

**Gary L. Wade**  
**320 Crescent Village Circle**  
**Unit 1396**  
**San Jose, CA 95134**  
**garywade@desisoftsystems.com**  
**949-505-3223**

## **KEY EXPERIENCES**

- Passion for developing solutions for Apple Inc.'s platforms (macOS, iOS, tvOS, watchOS)
- Innovative in usability design and testing, and refactoring for performance
- Design and development of multi-platform client/server/cloud software solutions
- Facilitate instruction for technical and non-technical staff including one-on-one mentoring

---

## **CURRENT TALENT SUMMARY (numbers are footnotes for positions referenced below)**

- Adobe Photoshop (24,17,10,9,8,1)
- AppKit/macOS (28,27,26,21,20,19,18,17,1)
- C (28,27,26,24,23,22,21,20,19,18,17,16,15,14,13,12,11,8,7,6,5,4,1)
- C++ (21,20,18,17,16,15,14,13,12,11,8,7,6,5,1)
- Client/Server/Cloud (28,27,26,25,24,22,21,20,19,18,17,16,13,7,6,4)
- Cocoa (28,27,26,25,24,23,22,21,20,19,18,17,1)
- CoreFoundation (28,27,26,22,21,20,18,17,16,15,14,13,12,1)
- Cross-platform (28,27,21,20,17,15,14,13,12,8,7,6,5,4,2)
- git (28,27,26,25,24,23,22)
- HTML (28,27,26,25,17,14,12,10,9,1)
- Int. Comp. for Unicode/ICU (28,27,25,22,17)
- Internationalization (28,27,26,25,24,23,22,18,17,15,14,12,7,3,1)
- JavaScript (28,21,12,10,9,1)
- JSON (28,27,26,25,24,22,21,20)
- LDML (17)
- Macintosh (Mac OS X) SDK (28,27,26,21,20,19,18,17,16,15,14,13,12,11,8,7,6,5,4,2,1)
- Objective-C (28,27,26,24,23,22,21,20,19,18,17,1)
- Perforce (21,20,15)
- Quality Assurance (25,24,17,3,1)
- QuickTime/AVKit (25,24,22,17,13,10,9,6,5)
- REST (28,27,26,25,24,22,21)
- Sales support (21,10,9,6,1)
- sqlite (23)
- Subversion (27,22,19,18,1)
- Swift (28,27,26,25,24)
- UIKit/iOS (28,27,26,24,23,22,1)
- UIKit/tvOS (25)
- Unicode (28,27,26,25,24,23,22,21,17,15,14,12,1)
- Unix (28,27,26,25,24,23,22,21,20,19,18,17,15,14,13,10,9,1)
- Xcode (28,27,26,25,24,23,22,21,20,19,18,17,1)
- XML (22,21,17,14,1)

**HISTORICAL TALENT SUMMARY** (numbers are footnotes for positions referenced below)

- Adobe Illustrator (10,9,1)
- Adobe PageMaker (10,9,8,5,1)
- AppleTalk (17,16,8,6,5)
- Carbon API (17,16,15,14,13,12,1)
- CGI (10,9,1)
- CodeBase (16)
- ColorSync/Color Matching (11)
- COM/OLE (14,12,5)
- CVS (17,16,14,12)
- Java (9)
- Metrowerks CodeWarrior (17,16,15,14,13,12,11,8,7,1)
- MFC (12,8)
- MPW (4,1)
- Oracle (9)
- Perl (17,14,10,9,1)
- PowerPlant Class Library (13,11,8)
- Project/Team management (10,9,7)
- Sybase (10)
- THINK C (7,6,5,4,1)
- THINK Class Library (7,6,5,4,1)
- Visual C++ (17,12,8,1)
- WASTE (16)
- Web browser plug-in SDK (14,13,12)
- Web server development (10,9,1)
- Windows SDK (17,12,8,5,1)
- XCMD (HyperCard) development (6)

---

## EMPLOYMENT AND CONTRACTS

### Personal Projects/Self-Employed as DesiSoft Systems – (1)

- Coined the phrase “Genocidal Refresh” as an anti-pattern after seeing the behavior prevalent amongst less experienced developers. Genocidal Refresh happens when a developer chooses to throw all objects away and rebuild everything when all that is needed is the addition or removal of a single instance or property, especially when rebuilding requires an extreme performance and/or memory hit.
- Developing an iOS app to give hints and answers to users of games like Scrabble, crossword puzzles, and 4 Pics 1 Word, that takes a set of possible letters, the number of required letters, the known letters in certain positions, and desired letters to show in the results, utilizing a self-designed trie structure that rivals in performance and memory footprint a number of Apple’s collection classes
- Developing an iOS app that uses Apple’s Data Detectors to parse text containing time sheet data (dates, time ranges, notes) into usable framework objects for synchronizing with events in Apple’s Calendar services, and utilizing swizzle-based overrides in NSLayoutManager to uniquely draw ranges of glyphs identified by application-defined attributes (e.g., an attribute identifying a set of text as being the notes for a day’s work), as well as custom push-gestured menu items to manually tag such items. Discovered a limitation in NSTextCheckingResult in Apple’s Data Detectors with respect to identifying dates or times without the other component, and developed a solution that works with current versions of iOS and OS X to the problem by introspecting Apple’s private runtime object data, and submitted a bug request to Apple to make the solution or one like it public in a future version of iOS and OS X, as well as blessing the current solution for prior versions.
- Developing a tool for use by a novel writers organization to show how far each member is from the local chapters using MapKit so chapter leaders may connect with the members closest to them.
- Developing a competitor to Contour by Mariner Software that overcomes limitations and usability issues to assist my novel writing, featuring auto-sizing text fields like OS X’s Contacts, navigation and progress indicators for each section of the three part novel structure, separate documents for each story, and hiding or showing of sections. Also generalizing it for other Q&A-based applications. For a demo video, go to <http://www.desisoftsystems.com/user/garywade/ResumeNovelCharacters.mov>
- Developing a play script application to assist actors in rehearsing their parts by themselves utilizing speech synthesis to speak other parts and a cadence indicator for all parts based on the user’s choice of voice speed per part. Play scripts are constructed by the application from HTML data by automatically inspecting the DOM tree provided by WebKit or manually by the user and categorized according to acts, scenes, parts portrayed, lines spoken, and directions given. Continuously updated tooltips were developed to display voice rate values and DOM structure information while navigating using slider controls. An overlay window was synchronized in size and position over each WebKit view to provide alpha-transparent color indicators per chosen item to prevent modification of the original HTML/CSS. For a demo video, go to <http://www.desisoftsystems.com/user/garywade/ResumeScriptPrompterDemo.m4v>
- Developing an Activity Monitor-like application whose current purpose is to provide a graphical means of changing the nice level (Unix priority value from 20 to -20) of each process. Provided the means to filter processes based on various architecture and runtime considerations. Included the ability to view icons of each process at the size chosen by the user. Included the use of the view-class SFAuthorizationView in order to elevate the user’s permission level in order to execute the renice

command. For a demo video, go to

<http://www.desisoftsystems.com/products/DSSProcessSpeedometer/DSSProcessSpeedometer-1.0a2.zip>

- Developing an iOS version of my DSS Name Convoluter OS X and Windows app that will allow user-defined rules beyond those characterized by the so-called “Star Wars Name” algorithm, and generalizing it for any information convolution, such as how some users choose unique passwords per service.

**Apple Inc.**, Cupertino, CA – (28)

Senior Software Engineer, macOS, iOS, tvOS

May 2018 – Present

- Using Swift and Objective-C, fixed bugs, refactored code, and developed new features including internal SDKs, for Apple’s media and productivity apps especially pertaining to internationalizing software with respect to bi-directional user interfaces (Arabic and Hebrew).
- Met with stake-holders and provided feedback by way of code reviews, team-internal bugs and feature requests, and provided solutions for many and software architectural designs for others.
- Developed unit and UI tests for new features.

**Apple Inc.**, Cupertino, CA – (27)

Senior Software Engineer, macOS, iOS

September 2017 – April 2018

- Using Swift and Objective-C, fixed bugs, refactored code, and developed new features for Cocoa-based apps running on macOS and iOS, and used by data scientists and new teams within Apple.
- Developed unit and UI tests for new features.
- Provided feedback by way of code reviews, team-internal bugs and feature requests, and provided solutions for many and software architectural designs for others.
- Developed a complete macOS app using a secure shell bastion to access some web sites and a per-WKWebView SOCKS proxy for others by adding per-instance SOCKS proxy support to the WebKit project’s source for future inclusion. Added to WKWebView the ability to modify contextual menu items before being presented to the user. Included support for test and production user authentication, and software update management. Included the ability to determine what web servers a user has access to through an authenticated service that returns its results in JSON including extra configuration options required for accessing the web site such as tunneling, SOCKS proxy, public key, type of user name.
- Developed a number of cross-application components in Swift and Objective-C including a network-connection indicator as a window affordance companion to the close, minimize, and zoom buttons; the ability to look up a user’s various accounts, credentials, and information; the means to automatically adopt the predominant color of an image as the background for the image’s superview (e.g., a red superview for a red icon, a blue superview for a blue icon); a view that encapsulates a set of any number of graphically-separate, segmented-control-like buttons that are self-resizable and capable of multiple themes and which may be instantiated completely inside Interface Builder with User Defined Runtime Attributes; the ability to manage authentication with a YubiKey; the ability to manage tunneling with multiple, individual secure shell instances in an app; a fairly comprehensive set of HTTP status codes used and suggested for inclusion in Foundation’s headers; a set of data conversion components available by selecting a contextual menu/floating window that goes beyond Data Detectors to show the selected text within a popover in other formats such as the hostname for an IP address, its IPv4 or IPv6 representation, or the local time for a UTC time string; the ability to use layer-based graphics values within a view’s User Defined Runtime Attributes in Interface Builder for pseudo-button backgrounds;

generic enclosing view/view controller discovery methods; a method to prepare multiline text for better readability in log messages; temporary file object support.

- Provided programmatic suggestions and solutions for users on various email groups.

**Apple Inc.**, Cupertino, CA – (26)

Senior Software Engineering Consultant, macOS, iOS (Intelliswift Software, Inc.)

March 2017 – September 2017

- Using Swift and Objective-C, fixed bugs, refactored code, and developed new features for Cocoa-based apps running on macOS and iOS used by data scientists and new teams within Apple.
- Developed unit and UI tests for new features.
- Developed both iOS and macOS-based server-side applications for the purpose of sending, receiving, and testing Apple Push Notifications, including research into using HTTP/2 for the new APNS services.
- Wrote a white paper about the protobuf-based data used by an internal application for third-party developers not using Google's protobuf generators, source code, nor any knowledge of the protobuf format.
- Provided feedback by way of code reviews, team-internal bugs and feature requests, and provided solutions for many and software architectural designs for others.

**Saddleback Church**, Lake Forest, CA/Telecommute from Irvine, CA – (25)

Senior Software Engineering Consultant, tvOS (DesiSoft Systems)

December 2016 – Present

- Using Swift, developing an Apple TV app for the church's membership that includes live and on-demand video and audio resources as well as inspirational stories and information about its campuses across the world, utilizing the currently-available REST and other web services for the content.
- Due to Apple TV currently not supporting web views as of tvOS 10, developed a means to view HTML aside and similar paragraph-classed tags into simulated aside content so they are displayed in the same position as they would appear on the desktop or iOS in a web view, by utilizing customized attributed strings and placeholder attributes in order to ensure proper positioning when the owning text view comes into focus and scrolled.
- Developed the code to convert the embedded version of YouTube content provided by the REST and web services into the corresponding YouTube video streams that can be natively displayed within the Apple TV app.
- Modified Swift's JSONEncoder classes to report when encountering keys not defined as part of a type's CodingKeys enumeration in order to work with JSON models that are continuously changing and not well-documented by a service provider.
- Developed an atomic property class to provide thread-safe access to Swift-based properties in a manner similar to how Objective-C defines atomic properties.
- Developed a method for asynchronously performing time-intensive string and graphics manipulation operations (plain text from rich text, typographical quote conversions, titlecasing, image loading, filtering, and rendering) so that content may display placeholders at the immediate time of need but progressively fill in content as it becomes available.
- Developed workarounds for size-class-based localized text for issues where the best choice for a particular screen width and related UI views wasn't being used.
- Developed workarounds for JSONEncoder handling of dates where the ISO 8601 dates returned were actually RFC 3339 and where multiple date formats were used in the same JSON model.

**Flickr Audition App (SharkFeed), Irvine, CA – (24)**

Senior Software Engineering Consultant, iOS (DesiSoft Systems)

September 2016 – Present

- Extended the basic functionality of Flickr's assigned SharkFeed audition app that I developed over a two-day weekend in July 2016 for personal use to support a more robust model controller subsystem with the intent of including other photo-sharing services, caching of photos/videos, preloading videos, animated GIFs and PNGs, display of extended information about photos and owners, geolocation of photos within a map view, quick filtering across media types (photos, videos, both, and none), secondary search of photos within the prescribed feed, infinite scrolling beyond Flickr's return limit of 4,000 photos, multi-state navigation bar similar to Apple News that differs well on various orientations and devices, custom transition animation from a photo's thumbnail to a full-screen light box view and back.
- Wrote a number of blog posts related to undocumented Flickr services and HD video formats, and how to do some atypical things with Apple's UIKit:  
<https://whatweretheythinkingblog.wordpress.com/2016/11/24/knowning-the-rest-of-the-story-flickr-video-formats/>  
<https://whatweretheythinkingblog.wordpress.com/2016/11/22/customizing-mkmapview-needs-apples-help/>  
<https://whatweretheythinkingblog.wordpress.com/2016/11/19/effectively-using-uisearchcontroller-with-uicollectionview/>  
<https://whatweretheythinkingblog.wordpress.com/2016/11/11/equally-spacing-uicollectionview-cells/>
- Converted some of the classes and categories from Objective-C, the language required by Yahoo, into Swift.
- Source code is available to the hiring manager for review as either a current snapshot or git repository.

**Reyes Beverage Group, Chicago, IL/Telecommute from Irvine, CA – (23)**

Senior Software Engineering Consultant, iOS (DesiSoft Systems)

April 2016 – March 2017

- Modernized iOrder, an enterprise-delivered iOS app used by beverage delivery personnel, to use the ARC memory model, 64-bit builds, modern frameworks for networked resources, and auto layout and storyboards.
- Fixed a number of bugs, memory leaks, and crashes, and added changes related to new and changed fields in the sqlite database used by the app.

**Panasonic Avionics, Lake Forest, CA – (22)**

Senior Software Engineering Consultant, iOS (MLS Technologies, Inc.)

June 2015 – September 2016

- Submitted a proposal to management to maximize profits and minimize development effort by building a single, customizable app rather than spending the time and effort in building individual apps per client airline that communicate with Panasonic Avionics ground and in-flight entertainment and backend services. The single app would be customized through automated build settings that would ultimately be chosen through a web interface simple enough to be operated by client airline marketing/UX representatives allowing them to select their unique color schemes, graphics, text and URLs, and desired features and services. The system would also provide the user the option to see their choices through visual feedback similar to how sites like VistaPrint do when choosing business card options, and when satisfied, the system would automatically build a set of apps practically ready to submit to Apple's and Google's app stores.

- Designed and developed a complete iOS app (usable on iPhone and iPad devices in all orientations) distributed through an enterprise license that provided airplane crew with the option to choose individual seats from an airplane-specific chart (the crew, based on preference, could choose the plane's nose position to be vertical or horizontal), or based on cabins, in order to send messages to passengers' seat-back displays, handheld devices, or personal electronic devices, using either a custom message or a predefined message. Also designed the JSON responses needed by the app from the REST services group for the predefined messages in available languages, the layout of seats and rows in relation to the decks of the plane, the available cabins, and the history of messages sent during the flight. For a demo video, go to <http://www.desisoftsystems.com/user/garywade/ResumeCrewMessaging.mov>
- Researched and developed integrating proprietary H.265 codec support within video players targeted for iPhones and iPads
- Submitted bug fixes to the open source project VLC by the Video LAN Organization
- Developed iOS-targeted SDKs along with sample and test apps for internal and external clients for use by the airline industry, both crew- and passenger-focused, using REST calls with airplane- and ground-based servers, and provided feedback as needed to the REST services group
- Provided UX and software architecture direction to other teams (iOS, Android, and REST services group) working on applications and services for aggregate news services, channel listings, weather, and automated purchase and deployment of customized apps for new and existing customers
- Researched various options and suggested the REST services group use <http://swagger.io/> instead of a static wiki so all teams may more easily test, understand, and use their REST APIs
- Suggested to management the need for use of best practices in better and more standard encapsulation of data, internationalization, and regional concerns to the lowest possible levels of the services stack rather than delivering everything in raw form to the application layer which had been requiring application developers to apply meaning to ambiguous data and often duplicating effort needlessly
- Suggested to management the need for customizable, constantly available, virtualized and physical in-flight services in order to test everything, from the server level services, through the REST APIs, to the SDK layer, and up to the application level so that developers at each level may be successful
- Suggested to management the need for streamlining the software build process from the current one that takes approximately a day, many intra-build management-level approvals, and repetitive and manual entry of reporting tickets to one that can be performed within Jenkins and similar tools through automated scripts that delay the need for management approval to when a build is finally completed and ready to deliver to the testing team, app stores, or client airlines

**Intuit Inc., San Diego, CA – (21)**

Senior Software Engineering Consultant, Mac OS X (Xtreme Consulting Group, Inc.)

July 2014 – June 2015

- Refactored TurboTax for Mac to use the latest Mac OS X technologies using Cocoa (Foundation, AppKit, etc.), Objective-C, and Xcode, including the use of fast enumeration, blocks vs invocations, saving log data to a private file vs the system log, reworked method signatures to be more ARC-friendly
- Assisted co-ops, QA, DevOps, and content development groups in utilizing the latest Mac OS X technologies in relation to their work for supporting and working on TurboTax for Mac
- Worked with the TurboTax for Windows software development group in identifying and smoothing out cross-platform issues including XML-based data and the correct encoding methods of Unicode required

- Integrated various Mac OS X, JavaScript, JSON, and REST components related to the 2014-mandated support of the Affordable Care Act into TurboTax for Mac
- Fixed bugs across the product, especially in the areas of responsive, correct, and consistent visual layout, made it possible for users to immediately view a selected state's related tax data from the refund monitor rather than searching for it manually, made it possible for users to view all states' refund-tracking data rather than just the first three, improved the performance and fixed previously-unknown bugs that appeared when supporting auto-layout when users work on raw tax forms vs the interview flow of the product
- Worked with other internal groups and Apple in submitting a sandboxed version and subsequent updates of TurboTax for Mac to the Mac App Store with in-app-purchase options different from the retail/download version
- Worked with customers and internal groups during high-peak usage times (mainly January when tax season began and April just before tax deadline) in resolving issues and identifying problems requiring workarounds or further investigation

**Symantec Corporation**, Mountain View, CA/Telecommute from Irvine, CA – (20)

Principal Software Engineer

November 2012 – June 2014

- Fixed a number of long-standing issues in the Norton Online Backup and Norton Zone products related to the networking models used, code signing, building procedures, compression, file hash calculation, memory management, objects needed, and improved the performance of the products by reducing the inherent complexity that had built up over time and which had been assumed to be necessary to manage a cross-platform codebase vs. a more efficient platform-specific one targeting Apple's frameworks utilizing asynchronous methods over threaded synchronous code.
- Investigated changes within Apple's frameworks across Lion, Mountain Lion, and Mavericks in order to more efficiently draw badges on icons of files and folders in the Finder, including reverse-engineering Apple's changes to their icon data format that supported retina-based icons, and fixed the inclusion of Norton Zone menu items in Finder contextual menus in Mavericks by introspecting the changes in the Objective-C classes used by the Finder application.
- Developed a runloop source and infrastructure for future similar services that provided a more efficient cross-thread method to save log messages to a non-system log file.
- Provided mentoring for junior and less-Cocoa-experienced software developers including pair programming, development direction, and architectural documents as needed.

**Apple Inc.**, Cupertino, CA – (19)

Senior Software Engineer

May 2012 – September 2012

Developed the Time Machine service for OS X Server and added the necessary foreground UI including views, bindings, and filtering predicates to display the options available to the user as well as the display of information related to historical backups on those versions of OS X that support it.

**Western Digital Corporation**, Irvine, CA – (18)

Staff Engineer – Software Development

June 2010 – May 2012

- Converted the previous backup engine's repository from a third-party structure and engine, to a new one using a self-styled format and engine.



- Mentored novice developers on Cocoa, Objective-C, and Xcode performance tools.
- Refactored the backup engine and UI to increase performance, be more maintainable (e.g., transitioned from using `vfs_fsevents` to `FSEvents`, paths to `FSRefs`, synchronous to asynchronous APIs), and fixed long-standing issues in the distributed objects (DO) interface, thread-timer-UI synchronization, symbolic links, memory leaks, and file system case-insensitivity issues.
- Developed a DO protocol allowing a number of clients (status menu, application, etc.) to communicate with the backup and restore engine to make changes to backup plans, request restore operations to the original or alternate locations, receive information about any current backup or restore operations, and the backup history of individual files.
- Provided feedback to Apple and tested the fix for a serious memory leak within OS X that was introduced in 10.6 and was present until 10.6.6, happening every time a file was copied.
- Architected and developed portions of a product similar to iDisk and DropBox for use with local and remote WD NAS devices using REST, AppleDouble, and a customized device selection dialog using Bonjour asynchronously filling in a collection view.
- Developed a solution to automatically substitute localized text for a single non-localized NIB and automatically resize UI objects by using parameterized key-paths in bindings; a localizer can also see the key and string table associated with a UI object by viewing tooltips with the caps lock down.
- Developed a bindings solution to enable controls based on the non-zero-count of an array controller's selection array.
- Developed an icon badging solution for package folders using a folder similar to the alias arrow badge presenting further backup and retrieval information to users.
- Developed an `NSBundle` category method using a reference string to choose the numerically correct string as defined by a localizer, such as might be required when a number to be included would be 0, 1, small-many, large-many, etc., thereby reducing the lines of code to retrieve a string to only one line and the supplied number; also designed it so the number could be alternately spelled out or in numerical form. Submitted the need for this to be more intrinsic within Apple's frameworks, which was eventually introduced in OS X Mavericks.
- Developed a number of conditional category methods to work around needed solutions in earlier versions of Mac OS X (e.g., 10.5) that were made available in later ones (e.g., 10.6), such as providing easier access to the Finder's method of sorting files based on embedded numbers.

**EFI PrintSmith**, Office in Scottsdale, AZ/Telecommute from Irvine, CA – (17)

Lead Senior Software Engineer

July 2004 – March 2010

- Removed Altura cross-platform support of Mac APIs used in the Windows version in lieu of native Windows APIs.
- Added platform-specific Unicode support for drawing, editing, and printing text for Macintosh and Windows.
- Added a runtime localization utility so translators can see their changes in the product immediately.
- Utilized IBM's ICU libraries and the Unicode Consortium's LDML XML data for globalization needs.
- Refined the product build process to accommodate automated builds.
- Developed runtime UI instantiation utilities to cut down on needed code.
- Added support for Mac OS X CoreFoundation sockets to replace OpenTransport functionality.
- Added Windows XP theme drawing to match Macintosh theme drawing.
- Wrote technical documentation on personally-developed features for end-users during beta testing that was used verbatim in final documentation.

- Converted some portions of the product from a Carbon-based application to use the Cocoa frameworks once Apple dropped the intended release of 64-bit Carbon.

**LearnStar**, Dallas, TX – (16)

Software Development Consultant (Matrix Resources)

September 2002 – November 2002

- Developed a C++ framework for the Macintosh version of LearnStar's teacher application based on the Carbon API set so the application would run on Mac OS 8.6, 9.x, and 10.x so full-time employees could add new functionality. The framework featured picture-based controls in place of standard UI elements to match the Windows version, interfaces to the third-party libraries WASTE and CodeBase (also fixed bugs in each library as they were discovered), and the ability to call Mac OS X-native code available only in Mach-O from the CFM-based build.
- Set up CVS as a source code control system for the Macintosh codebase.
- Modified the existing LearnStar student application to use the Carbon API set so it would run on Mac OS X natively.

**Macromedia**, Richardson, TX – (15)

Senior Software Engineer

October 2000 – October 2001

- Originally hired into the Enabling Technologies Group to bring common software solutions to all of Macromedia's products, including product serialization and registration, internationalization, and localization mainly for the Macintosh platform, but with an eye towards cross-platform concerns with the Windows products.
- Due to the rest of the team being laid off, assisted the FreeHand team in porting legacy Macintosh code to the latest operating system version, Mac OS X, including developing new features to take advantage of the system resources and safeguarding against formerly safe coding practices that were no longer applicable.

**Netscape**, Mountain View, CA – (14)

Software Development Consultant (DesiSoft Systems)

June 2000 – September 2000

Assisted the internationalization team in implementing fringe features for the Mozilla project's Macintosh codebase, including the user-defined character set, performance optimizations, and porting a French company's character-set sniffing technology (the source code, comments, and variables were written in French).

**Piranha, Inc.**, Dallas, TX – (13)

Software Development Consultant (DesiSoft Systems)

March 2000 – June 2000

- Developed the Macintosh version of the Piranha Byte application and its related installer.
- Made changes in the cross-platform codebase that also supported Windows and Linux.
- Ported the Piranha Net web browser plug-in from Windows to Macintosh.
- Provided expertise in cross-platform QuickTime SDK issues for the Piranha Stream product.
- Demonstrated product features for prospective clients in London, England.
- Tutored a remote group of Windows developers about porting issues going between Windows and the Macintosh platform.

**Microsoft Corp., Plano, TX – (12)**

Software Development Consultant (Riccione & Associates/Volt)

October 1998 – January 2000

- Ported the MLang COM library to the Macintosh platform for use in Microsoft Word and Internet Explorer, utilizing Unicode, Windows codepages, Apple's Text Encoding Converter, and other native Mac APIs where relevant.
- Fixed bugs and added functionality with respect to text display using multiple script systems and languages.
- Internationalized the JScript and XML engines of Internet Explorer 5.0 for the Macintosh.
- Added Macintosh implementations of needed Windows APIs to simplify porting portions of Internet Explorer to the Macintosh.

**Retina Foundation of the Southwest, Dallas, TX – (11)**

Software Development Consultant (Technical Programming Services)

August 1998 – October 1998

Designed and developed a C++-based application using the PowerPlant object-oriented framework and ColorSync in order to allow testing fine-tuned color and light testing perception of eye patients.

Features of the application: the user could choose which monitor tests should be performed on using a similar UI as the Monitors control panel, the color depth to use on the chosen monitor, parsing of color calibration tables using a finite state machine syntax checker, calculation windows allowing a user to convert between color spaces of monitors, such as RGB, CIE, XYZ, CIE Yxy, and Judd normalizations of these, and the color spaces recognized the human eye, such as LMS and lsy, and a stimulus-testing C++ class that could be used by internal developers to add in a modular way new tests with little extra coding by taking care of fundamentals like preparing a subject for the coming stimulus, asking for testing parameters, displaying the stimulus, reporting feedback to disk and printer, and providing final output of the test results.

**CyberQuest, Inc., Dallas, TX – (10)**

Manager of Information Technologies

June 1997 – June 1998

Managed and directed the design, development, implementation, purchase, and installation of the Internet, e-commerce, web development, and personal workstation configurations, and technical recommendations and standards of the company, especially with respect to the flagship product, bid4it, an online exchange web site modeled on the NASDAQ stock exchange, designed to sell commercial products.

**Nortel, Richardson, TX – (9)**

Internet Consultant (MacTemps)

August 1996 – June 1997

- Assisted the manager of Nortel ICS in department staffing by interviewing professionals to develop graphics, HTML, web software, and system administration, negotiating contract rates with agencies, and orienting staff to department procedures and in initially contacting customers.
- Configured web servers using Sun workstations, Netscape server software, Oracle databases, and other needed products, such as Java and Perl runtime environments.
- Developed customized server applications and web sites according to customers needs.

**Global Software Development, Ft. Worth, TX – (8)**

Software Development Consultant (DesiSoft Systems)

April 1995 – October 1996

- Designed, developed, and assisted in the marketing of the Macintosh and Windows versions of Photobot, an Adobe Photoshop add-on that automated many tasks performed in Photoshop.
- Added customized serialization of the product along with network copy detection of the same version and serial number over the local area network.
- Developed the installer programs and scripts necessary for both platforms as well as provided a demo version.

**IBM Porting Center, Roanoke, TX – (7)**

Software Development Consultant (Decision Consultants, Inc.)

August 1994 – August 1996

- Trained and assisted IBM software engineers in Macintosh software development in order to port an IBM client/server product, Time and Place, to the Macintosh from the OS/2 and Windows platforms.
- Designed the framework of the Macintosh application.
- Developed a number of C++ classes utilizing the THINK Class Library framework to perform certain UI operations in order to reduce the amount of time needed to port the UI portion of the application.

**AMX Corporation, Dallas, TX – (6)**

Lead Macintosh Software Engineer (DesiSoft Systems)

January 1994 – January 1995

Developed a Macintosh (both 680x0 and PowerPC) client product to communicate with a Windows-based server so teachers could schedule media, control AMX hardware, and create and modify media groups.

**Aldus Corporation, Dallas, TX – (5)**

Macintosh Software Engineering Consultant (Software Dynamics)

July 1992 – January 1994

- Designed and developed platform-specific and cross-platform code in Aldus Persuasion 3.0 and Player 3.0.
- Discovered bugs in Apple's Toolbox and developed unique workarounds in Persuasion and Player.
- Developed callback functions for the Aldus FreeHand team for use with Aladdin's Stuffit InstallerMaker.
- Took ownership of the Aldus Table OLE applet included with Aldus Persuasion 3.0 from a third-party developer and modified as needed for the new release.

**Ideal Learning, Inc., Irving, TX – (4)**

Macintosh Software Developer

January 1991 – July 1992

Developed major revisions to a client/server curriculum management system called Podium, including separating out UI and file-based operations across functional boundaries so a client product could be developed that allowed teachers to perform the same operations as on the server but on a separate computer across an AppleTalk network.

**IBM Corporation**, Roanoke, TX – (3)  
Consultant (AGS Information Services)  
June 1989 – January 1991

Tested and revised various portions of the user interface of OfficeVision/VM in over ten international versions to ensure compatibility with the American version, worked with international translators, wrote and modified various utility programs to facilitate a better personal testing environment, inspected, wrote, and executed test cases, provided assistance in determining problems and solutions, attended code and test case inspection meetings, and assisted new employees in getting familiar with their duties and OV/VM.

**Earth Scientists, Ltd**, Coffeyville, KS – (2)  
Programmer  
August 1984 – January 1987

Modified and ported an existing set of software for geological production forecasts designed for the Apple II to work on a variety of personal computer systems including the Apple ///, PC-DOS, and the Macintosh.

---

## **EDUCATION**

B.S. degree in Computer Science, Applied Science Option, with emphasis in Mathematics, Southwest Baptist University; Bolivar, MO 65613. GPA in Computer Science: 3.77; Overall GPA: 3.29.