

Emily Berk
P.O. Box 370588
Montara, CA 94037
(650) 728-0376
emily@armadillosoft.com

*I believe that my extensive experience in **designing and implementing complex software systems** that interface to sophisticated hardware will allow me to serve your company well as a Systems Analyst/Software Architect.*

Technology Experience

Computer languages	C++, C, Java, HTML, JavaScript, PHP, Flash, 6809, BAL, and other assembly languages, PL/1, Fortran, Basic, Pascal, Visual Basic, others
Database languages & products	MySQL (SQL), Oracle, Sybase, Toad, MS Access
Presentation tools I use extensively	DreamWeaver (although I prefer my HTML raw), FrameMaker, Acrobat, Visio, WebWorks, FireWorks, Power Point
Source code control, bug tracking	Bugzilla, ClearCase, CVS, Perforce, PVCS, Source-Safe, Vantive
Installer scripting	InstallShield, Wise
Other products and technologies I use (This is the buzzword-compatible section)	J2EE (BEA WebLogic 5.1 & 6.x, Sun RI, other application servers), XML, SGML, Object-oriented design & programming (OOP), Structured Design, Structured Programming, CORBA -- Visibroker, MS Project, UML (Mostly TogetherSoft, some Rational), Rapid Software Development (RAD), Extreme Programming/Development, ERWin, MQSeries, variety of other legacy backends

Education

Sun University	Java Architects' Course
University of Pennsylvania, Philadelphia, PA	B.S.E., Computer Science Engineering Tau Beta Pi, Engineering Honor Society
Dickinson College, Carlisle, PA	B.A., Russian Studies

Requirements Gathering

As a systems designer/consultant, I have been creating functional specifications that some of my clients consider "perfect" for many years. To write good specifications, one must first know how to gather the requirements. I am adept at interviewing internal and external "special interest groups" to learn what's needed.

I speak efficiently with various constituencies (developers, marketing, data entry, end users, etc.) to learn:

- What the software needs to do (prioritized functionality)
- How it should do it
- How to teach developers to use it, including design of GUI and examples when necessary
- What the real budgets are for: time, money, resources and tools
- What the development team can realistically accomplish given these budgets

Some of my most interesting requirements gathering projects have included:

- **Designed and implemented UNIX-based teleconferencing server (client: Syntellect)**
End users were unskilled (primarily non-English-speaking) telephone operators. Designed Graphical User Interface (UI) to display state (muted, talking, on-hold, etc.) of thousands of callers on single screen. Allowed operators to interactively move callers between teleconferences, disconnect them, put them on hold, monitor them, etc. Involved realtime parsing of messages received from UNIX server and generation of messages to UNIX server. System was installed internationally within weeks after coding was completed. Managed team of four programmers. All code written in C++.
Hardware: Dedicated UNIX telephone switch
- **Project Leader (client: RCA Laboratories)**
In charge of software development for one of RCA Lab's (Now Intel's) DVI/ CD-ROM applications. Purpose of the application was to demonstrate the advanced graphical capabilities of a graphical processing chip. Application was cleverly disguised as a children's game. Managed ISVs (Independent Software Vendors) hired by RCA to develop software out of house. Job involved game and software design, programming, human interface research, supervision of video production, and contract negotiations, while keeping track of changes in processor's capabilities and interfaces. Code written in C.
Hardware: DVI (high-speed, real-time graphics processor)
- **Software architect (client: Zoomedia)**
Designed second-stage architecture and wrote functional specification for integrated Internet-based film production/costing application.
- **Functional specification writer (client: Salestar)**
Defined long and short-term enhancement strategy, wrote functional specification for surprisingly complex cost-comparison application for telecom industry.
- **Consulting Systems Engineer (client: Edwin Schlossberg Inc.)**
Created budget and specifications for purchase of hardware and development tools and design and production of custom game software for a multi-million dollar entertainment center.
- **Consulting Software Designer (client: Children's Television Workshop)**
Designed and implemented (in 6809) educational games and artists' tools on Radio Shack Color Computer. The artists' tools included a font editor and a bitmapped-frame animation tool. The user interface for the artists' tools was developed in close consultation with talented artists who had had limited exposure to computer-based tools.

System Architecture/Specification/Scheduling

Create functional and technical specifications and object-oriented program design for C++, C, Java, PHP, and Flash-based Internet and non-Internet APIs and applications in fields of multimedia, healthcare, teleconferencing, embedded systems, pattern recognition and videoconferencing. Often responsible for scheduling, project management and resource allocation of programming and quality assurance staff. Define requirements for and implement source code control and bug tracking systems for clients, many of whom had not ever used these tools before. Supervise teams of programmers who implement these projects; serve as chief programmer on projects. Most require real-time message-processing and/or communications running in the background. Often, projects require on-going support for legacy DOS-based back-end systems. Most interface to large multimedia databases and many to weird hardware. All present challenges in terms of user interface design, database design and response time.

Projects have included:

- **Architect/project leader www.emarker.com (client: Sony)**
Designed object-oriented software architecture that allowed two separate development teams, one in San Francisco and one in Seattle to independently develop Flash/HTML/EJB/Oracle modules that came together with minimal integration effort. **Hardware:** Data inputs included: dedicated USB device, a database import that occurred every five minutes or so, and a larger, less frequent database import.
- **Enterprise Application Technical designer (client: eStamp)**
Participated in technical design of UI for EON's eStamp's Web-based postage stamp generator, in technical design and specification of the eStamp enterprise architecture and in technical design and specification of the eStamp API. **Hardware:** Dongle on client PC held postal license and account information. System interfaced with postage printers, scales, servers at eStamp and at Postal Service.
- **Technical project management/documentation of EFI's print server API (client: EFI)**
Helped plan for release of API. Created code examples in Java and C++. **Hardware:** API interfaced with EFI's Fiery print servers.
- **Enterprise architect (client: Matthew Bender)**
Created technical specification/architecture/project plan for re-design of production process such that a huge legal database could be re-linked and stored on CD in time allotted. (C/ SGML/Sybase/Folio)
- **Real-time/Embedded system programmer (client: Moberg Medical)**
Implemented host processing software and print-out for system that monitored brain waves of anaesthesia patients. Ported 12-year-old host processing code from pre-ANSI C to ANSI C with few supporting tools. Implemented printing of brain waves from RTOS; interfaced with a separate CPU for user interface and to an HP printer's internal printing language for printing. Truly a project in which software bugs could profoundly affect the lives of users. **Hardware:** Individual processors in brain monitor (Graphics processor, DSP, PSOS/RTOS host processor), HP printer
- **Architect/programmer of image database management system (client: PictureWare)**
Project required careful process and database design (it takes time to sequence and store 57,000 analog images before they can be written to CD). Windows code written in C and MS Assembler. **Hardware:** System output image database to videotape, from which analog CDs were produced.
- **Co-editor, Hypertext/Hypermedia Handbook (published by McGraw-Hill)**
Pioneering examination of methods for structuring electronic information for ease of navigation.
- **Designed user interface to handprint (pattern) recognition software (client: Nestor)**
Wrote software to convert between graphics file formats (e.g., TIFF to PCX), compress and decompress bitmaps and files. UI design and implementation. Client code interfaced with neural net running on transputer multi-processing board. Participated in design/early development of Windows-based front end for a pattern recognition chip. Also debugged code on the transputer. Code written in MS C. **Hardware:** Transputer board; unnamed pattern recognition chip
- **Programmer/Analyst (client: Qyx/Exxon)**
Structured analysis and design of operating system of microprocessor-based typewriter/word processor.

For more information, please see:

<http://www.armadillosoft.com/emily/emResume.php?newTopic=analysis>