating Note 37, Symbolic Comp
- J. Fitch and J. Marti. SLISP - The B-Interpreter. Symbolic Computation Group, University of Utah, 1978.
- J.B. Marti and J. Fitch. SLISP, A Standard LISP Implementation in a High Level Language. *REDUCE News*
- J. Fitch and J.B. Marti. SLISP – A Standard LISP Implementation for the Burroughs 1700. Operating Note

- Martin L. Griss, Robert R. Kessler, and Gerald Q. Maguire, Jr. TLISP ? A Portable LISP Implemented in P-c Computation, June 1979, pages 490-502.
- I.B. Frick. A Portable Lap and Binary Loader. Operating Note 52, Utah Symbolic Computation Group, Univ
- Martin L. Griss and Anthony C. Hearn. A portable LISP compiler. *Software--Practice and Experience*, Volu

Originally appeared as Report UCP-76, Utah Symbolic Computation Group, University of Utah, June 19

- Eric Benson and Martin L. Griss. SYSLISP: A Portable LISP-based Systems Implementation Language. UCI revised 7 April 1981), 12 pages.
- Martin L. Griss, Eric Benson, and Anthony C. Hearn. Current Status of a Portable LISP Compiler. *Procedin* pages 276-283.

Originally appeared as Report UCP-82, Utah Symbolic Computation Group, Department of Computer S

- W. Galway and M.L. Griss. An Editor Interface for a Portable Lisp Programming Environment. Opnote 64, of Utah, January 1982.
- M.L. Griss. Some Notes on the Basic PSL Timing Tests. Opnote 66, Utah Symbolic Computation Group, De
- Martin L. Griss, Eric Benson, and Gerald Q. Maguire Jr. PSL: A Portable LISP System. In *Proceedings of th* DL [<http://portal.acm.org/citation.cfm?id=802139>]

Originally appeared as Report UCP-83, Department of Computer Science, University of Utah, May 1982

Applications

- A.C. Hearn. REDUCE 2 User's Manual, Memo No. AIM-133, Stanford Artificial Intelligence Project, Octobe
- A.C. Hearn. REDUCE 2: A system and language for algebraic manipulation. In *Proceedings of the Second S* pages 128-133. *ACM DL* [<http://portal.acm.org/citation.cfm?id=806277>]
- A.C. Hearn. REDUCE 2 Users Manual. Report UCP-19, Symbolic Computation Group, Department of Comj

Franz Lisp

Source code

- Franz Lisp Opus 38.93b, the last public domain release available from the ucbvax ftp site before its demise. [ftp://ftp.cs.cmu.edu/user/ai/lang/others/franzlsp/op38_93b/0.html]

Documentation

- John K. Foderaro, Keith L Sklower, and Kevin Layer. The FRANZ LISP Manual. University of California, Be

Common Lisp

Coming soon -- want to help?

Implementations

[See pointers at <http://linuxfinances.info/info/commonlisp.html>]

Allegro CL

- <http://www.franz.com/> [<http://www.franz.com/>] [commercial]

CMU Common Lisp (CMUCL)

- <http://www.cons.org/cmuc/> [<http://www.cons.org/cmuc/>]

GNU CLISP

- <http://clisp.cons.org/> [<http://clisp.cons.org/>]

LispWorks [originally Harlequin CL]

Steel Bank Common Lisp (SBCL)

Forked from CMUCL in 1999.

- <http://sbcl.sourceforge.net/> [<http://sbcl.sourceforge.net/>]

Corman Lisp

- <http://www.cormanlisp.com/index.html> [<http://www.cormanlisp.com/index.html>]

References

- Guy L. Steele. Common Lisp the Language, 2nd Edition, Digital Press, 1990, 1029 pages. [Online at www-2.cs.cmu.edu/~dst/CommonLisp/](http://www-2.cs.cmu.edu/~dst/CommonLisp/)
- Kent Pitman. Common Lisp HyperSpec, version 7.0. LispWorks Ltd., 2005. [Online at www.lispworks.com](http://www.lispworks.com/documentation/HyperSpec/) [<http://www.lispworks.com/documentation/HyperSpec/>]

Bibliographies

- Herbert Stoyan. A complete LISP-Bibliography. University of Erlangen D-852. [HTML](http://www8.informatik.uni-erlangen.de/~stoyan/lisp/) [<http://www8.informatik.uni-erlangen.de/~stoyan/lisp/>]

Papers about LISP history

Precursors of LISP

- A. Newell and J. C. Shaw. Programming the logic theory machine. *Proceedings Western Joint Computer Conference*, 1956, pp. 339-352.
- J. McCarthy. Programs with common sense. Paper presented at the *Symposium on the Mechanization of T*. 1958. (Published in Proceedings of the Symposium by H. M. Stationery Office).

By LISP project members

- John McCarthy. LISP: a programming system for symbolic manipulations. Preprints of papers presented at pages 1-4. [ACM DL](http://doi.acm.org/10.1145/612201.612243) [<http://doi.acm.org/10.1145/612201.612243>]
- John McCarthy. Recursive functions of symbolic expressions and their computation by machine, Part I. [Collected Papers](http://doi.acm.org/10.1145/367177.367199) [<http://doi.acm.org/10.1145/367177.367199>] online at Stanford [<http://www-formal.stanford.edu/jmc/recursive.pdf>]
- Berkeley and Bobrow, editors. *The Programming Language Lisp : Its Operation and Applications*. Inform Science Press, 1978.
- John McCarthy. History of Lisp, *Proceedings of the first ACM SIGPLAN conference on history of programming languages* [<http://www-formal.stanford.edu/jmc/history/lisp/lisp.html>]; appears in: R. Wexelblat, editor. *Lisp: An Introduction*. 1981 PDF
- John McCarthy. LISP – notes on its past and future. *Proceedings of the 1980 ACM conference on LISP and Symbolic Computation* [<http://portal.acm.org/citation.cfm?id=802782>] online at Stanford [<http://www-formal.stanford.edu/jmc/lisp/notes.pdf>]

By others

- Herbert Stoyan. Early LISP history (1956-1959). *Proceedings of the 1984 ACM Symposium on LISP and Symbolic Computation* [<http://portal.acm.org/citation.cfm?id=802047>] HTML at University of Erlangen [<http://www8.informatik.uni-erlangen.de/~stoyan/lisp/>]
- Guy L. Steele, Jr. and Richard P. Gabriel. The evolution of Lisp. *Proceedings of the second ACM SIGPLAN conference on LISP and Symbolic Computation*, 1979, pp. 1-10.

- pages 231-270. [ACM DL \[http://doi.acm.org/10.1145/154766.155373\]](http://doi.acm.org/10.1145/154766.155373) PDF at Dreamsongs.com [http://www.dreamsongs.com/NewFiles/HOPL2-Uncut.pdf] Slides PDF at Dreamsongs.com [http://www.dreamsongs.com/NewFiles/HOPL2-Uncut.pdf]
- http://www.lispworks.com/documentation/HyperSpec/Body/01_ab.htm [http://www.lispworks.com/doc
- Anonymous. Lisp programming language. Wikipedia. [online at wikipedia.org \[http://en.wikipedia.org/wiki](http://en.wikipedia.org/wiki/Lisp)

Published biographies and interviews

- etc.

Correspondence and memoranda

- Edmund C. Berkeley Papers, 1928-1988, Charles Babbage Institute, CBI 50. [Finding aid at CBI \[http://www](http://www.cbi.harvard.edu/)

Photographs

- John McCarthy, black&white photograph, date unknown (late 1950s?), from collection of Herbert Stoyan. [Online at Stanford University \[http://www-fc](http://www.fcs.stanford.edu/people/stoyan/)
- Six participants in the 1960 Algol conference in Paris. The picture was taken at the 1974 ACM conference on [Wegstein. Bottom row: John Backus, Peter Naur, Alan Perlis. Online at Stanford University \[http://www-fc](http://www.fcs.stanford.edu/people/stoyan/)
- John McCarthy, black&white photograph, date unknown (contemporary). [Online at Stanford University \[http://www-fc](http://www.fcs.stanford.edu/people/stoyan/)
- John McCarthy, color photograph, date unknown (contemporary). [Online at Stanford University \[http://www-fc](http://www.fcs.stanford.edu/people/stoyan/)
- John McCarthy, color photograph by Bart Nagel for Time Digital, date unknown (contemporary). [Online at Stanford University \[http://www-fc](http://www.fcs.stanford.edu/people/stoyan/)
- Speakers at the 2002 International Lisp Conference in San Francisco. [Online \[http://www.international-lisp](http://www.international-lisp.org/)

Audio/video

- Excerpts from LISP session. *History of Programming Languages Conference*, 1-3 June 1978, Los Angeles, [\[http://tennessee.cc.vt.edu/~hopl/lisp.html\]](http://tennessee.cc.vt.edu/~hopl/lisp.html)

Related resources

- MIT Artificial Intelligence Memos. http://www.ai.mit.edu/research/publications/pubs_browse.shtml [http://www.ai.mit.edu/research/publications/pubs_browse.shtml]
- Stanford University Computer Science Technical Reports, 1963-1968. [Google cache \http://www.google.com/search?q=cache:_FHcxAQff70J:z.baumgart.org/saildart/prog/CSR/html/00056
- Herbert Stoyan. *Lisp History* [http://www8.informatik.uni-erlangen.de/html/lisp-enter.html]
- Les Earnest, Editor. Final Report : The First Ten years of Artificial Intelligence Research at Stanford. Stanford Science Department Report No. STAN-CS-74-409, July 1973. [Online at historical.ncstrl.org \[http://historical.ncstrl.org\]](http://historical.ncstrl.org)