

Allen I. Holub

1041 Shattuck Ave., Berkeley, CA 94707

allen@holub.com

www.holub.com

510.859.3620 (home, cell, sms)

510.528.3620 (office, fwd to cell)

Computer professional with 30 years experience in all aspects of computer technology and software engineering.

- Expert programmer. Guru-level OO design, Agile process, software-architecture skills.
- Skilled at forming and guiding highly-functional Agile development teams. Demonstrated competence in strategic product development, hiring, mentoring and motivating technical staff.
- Exceptional teaching, coaching, mentoring skills.
- Exceptional written-and-oral communication skills. Widely published (100+ magazine articles, 8 books). Known for the ability to effectively explain complex technical topics to both technical and non-technical audiences.
- Expert in software and design analysis and assessment.

Professional Profile

Technical

Highly skilled programmer, have worked on projects ranging from highly interactive AJAX Web 2.0 front ends to traditional Web 1.0 web sites, to back-end database servers, compilers, and tool libraries.

Design	Programming	Web
<i>Object-Oriented Design (guru)</i>	<i>Java (since 1995, guru level)</i>	<i>AJAX, HTML, XHTML, HTML5,</i>
<i>UI Design</i>	<i>J2EE, EJB, JDBC, JSTL, JSP, Threads, RMI/Networking,</i>	<i>GWT, CSS,</i>
<i>Design Patterns</i>	<i>Servlets, JSP, security, UI, Struts, JPA, Hibernate</i>	<i>JavaScript Perl, PHP, Python</i>
<i>UML</i>	<i>C++ (since 1989), C (since 1978)</i>	<i>Apache/Tomcat</i>
<i>RUP</i>	<i>CVS, Subversion, Ant, JUnit</i>	<i>Linux</i>
<i>Agile Process, XP</i>	<i>KSH, CSH, Bash, Awk</i>	<i>XML</i>
	<i>SQL (PostgreSQL & MySql)</i>	<i>Web Services</i>

Technical Assessment & Due Diligence

Highly skilled in evaluating software and design quality, evaluating and improving development processes, and evaluating technology for possible adoption or appropriateness in solving business problems.

Teaching

Among the most respected technical educators in the San Francisco area. Teach on a consulting basis, and present some of the most technically rigorous classes offered by Univ. of Calif. Extension.

Communication

An effective and dynamic communicator, speak regularly at large conventions (JavaOne, Software Development); Consistently receive highest evaluations. Excel at explaining complex technical topics in an easily understandable way.

Written Communication Project Management

Have published 9 books and 100+ magazine articles.

Regularly help CTO/VPE-level individuals implement software-development processes. Thoroughly understand both business and issues. Skilled team builder. Work well in a collaborative environment.

Management

Experienced in budgeting and estimation. Have run my own business for 20 years. Skilled at interfacing between internal organizations (Marketing and Engineering, for example).

Mentoring

Have mentored many companies through the OO-design-adoption process. Have helped many companies set up OO-compatible software-development processes.

Recruiting

Have recruited in prior jobs. My public teaching and presenting work puts me in an excellent position for recruiting. Am often asked to refer students to job opportunities.

Strategic

Understand both the business and technical issues. Can easily identify and communicate the core problems. Early adopter of Java, etc.

Recent Experience

Allen Holub & Associates, Inc. (President and CEO) 1983-Present

Consulting, mentoring, due-diligence, and training in object-oriented design, UML, software-development processes, and Java to a diverse client base; companies and organizations ranging from Internet startups to Fortune 500 companies. Application and class-library design and development (Web 2.0/OO/Java)

Representative Projects:

iExperiment.net (2009-2010)

Web 2.0 Application Designed and implemented web-based electronic-notebook for biotech researchers, a highly interactive AJAX Web 2.0 application based on the Google Web Toolkit (GWT), JavaScript, Java Servlets, and MySQL. Designed system, working with scientist domain experts, managed implementation team, programmed extensive user-interface component.

Justatodolist.com (2008)

Web 2.0 Application Designed and implemented online to-do list application. This is a highly interactive AJAX application based on the Google Web Toolkit (GWT), Java Servlets, and MySQL.

coValuate: smartcorriders.com (2007-2008)

Web 2.0 Application Analyzed requirements, developed architecture, and implemented portions of a Web 2.0 (AJAX) application that displayed traffic-flow data from live telemetry and camera feeds. UI customized for both public users and traffic engineers.

Austin Ventures

Due Diligence. Analyzed and critiqued a Java (J2EE) ecommerce software system, focusing on code quality, reliability, and maintainability. Estimated effort required to make the software production ready. Determined dependencies on third-party software and effort required to remove those dependencies. Analyzed database schema as well as Java source code.

interPro, StockPower

Helped formulate OO requirements document for next-generation product. Reviewed and validated use cases, static and dynamic models. Helped institute new design/development processes to ease the transition to OO. Recommended appropriate technology for implementation.

StockPower, interPro, EBMUD, Barclays Global, Intuit, etc.

Training in OO-Design. Trained either entire staff or large group within organization..

Symantec

Evaluated class library and made recommendations that led to the library being abandoned. Developed in house and developed training curriculum and materials.

ProBusiness

Evaluated existing system and made recommendations for next-generation system, both process and architecture. Identified Java as the most appropriate solution to their problem at a time when Java was seen as too new a technology to be taken seriously.

iMind

Identified requirements and aided in the UI design for mission-critical software. Critiqued system architecture. Worked collaboratively with the VC and CEO to identify current organizational problems and develop strategy for putting company back on course.

KLA/Tencor

Training in C++, Java, and OO Design. Developed migration strategy for moving existing product line to new software architecture and operating environment.

OO Design Workshop

Train senior staff (CTO, VPE, Senior Engineers) in OO methodology and process. Put together small design teams that take a real problem from requirements gathering through to a implementation-level design. Covers life-cycle, process (XP and RUP), use cases, static and dynamic modeling using UML.

Holub Java-Threading Library

Designed and implemented (in Java) open-source threading library that considerably expands Java's threading capabilities. (<http://www.holub.com/taming.java.threads.html>). Classes included in many commercial products.

Partial List of clients:

American Telecorp	Fireman's Fund	OnDisplay, Inc.
Alcatel USA	Genentech	Pacific Bell
Aptivity/Progress Softw.	Golden Gate Software	PeopleSoft
Autodesk	The Govt. of the Yukon	ProBusiness
BARRA Corp.	Hewlett Packard	PE Applied Biosystems
Barclays Global	Integrated Experimental	Scopus Technology
Bell-Atlantic Health Care	Sys.	Shulumberger/Geoquest
Calif. SCIF	Illuminate Corp.	Stanford Linear Accelerator
Canon Info. Systems	Intuit	Sybase
Chordiant Software	i-mind, Inc.	Stock Power, Inc.
Cimtec Corp.	Intel Corp.	Sun Microsystems
Contra Costa Water District	Intellicorp	Technicon
CoValuate	Interpro	Symantec
Caere Corp.	Identitech	Synon
Concur, Inc.	Island Graphics	TRW Financial Systems
Charles Schwab and Co.	IA Corporation	Teknekron Inc.
Documentum, Inc.	JT3	3-Com Corp.
Dynamedix	KLA/Tencor	UC Berkeley Library
East Bay Municipal Utility District	MDL Information Systems	UCSF Pharmacy Dept.
eSignal	Morgan Stanley	U.S. Navy
Fujitsu	Novell	ViewStar
	Oacis Health Care	Wind River Systems

NetReliance (CTO/VP Engineering) 2000-2001

Designed physical and software architecture for a cross-jurisdictional certificate infrastructure for use in eBusiness. Responsible for product design, building a technical team, and for infusing a culture of engineering best practices throughout the organization. Mentored the marketing department in Use-Case analysis and OO requirements-gathering techniques using UML.

Teaching Experience

UC Berkeley Extension (Instructor) 1983-present

My classes are among the most technically rigorous offered by the Extension and are always highly regarded. Students are senior-level professionals (CTO, VPE, Senior Engineers, Senior Analysts, etc.). I identify needs for courses based on predicting the technology that will be hot in the future, research and develop course materials, and teach the class.

Recently taught classes include:

- *Migrating to Java*. A 10-week intro to OO design and Java for C and C++ programmers.
- *OO Design I: Life Cycle, Requirements Gathering, Notation, and Analysis*. Covers the OO design and development process with a focus on CMMI, RUP, and Extreme Programming, use cases, static and dynamic modeling, and the Unified Modeling Language (UML).
- *OO Design II: Patterns, Class hierarchy, and Structure*. Covers the "Gang-of-Four" design patterns and implementation-level design.
- *Taming Java Threads*. Java threading pitfalls and solutions.

- *Serving the Java UI*. Covers server-side UI architecture and technology. Server architecture, Apache/Tomcat, Servlets, Servlet Filters, Java Server Pages (JSP) and the Java Standard Template Library (JSTL), including custom tags, Applets, XML and XML parsing, Model/View/Controller and Presentation/Abstraction/Control architectures.
- *Java Networking for the Enterprise*. Sockets, RMI, CORBA, EJB, URLs, etc.
- *Java AWT/Swing Programming*. Covers OO user-interface architectures and Java implementation.

Past Classes include:

- *C Language Programming for Professional Programmers*
- *C++ Programming for Professional Programmers*
- *Java for Professional Programmers*
- *Compiler Design and Implementation*
- *Microsoft-Foundation-Class Programming*
- *Microsoft OLE Programming with MFC*.

Prior Engineering Experience

Cimtec (Senior Engineer) 1982-1983

Automated machine-tools and industrial process control. Developed dedicated signal-processing hardware and software for detecting failure in automated machine-tool systems. Developed and implemented user interface for automated machine tools. Developed symbolic language and compilers for specifying UI configuration.

TERA/TRW Financial Systems (Lead Software Engineer) 1980-1982

Responsible for the development of a real-time operating system kernel, systems programs, and device drivers for a distributed robotic system. Developed network protocol for communicating between distributed document-control system and mainframe databases. Developed symbolic language for describing finite state machines and implemented compilers for same. Headed a team of 5 programmers, responsible for recruiting and hiring. Developed primarily in C. Developed various UNIX system utilities.

Teknekron, Inc. (Engineer) 1979-1980

Designed and built robotic document-management systems. Designed controller cards and C-language device drivers for auto-focus feature of wafer-inspection microscope and implemented prototype system. Developed controller software for automated microfiche storage and retrieval systems using Intel RMX/80 and PL/M. Wrote user documentation. Implemented robotics control software in TI9900 assembly language.

Cylotomics (Jr. Engineer) 1978-1979

Error-correcting-and-detecting hardware for high-speed data transmission. Aided in the development of a error-correcting data-transmission system for the Hubble Space Telescope. Implemented and tested prototype.

Speaking Experience

Major Conferences:

JavaOne:

Taming Java Threads (three times on successive years. 4,000 programmers in the audience.

Software Development Expo:

I am the Security-Track chair for the Software Development Expo conference. Talks include:
The Google Web Toolkit
How Compilers Work

Object-Oriented UI Architectures
The Object-Oriented-Software-Development Environment
Inside the Java Virtual Machine: A programmers look at the JVM.
Cryptography 101: An introduction to cryptography.
Security 101: An introduction to security
The Java Security APIs
Everything you know is wrong: Why extends and getter/setter methods are evil.

Slides for many of these talks are available at
http://www.holub.com/publications/notes_and_slides.

Other Speaking Engagements:

Am a regular invited speaker at local technical SIGs (Special Interest Groups).

Computer Programs

The following list shows a few of the complete programs or libraries that I've built. It does not include systems for client companies to which I contributed. Some of these projects were open source utility packages, and are incorporated into various commercial products by IBM, Peachtree, Ask Jeeves, and others.

- Highly interactive Web 2.0 AJAX applications based on GWT, Java, MySQL
- Embedded SQL database engine for Java, including JDBC drivers.
- Various robotic control systems.
- SH: A UNIX C-Shell implementation for Microsoft Platforms.
- Curses for MSDOS. A Port of UNIX Curses to the Microsoft Platform.
- NR/TR: Version of the UNIX nroff and troff utilities for Microsoft Platforms
- A full-featured embeddable Real-Time Multitasking kernel/
- Eunuchs: A Unix-Like Operating system for Motorola 68000-series processors.
- SM: A symbolic language and compiler for building hardware state machines.
- Several specialized C Compilers.
- OCCS compiler-development toolkit. Versions of the UNIX yacc and lex utilities for building LL(1) and LALR(1) table-driven parsers. Included a "visual" parser that let you debug the grammar by watching the parser work.
- Arachne: A "Literate Programming" system for C and troff, that allows C source code and it's documentation to reside in the same file.
- HML: A "Literate Programming" system for Java and HTML.
- A C++ Collection-Class library.
- Various utilities (C, C++, Java) for UI construction, hyphenation, sorting, searching, command-line parsing, etc. Many of these utilities have been incorporated into commercial products and class libraries.
- Java Zip-file management utilities.
- Java Threading Library
- The Bank of Allen: (Commercial educational software, in progress.)
- Smartcorridors (Web. 2.0 AJAX application).

Publications

Have published nine books. My most recent book is Holub on Patterns (Apress). Other books include: *Taming Java Threads* (APress), *Compiler Design in C* (Prentice Hall), *Enough Rope to Shoot Yourself in the Foot* (McGraw Hill), and *C+C++: Programming with Objects in C and C++* (McGraw Hill).

I wrote a regular monthly column for *JavaWorld* (since 1998) and a bi-monthly column for *SD Times* (April, 2004 to December 2006). I wrote the influential *C Chest* for Dr. Dobb's Journal, and have published many stand-alone articles. I have also written for *IBM Developer Works* and other magazines, and served as the Technical Editor for *Java Solutions*.

Books

1. *Holub on Patterns: Learning Design Patterns by Looking at Code.* (Apress, 2005).
2. *Taming Java Threads.* (Apress, 2000)
3. *Enough Rope to Shoot Yourself in the Foot: Rules for C and C++ Programming* (McGraw-Hill, 1995)
4. *C++: Programming With Objects in C and C++.* (McGraw-Hill, 1992)
5. *Compiler Design in C.* (Prentice Hall, 1990)
6. *The C Companion.* (Prentice Hall, 1987)
7. *The C Chest and other C Treasures.* (M&T Books, 1987)
8. *On Command: Writing a UNIX-like Shell for MS-DOS.* (M&T Books, 1986)
9. *Dr. Dobb's Toolbook of C.* (M&T Books, 1986)

Many of these books have been translated into other languages.

Magazines

Find hot links to many of these articles, along with the accompanying source code <http://www.holub.com/publications/articles/>. Some of these articles have been read by 150,000+ people (based on measured "hit counts").

1. JavaWorld Jan 2004: More on getters and setters
2. JavaWorld Nov 2003: Create client-side user interfaces in HTML (part 2)
3. JavaWorld Oct 2003: Create client-side user interfaces in HTML (part 1)
4. JavaWorld Sep 2003: Why getter and setter methods are evil
5. JavaWorld Aug 2003: Why extends is evil
6. JavaWorld Jul 2003: Solve the date-selection problem once and for all
7. JavaWorld Jan 2002: When it comes to good OO design, keep it simple
8. JavaWorld Feb 2001: Warning! Threading in a multiprocessor world
9. JavaWorld Mar 2000: User interfaces for object-oriented systems, Part 6: The RPN calculator.
10. JavaWorld Oct 2000: Modify Archives Part 2
11. JavaWorld Jul 2000: Modify Archives Part 1
12. JavaWorld Jan 2000: Misc. Useful Stuff (Assert, Tester, bit_bucket, Std, Log, Align.)
13. JavaWorld Jul 1999: What is an object?
14. JavaWorld Aug 1999: XMLOutputStream
15. JavaWorld Sep 1999: Implementing the Visual-Proxy Architecture
16. JavaWorld Oct 1999: The Incredible Transmogrifying Widget
17. JavaWorld Nov 1999: Menu Negotiation
18. JavaWorld Jun 1999: Reactors, Active Objects, and Consoles
19. JavaWorld May 1999: Blocking Queues, Thread Pools, An OO Look at Threads
20. JavaWorld Apr 1999: Reader/Writer locks, Singletons, JDK 1.1 Class-Unloading Bug
21. JavaWorld Mar 1999: Observer and the Mysteries of the AWTEventMulticaster
22. JavaWorld Feb 1999: Timers
23. JavaWorld Dec 1998: Condition Variables and Counting Semaphores
24. JavaWorld Nov 1998: Semaphore, Lock_manager, and Mutex
25. JavaWorld Oct 1998: Common multithreading Pitfalls (Deadlock, etc.)
26. JavaWorld Sep 1998: Threading Architectures
27. IBM DeveloperWorks Jul 2000: The OO-Design Process: Getting started: How to prioritize
28. IBM DeveloperWorks Aug 2000: The OO-Design Process: Beginning to design software.
29. IBM DeveloperWorks Sep 2000: The OO-Design Process: Refining the problem definition
30. IBM DeveloperWorks Oct 2000: If I were king: A proposal for fixing the Java programming language's threading problems.
31. IBM DeveloperWorks Dec 2000: The OO-Design Process: Verifying the analysis
32. IBM DeveloperWorks Jan 2001: The OO-Design Process: Use cases, an introduction
33. IBM DeveloperWorks Mar 2001: The OO-Design Process: Use-case planning
34. IBM DeveloperWorks Apr 2001: The OO-Design Process: Use-cases applied, Part 1

35. IBM DeveloperWorks May 2001: The OO-Design Process: Use-cases applied, Part 2
36. SD Times 2004-05-15: You've Gotta Have Faith
37. SD Times 2004-06-01: When Hiring, Smarts Beat Skill Lists
38. SD Times 2004-06-15: Good Tech Support Yields Better Software
39. SD Times 2004-07-01: More Thoughts on Tiger
40. SD Times 2004-07-15: Java Still Struggles with Persistence
41. SD Times 2004-08-01: JavaOne, Community and Dumb Ideas
42. SD Times 2004-08-15: From JavaOne to JavaOne-Half
43. SD Times 2004-09-01: The Dark Underbelly of Community
44. SD Times 2004-09-15: First, or Best, to Market?
45. SD Times 2004-10-15: Schizophrenic Development
46. SD Times 2004-11-01: The Protection Racket Shakedown
47. SD Times 2004-11-15: Reading About Design Patterns
48. SD Times 2004-12-15: Do we really need the JCP?
49. SD Times 2005-01-01: Open-Source Context
50. SD Times 2005-01-15: Teach Your Programmers Well
51. SD Times 2005-02-01: The Prince of Purity
52. SD Times 2005-02-15: Tools that work Against You
53. SD Times 2005-03-15: Is Software Engineering and Oxymoron
54. SD Times 2005-04-01: Java Annotation
55. SD Times 2005-04-15: What's not Coming in Mustang
56. SD Times 2005-05-01: The Terror of Code in the Wrong hands
57. SD Times 2005-05-15: XML Digital Signatures in Mustang
58. SD Times 2005-06-01: Finding PathFinder
59. SD Times 2005-06-15: JavaOne and Stalinist management
60. SD Times 2005-07-01: A Cautionary Tale
61. SD Times 2005-07-15: Visual Java
62. SD Times 2005-08-15: Java Studio Creator
63. SD Times 2005-09-01: Learning UML 2
64. SD Times 2005-09-15: Requirements Gathering
65. SD Times 2005-10-01: A UI Designer's Reading List
66. SD Times 2005-10-15: A Conversation with Creator's Creator
67. SD Times 2005-11-01: Jumping Off the Bandwagon
68. SD Times 2005-11-15: Coordinating the Evolution of Java
69. SD Times 2005-12-01: Culture Clash
70. SD Times 2005-12-15: JasperReports Disappoints
71. SD Times 2006-01-01: A Bolshevik Take on Computer Security
72. SD Times 2006-01-15: I Text, You Text, We All Text for iText
73. SD Times 2006-02-01: The Next Big Thing
74. SD Times 2006-02-15: Scripting and Java
75. SD Times 2006-03-01: AJAX is No Panacea
76. SD Times 2006-03-15: Two Sides to AJAX Toolkits
77. SD Times 2006-04-01: The End of the World as We Know It
78. SD Times 2006-04-15: A Culture of Fear
79. SD Times 2006-05-01: Stomping the Bugs
80. SD Times 2006-05-15: The Clearinghouse Model
81. SD Times 2006-06-01: A Taxonomy of Coding Errors
82. SD Times 2006-06-01: Rent Before You Buy Into Offshoring
83. SD Times 2006-06-15: A Sleepy JavaOne
84. SD Times 2006-07-01: The Google Web Toolkit
85. SD Times 2006-07-15: TestNG is A-OK
86. SD Times 2006-08-01: There's No Avoiding Politics
87. SD Times 2006-08-15: The Next Big Thing
88. SD Times 2006-09-01: just Say No to XML
89. SD Times 2006-09-15: The Finesse of FitNesse]
90. The UML Reference Card
C/C++ Users Journal Oct 2003

91. Is It Time for Java++? (Editor's Forum).
C/C++ Users Journal Oct 2002 v20 i10 pS2(1)
92. Know Thy Machine. (Editor's Forum). (Editorial).
C/C++ Users Journal August 2002 v20 i8 pS2(1)
93. Web services: just say no (for now). (Editor's Forum).
C/C++ Users Journal June 2002 v20 i6 pS2(1)
94. Adopting OO. (Editor's Forum). (Editorial).
C/C++ Users Journal Feb 2002 v20 i2 pS2(1)
95. Is There Anything to JavaBeans but Gas? (Editor's Forum).
C/C++ Users Journal Dec 2001 v19 i12 pS2(1)
96. Editor's Forum. (Editorial).
C/C++ Users Journal Oct 2001 v19 i10 pS2
97. Hide your data and make objects responsible for their own user interfaces, part III.
(Technology Tutorial)(Cover Story).
Microsoft Systems Journal Jan 1997 v12 n1 p67(12)
98. Hide your data and make objects responsible for their own user interfaces, part II.
(Technology Tutorial).
Microsoft Systems Journal Dec 1996 v11 n12 p35(8)
99. Hide your data and make objects responsible for their own user interfaces. (includes
related article on Booch notation)(part 1) (Technology Tutorial)(Tutorial).
Microsoft Systems Journal August 1996 v11 n8 p53(17)
100. Roll your own persistence implementations to go beyond the MFC frontier.
(developing a persistence implementation that is not integrated with MFC, but that can
coexist with MFC) (Technology Tutorial)(Tutorial).
Microsoft Systems Journal June 1996 v11 n6 p31(16)
101. Rewriting the MFC Scribble program using an object-oriented design approach.
(Microsoft Foundation Classes)(Tutorial).
Microsoft Systems Journal August 1995 v10 n8 p17(19)
102. Using C++ for directory management: use operator overloading to build C++ tools
for managing your directories. (Tutorial)
Byte Dec 1993 v18 n13 p213(6)
103. Visual C++: its compiler, language implementation, and code quality. (summary of
feature article) (Brief Article).
Microsoft Systems Journal June 1993 v8 n6 pCOV(1)
104. Visual C++: its compiler, language implementation and code quality. (Microsoft
Visual C++ application development environment).
Microsoft Systems Journal June 1993 v8 n6 p65(11)
105. The power of inheritance. (avoiding multiple inheritance problems in C++) (Some
Assembly Required).
Byte May 1993 v18 n6 p221(4)
106. The C majors. (eleven DOS-based compilers perform in a virtuoso test) Scott
Lewis, Kenji Hino, Allen Holub, Scott Ladd, G. Michael Vose, Robert Ward.
Computer Language Feb 1989 v6 n2 p31(25)
107. C Chest: Shell archives. (C language programming technique)
Dr. Dobbs Journal of Software Tools for the Professional Programmer. July 1988
v13 n7 p116(9)
108. C Chest: Stalking the wild memory allocator. (problem-solving)
Dr. Dobbs Journal of Software Tools for the Professional Programmer. June 1988
v13 n6 p80(9)
109. C Chest: Postfix notation and common-subexpression elimination.
Dr. Dobbs Journal of Software Tools for the Professional Programmer. May 1988
v13 n5 p72(8)
110. C Chest: Formatted print functions: the innards.
Dr. Dobbs Journal of Software Tools for the Professional Programmer. April 1988
v13 n4 p98(4)
111. C Chest: Hiding configuration information.
Dr. Dobb's Journal of Software Tools Feb 1988 v13 n2 p92(6)

112. C Chest: A preemptive multitasking kernel continued, Lattice dBC, and compiler controversies.
Dr. Dobb's Journal of Software Tools Jan 1988 v13 n1 p72(9)
113. C Chest: A preemptive multitasking kernel and more mean subroutines.
Dr. Dobbs Journal of Software Tools for the Professional Programmer. Dec 1987 v12 n12 p126(9)
114. C Chest: Using the Unix-ANSI time functions.
Dr. Dobbs Journal of Software Tools for the Professional Programmer. Nov 1987 v12 n11 p116(2)
115. C Chest: Language wars over C's.
Dr. Dobbs Journal of Software Tools for the Professional Programmer. Oct 1987 v12 n10 p124(4)
116. C Chest: The ultimate metronome: writing interrupt service routines in C.
Dr. Dobbs Journal of Software Tools for the Professional Programmer. Sept 1987 v12 n9 p106(9)
117. C Chest: Subroutines with a variable number of arguments.
Dr. Dobbs Journal of Software Tools for the Professional Programmer. Aug. 1987 v12 n8 p100(6)
118. C Chest: Curses: Unix-Compatible windowing output functions.
Dr. Dobbs Journal of Software Tools for the Professional Programmer. July 1987 v12 n7 p94(19)
119. C Chest: Priority Queues.
Dr. Dobbs Journal of Software Tools for the Professional Programmer. June 1987 v12 n6 p102(13)
120. C Chest: Statistical application of digital low-pass filters, exec bug in Microsoft C.
Dr. Dobbs Journal of Software Tools for the Professional Programmer. May 1987 v12 n5 p102(31)
121. C Chest: Nr: a C implementation of Nroff, part 3. (continuation of user manual for Nr)
Dr. Dobbs Journal of Software Tools for the Professional Programmer. April 1987 v12 n4 p130(8)
122. C Chest: Nr: a C implementation of Nroff, part 2.
Dr. Dobbs Journal of Software Tools for the Professional Programmer. March 1987 v12 n3 p96(28)
123. C Chest: Nroff: hashing, expressions, and roman numerals.
Dr. Dobbs Journal of Software Tools for the Professional Programmer. Feb 1987 v12 n2 p90(23)
124. C Chest: Shrinking.EXE file images.
Dr. Dobb's Journal Jan 1987 v12 n1 p104(7)
125. C Chest: Sets in Microsoft C, version 4.
Dr. Dobb's Journal Nov 1986 v11 n11 p14(7)
126. C Chest: more, a file-browsing utility.
Dr. Dobb's Journal Oct 1986 v11 n10 p22(3)
127. C Chest: directory traversal, trailing Zs, and horrifying experiences.
Dr. Dobb's Journal Sept 1986 v11 n9 p14(7)
128. C Chest: An AVL tree database package. (Adelson-Velski and Landis)
Dr. Dobb's Journal August 1986 v11 n8 p20(7)
129. C Chest: trees and more on Microsoft and Lattice compilers.
Dr. Dobb's Journal July 1986 v11 n7 p18(6)
130. C Chest: sort - a general purpose sorting program.
Dr. Dobb's Journal June 1986 v11 n6 p22(6)
131. C Chest: Assessing IBM video display memory and a Microsoft bug.
Dr. Dobb's Journal May 1986 v11 n5 p18(5)
132. C chest: redirection - the -dev Directory, SWITCHAR, and Touch.
Dr. Dobb's Journal April 1986 v11 n4 p18
133. C Chest: the last of the shell support routines.
Dr. Dobb's Journal March 1986 v11 n3 p14(8)
134. C Chest: A New Shell for MS DOS (continued).
Dr. Dobb's Journal Feb 1986 v11 n2 p16

135. C Chest: A Unix-Like Shell for MS DOS.
Dr. Dobb's Journal Jan 1986 v11 n1 p18-20
136. C Chest: A New Shell for MS DOS, Part 1
Dr. Dobb's Journal Dec 1986 v10 n12
137. C Chest: How Compilers Work -- A Simple Desk Calculator
Dr. Dobb's Journal Aug 1986 v10 n9
138. C Chest: Make
Dr. Dobb's Journal Aug 1985 v10 n8
139. C Chest: An implementation of the UNIX Is utility for MSDOS
Dr. Dobb's Journal Jul 1985 v10 n7
140. C Chest: Queues and Bit Maps
Dr. Dobb's Journal Jun 1985 v10 n6
141. C Chest: Command Line Processing
Dr. Dobb's Journal May 1985 v10 n5
142. C Chest: Sorting Routines: Shell Sort and Quicksort
Dr. Dobb's Journal Apr 1985 v10 n4
143. C Chest: Pipes, Wild-Card Expansion, and Quoted Arguments
Dr. Dobb's Journal Mar 1985 v10 n3
144. Grep.c--A Unix-like Generalized regular Expression Parser.
Dr. Dobb's Journal Oct 1984 v9 n10
145. A Window System for Text Editing.
Programmer's Journal Nov/Dec 1991 v9 n6
146. Window Layers.
Programmer's Journal Sep/Oct 1991 v9 n5
147. Window Viewports.
Programmer's Journal Jul/Aug 1991 v9 n4
148. Text Buffers in Windows.
Programmer's Journal Jan/Feb 1991 v9 n1
149. Virtual Memory Paging.
Programmer's Journal Nov/Dec 1990 v8 n6
150. A Virtual Memory System in C/C++.
Programmer's Journal Sep/Oct 1990 v8 n5
151. Using C++ Features in C.
Programmer's Journal Jul/Aug 1990 v8 n4
152. Programming with State Machines.
C Gazette. 1990?
153. Solving the Scheduling Problem Using Game Theory.
C Gazette. 1990?

Have also written articles for Mac Tech Journal (4 articles)

This list may not be up to date. I typically publish one or two articles per month.

Service, Committees, Editorships, etc.

Security-Track Chair, Software Development Expo conference.

Member, Board of Advisors, Ascenium Corp., Ontometrics, Inc..

Advisory Council, University of Calif. Extension, Computer Information Systems Dept.

Technical and Contributing Editor, *Dr. Dobb's Journal*, *SD Times*

Senior Editor, *C/C++ User's Journal*, *Java Solutions*

Contributing Editor, *SD Times*, *JavaWorld*, *Programmer's Journal*, *Dr. Dobb's Journal*

Technical reviewer, Kernighan and Ritchie, *The C Programming Language*.

Technical reviewer, Spolsky, *User Interface Design for Programmers*.

Professional Organizations

IEEE, ACM

Education

University of California, Berkeley: B.A. in Computer Science and Medieval-European History, 1981.

Personal

Instrument-rated commercial pilot, luthier, composer/musician (percussion and keyboards), painter, and former modern/ballet dancer.