Richard Mlynarik
436 Alvarado Street
San Francisco, CA 94114
415-648-5457

Nationality: Australian **US Permanent Resident**

December 1991 – present	**Xerox Palo Alto Research Center / Xerox Advanced Document Systems,** Palo Alto CA Implemented and maintained, with others, a suite of image-processing code for use in "paper user-interface" applications (e.g. fax, digital copiers.) Extended and maintained local enhancements to Ghostscript, a Postscript interpreter and imaging system. Contributed to the design of a system architecture for "advanced office document systems" applications. Our software system was the basis for Fuji Xerox's award-winning "IM-200" copier product, which launched in September 1996. Participated in the redesign of the document-applications system for reimplementation in Java. Reimplemented the core of the system in Java (including native-library interfaces.)
November 1990 – November 1991	**Software Productivity Research**, Burlington MA (Contractor.) Assisted with the implementation of Checkpoint, SPR's IBM-PC-based software estimation tool. Responsible for the development of various program-generation tools, interpreter engines, binary-data formats. Maintained the script-driven user-interface kernel around which the product is based.
January 1989 – March 1990	**Apple Computer**, Cambridge MA Evaluated various hardware platforms with regards to their suitability for "dynamic language" (garbage-collected, explicitly-typed data) systems. Designed the data formats, calling conventions, etc., for a dynamic language implementation on new processor platforms. Designed and partially implemented the runtime system for that implementation. Wrote simulation and debugging tools. Worked on the design of a new programming language which would later become "Dylan". Designed strategy for prototyping the language under Apple's Common Lisp. Wrote machine code to implement the language's object system. Also: Common Lisp implementation.
November 1988 – December 1988	**Coral Software**, Cambridge MA More Common-Lisp implementation. Coral was purchased by Apple in December 1988.
September 1987 – December 1988	**Palladian Software**, Cambridge MA Designed and implemented new, platform-independent hardcopy and window system interfaces for a very large Lisp application. Helped debug and maintain the extensive and poorly-defined interfaces between Palladian's and various vendors' code.

September 1987 – September 1989	**Center for Biological Information Processing, M.I.T.**, Cambridge MA Kept a network of Suns, Symbolics Lisp Machines and Macintoshes from falling apart. Kept TCP network working.
Summer 1987	**Lucid**, Menlo Park CA Designed a mechanism for supporting Lisp "light-weight processes" on a multiprocessor computer. Suggested and implemented numerous changes to the Lisp system to increase paging performance and decrease system size. Common Lisp implementation.
September 1985 – March 1986	**LMI**, Cambridge MA Responsible for the maintenance and development of a large part of the software of the LMI Lisp Machine system, including a Lisp compiler, text editor, virtual memory system and garbage collector written largely in microcode. In addition to maintenance on the system as a whole, assisted in debugging a new garbage collector and paging system. Worked on the design and implementation of a new file system. Assisted with the design of a generic tape interface facility.
1986	**Laboratory for Computer Science, M.I.T.** Wrote a Common-Lisp package system for NIL, a Lisp system running under VAX/VMS.
December 1983 – June 1985	**Artificial Intelligence Laboratory, M.I.T.** Implemented enhancements to Lisp Machine system software as requested by the Vision Group. Duties included software support, bug-fixing, and user consultation.
October 1982 – June 1983	**Visible Language Workshop, M.I.T.** Designed and implemented an interactive typeface design and editing system.
Distant past to present	Helped with the implementation and maintenance of GNU Emacs, a text editor written in C running under Unix. Wrote a spelling correction program for use with Emacs. Wrote a number of Unix utilities for use in the GNU system. Was extensively involved in redesigning and reimplementing large parts of Emacs to form Lucid Emacs (later known as XEmacs) while maintaining backward-compatibility with a huge installed customer base.