

Timothy Daly
146 Century Farm Rd
Clinton, PA 15026
(412) 418-0751
daly@axiom-developer.org
<http://daly.axiom-developer.org>

WORK EXPERIENCE

Literate Software: (Feb 2011 - Present)

Consulting contracts:

- Develop a management plan for conversion of online course material to mobile devices (iPad/Objective C), (Android/Java)
- Develop a multi-touch vision tracking system (C++)
- Design parallel DNA sequence alignment using GreenArrays hardware chips (Forth, C)
- Design literate software extraction program (Lisp)

Carnegie-Mellon University: (July 2005-Feb 2011)

Top Secret Security Clearance

The Carnegie Mellon Software Engineering Institute (SEI) is a federally funded research and development center. The SEI staff has advanced software engineering principles and practices in software engineering, computer security, and process improvement.

- Function Extraction (FX) is a disruptive new technology that will substantially improve the economics of software development and increase the dependability of software systems. Function Extraction applies mathematical foundations to automate calculation of the behavior of software, the objective being to replace slow and fallible manual methods of code reading and inspection with fast and correct computation of behavior. My main focus was the definition of semantics for the Intel Instruction set. Programmed in Lisp.
- XNET is a CERT “train as you fight” exercise network. It provides a cyber-security platform for doing “live-fire” exercises to replicate real-world cyber attack, defend, and respond missions. My main focus involved writing a dynamic IPTable module to route remote users to back-end servers. Programmed in C and Intel Assembler.
- Forensics focuses on writing tools for Law Enforcement Officers. My main focus was developing cross-platform tools to scan for Credit Card information in a variety of file formats. Programmed in Python.
- Network Situational Awareness (NetSA) has developed and maintains a suite of open source tools for monitoring large-scale networks using flow data. The System for Internet Level Knowledge (SiLK) is a network flow collection and storage infrastructure that will accept flow data from a variety of sensors. My main focus was developing a graphical Configuration Editor. Programmed in Python and Bash shell.

Member of Board of Directors of the Association of Lisp Users

(Feb 2009 - May 2010)

CAISS Institute City College of New York: (Sept 2002 - May 2005)

Researcher leading three open source mathematics projects: Axiom, Magnus, and Doyen. Managed 7 students. Worked with ACM and ISSAC. Individual Research in Literate Programming. C++ and Lisp programming.

- Lead developer on Magnus, an infinite group theory package for exploratory Infinite Group Theory (Gilbert Baumslag, Marianna Papaleo, Xiawei Xu, Marcos, Zyman, Peggy Dean). 150k lines of C++ source code.
<http://sourceforge.net/projects/magnus>
- Lead developer on Doyen, a liveCD science platform (Wilken Rivera, Jose Alfredo Perez) <http://sourceforge.net/projects/doyencd>

- Lead developer on Axiom, a general purpose computer algebra system. <http://axiom-developer.org>
- Principle Investigator. NSF Grant on Indefinites (William Sit, Gilbert Baumslag, Sean Cleary, Douglas Troegar)
- Conference Chair “The Future of Scientific Computation” (CCNY 2004)

Worldcom: (Oct 2000 - July 2002)

Primarily responsible for validating data center network infrastructure. Test and certify new routers, monitor for equipment failures, and generate customer billing information. Programming in Java and Lisp.

- Team Lead to install monitoring systems for all data center switching within Worldcom (93 data centers worldwide). SNMP monitoring for failure alerts and billing. Wrote Java/XML program to audit actual equipment versus monitoring database.
- Managed a lab of networking equipment and racks of Linux and Sun servers to test management software and network configurations.
- Wrote performance and stress testing software for high-end backbone routers and switches.

Cold Spring Harbor Labs: (Oct 2000 - April 2001)

Consulting work for Stein Labs developing a program to enable Biochemistry researchers to search and annotate DNA sequences. My effort involved web-enabling SNP searches in DNA sequences. Programming in Javascript and C.

Centerport: (Feb 2000 - Oct 2000)

Developed a converged user interface for four internal systems that are used to deploy web based measurement and marketing programs. Programming in Java and Perl.

UUNET: (Dec 1998 - Feb 2000)

Responsible for designing programs to monitor the status of servers and databases on the UUNET global hosting network (3500 nodes). I wrote a multithreaded, distributed monitoring program to show the status, display a “trouble board”, and interface to a Remedy trouble-ticket system. Programmed in Java.

Stood up a networking lab to test Gigabit switching networks from multiple vendors for deployment in UUNET data centers.

Approach: (Dec 1996 to Dec 1998)

Approach was a consulting company. I developed a technology demonstration for Comcast (J++ and DCOM) showing statistics in a Java applet on a web page.

I wrote a business case detailing a plan for EDI consulting. I created Java classes that constructed the BAPLIE (shipping) and JAPRES (job scheduling) message sets.

I was a member programming team to automate paying commissions to sales people at Comcast. I built a proof of concept in Java, wrote SQL and stored procedures to access and validate data. I wrote a CLIPS compatible rule-based program in the business layer. I jointly helped build the central payment system in C++ and Visual Basic.

I was the co-lead for developing an ECommerce guide for use in pre-sale client interviews.

I did a systems analysis for requirements gathering and use-case design for streaming video presentations.

I developed a reusable 3-tier architecture using CORBA and Java, interfacing with COM objects. It used UML patterns to automate the construction of large portions of the archi-

ture for client applications.

IBM Research: (August 1978 - Dec 1995)

Natural Language Understanding (1995): Developed a CHART parser and knowledge representation program (KREP-based) to read mail, extract sentences involving priority phrases, scheduling phrases, and meeting phrases to sort, prioritize and schedule calendar events.

Rewritable Paper (1995) Independent research to invent, develop, and patent rewritable paper based on organic materials.

Axiom Symbolic Computation System (1989-1995): Research and development of a commercial computer algebra system. Primary task was the development of system interpreter internals, written in Lisp. Developed algebra for Denavit-Hartenberg Matrices.

ECLPS (1986-1989): Develop commercial rule-based programming language for Expert Systems. Written in Lisp and OPS5.

FAME (1986-1989): Wrote KROPS, the implementation language for FAME (Financial and Marketing Expert) based on OPS5 for rules and KREP for knowledge representation.

BOXER (1984-1986): Research in Robot Planning. Invented and developed a system to reason about solid models and generate robot motion to perform automatic assembly as a "design-to-build" system.

AML (1983-1984): Co-author of AML Robot Language, an IBM commercial product for real-time robot control. Received Patent for Slip Sensor (4,605,354)

VM/370 Systems Programmer (1980-1983): Mainframe operating system performance and tuning. Co-authored subsystem to manage free storage, shipped as part of high performance upgrade to VM/370.

Archive (1978-1979): Designed and developed an online archival storage and retrieval system using a hierarchy of disk and tape backups.

Unimation: (May 1976 - July 1978)

VAL (Vicarm Assembly Language) - Co-developed real-time commercial robot programming language.

Open Source Work: (1997 - present)

Clojure in Small Pieces – A literate program implementation of the Clojure programming language. (<http://daly.axiom-developer.org/clojure.pamphlet>)

Axiom - Computer Algebra System, originally written at IBM Research and sold as a commercial competitor to Mathematica and Maple. Open sourced in 2001. I am the Lead Developer (<http://axiom-developer.org>).

Magnus - Infinte Group Theory Computer Algebra system, originally written at City College of New York CAISS Institute. I am the Lead Developer (<http://sourceforge.net/projects/magnus>)

Doyen - Scientific Platfor on Live CD. Used by Sage, Axiom, and Magnus. Lead Developer (<http://daly.axiom-developer.org/doyen>)

Pinger - Protocol-agile real-time monitoring of internet backbone routers and data center server racks. Used for trouble detection at Worldcom. Lead Developer (1997, project defunct).

TEACHING EXPERIENCE

City College of New York, NY

Open Source Programming (Fall 2004)

Vassar College, NY

Compilers (Spring 1990)

Programming Languages (Fall 1990)

William Patterson College, NJ

Data Structures (Fall 1981)

EDUCATION

Ph.D. program (thesis incomplete) Polytechnic Institute of New York, Brooklyn, N.Y.
(Sept 1997) Topic: "Provisos – Conditional Computational Algebra"

Masters in Computer Science Fairleigh Dickinson University, Teaneck, N.J. (May 1980)

Bachelor of Science (Mathematics) Montclair State College, Montclair, N.J. (June 1975)

PUBLICATIONS

Daly, Tim *Publishing Computational Mathematics* Notices of the American Mathematical Society, Vol 59, No. 2 Feb 2012 pp 320-321

Daly, T., Linger, R., Prowell, S., Burns, L. *Concordia: Google for Malware* Journal of Information Systems and e-Business Management (accepted for publication)

Linger, R., Sayer, K., Daly, T., Pleszkoch, M. *Function Extraction Technology: Computing the Behavior of Malware* Proceedings of Hawaii International Conference on System Sciences, Kuauai, Hawaii, January 2011

Daly, T., Linger, R. *Test and Evaluation: Robot Forward Kinematics* CERT IRAD report, Feb 2010

Daly, T., Burns, L. *Concordia: A Google for Malware* 6th Annual Cyber Security and Information Intelligence Research Workshop, Oak Ridge National Labs, April 21-23, 2010

Daly, T. Burns, L. *Concurrent Architecture for Automated Malware Classification* Proceedings of Hawaii International Conference on System Sciences, Kuauai, Hawaii, January 2010

Sayer, K., Pleszkoch, M., Daly, T., Linger, R., Prowell, S. *Function Extraction for Malicious Code Analysis* CERT 2009 Research Report (R. Linger, Ed.) Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, 2010

Linger, R., Daly, T. *Advanced Technology for Test and Evaluation of Embedded Systems* CERT 2009 Research Report (R. Linger, Ed.) Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, 2010

Burns, L. Daly, T. *FXplorer: Exploration of Software Behavior: A New Approach to Code Understanding and Verification* Proceedings of Hawaii International Conference on System Sciences, Kona, Hawaii, January 2009

Bartholomew, R., Burns, L., Daly, T., Linger, R., Prowell, S. *Function Extraction: Automated Behavior Computation for Aerospace Software Verification and Certification* American Institute of Aeronautics and Astronautics, Sonoma CA, 2007

Linger, R., Burns, L., Daly, T., Pleszkoch, M., Prowell, S., Sayer, K., Walton, G. *Function*

Extraction Invention Disclosure, 2007

Mili, A., Daly, T., Pleszkoch, M., Prowell, S. *A Semantic Recognizer Infrastructure for Computing Loop Behavior* Sept 2006

Linger, R., Daly, T., Pleszkoch, M., Prowell, S., and Walton, G. *Function Extraction (FX) Technology: Automated Calculation of Program Behavior for Software Assurance* Proceedings of NSA High Confidence Software and Systems Conference, Ft. Meade, MD, 2006

Daly, T. "Axiom: The 30 Year Horizon. A Tutorial", Dec 2005, 288 pages, Lulu.com ISBN 1-411-66597-X

Daly, T. *Axiom Computer Algebra System* ACM SIGSAM Bulletin, Vol 36, No. 1, March 2002 Workshop on Open Source Computer Algebra, Lyon France (Invited Paper)

Daly, T. *Provisos: A Programming Extension for Conditional Computational Algebra* (Ph.D. Thesis, Incomplete) 1995

Daly, T. *Susan: Adding Meaning to Desktop Folders* IBM Internal Publication, May, 1992

Jenks, et. al. "Axiom: The Scientific Computation System" Springer-Verlag, NY ISBN 0-387-97855-0, May 1992

Daly, T. *Axiom Library Compiler* Book Chapter 21, NAG LLC, England, May 1992

Daly, T., Kastner, J., Mays, E. *Integrating rules and inheritance networks in a knowledge-based financial marketing consultation system* Proceedings of Hawaii International Conference on System Sciences, Jan 1988, pp496-500

Daly, T. *Verification and Validation of Rule Based Programs* American Society of Mechanical Engineers, June 1989

Daly, T. *Integrating Rules and Inheritance Networks* IBM RC 12144, Sept 1987

Daly, T. *Boxer – A Design-to-Build System* June 1985 MIT Robot Symposium (Invited Paper)

Daly, T. *Boxer – A Design-to-Build System* IBM RC 11096

Bozman, G., Bucu, W., Daly, T., Tetzlaff, W. *Analysis of free storage algorithms – Revisited* IBM Systems Journal, (Feb 1984) pp44-64