# RICHARD BOYD

11824 80th Avenue East

Puyallup, WA 98373-4761

253-906-6830; richard.v.boyd@comcast.net

***Like challenge of solving problems***

More Details in **Online Resume** at … <http://rem7.info/resume/>

### TOC

[.NET](#_.NET_1)

[FSHARP](#_FSHARP__)

[CLOJURE](#_FSHARP/CLOJURE)

[JAVA](#_JAVA_1)

[SMALLTALK](#_SMALLTALK)

[LISP](#_LISP/SMALLTALK_1)

[GIS](#_GIS)

[DATABASE](#_FSharp_Clojure)

[EDUCATION](#_EDUCATION)

[PUBLICATIONS](#_PUBLICATIONS)

[ADDITIONAL EXPERIENCE](#_ADDITIONAL__EXPERIENCE)

### GOALS

***Software Development or Teaching***

**Preference for,**

* **F#**, **Clojure**, and Functional Programming
* **Hybrid** Programming … (C# + F#) … (Java + Clojure)
* .NET 4.5 suite of technologies
* Web based multimedia educational projects with AI backend (Intelligent Agent/Tutor)
* GIS related projects
* Telecommute opportunities

### CAREER

**Functional Projects**

* **F#** 3.0 exploration of Financial Trading data.
* **Clojure** Web Crawler to search Amazon media info.

**.NET Projects**

* Azure Cloud development, Faro Scene 3D, Entity Framework, OData, JSONP, JavaScript, HTML5, CSS3, ISAPI, redirect calls from IIS to Tomcat, Sencha Touch Mobile Framework.
* Convert complex algorithms from Mathematica to C#.
* Microsoft project to connect .NET client to Java server via SOAP web services.
* ASP.NET app to monitor Microsoft vendor/partners.
* Silverlight/Bing kiosk for UW campus.
* Silverlight/Flex UI and Web Services for utility district.
* Embed F# within C#.

Skills: .NET 4 experience with C#, Silverlight, LINQ, WPF, TFS, WCF, TPL, Bing Maps, Expression Blend, SketchFlow, VS 2012, SQL Server 2012, SQL Server 2008 Geospatial, GIS, Windows 8 Pro, Windows Server 2012.

**Java Projects**

* Server, middleware, UI
* Servlets, JSP, Applets, Flex, Java2D.
* Real time weather application for NASA (Space Shuttle).
* Multicast IP Address to broadcast weather data.
* Custom JavaBeans for visual and non-visual components.
* FAA Prototype using ESBs.
* Animation + audio for web-based educational software.

**Misc Experience**

* Agile style development from my Smalltalk and Lisp background.
* Much of design patterns, unit testing, agile development, rapid prototyping, and short delivery cycles arose from the Smalltalk community, of which I was a member.
* Transition to Java in 1996 and C# in 2003.
* Hybrid development with Python, Ruby, and Groovy … embedded within Java.

# .NET

Dec 2012—Jun 2013

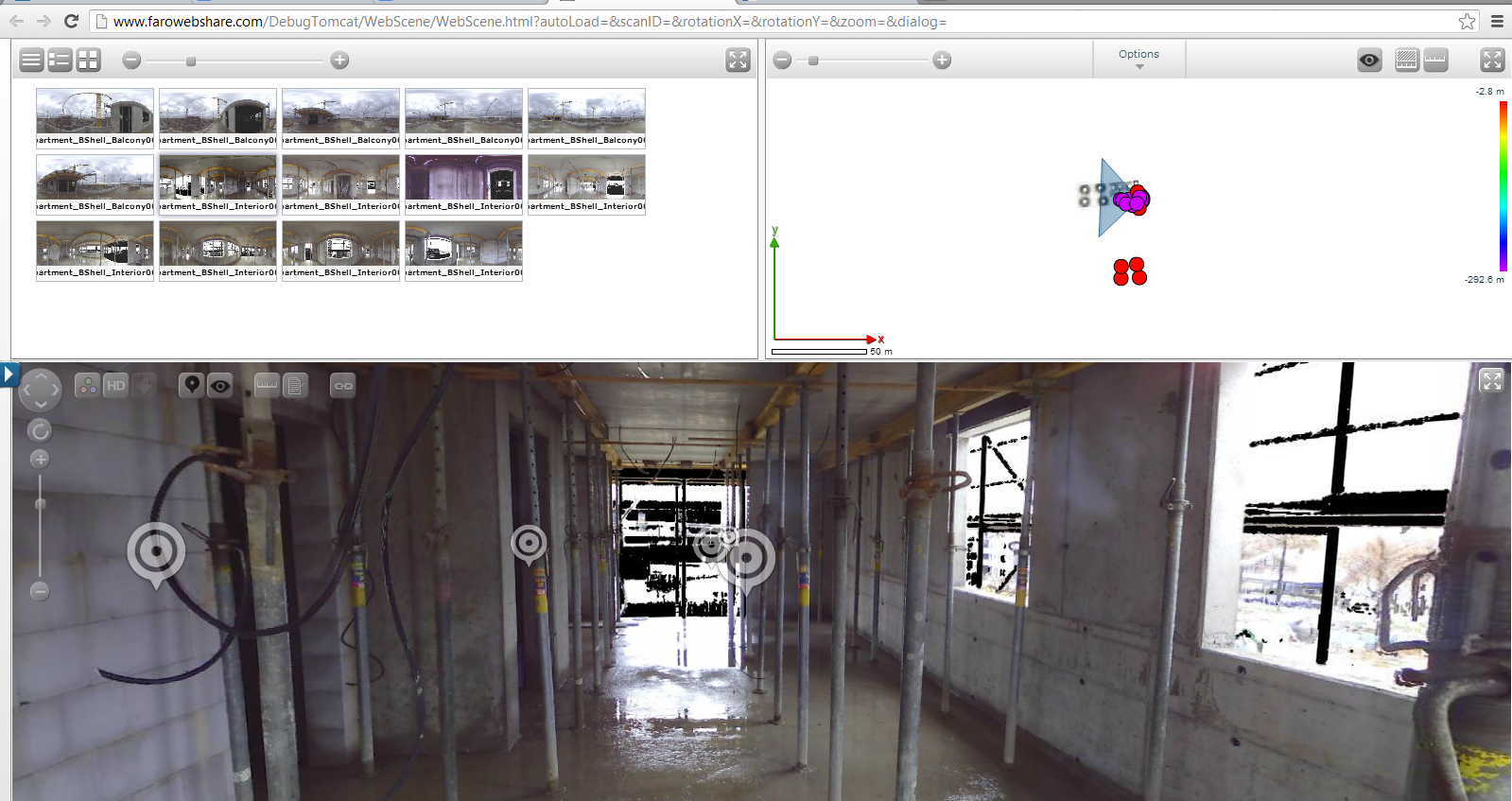
**Rich Geographic Internet Applications** *Seattle, WA*

*Remote Project*

Summary: **Convert project** from Silverlight to Mobile Framework (**Sencha**) … VS2012 MVC4, Web API, OData, JSON, JSONP, Entity Framework.

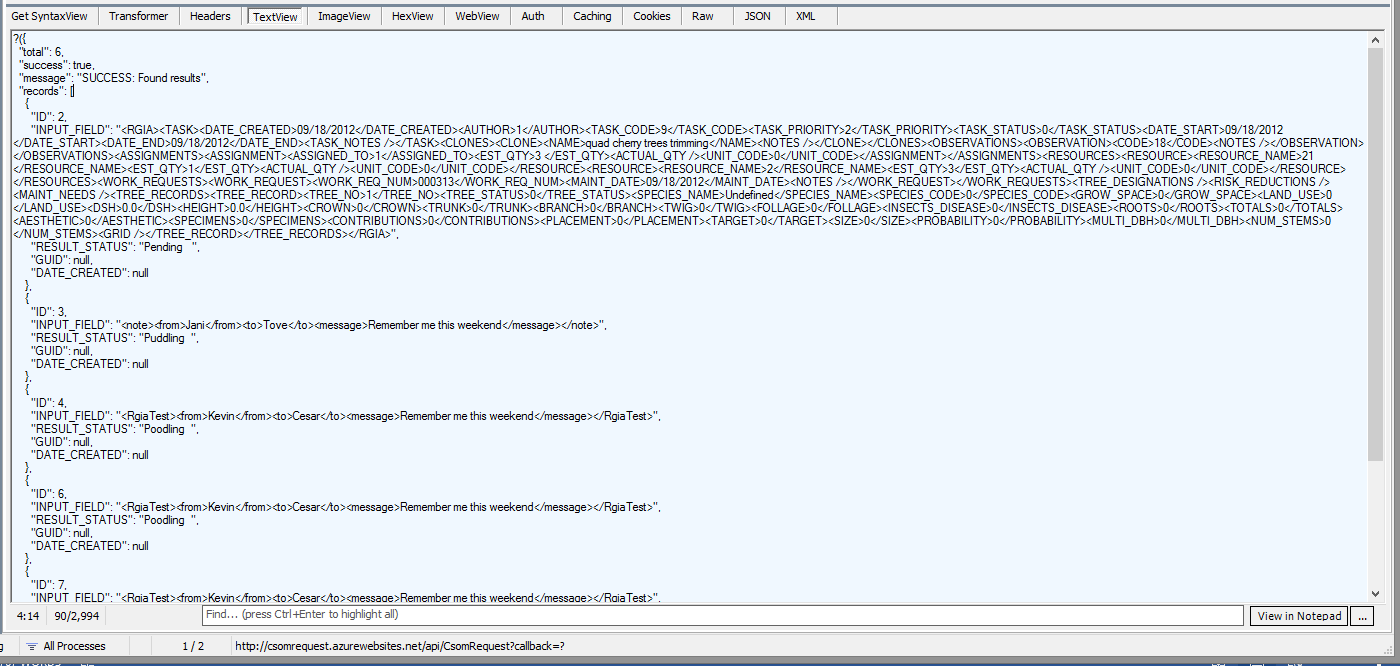
[Web Kiosk](http://www.rgia.org/) for University of Washington campus. Display assets … buildings, trees, piping, sewer, bollards, etc. Represent each asset visually and link to database for tracking, maintenance, deployment, monitoring. Allow Facilities crew to make onsite evaluation and instantly update central repository.

1. Work remotely … **RDC** (Remote Desktop Connect), **Skype**, **Join.Me**.
2. **Convert** web project from **Silverlight** to **Mobile Platform** using [Sencha Touch](http://www.sencha.com/).
3. Install, debug, configure [XAMPP](http://www.apachefriends.org/en/xampp.html) Apache distribution as per instructions for running Sencha Framework.
4. Install and Configure [Faro Scene 3D](http://www.faro.com/en-us/products/faro-software/scene/overview) software used to scan underground utility tunnels … in order to provide **3D Viewing** facility of UW Utility Tunnels similar to [Google Maps Street View](http://maps.google.com/intl/en/help/maps/streetview/#utm_campaign=en&utm_medium=van&utm_source=en-van-na-us-gns-svn).
5. **Faro** runs as **Java Servlet** on **Tomcat** … Client calls via **IIS** … so configure **Apache Tomcat Connector** **to redirect** from IIS to Tomcat.
6. Problems with **ISAPI** filter … switch to [BonCode](http://boncode.net/connector/webdocs/Tomcat_Connector.htm) which is an [IIS to Tomcat Connector](http://tomcatiis.riaforge.org/) … essentially upgrade from **C++** to **C#**. Worked better … could now redirect calls from IIS to Tomcat to Faro 3D Server.



Integrate **SQL Server 2012** database running **Windows Server 2012** on Domain #1 … server code running on **Azure** (Domain #2) … **JavaScript Client** on Domain #3.

1. Use Microsoft’s [Web API](http://www.asp.net/web-api) framework to build **REST**ful applications using **VS2012 MVC4** project.
2. **Entity Framework** to build **CRUD** services against database.
3. Initially incorporate [OData](http://www.odata.org/) Protocol to standardize REST-based CRUD operations against disparate UW campus-wide resources exposed as data services.
4. **OData** supports two formats … **XML**-based **AtomPub** and **JSON.** This was a Problem because cross domain requires **JSONP** … Solution was to add third format via **Custom JSONP Formatter**.
5. Drop OData approach when discovered Sencha only supported OData for **SAP** … so OData for anything else became a moot point.
6. Standard way Sencha Controls want data returned is wrapped with additional info. Used **Json.NET** to **Serialize** data and **JsonWriter** to wrap it … then **Deserialize** Response. Doing this made the **Queryable Attribute** meaningless … further eliminating any chance of using OData.
7. Create **Azure** website … publish VS2012 JSONP server code to Azure … test **CRUD** from **C#** client.
8. Install [Aptana Studio 3](http://www.aptana.com/) as development environment for Sencha. Build simple Sencha test app using [Ext.data.JsonP.request](http://docs.sencha.com/extjs/4.2.0/#!/api/Ext.data.JsonP-method-request). JSONP request to Azure worked OK. After confirming that Sencha was consuming JSONP data from Azure server … next step was to use [Sencha Command](http://www.sencha.com/products/sencha-cmd/download) to build framework for [Sencha MVC](http://www.sencha.com/learn/the-mvc-application-architecture/) app to test JSONP using [Sencha Data Stores](http://www.sencha.com/blog/sencha-basics-an-introduction-to-data-stores/).



Skills:C# 4.5, VS2012, Azure Cloud, SQL Server 2012, LINQ, IIS, Web Services, ESRI, GIS Spatial Data, Entity Framework, Stored Procedures, MVC4, JavaScript, HTML5, CSS3, Sencha Touch, XAMPP, Firebug, ExtJS, jQuery, Aptana Studio 3, Windows 8 Professional, Windows Server 2012, Faro Scene 3D Software, NuGet Package Mgr., Json.NET, Fiddler.

Jun 2012 – Oct 2012

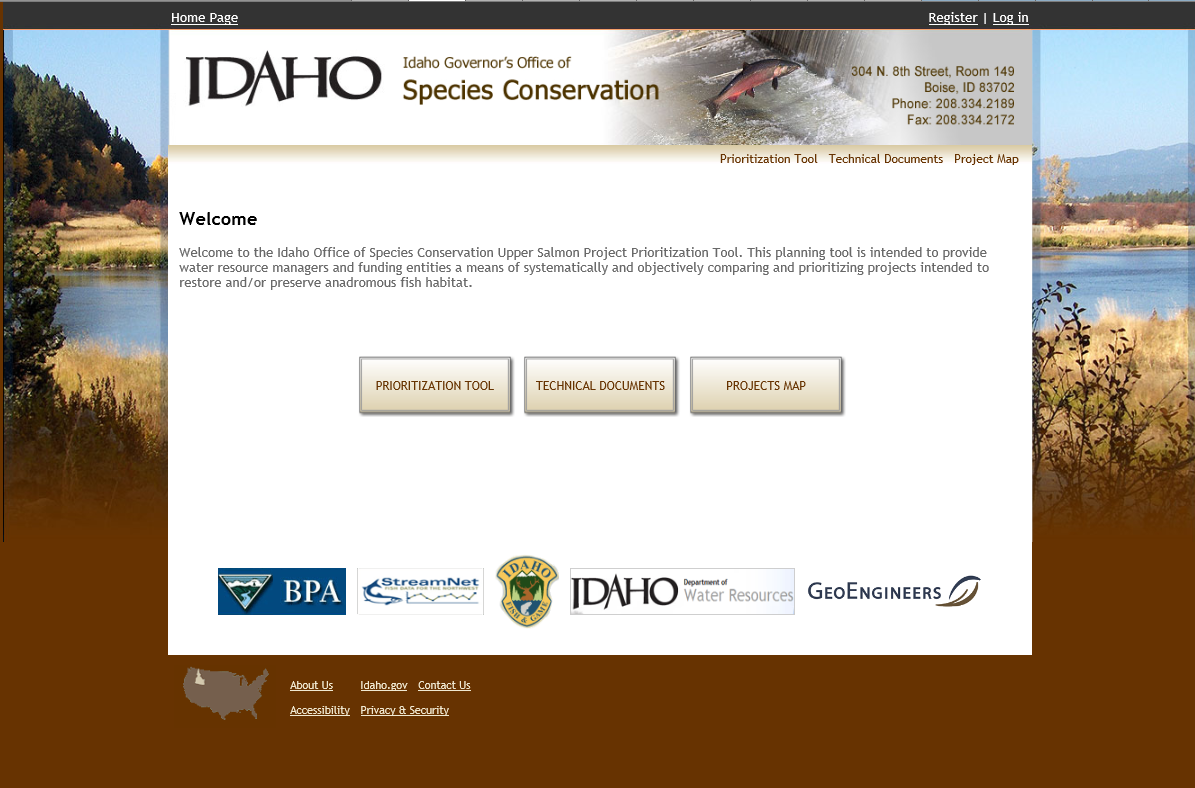
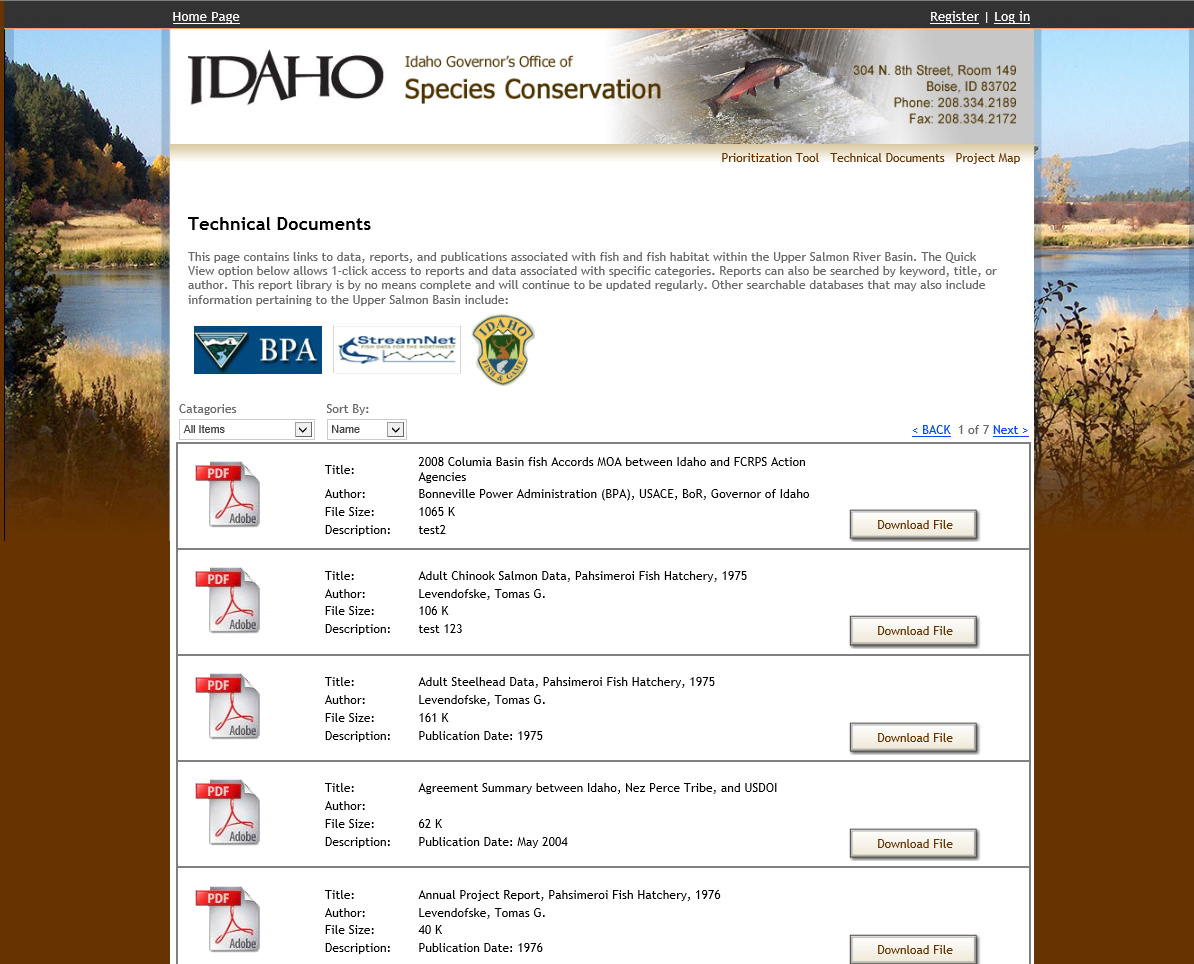
**GeoEngineers** *Tacoma, WA*

Web Applications Developer

*Remote Project*

Summary: [GeoEngineers](http://www.geoengineers.com/) specializes in environmental, ecological and geotechnical solutions for the Water and Natural Resources, Transportation, Energy, Federal and Development markets. I was hired for a *casual, on-call position* to develop desktop, web and/or mobile applications … integrate applications with databases and GIS applications … develop web services, and enterprise solutions using ESRI’s Adobe Flex API, JavaScript API, and Microsoft Silverlight API.

1. VPN in remotely for **Office of Species Conservation** (**OSC**) project, which was a planning tool to allow comparing and prioritizing fish habitat restoration for the State of Idaho in the Upper Salmon River Basin.
2. Project initially prototyped using [DotNetNuke](http://www.dotnetnuke.com/) …I explored and prototyped conversion to **Visual Studio 2010** **MVC4** and **LINQ /** **Entity Framework** for data.
3. Used **Expression Web 4** to do basic layout.
4. **User Validation** in MVC4 to control behavior depending on roles was important aspect for this project.
5. I researched and implemented all these technologies in first 3 weeks … but by that time GeoEngineers felt it was all too much new technology to “risk” on the short timeframe allowed.
6. GeoEngineers was new to MVC style, so they decided to fall back to their “comfort zone” … i.e., not use Razor view engine, go back to \*.aspx rather than \*.cshtml, write SQL rather than use Entity Framework, stick with JavaScript/HTML/CSS approach, cut-n-paste code from prior ASP.NET project to provide Role management.
7. Complete this project via “old style” technology stack.

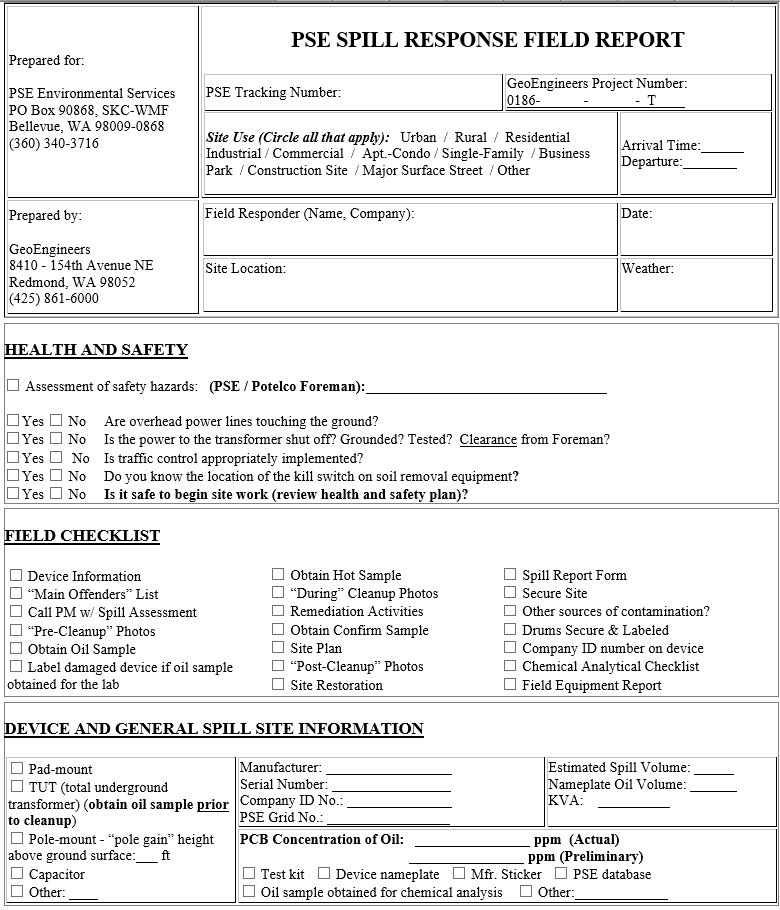
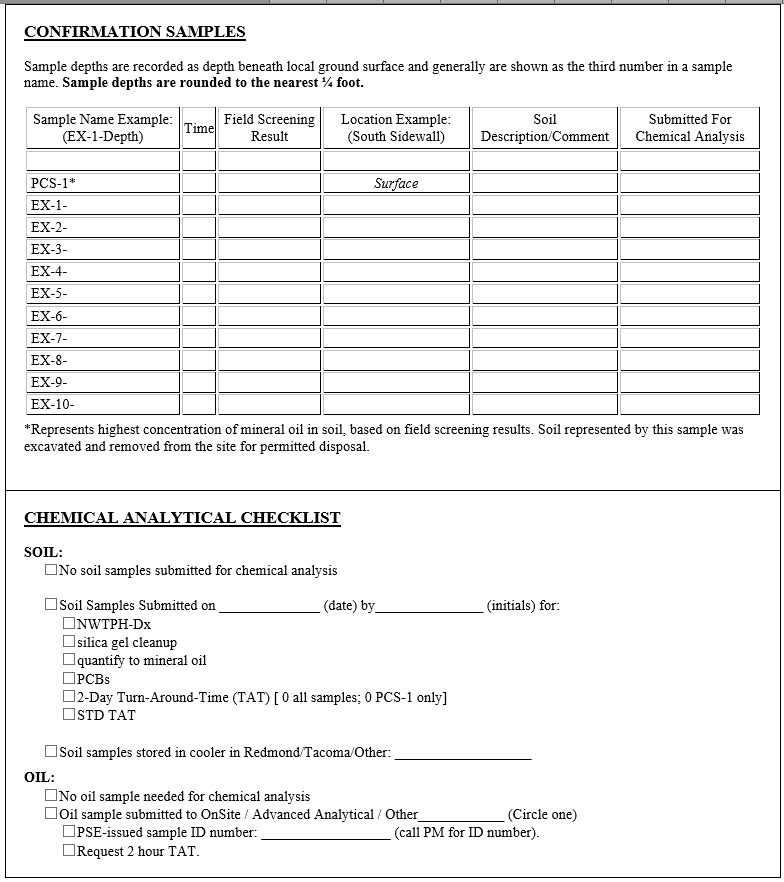
Oregon **Department of Environmental Quality** (**DEQ**) was conducting Cleanup Training Seminars focusing primarily on petroleum cleanups. I built website to announce, promote, and register for this seminar.

1. Develop code as per **Use Case** document.
2. Insert **Google Maps**, **Pushpins**, **GeoLocation**, Hover info **Popups**.
3. **Accept** user registration info.



**Puget Sound Energy** (**PSE**) project.

1. Convert **PDF** of PSE *Spill Response Field Report* to **JavaScript/HTML/CSS** online web form.
2. Initially import PDF into Word Doc … and then into **Expression Web**.
3. First Pass -- Timed myself … **11 minutes** starting from scratch … to create & publish to web … the converted forms for the entire PSE PDF doc … using Expression Web for auto conversion.
4. Looked great … but HTML was pretty “Ugly”!
5. So back to building **JavaScript/HTML/CSS** equivalent form.
6. **Layout** was biggest issue … HTML + CSS is not nearly as user friendly and precise as using **XAML**.

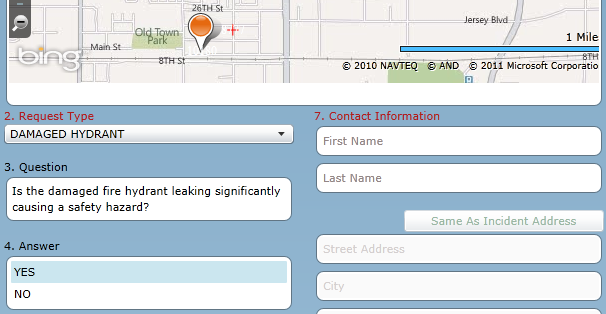
Skills:C# 4.0, VS2010, SQL Server 2008, LINQ, Expression Blend 4, IIS, Web Services, ESRI, GIS Spatial Data, Entity Framework, MVC4 Roles, XAML, JavaScript, HTML, CSS.

Mar 2011 – Nov 2011

**Weston Solutions** *Seattle, WA & Rancho Cucamonga, CA* **(via Techstaff** *Seattle, WA***)**

*Remote Project*

Summary: [Cucamonga Valley Water District](http://www.cvwdwater.com/): Needed to add a section to their website where a customer could enter a Service Request and track its Status. A Silverlight app was built that is embedded within current site. Bing Map allows user to locate request. Employed GIS to Reverse Geolocate map point/street address. Web Service backend for Request Categories, Questions, Answers, Instructions, and Comments. Confirmation Page summarizes Request. Email provides link for customer to monitor Status of request.



1. Build **Silverlight** UI for Cucamonga Valley Water District (CVWD) web based Customer Requests System that facilitates public entry and tracking of service requests.
2. Embed Silverlight app within current CVWD website.
3. Address entered in text field … **Geolocate** to translate Street Address to Longitude/Latitude and display Pushpin on **Bing Map**.
4. Pushpin dropped on Bing Map … **ReverseGeolocate** to convert Longitude/Latitude to Street Address displayed in text field.
5. Convert ESRI Shapefile to SQL Server 2008 Spatial GML format using Shape2SQL tool.
6. Extract boundary from imported ESRI Shapefile.
7. Implement algorithm, using polygon created from boundary points, to calculate if an address lies within or outside CVWD Service Area (could have done this in SQL Server 2008, but client runs 2005 which does not have Spatial features).
8. Alert user if address falls outside CVWD Service Area.
9. **Web services** to select applicable problem descriptions, present list of services, provide clarification questions, capture customer answers.
10. Validation via Silverlight ValidationSummary control and bindings Mode.
11. **Email confirmation** (via Web Service) with unique ID sent to customer so request status can be tracked.

Skills:Silverlight 4.0, C# 4.0, VS2010, SQL Server 2008 Geospatial, LINQ, Expression Blend 4, SketchFlow, Bing maps, IIS, Web Services, ESRI Shapefiles, GIS Spatial Data.

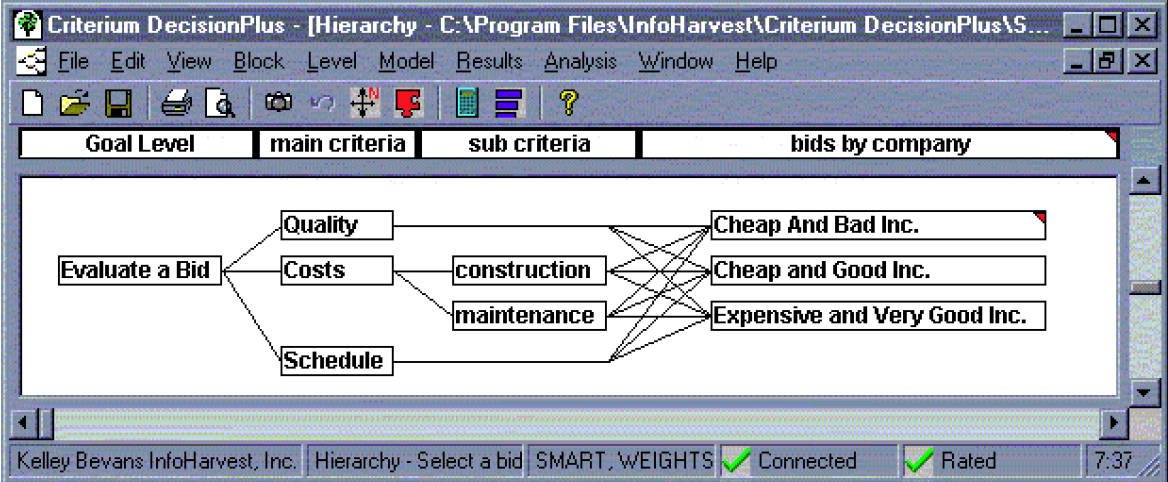
May 2011 – Dec 2011

**InfoHarvest** *Seattle, WA*

*Remote Project (part time)*

Summary: [InfoHarvest](http://www.infoharvest.com): Developed visual tool that assists making complex decisions. Current work is in conjunction with the [Redlands Institute](http://www.redlands.edu/innovation-research/7417.aspx) which conducts applied research into complex environmental and social systems using GIS and associated technologies. Core decision making technology is written in VB. Goal of this project is to update to C# and current .NET Framework. Convert WinForms to XAML. Use ObservableCollection to decouple bindings. **Problem**: Dynamically select database at runtime, but LINQ needs to do database mapping at compile time. **Solution**: ADO.NET creates DataSet … then query via *LINQ to DataSet*.

Screenshot of “decision” in VB6 version



1. Create light version of InfoHarvest’s Multi-Criteria Decision Analysis object to replace current wrapper for COM WDObj decision analysis DLL.
2. Convert **VB 6** code to **C# 4** … WinForms to XAML.
3. Decouple binding between View and Model using ObservableCollection.
4. **Problem**: Database selected **at runtime**, but LINQ needs to do mapping at compile time.
5. ***Slow* solution**: System.Diagnostics.Process to fire SqlMetal.exe in order to create \*.dbml mapping file. Similar process to create \*.cs file from \*.dbml. System.CodeDom.Compiler namespace to dynamically compile \*.cs file and return compiled assembly. Dynamically load assembly.
6. ***Fast* solution**: ADO.NET creates DataSet … then query using *LINQ to DataSet*.
7. List of tables in database from which user selects one table.
8. Populate list of column names from table.
9. Load Decision Analysis Model and use Regex to match Model columns to Database columns … allowing for white space, capitalization, etc.
10. Filter to select columns with unique entries that can be used as Foreign Keys.

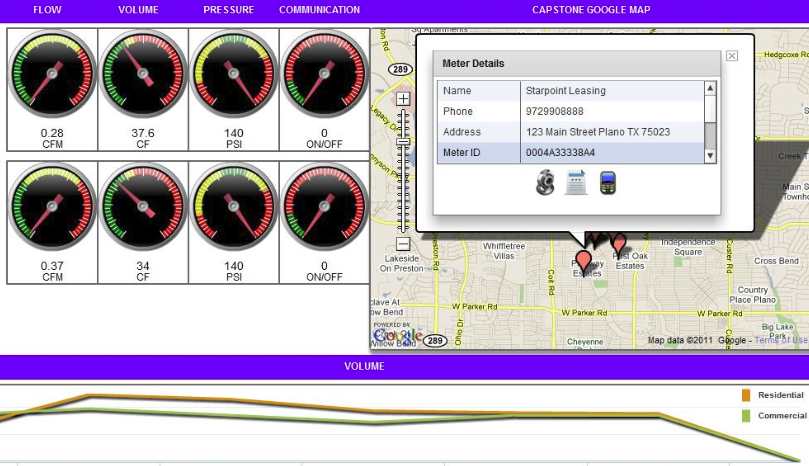
Skills:C# 4.0, VB6, VS2010, SQL Server Compact, LINQ, WPF

Jan 2011- Mar 2011

**Intellacuity**  *Dallas, TX* **(via Consultis** *Dallas, TX***)**

*Remote Project*

Summary: Create application for Capstone Metering (CM) to monitor and manage their smart water meter. Business Analytics has Adobe Flex interface, so UI built in Flex. Prototype was built in SketchFlow. Data was read from SQL Server 2008 and parsed with C#, LINQ, RegEx. Create web services (\*.asmx) in VS2010. Construct water meter widget using ActionScript. Pushpins and Info popups were inserted on Google Map. Live WebCam. UI was iPhone-like display.



## Build prototype Dashboard for real time management of water systems (residential & commercial).

1. Prototype process using **SketchFlow** (Expression Blend 4).
2. Read data from **SQL Server 2008** … parse with **LINQ**.
3. Create **Web Services** in **VS2010**.
4. **Adobe Flex 4** UI consumes **Web Services**.
5. Construct water meter widget using **ActionScript**.
6. Populate Pushpins on **Google Map** via Google Flash API.
7. PushPins can popup an InfoWindow using Flex DataGrid to display meter data … with links to Reports, live **WebCam**, and meter data displayed on face of *iPhone*-like display surface.
8. Data Forms popup to capture user input.

Skills:Adobe Flex, ActionScript, C#, VS2010, LINQ, Google Map, Expression Blend, SketchFlow, SQL Server 2008, Web Services.

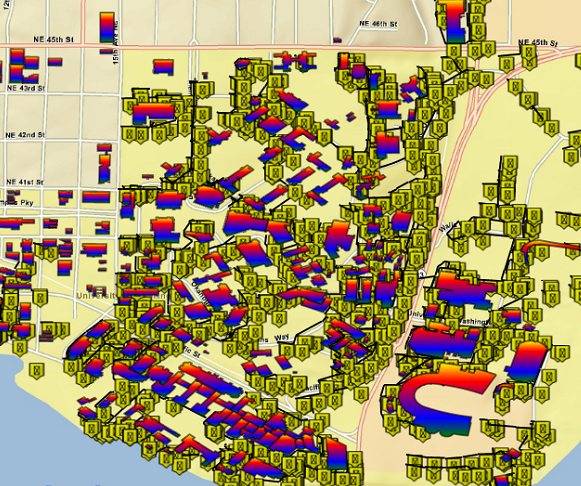
Fall 2009—Nov 2010

**Rich Geographic Internet Applications** *Seattle, WA*

*Remote Project*

## Director Research & Development

Summary: [Web Kiosk](http://www.rgia.org/) for University of Washington campus. Display assets … buildings, trees, piping, sewer, bollards, etc. Represent each asset visually and link to database for tracking, maintenance, deployment, monitoring. Allow Facilities crew to make onsite evaluation and instantly update central repository. Original project in Open Laszlo + Google Maps. Convert to Adobe Flex + ESRI Maps. Because of performance (thousands of icons), convert to Silverlight + Bing Maps.



1. [RGIA](http://www.rgia.org/) builds a rich interactive mapping application (**RIA**) to locate and plot assets and link to databases for tracking, maintenance, deployment, and monitoring. The **AssetMapper** application is targeted for the University of Washington.
2. Establish communication between multiple **Flash** instances.
3. Setup [Red5 Media Server](http://osflash.org/red5) an open source **Flash** Server.
4. Build **Flex 3** and **ActionScript 3** applications as templates for connecting to **Red5** shared objects.
5. [Concept Maps](http://cmap.ihmc.us/) as tool for capturing Architecture, Design, Agile Planning and Use Cases.
6. **Expression Blend 4** and **SketchFlow** to prototype UI.
7. Setup **IIS** as web server … and **TFS** (Team Foundation Server) as code repository.
8. Initially convert **OpenLaszlo** **+ GoogleMap** application to **Flex/Flash/ActionScript + ESRI Maps.**
9. Convert to Flex4/FlashBuilder 4.
10. Convert Flex to Silverlight + ESRI Maps because of performance and wider range of components.
11. **Flex** is single threaded … **Silverlight** is multithreaded. Use threads available with Silverlight to run processes in BackgroundWorker threads. Invoke behavior back on UI thread via *Deployment.Current.Dispatcher.*
12. [Shibboleth](http://shibboleth.internet2.edu/) single sign-on for security.
13. Modify basemap to incorporate both **ESRI** and **Bing** mapping … via Silverlight API.
14. Parse XML query results using **LINQ** and **RegEx**.
15. From parsed XML, render Points, Lines, Polygons on map layers to represent features such as trees, bollards, water/sewer/high pressure lines, buildings, parking lots.
16. Build and Consume **Web Services** via **WCF 4**.
17. Animations and Transitions via XAML.
18. **F#** to explore dynamic routing of King County bus routes.
19. **Silverlight** animations, sliding panel, video brush.
20. Switch out **Bing** basemaps, drop points on map, right mouse context menu, popup info window.

Skills: VS2010, SQL Server 2008, Silverlight 4, XAML, LINQ, RegEx, C#, F#, WCF, Web Services, WPF Expression Blend 4, SketchFlow, TFS, ESRI Maps, Bing Maps, Shibboleth … Java, Flex 3/4, Flash, ActionScript, Red5 Media Server, Tomcat.

Sept 2003--Feb2004

**InfoHarvest** *Seattle, WA*

Summary: [WOPR](http://www.daylightdecisions.com/ddweb/WOPR_Report.pdf): Bureau of Land Mgmt developed plan to manage Oregon lands. Build website to gather feedback/comments from residents so that [InfoHarvest](http://www.infoharvest.com/) Decision Analysis could be used on results to enable fine-tuning of plan. **MSFT:** Build website for Microsoft, using Decision Analysis techniques, to monitor Microsoft vendor/partners in order to determine needs/wants/issues.

1. Build website to explain Draft Environmental Impact Statement (DEIS) for BLM's Western Oregon Plan Revision. Website tracked user identity and feedback via Interactive Map Browser and comments users made in online documents. **C#**, **SQL Server 2005**, **ASP.NET**.
2. Upgrade **ASP HTML** controls to **Server** (and **HTML**) **controls** for **ASP.NET 2003**.
3. *Refactor* client side script to **C# code-behind**.
4. Develop web application for Microsoft to monitor vendor/partners. Data access for controls via **ADO.NET** **DataSets**. Automate **SQL Server database** table generation using **DataSet** that was populated, via **Data Adapter**, from **Excel spreadsheet** description.
5. Emulate database tables via creation of **XML files**.
6. Manipulate and extract data via **DataViews**.
7. Author **Stored Procedures** and use them to create **DataAdapters**.
8. **SQL** **Table** creation.
9. **XSLT** **Transform** of **XML** files for input to Excel.
10. Capture *database script changes* via VS.NET **Database project**.
11. **Hyperlink** and **Template** columns.
12. Web **Tree Control**. **Deploy** projects to Web.
13. **Mentor** developers on upgrade path using **Visual Studio .NET 2003**.

Skills: C#, ASP.NET, VS2003, SQL Server 2005.

July 2003--Sept 2003

**WIRB** *Olympia, WA*

Summary: [WIRB](http://www.wirb.com/): Research and Review of FDA related products, processes, regulations. Ensure compliance. WIRB was early adopter of beta VB.NET to automate process. New system of 500,000 lines of VB.NET code suffered from severe memory leaks that WIRB had worked with Microsoft for about six months, and were unable to solve. I debugged the system and found 163 classes that were creating instances which were never garbage collected. Fix resulted in considerable improvement in performance.

1. [WIRB](http://www.wirb.com/) does independent evaluations of tests involving drugs, vaccines, and medical devices for health care industry.
2. WIRB was early adopter of **beta VB.NET** to convert massive manual process for tracking and coordinating documents and processes. New system had about **500,000** lines of code and **36 GB** of data in **SQL Server** … but had **memory leak performance problems.**
3. WIRB worked first half of 2003 with Microsoft to solve this, but was unsuccessful. Microsoft **PerfMon** had problems with **.NET 1.0 Framework.**  **DevPartner Studio** memory profiler not available until Fall 2003.
4. Solution: Upgrade BIOS in obsolete PCs and setup automated test framework using **WinBatch** + **.NET Memory Profiler** to debug application in both 1.0 and 1.1 .NET Frameworks.
5. Found **163 classes** that created thousands of instances of custom controls not being released (Garbage Collected) because of way application was using cloning.
6. Debug and fix other tasks using **Visual Studio .NET 2003**.

Skills: VB.NET, VS2003, SQL Server, PerfMon, Memory Profiler

Apr2003--Jun2003

**ScienceOps** *Lynnwood, WA*

(contract via **Saltmine** *Seattle, WA*)

Summary: [ScienceOps](http://www.scienceops.com/): PhDs only consulting firm that develops algorithms and tackles complex problems that no one else in the area would be able to solve. This particular project involved calculating the optimal layout for long strips of 2 inch wide electrical floor heating elements … in order to cover the most floor area with the least waste. The client was off-shoring this work to teams in India, to solve manually … which resulted usually in a 24 hour delay. Client wanted computerized system to solve problem “instantly” while customer was still in showroom. Solution was solved and prototyped in Mathematica, then converted to C# and put on the web.

1. Convert **complex algorithms** from **Mathematica** **4.2** to **C#**.
2. Develop optimization algorithms in Mathematica.
3. Mathematica lacked the rich data structures of C#, thus often becoming a roadblock … requiring rethinking the process.
4. Analyze performance problems in **unmanaged** **.NET C++** code that had originally been developed in Mathematica.
5. Data representation in **.NET XML**.
6. Extensive debugging of **.NET C#** and **C++** code.
7. Evangelize need for infusing software engineering principles into client's development process.
8. Proofread, analyze, and write technical reports.
9. Begin development of technical websites using **WebMathematica**.

Skills: .NET, C#, C++, XML, Mathematica.

2003

**InfoHarvest** (part-time) *Seattle, WA*

Summary: [Criterium DecisionPlus](http://www.infoharvest.com/ihroot/infoharv/products.asp#CDP30) software applies structured methodology to decision making. Originally written in older VB, it needed upgrade to new .NET Framework. Developed partial upgrade of key components … enough to get system functional.

1. Partial upgrade of [InfoHarvest's](http://www.infoharvest.com/) Decision Analysis application from **VB6/ASP** to **VB.NET/ASP.NET**.
2. Develop multi-tier system including **ASP.NET WebServices** and **ASP.NET Application**, using **MS SQL Server** database as backend.
3. Upgrade existing VB6 modules to VB.NET.
4. Convert **ADO** to **ADO.NET**.
5. Ensure safe execution of single-threaded **VB6 COM** components within **.NET**.

Skills: C#, VB.NET, ASP.NET, ADO.NET, VB6, COM, Web Services, SQL Server.

Nov2002--Jan2003

**Mircosoft** *Redmond, WA*

*(contract via* ***Saltmine*** *Seattle, WA*)

Summary: [Microsoft](http://www.microsoft.com/) had client who wanted to link their Java environment to a .NET client. Several developers were contacted about this project, but decided the timeframe was too short. I accepted the “challenge” and brought in a PhD friend who helped me build a .NET client that used SOAP to communicate with a Java server running the standard Java Pet Store demo. Prototype system was built running on RS6000 using AIX and DB2. Much coding was done in Notepad while browsing W3C standards. Prototype was finished on time, and was a real challenge.

1. Develop prototype for Microsoft to demonstrate **WebService** used by **.NET** **Client** to communicate with **Java Server**.
2. **SOAP** to send **XML** messages from **Win2000 .NET Client**, via **HTTP**, to **IBM WebSphere EJB Application Server** running **AIX** on **RS6000,** which queried **DB2** **Database** using **JDBC** ... pass result back to.NET Client.
3. Initial prototyping done using **Redhat Linux 7.2** on **WinX Intel** box, then ported to IBM's **AIX** **Unix** on the RS6000.
4. **Session** and **Entity Beans** developed on server side.
5. **Visual Basic.NET** and **ASP.NET** employed on client side.
6. **JSP** used for testing **WebService**.
7. Considerable **debugging** because of Application Server **Deployment** bug.

Skills: VB.NET, Java, JDBC, JSP, Web Services, SOAP, XML, HTTP, WebSphere, AIX Unix, RS6000, DB2, Redhat Linux, Session/Entity Beans.

Jan2001--Mar2002

**Advanced Biometrics** *Tacoma, WA*

*Mgr R&D / Architect / Developer*

Summary: [Advanced Biometrics](http://www.ereleases.com/pr/advanced-biometrics-employees-file-suit-against-company-in-federal-court-1032): Patent for using infrared to read subsurface blood vessels in palm of hand. Project was essentially a subsurface fingerprint of the hand. Think of those hand scans in science fiction movies. Between time I was hired, and time I started, the technical people left the company. I suddenly became in charge of technology. I built team to develop algorithms for pattern recognition of combination of surface features of palm along with subsurface blood vessels. The prototype was built using Microsoft Visual C++. By that time, poor management of company finances resulted in lawsuits. Company was bought out and it moved to another state.

1. Build and manage new software team for biometric identification of hand using infrared and pattern recognition.
2. Lead development of proprietary pattern recognition algorithms and lead team development process using modified **eXtreme Programming** methodology.
3. Architect new application framework.
4. Debug undocumented code for **C/C++** embedded application using **Microsoft Visual C++,** **Metrowerks Codewarrior**, and **Understand For C++.**
5. Explore using **Python**.

Skills: C/C++, Python, Biometrics, XP.

# FSHARP/CLOJURE

Mar 2012 – Present

Startup Project – off-hours

**Summary**: Financial Trading application to discover hidden patterns and cycles within the financial markets through use of mathematical relationships (Fibonacci & Lucas Numbers, etc.), Gann Theory, Elliott Waves, Genetic Algorithms, and Parallel Processing. Use **F# 3.0** for analysis, **F# Type Provider** for data and services, **FSharpChart** for display of results. **TPL** for parallel processing.

**Skills**:F# 3.0, F# Type Providers, FSharpChart, .NET 4.0, TPL, LINQ, SQL Server Compact, Genetic Algorithms, Functional Programming, Financial Algorithms.

Feb 2012 – Mar 2012

Short Startup Project

**Summary**: Build Web Crawler to search Amazon for media titles/info. Amazon API was too restrictive for client usage … so build tool to dynamically call searches based on movie titles read from **MySQL** database. Code in **Clojure** using **regex** for filtering. **GNU Wget** for retrieving files. **Git** for version control. **Leiningen** for automation and build. **IntelliJ** + **La Clojure** as IDE. **Korma** is Clojure DSL for working with MySQL. **Hiccup** for representing HTML in Clojure. **Enlive**, using CSS-like selectors, as extraction and transformation library for HTML and XML docs written in Clojure. **Fiddler2** and **Firebug** for debugging. Half of project time was spent getting back up-to-speed in my Lisp from 1980’s! … and setting up environment for open source tools.

**Skills**:Clojure, regex, MySQL, Wget, GIT, Leiningen, IntelliJ, La Clojure, Korma, Hiccup, Enlive, Fiddler2, Firebug.

# JAVA

Mar2009--Aug2009

**General Dynamics Information Technology** *Ft Lewis, WA*

Summary: [Telehealth & Technology](http://t2health.org/) (T2) researches and develops technology for Psychological Health and Traumatic Brain Injury ([PHTBI](http://cdmrp.army.mil/phtbi/)) across the Department of Defense, for military personnel and their families. Think of it as WebMD for the DoD. I was hired as a Sr. Java Developer to update the technology and mentor Jr. Developers. First project was to build application to browse and coordinate multiple research databases that supported the DoD mandated *Psychological Health and Traumatic Brain Injury* program. Scarce documentation, so mock UI in Adobe Flex as vehicle to communicate with domain expert on East Coast. While waiting for Oracle database, export Microsoft Access database to XML file to act as web pseudo-database. Populate Flex AdvancedDataGrid via HTTPService that accessed online XML file. SWFObject to embed Flex with HTML pages. Second project was Online Web Locator that could be embedded in HTML page to allow search of Oracle database for info about DoD healthcare websites. Use [soapUI](http://sourceforge.net/projects/soapui) to test and debug SOAP connection. Build prototype based on [TED Metaphor](http://www.ted.com/). Use Flex transition effects. Third project was to build Web Service front-end to research database project as alternative to what had been done in Struts 2 and HTML. Capture Technology Overview (Java, .NET, Agile Methodology) as series of 10 mind maps. Begin work on a T2 (Telehealth & Technology) wide tutorial website to educate psychologists and developers in latest technologies. Mentor personnel on Java, Flex, and Agile Methodology. Explore alternative approach using VS2008, C#, Silverlight, and [Telerik](http://www.telerik.com/) commercial controls … (much superior!).

Summary: [Telehealth & Technology](http://t2health.org/) (T2): Research and develop technology for Psychological Health and Traumatic Brain Injury ([PHTBI](http://cdmrp.army.mil/phtbi/)) across the Department of Defense, for military personnel and families. Think of it as WebMD for the DoD. I was hired as a Sr. Java Developer to update the technology and mentor Jr. Developers. **PHTBI:** Application to browse/coordinate multiple research databases that supported the DoD mandated PHTBI. Mock UI in Adobe Flex as vehicle to communicate with domain expert on East Coast. Export Microsoft Access database to XML file that acted as web pseudo-database until Oracle was ready … accessed via Flex DataGrid + HTTPService. SWFObject to embed Flex. **OWL:** Flex app embedded in HTML page to search Oracle database for info about DoD healthcare websites. [soapUI](http://sourceforge.net/projects/soapui) to test/debug SOAP connections. **Struts 2:** Prototype based on [TED Metaphor](http://www.ted.com/) using Flex transition effects. Flex UI and Web Services as alternative to Struts 2 approach. **Presentation:** Technology Overview as series of [Mind Maps](http://www.xmind.net/). **Education:** Begin T2 website to educate psychologists and developers in latest technologies. **.NET:** Explore PHTBI alternative approach using VS2008, C#, Silverlight, and [Telerik](http://www.telerik.com/) commercial controls … (*much* superior).

1. [Telehealth & Technology](http://t2health.org/) (T2) researches and develops technology for Psychological Health and Traumatic Brain Injury across the Department of Defense, for military personnel and their families.
2. Capture and communicate analysis/brainstorming via [XMind](http://www.xmind.net/) **Mind Mapping**.
3. Convert Microsoft **Access** database prototype to online version using Adobe **Flex**, Java, and **Oracle**.
4. Documentation was scarce, so **mock** UI in Flex as vehicle to communicate with domain expert. Export Access database tables to **XML** … place files on **Tomcat** server … use Flex **HTTPService** to link temporary server data (XML files) with Flex **AdvancedDataGrid** component and other widgets. Domain expert, on East Coast, was able to login to my server and run working application. This provided working screens to glean feedback and insight.
5. Layout **Flex** screens that paralleled appearance of HTML versions … using Flex layout managers, gradients, styles.
6. **SWFObject** to embed Flex within HTML pages.
7. Explore alternative approach using **Visual Studio 2008**, **C#**, **Silverlight**, and [Telerik](http://www.telerik.com/) commercial Silverlight controls.
8. **Mentor** personnel on Java, Flex, object oriented, Agile Methodology.
9. Project OWL: Online Web Locator. Build Flex application that could be embedded in HTML page … allows user to select/enter options, and query Oracle database for matching DoD healthcare websites. Display results in Flex DataGrid. Build **Web Service** as middleware between Flex and Java/Oracle. Use [soapUI](http://sourceforge.net/projects/soapui) to test and debug SOAP connection.
10. Build prototype of OWL based on [TED Metaphor](http://www.ted.com/). Use Flex transition effects to **animate** opening of sub-panels.
11. Begin work on building **Web Service** front-end to research database project that was being done only in Struts 2 and HTML. Use [enunciate](http://enunciate.codehaus.org/) for building web services and auto-generating ActionScript code that would be consumed by Flex components. Connect Flex UI via Web Services to Java/Struts 2/Oracle … to provide an alternative approach to the older technology the team was following. Older approach required complete page refresh from server for even the smallest edit. The Struts validations were not working because of the addition of REST, whereas validation is one of Flex's strengths. The Flex results grid allowed multi-column sorting, whereas the HTML version was more limited in features. The team was having problems with how HTML version would display on IE6, 7, 8 and Firefox and Safari. Using Flex version eliminates these and many other problems. A far superior approach … and more user friendly UI.
12. Capture **Technology Overview** (Java, .NET, Agile Methodology) as series of 10 mind maps. Present information to T2 mgmt. Suggest ways to improve their s/w development.
13. Begin work on a T2 (Telehealth & Technology) wide tutorial website to educate psychologists and developers in latest technologies. Began architecture that would use Flash animation, mind maps, interactivity, Q/A, quizzes, demo applications, code, dashboards to provide interactive 24/7 access.

Skills: Java, Adobe Flex, ActionScript, Oracle, Struts 2, Web Services, XML, Tomcat, VS2008, C#, Silverlight.

Jun2008--Feb2009

**Topia Technology** *Tacoma, WA*

Summary: [Topia](http://www.topiatechnology.com/): Developed software mobile objects that can handle network tasks. **FAA:** Prototype to demonstrate Topia mobile objects in ServiceMix ESB to handle monitoring and maintenance. Allow dynamic re-configuration of remote system similar to OSGi. Build producer module and deployment document. Target was aviation weather for cross-country flights. Build admin tool to deploy Topia mobile objects. Write code to deploy filters to ESB. Dynamically move filters in accordance with traffic load in order to reduce network bandwidth. **NetCDF:** Prototype and parse NetCDF weather data. Prototype code to create arbitrary number of HTTP clients so that metrics could be taken. **Graphics:** Prototype Adobe Flex application to demonstrate online, interactive graphics as option to Topia’s current Java 2D tool. Use Adobe BlazeDS to push events over HTTP from Flex client to ESB. **Plat Map:** Debug *buggy* Plat Map application that had been outsourced … unravel logic, fix bugs, refactor code.

1. [Topia Technology](http://www.topiatechnology.com/) leverages power of software mobility to handle complex network requirements.
2. Prototype system for FAA to demonstrate interoperability of Topia's Kolona [Mobile Objects](http://www.topiatechnology.com/flash/Introduction_MOT.html) with the **ServiceMix/Iona Fuse Enterprise Service Bus**.
3. Goal of prototype was to demonstrate monitoring and maintenance strategies within an **ESB** service container environment using Topia's mobile object technology.
4. Apache **ActiveMQ** used for asynchronous messaging.
5. Data exchanged between heterogeneous systems.
6. **JMX** instrumentation attached at various points in message flow.
7. System management enterprise integration patterns.
8. Mobile object technology layered on top of standard monitoring and maintenance strategies.
9. Mobile objects allowed dynamic re-configuration of a remote system (similar to **OSGi**).
10. Build producer module, debug, test, author deployment document.
11. Target environment was **CIWS** and **ITWS** weather for FAA.
12. Topia's Kolona Engine was added as a plugin to Iona Fuse via **JBI**.
13. **Microburst**, **Gust**, and **EchoTop** weather simulated and fed to endpoints in ServiceMix ESB.
14. Build admin tool that deployed Topia Mobile Objects to ESB in order to monitor traffic.
15. Wrote code for Topia **MDCI** (remote components) which deployed filters to ESB.
16. Based on traffic load, filters were dynamically moved from the first ESB and over to ESB closer to the weather source … reducing network bandwidth.
17. Prototype code to read and parse **NetCDF** weather data (**XML** and **binary**).
18. Prototype code to create arbitrary number of **HTTP clients** using the **JBI** compliant **HTTP/SOAP** binding component named **servicemix-http**. This allowed the ESB system to be loaded so that metrics could be taken.
19. Build **JTable** console application for FAA demo … remote deployment and execution of Java Mobile Objects in Open Source ESB environment.
20. Popup window for selected cell to display XML pipeline configuration file.
21. Dynamic ability to add new table rows as XML configuration files were added.
22. Prototype **Adobe Flex 3** application to demonstrate online, interactive graphics capability that could be used as an option to to Topia's current **VizTool** (which used **Java 2D**).
23. Wrote code to connect Adobe Flex client to the Java ESB via **Java Message Service (JMS)** using Adobe's **BlazeDS** … which provided the “*pushing of event*s” over HTTP.
24. Debug **Flex** plat map (land surveys) application that had been outsourced, but not working. Buggy code, little documentation, poorly organized. Plat map application was a combo of **Flex**, **ActionScript**, **HTML**, **Java**, and **Struts**. Unravel logic.
25. Format and *diff* versions of **ActionScript** plat map code to gain understanding of what changes were made version-by-version. Fix bugs. Refactor code.

Skills: Java, Open Source, ESBs, Spring, ActiveMQ, JMS, JMX, ServiceMix-3/4, Iona Fuse-3/4, MacOSX, SOAP, Adobe Flex 3, BlazeDS.

Sep2007--Apr2008

**Reality Works Company** *Bellevue, WA*

Summary: [RWC](http://beschoolready.com/): Created computer based cognitive learning program to give children confidence and motivation to learn. **Status:** All development had been outsourced … I was hired as first full time software engineer. Hierarchical database from another startup was being used … which compounded problems. Development had been *ad hoc* and undocumented. **Setup:** Pull together current status into a coherent model. Add code repository and bug tracking. **Problems:** Hand crafted animation was unwieldy and often caused audio to be out-of-sync with video. Audio was being downloaded in uncompressed format. Duplicate files were being downloaded. ALL THIS had to be fixed. **Solution:** Convert audio to compressed OGG format. Automate manual animation scripting file process. Add peer-to-peer architecture to download files to local repository to reduce network traffic and increase response time. **Alternatives:** Once basic process had been cleaned up, begin evaluating alternative approaches … primarily JavaFX and Adobe Flash/Flex. Eventually convince RWC to switch to Flash/Flex.

1. Reality Works Company (RWC) created a *computer based cognitive learning program* whose goal was to give children more confidence and motivation to learn.
2. All development work had been out sourced … I was the first full time software engineer to be hired.
3. Initial development process was somewhat *ad hoc* and undocumented, so first tasks was to pull together all the pieces into a coherent description using **Enterprise Architect** to reverse engineer **UML** diagrams from Java, and [IHMC CmapTools](http://cmap.ihmc.us/) to create concept maps describing relationships among entities.
4. No code repository was being used, so install **Fedora Linux** on a spare workstation and setup [CollabNet Subversion](http://www.collab.net/) as repository.
5. Help setup **Mantis** as the bug tracking system.
6. RWC used a new database called [POD](http://www.pointofdata.com/) which was written in **C** using **Forth** as the glue language. I was the only one who knew **Forth**, so managing the database became one of my responsibilities.
7. Primary development in **Java**, with scripting/glue languages in **Perl** and [Sleep](http://sleep.dashnine.org/).
8. I was also the only one with development background in dynamic languages, so managing the **Sleep** code also became my responsibility.
9. Graphics were handled as standard **JPG** files.
10. *Animation* and audio was controlled by **Sleep** using ASCII cast and script files, with graphics handled by **Java 2D**.
11. To reduce the large amount of audio/video files/jars each student downloaded, **Peer-to-Peer** architecture was added using Sun's [JXTA](http://jxta.kenai.com/), which basically created a *distributed database* among the local student workstations.
12. Initial audio development used **AU** *non-compressed* sound format. To improve network performance, RWC decided to convert **AU** audio to compressed [OGG](http://www.vorbis.com/).
13. Integrate **OGG** audio format into existing software ... making sure that issues such as timing were not adversely affected.
14. *Preload* audio into memory to reduce delays when the files are played.
15. Wrap audio clips with **Threads** to allow a clip to be immediately stopped, paused, or looped ... with thread dying at completion of clip.
16. Optimize download of large jar files to reduce network load time by a factor of 2 to 3.
17. Use [REBOL](http://www.rebol.com/) to build utilities that automated cast, script, and jar file editing.
18. Verify Adobe's licensing of **MP3** … to allow RWC to use MP3 in their applications, via Flex, without fear of infringing on Thomson's MP3 license.
19. PROBLEM: Single script file for one long activity was 230+ pages of single space ASCII instructions ... essentially hand generated via copy-n-paste.
20. Hodge-podge of languages/technologies made code very unmanageable.
21. Compare pros/cons of Sun's [JavaFX](http://javafx.com/) vs. Adobe's [Flex 3](http://www.adobe.com/products/flex.html).
22. Prototype both languages ... **Flex** development was faster and produced better, more game-like UI experience.
23. Implement subset of a current RWC/Java Activity into Flex 3/ActionScript 3 as a demo.
24. Convince RWC to drop present, crippled approach and switch to **Flash/Flex 3**.
25. BOTTOM LINE: Numerous problems (*too many different technologies*) with current animation framework. *Example*: animation controlled by **Sleep** script was **time-based** rather than **keyframe/declarative approach** as in traditional animation.

Skills: Java, Forth, Sleep, Perl, C, Flash/Flex 3/ActionScript 3, JXTA, OGG, Mantis, Subversion, Fedora Linux, REBOL.

Mar2004--Sep2007

**Coastal Environmental** *Seattle, WA*

Summary: [Coastal](http://www.coastalenvironmental.com/): Designs professional weather sensors for aviation, naval, military, hazmat, and Antarctic. **Air Force Bombing Range:** Convert hybrid system (VB4, C, Assembly, Scheme, Tcl) to Java. By end of Feb 2004, project was in great danger … only three weeks remained until delivery. I joined Coastal on 01Mar2004 … started mentoring & development. Work almost 24/7 to successfully make 17Mar2004 deadline. **Space Shuttle:** May 2004 delivery of real time Java weather application to NASA. **Swedish Military:** Embedded h/w, new weather algorithms written in Java, complete Java UI, database, distributed access, voice. **Canada:** Remote, autonomous weather stations to provide nationwide weather for Canada. **Tools:** CommAPI 2.0 for serial port data streaming, NIO, Multicast IP Addressing, MySQL, SVN, Bugzilla, JProfiler, OpenSSH, PuTTY, Log4J, JavaBeans, XStream, ANTLR, METAR weather. **Re-Architecture:** In first month I suggested a complete re-architecture of system into Java was needed. Coastal decided to continue on path using [s/w tool](http://diamondedge.com/) to automatically convert VB4 code to Java. Result was costly in terms of poor architecture, debugging, and maintenance … which caused loss of business due to reduced features and missed deadlines.

1. [Coastal Environmental](http://www.coastalenvironmental.com/) designs, manufactures, and installs professional weather stations for aviation, naval shipboard, military weather, hazmat, Antarctic.
2. Coastal licensed weather software from another company … but that company was experiencing problems converting their **VB** application to **Java**.
3. Existing software system in 2004 was hybrid of *VB4, C, Assembly, Scheme, Tcl*.
4. Coastal decided to deliver new applications in **Java** and, in Fall 2003, began first Java project (**Air Force** bombing range) ... but by the end of Feb 2004 the project was in great danger ... only three weeks remained until delivery on 17Mar2004.
5. On 01March I joined company and started mentoring & development on Air Force Java project, and we worked 24/7 to successfully make the 17March deadline.
6. In May 2004 we delivered a real time Java weather application to **NASA** for the **Space Shuttle**.
7. Next major project was delivery of aviation weather application for **Swedish Military** which included embedded hardware, new weather algorithms written in Java, complete Java UI, database, distributed access, voice.
8. A large project was system of remote, autonomous weather stations to provide nationwide weather for Canada.
9. Within first month (March 2004), I suggested complete re-architecture of application into Java was needed, but Coastal decided to continue on the path they had already started using a **VB4 to Java conversion** software from **DiamondEdge**. Their decision proved costly in terms of poor architecture, debugging, maintenance.
10. Use Sun’s **CommAPI 2.0** to access serial port data streaming in from weather sensors, parsed this data stream, wrote it to file using Java’s **NIO**, and also **multiplexed** it out on multiple serial ports to other monitoring stations.
11. Setup **Multicast IP Address** using **MulticastSocket** so weather data could be broadcast via Intranet. This allowed capture of weather data that could be played back for **regression testing**.
12. No database or source code repository was being used. I led effort to use **MySQL** and assisted in setting up **SVN** as the source code repository.
13. Setup **Bugzilla** for bug tracking, **JProfiler** for memory profiling, and **Enterprise Architect** for **UML** diagrams.
14. Setup and install **Win2000** and **WinXP** **System** **Services** to auto run Java applications, secure **SSH** network connectivity to **Debian Linux** platform using **OpenSSH**, **Pageant Agent** and the **PuTTY** SSH client.
15. Build custom **JavaBeans** for both visual and non-visual weather components … example: custom built graphical components for compass wind direction and lightning graphs.
16. UI development using **JFormDesigner** and **Jigloo.**
17. Java charts using **JCCKit**. **Web Services** using **Axis.**
18. Exploration and spikes using **Groovy.**
19. Logging via **Log4J**.
20. Java **Threads** to control multiple concurrent processes (database access, charts).
21. R**everse engineer** VB4 binary files, and wrote code to convert between VB4 and Java (*Endian* differences, varying byte lengths for data types, no documentation, **challenge of deciphering** VB structures from sparse binary files).
22. Use **XStream** to marshal and unmarshal **XML** weather data to and from Java objects.
23. Convert **METAR** (**MET**eorological **A**viation weather **R**eport) to **BUFR** (**B**inary **U**niversal **F**orm for **R**epresentation of meteorological data).
24. Setup **VMware** to run **Ubuntu** **Linux** on **WinXP** in order to explore Unix-only versions of BUFR parsers.
25. Converted **PDF** BUFR table to **ASCII** text format using **Groovy**.
26. Begin building **ANTLR** grammar to parse METAR.

Problems:Coastal resisted using **Ant** and **JUnit** testing ... a decision which greatly hindered projects ... as did their decision to use the above mentioned VB4 to Java conversion utility … and their decision not to re-architect the code base.

Skills: Java, VB4, C#, Scheme, Tcl, C, Groovy, Web Services, XStream, COM, NIO, JavaBeans, MulticastSocket, MySQL, SVN, Bugzilla, JProfiler, UML, Debian Linux, Ubuntu Linux, VMWare, OpenSSH, PuTTY.

Jun2000--Jan2001

**ProductSight Corp** *Bellevue, WA*

Summary: [ProductSight](http://findarticles.com/p/articles/mi_m0EIN/is_2003_July_16/ai_105495205/): eCommerce startup to accelerate decision flow for OEM and suppliers of custom parts. **Agile:** Help establish XP (*eXtreme Programming*) team. Participate in planning, architecture, analysis, design, testing. Unit testing and pair programming. **PLM:** Build Product Lifecycle Mgmt tool to track and index product data scattered across multiple systems. Web app provides secure access, workflow, and sourcing. Manage change orders and communicate with ERP systems. **Tools:** IBM VisualAge for Java, Oracle, TopLink ORM, CVS, JSP, Servlets, Tomcat.

1. [ProductSight](http://findarticles.com/p/articles/mi_m0EIN/is_2003_July_16/ai_105495205/) developed collaborative eCommerce application to accelerate decision flow for OEM and supplier companies to engineer products and custom parts.
2. As member of **Java XP (eXtreme Programming) team**, participate in planning, architecture, project management, analysis, design, coding, and testing.
3. Control scope, features, quality, and deadlines via XP.
4. **Unit testing** and **pair programming**.
5. **Oracle** database via **TopLink** object-to-relational mapping.
6. UI was **HTML** generated by **JSP** and **Servlets** using custom state machine architecture to handle the stateless **HTTP** environment. Versioning via **IBM VAJ Repository** and **CVS**.

Skills: Java, XP, Unit Testing, Pair Programming, Oracle, TopLink ORM, HTML, CVS, VisualAge for Java, JDBC, Tomcat, JRun.

Jan1999--Jun2000

**Appworx Corp** *Bellevue, WA*

Summary: [AppWorx](http://www.uc4.com/about-us/news-events-room/history/appworx-merges-with-uc4-software.html): Developed job scheduling and IT process automation targeted at Oracle databases. **4GL:**  Existing app was written in a proprietary 4GL language. Task was to port 4GL to Java, which at that moment was JDK 1.0. **Java:** Architect conversion process and evaluate best technologies. Select architecture incorporating Applets, RMI, JDBC … and still-to-be-released Java 2.0. Design object model. Java Swing to provide richer UI than provided by HTML. Design patterns. Pluggable architecture via Java Interfaces**. Borland:** Current Java environments did not support Oracle Stored Procedures, therefore became early adopter and tester for Java 2 version of Borland’s JBuilder.

1. [AppWorx](http://www.uc4.com/about-us/news-events-room/history/appworx-merges-with-uc4-software.html) developed job scheduling and IT process automation targeted at Oracle database.
2. Port enterprise process automation, used by major corps for scheduling and services, from proprietary **4GL** to **Java JDK 1.2.2, 1.3Rc3**.
3. Architect conversion process.
4. Evaluate best technologies and best network architectures.
5. Design underlying object model.
6. Select architecture incorporating **Applets**, **RMI**, **JDBC**, **Java2**.
7. UI redesign to incorporate **Swing** features such as trees, tabbed panels, drag-n-drop, Java 2D.
8. **Design patterns** such as Facade, Abstract Factory, Singleton.
9. **Java interfaces** to provide pluggable architecture.
10. **JUnit** testing.
11. Debug early **Borland JBuilder** Java2 code in order to implement **Oracle 8 Stored Procedures**.

Skills: Java1/2, Proprietary 4GL, Applets, RMI, JDBC, Swing, JUnit, Oracle, Design Patterns.

# SMALLTALK

Jun1998--Dec1998

**Washington Mutual Bank** *Seattle, WA*

(contract via **Cordada Group** *Bellevue, WA*)

Summary: [WaMu](http://en.wikipedia.org/wiki/Washington_Mutual): Developed [Smalltalk](http://www.cincomsmalltalk.com/main/) loan application that was deployed to all their branches. Task was primarily maintenance of existing system. **Visual Smalltalk:** Interface developed using Digitalk’s PARTS Workbench which allowed layout of components via visual (event) links to build components which can be nested. This allowed very easy and rapid development of visual interfaces for small systems. It also meant it made a *complex mess* for larger systems. WaMu’s loan system was of the latter category. It was deployed to all branches and literally took 20 minutes to boot. **Upgrade:** Main task of six month contract was to maintain, debug, and upgrade existing system. **Java:** Because of sluggish Smalltalk system, I suggested WaMu explore conversion to Java. Suggestion was ignored.

1. Upgrade development for [Washington Mutual Bank’s](http://en.wikipedia.org/wiki/Washington_Mutual) **Visual Smalltalk-to-DB2** client/server loan application.
2. Maintain and debug PARTS Workbench code used to build visual interface.
3. Suggest conversion to Java because of 20 minute load time of large Smalltalk image.

Skills: Smalltalk, OS/2, DB2.

Oct1996--Mar1998

**Tellus WebWorks** *Seattle, WA*

*Founding Partner/Smalltalk Developer*

Summary: **Tellus WebWorks:** Startup aimed at online classified ads. **Classifieds:** Develop *agent* to assist in automobile search. Newspaper classified ads list specific cars. Salesman on car lot will suggest alternatives. Ex: Customer wants 2007 Green Chevrolet. None on car lot, but salesman might suggest 2008 Blue Ford. Similar year, color, make, price, functionality. Online app was aimed at latter approach. Target was media outlets looking for alternatives/supplements to existing newspaper classified ads. **Fuzzy Logic:** Use fuzzy logic to build *meta-models* of cars. Customer enters info for specific car … which is mapped to *meta-model* … which, in turn, maps to specific, available vehicles. Fuzzy code written as C++ DLL and embedded within Smalltalk of GemStone database. This provided “intelligent queries”. **Tools:** Visual Works Smalltalk, GemStone object oriented database (also Smalltalk), C++, CGI, HTML, Unix, Windows NT.

1. Integrate client-side **Java** with both server-side **Visual Works Smalltalk** and **GemStone 5.0** object oriented database, for internet online classified ads.
2. Embed **fuzzy logic** **C++ DLL** within **GemStone** OO database to handle ***fuzzy* database queries** which provided intelligent query system allowing users to specify fuzzy concepts when querying a database, such as, … “*a recent economy family car with low mileage that is near my home.”*
3. **Convert** protions of C code to Java. Prototype **VisualWorks Smalltalk** server running under **Windows NT** which processed **CGI** requests via a **socket connection** from **Unix** client.
4. User interacted with **HTML** pages, but complex processing handled by **Smalltalk** server.

Skills: Java, Smalltalk, GemStone OODB, C/C++, HTML, CGI, Fuzzy Logic.

Apr1997--May1997

**Kangaroo** *Seattle, WA*

*OO Architect/Analyst/Smalltalk Developer*

Summary: [Kangaroo](http://brianlivingston.com/windowmanager/archive/cgi-bin/new/livingst/970915bl.htm): Startup with patented push technology for updating software via internet. Architect framework to implement patent. Analyze design alternatives … and Java vs Smalltalk implementation. Co-author white papers. Build relationships with vendors. Build prototype.

1. Architect framework to implement patented [push technology](http://brianlivingston.com/windowmanager/archive/cgi-bin/new/livingst/970915bl.htm) product for updating software via internet.
2. Establish and justify design alternatives.
3. Analyze strengths and weaknesses of **Java vs. Smalltalk** implementation.
4. Co-author white papers on topics: overview, architecture, problem definition, specifications, features, use cases, database, graphical scenarios, productivity tools, object oriented perspectives.
5. Build relationships with consultants and vendors.
6. Help correlate technical and business goals.
7. Build Prototype.

Skills:Smalltalk, Java, GemStone, Objectivity, WinNT.

Mar1995--Jun1996

**CAP Gemini America** *Seattle, WA*

*Principal Consultant*

Summary: [CAP Gemini](http://www.capgemini.com/): Both developers left in middle of *City of Seattle* project. I was hired take over, continue development, and deliver project. **LCS:** Labor Collection and billing System was online app to gather daily hours, tasks, comments, etc. for City of Seattle employees. **Status:** Decipher existing code using Kent Beck’s *Object Explorer*, build new development team, mentor personnel, and develop *Brown Bag Seminars* to educate client personnel. Deliver on time. **Ameron:** Develop engineering and estimation prototype for pipe configuration system. **Nike:** Convert spreadsheet sales data to web version that provided online access for about 20 European countries/locations. Prototype system in Smalltalk. Implement in C. Parse input files and insert HTML tags. Generate over 300 HTML output files. Evaluate potential of Java. **Evangelize:** Promote and educate about advanced technologies with CAP Gemini … endeavor to build vendor alliances.

1. Take over and deliver OO Labor Collection and billing System (LCS) for the City of Seattle, using ParcPlace **VisualWorks** **Smalltalk** accessing **Oracle** database.
2. Mentor apprentice personnel in **Smalltalk** and **Object Oriented Development**.
3. Develop *Brown Bag Seminars* to educate client about technical issues.
4. Develop engineering and estimation prototype for Ameron Corp. pipe configuration system, using ParcPlace **VisualWorks** **Smalltalk** accessing **Oracle** database.
5. Prototype system for **Nike** to convert spreadsheet info to **intranet** that provided online access to all their European sales data. Prototype with Digitalk **Visual Smalltalk**, and then implement in **C**. Parse input files and insert **HTML** tags to convert text data into web pages. Generate over 300 **HTML** output files.
6. Evaluate potential of **Java**.
7. Promote and educate within CAP Gemini regarding **advanced technologies** and endeavor to build alliances with major technology vendors.

Skills:Smalltalk, C/C++, Java, HTML Oracle.

Feb1990--Aug1994

**U.S. Army Information Systems Command** *Ft. Huachuca, AZ*

*Advanced Technologist*

Summary: [USAISC](http://www.huachuca.army.mil/site/Visitor/index.asp): On staff at Ft. Huachuca, AZ. Author papers and perform analysis to support Army Information Systems. **Status:** Integrate multiple technologies into client/server system that provided graphical UI to status of Army communications worldwide. Perform knowledge engineering, analysis, design, coding, testing, documentation, and training. Emails stripped and parsed. Expert System to perform analysis and interpretation. Results emailed to Army personnel who then viewed via [Asymetrix Multimedia Toolbook](http://www.economicsnetwork.ac.uk/cheer/ch10_1/ch101p36.htm) application. Instant access to visual data, rather than two week delay for manual analysis. Awarded IEEE Engineer of Year for this work. **Manpower:** Prototype to access, display, and edit Army manpower database in Windows 3.1 as alternative to UNIX version which had experienced problems. [KnowledgePro](http://www.highbeam.com/doc/1G1-8286479.html) programming environment combined with expert system to build data rules as opposed to procedural UNIX version. SQL calls via DDE. **Author:** Convert *Army Long Range Planning* document to hypermedia electronic format using KnowledgePro. Generate dynamic links rather than static HTML links. Author 300+ page Tech/Standards volume for *Army Information System Architecture* series.

1. Integrate object oriented, expert system, multimedia, communication, and database software applications in a client/server configuration to provide intelligent graphical user interface to Army global communications status.
2. Perform analysis, design, coding, testing, implementation, documentation, and training.
3. Strip and parse incoming emails using **Visual Smalltalk**, inference via **DDE** connect to **Eclipse++** expert system, persistence via **DLL** connect to **dBase** database, email results to Army personnel.
4. Use [Asymetrix Multimedia Toolbook](http://www.economicsnetwork.ac.uk/cheer/ch10_1/ch101p36.htm) to create viewer for emailed files. Awarded 1992 Ft. Huachuca **IEEE Engineer of Year** for this work.
5. Result was instant access to visual representation of data, rather than two week delay (which manually did analysis to produce hardcopy).
6. Rapidly prototype application to access, display, and edit Army manpower database under **Windows 3.1** as alternative to **UNIX** version which had been experiencing numerous problems.
7. Used [KnowledgePro](http://www.highbeam.com/doc/1G1-8286479.html), which is a Microsoft Windows programming environment combined with an expert system, to build data rule input templates as opposed to procedural approach under **UNIX** version.
8. **SQL** calls to database via **DDE**.
9. Convert hardcopy of *Army Long Range Planning* document to **hypermedia** electronic format using **KnowledgePro for Windows**.
10. Generate dynamic hyperlinks via **KnowledgePro expert system**, rather than static **HTML** links.
11. Author 300+ page Tech/Standards volume for *Army Information System Architecture* series.

Skills: Smalltalk, DDE, DLL, dBase, Asymetrix Toolbook, Windows 3.1, Unix, KnowledgePro, SQL, HTML.

# LISP/SMALLTALK

Jan1987--Jan1990

**Naval Undersea Warfare Engineering Station** *Keyport, WA*

*Advanced Technologist*

Summary: [NUWES](http://www.navsea.navy.mil/nuwc/keyport/default.aspx): **M.O.R.E:** Take over development of *Missing Ordnance Recovery Expert*, which was expert system to emulate activity of Navy Range Officer as he searched for torpedoes that went off-range during ocean tests off Canadian Coast. Consider depth, tides, drift, expanding search area, search vehicles available, previous searches, knowledge of ocean currents and sea floor. Soviet submarines in area … so Navy had to find torpedo before *they* did. [Symbolics 3640 Lisp Machine](http://en.wikipedia.org/wiki/Symbolics) and [OPS5](http://en.wikipedia.org/wiki/OPS5) production rule system. Access test range interface buffer for real-time data acquisition. Results plotted graphically … ships, helicopters, airplanes, torpedoes. **Datanet8:** Perform all analysis, planning, coding, knowledge acquisition, and documentation for [Smalltalk](http://en.wikipedia.org/wiki/Visual_Smalltalk_Enterprise) system to configure and optimize hundreds of computers, terminals, printers, etc. that were inputs to Datanet8 front-end of Honeywell mainframe. Wrote 5 volumes of documentation. Presented paper at HLSUA conference … won award as best presenter (Zenith laptop).

1. Take over development of [Navy](http://www.navsea.navy.mil/nuwc/keyport/default.aspx) **expert system** (M.O.R.E. - *Missing Ordnance Recovery Expert*) to locate torpedoes that go off-range during ocean tests.
2. Integrate functional ([Symbolics](http://en.wikipedia.org/wiki/Symbolics) **ZetaLisp**), production rule ([OPS5](http://en.wikipedia.org/wiki/OPS5)), and object oriented programming (**Lisp** **Flavors** OO Extension).
3. Access test range interface buffer for real-time data acquisition and plotted results graphically (as ships, helicopters, airplanes, torpedoes) on a **Symbolics 3640 Lisp Machine** running the **Genera** operating system.
4. Perform all project planning, knowledge acquisition, and [Digitalk Smalltalk](http://en.wikipedia.org/wiki/Visual_Smalltalk_Enterprise) object oriented development for Navy system to configure and optimize hundreds of computers, terminals, printers, etc., that were inputs to the **Datanet8** front-end of a **Honeywell** **mainframe**.
5. Presented paper at HLSUA Forum 49 Conference in 1989 and **won Best Presenter Award** (Zenith laptop).

Skills: Lisp, Smalltalk, OPS5, Symbolics 3640 Lisp Machine, Genera, Honeywell Mainframe.

# GIS

**Summary**:

[Cucamonga Valley Water District](http://www.cvwdwater.com/) (CVWD) is a retail water provider serving 185,000 customers within a 47 square mile area. A Silverlight app was built using **Bing** Map to allow users to locate requests. GIS to Reverse Geolocate map point/street Address. Convert **ESRI** Shapefile to SQL Server 2008 Spatial GML format using Shape2SQL tool. Implement algorithm, using polygon created from boundary points, to calculate if an address lies within or outside CVWD Service Area.

For Redlands Institute, work on decision analysis UI to interface complex environmental systems with **GIS**.

Create app for Capstone Metering to monitor and manage smart water meter. Create Pushpins and Info Popups on **Google** Map via Google Flash API. Live WebCam.

Build rich interactive mapping application for University of Washington campus to locate and plot assets and link to databases for tracking, maintenance, deployment, and monitoring. Convert OpenLaszlo + **GoogleMap** … to Flex/Flash/ActionScript + **ESRI** Maps. Convert Flex … to Silverlight + ESRI Maps because of performance (Flex single threaded vs. Silverlight multi-threaded). Modify **Basemap** to incorporate both ESRI and Bing mapping. From parsed XML, render **Points**, **Lines**, **Polygons** on **Map Layers** to represent features such as trees, bollards, water/sewer/high pressure lines, buildings, parking lots.

**Skills**: Bing Maps, ESRI Maps, ESRI Shapefiles, SQL Server 2008 Spatial, Shape2SQL, GIS Spatial Data, Google Maps.

# DATABASE

Exposure to Object Oriented Databases ***before*** Relational Databases.

## Object Oriented

[GemStone](http://www.smalltalk.org/versions/Gemstone.html) (Smalltalk)

[Objectivity](http://www.objectivity.com/)

Tensegrity

[POET](http://en.wikipedia.org/wiki/Object_database)

[Versant](http://en.wikipedia.org/wiki/Object_database)

## Relational

SQL Server (most experience)

MySQL

Oracle

DB2

PostgreSQL

## Spatial

SQL Server 2008 Spatial

## Reading About

[Hadoop](http://hadoop.apache.org/)

[Mongo](http://www.mongodb.org/)

[NoSQL](http://nosql-database.org/)

## Current

[LINQ](http://msdn.microsoft.com/library/bb308959.aspx) (closest experience to using OODB)

[Entity Framework](http://msdn.microsoft.com/en-us/library/bb399572.aspx)

# EDUCATION

University of Arizona - Tucson, AZ

1991-1993

**PhD Geophysical Engineering -- 3.9 GPA**

*Research to automate subsurface image interpretation for Martian probe. Simulation of simple human decision making process.*

*Developed client/server,* ***distributed processing system*** *between two Sun SPARCstations to interpret ground penetrating radar signatures.* ***Object oriented*** *and* ***rule based*** *processing on SPARC1 -* ***neural network*** *pattern recognition on SPARC2. Embedded* ***C*** *primitives within* ***Smalltalk.***

*Converted neural network pattern recognition to* ***fuzzy theory****. Fine tune via* ***Genetic Algorithms****.*

University of Arizona - Tucson, AZ

1982-1986

**M.S. Electrical and Computer Engineering -- 3.9 GPA**

*Artificial Intelligence Plan Generation using Prolog*

*Minor in AI*

Trinity Evangelical Divinity School - Deerfield, IL

1971-1974

**M.Div. Theology – 3.5 GPA**

*Greek, Hebrew, Theology*

Faith Evangelical Lutheran Seminary – Tacoma, WA

1969-1970

*Greek, Theology*

University of Washington - Seattle, WA

1960-1964

**B.S. Aeronautical Engineering**

### HONORS

Tau Beta Pi (national engineering honor society)

Golden Key National Honor Society

National Dean's List

IEEE Engineer of the Year Award (USAISC, Ft. Huachuca, AZ)

Best Presenter at Honeywell HLSUA Conference

### SHORT COURSES

**C++** [1989] Dr. Ira Pohl at University of California Santa Cruz.

**Parallel Distributed Processing** [1989] Dr. David Rumelhart at Stanford University.

### LANGUAGES

If I had to pick only one: **F#** (**Clojure** is runner-up)

Used on most projects: Java, C#, Smalltalk, Silverlight, ActionScript, Flex

Occasionally used: Lisp, C/C++, VB.NET, Python, Ruby, Groovy, JavaScript, REBOL

Light usage: Forth, Clojure, Scala

### CURRENT INTERESTS

* F#, Clojure,   Functional Programming, Hybrid Programming
* Fuzzy Logic Modeling
* Intelligent Agents
* Uncertainty
* Semantic Web
* Cloud
* Silverlight/WPF
* LINQ/PLINQ
* Entity Framework
* TPL Parallel Programming
* GIS

# PUBLICATIONS

**A Unified Analysis Method for the Calculation of the Steady-State Aeroelastic Airload Distribution** – *Boeing Document D6-23828TN*. FORTRAN simulation of airloads on Boeing SST.

**Interpreting Synthetic Ground-Penetrating Radar** – PhD dissertation submitted to *Mining and Geological Dept. at University of Arizona*. Research aimed at developing base technology for continuous profiling geophysical systems to determine “interesting” subsurface features. Hybrid development combining object oriented (Smalltalk) and procedural programming (C), neural networks, fuzzy theory, genetic algorithms. Possible applications included unmanned Mars Rover to scan subsurface Martian landscape as preliminary step to seismic imaging.

**Plan Generation and Prolog** – M.S. thesis submitted to University of Arizona Dept. of Electrical and Computer Engineering. Give introduction to concepts of plan generation, an overview of Declarative Languages, and suggestions to seek blend of best features of LISP and Prolog.

**POLOG: A Mineral Exploration Expert System** – Co-authored paper presented to *IEEE International Conference on Systems, Man & Cybernetics*. POLOG was expert system developed as a graduate student at University of Arizona to simulate a geological expert searching for a Kuroko type massive sulfide deposit. Developed using **Prolog** computer language.

**Expert Systems and Uncertainty** – Presentation to *Management Science* faculty, PhD students and other interested parties at University of Washington. Review of numerical and non-numerical methods for handling uncertainty in expert systems. Contrast with claims made against use of probability. Describe where topic fits within overall context of logic. Discussion of modified Bayesian method as implemented in expert system designed to search for Kuroko type massive sulfide deposit that was developed at the University of Arizona.

**Labor Collection System** (LCS). Co-authored LCS user manual. LCS was system developed for the City of Seattle to collect labor hours and pass these on to one or more financial systems. LCS replaced a manual process.

**Technology Articles** for CAP Gemini – Articles written to promote Emerging Technologies for the *CAP Gemini Newsletter*, Pacific Coast Unit. *Example*: An Evaluation and Comparison of Object Oriented approaches, circa 1996.

**Technology and Standards** – Authored 330 page Volume 3 (of 14) for the U.S. Army’s *Information Systems Architecture*. Provided detailed description of key technologies and standards that support implementation of the Information Systems Architecture. Focus on emerging technologies for *United States Army Information Systems Command* (**USAISC**).

**ASR 25-6 Prototype** – *Army Supplemental Regulation* required status of outages and hazardous conditions of specific types on equipment to be reported to USAISC HQ on daily basis. ASR 25-6 is one of the methods USAISC uses to gather information in order to manage Army networks and systems. Co-authored ~200 page document describing development of prototype via Smalltalk and Forward Chaining Expert System. UI via Asymetrix Toolbook. Automated two week manual process into system that dynamically “read/parsed” emails and expert system that “interpreted” parsed results.

**DATANET 8 Configurator** -- *Naval Undersea Warfare Engineering Station*. Authored 5 Volume documentation describing Smalltalk application to automate configuration process for Honeywell DPS 8 Mainframe, and handle communications interface between mainframe and all the terminals, printers, etc. attached to it. Presented as paper to **HLUSA** Forum 89 (Honeywell Mainframe User’s Group). Won Best Presenter Award (Zenith laptop).

# ADDITIONAL EXPERIENCE

2002

**AutomationIQ** (*startup*)

AutomationIQ was an short-lived startup focusing on the integration and engineering of the best of breed home networking technologies to produce a "smart" home. These technologies included X10, CEBus, 80211.x, Bluetooth ... and related technologies in order to network a home over powerlines, phonelines, cable, or wireless. Home networking embraces home theater, office, intelligent appliances, smart objects, telecommunications ... as well as subsystems which control security, heating, cooling, lighting, and energy management.

Primary focus was to analyze and architect a Java and Embedded Linux application that would interface to the X10 powerline protocol. X10 controllers send signals over existing AC wiring to receiver modules and X10 adapters connected to AC outlets in order to control simple electrical devices. This task involved the analysis and study of the basic X10 theory in order to understand the protocols so that I could use Java's binary bitwise operators to construct the byte code commands to be sent over a Serial Port to the X10 controller. This task involved reverse engineering of hardware and software.

2002

**Database Administrators** (*startup*)

Develop web crawler to dynamically search web for potential customers that could use specialized image compression software. Use **REBOL** computer language to capture URL, open port to URL, check for redirected URL, select actual URL, read webpage, parse webpage based on HTML image tags, extract image size, capture max image size found, check for min and max image thresholds. Based on results, the crawler may or may not go one level deeper and continue the search. Email results of parsing and search to Database Administrators.

1999

**LifeCycle Systems** (*startup*)

Develojp BPR tool for LifeCycle Systems to control and graphically display processes, behavior, state, rules. Used Digitalk's **Visual Smalltalk**, **GF/ST** graphics framework, **Tensegrity** OODBMS. Extensions discussed included **ExperTalk** expert system, **Model Quest** (abductive modeling … neural net/statistical), **Decision Plus** fuzzy logic, **Evolver** genetic algorithm DLL, **Raven Write** hypermedia DLL.

1998

**Financial Trading** (*startup*)

Partial port of financial trading application from **VisualWorks Smalltalk** to **Dolphin Smalltalk** so that it could be delivered as a commercial Microsoft Windows application. Modify data persistence from binary files to **Pachyderm** object oriented database. Modify graphics from 2-D Smalltalk to 3-D **DirectX**. Build separate process to monitor daily stock data files and transfer them via socket to OO database. Modify architecture to take advantage of improved behavior offered by OO database and improved graphics offered by Microsoft DirectX. Problems encountered with unstable support for Smalltalk. Database was withdrawn from market, DirectX support was limited to version 3.0, etc. Java offered a more stable platform. Begin Java port.

1998

**Financial Trading** (*consultant*)

Develop financial trading application similar to commercial products such as Wall Street Analyst or AIQ Trading Expert. Client had 2000 stocks which were monitored daily, but no existing trading application had specific features needed. Full application development from specs, analysis, design, coding, test, delivery. Two modes: [1] graphical display of stock data using object oriented graphics framework (**GF/ST**); [2] monitoring mode which tracks stocks and reports key states to log file. Read from binary or ASCII files, graphical display which allows user interaction via menus and which varies display in response to changes in financial parameters (ex. greater volume displays as wider graphic bar ticks, or as different colors), VisualWorks 3.0 **Smalltalk** development environment.

1996-1998

**Tellus WebWorks** (*startup*)

Develop proprietary Internet application requiring high volume and interactivity. **Java** client with **Smalltalk** server linked to **GemStone** 5.0 object oriented database. Embed fuzzy theory C++ DLL within GemStone OO database to handle *fuzzy* database queries. Alpha, Beta, and Release 1.x developer for *Classic Blend*, a unique Internet development environment which integrates server application in Smalltalk with client GUI in Java. Java Builder initially downloaded to client. Thereafter, Java Spec shipped to client, from which client's Java Builder would build Java Applet … thus reducing network traffic. Prototyping also using **Objectivity** 4.0 ODBMS.

**Fuzzy logic** modeling in **C**, **Java,** and **Smalltalk** of an intelligent query system which allows a user to specify fuzzy concepts when querying a database … such a questions as *… a recent economy family car with low mileage that is near my home.* Convert C code to Java implementation.

Prototype VisualWorks **Smalltalk** server running under Windows NT which processed CGI requests via a socket connection from Unix client. User interacts with HTML pages, but complex processing is handled by Smalltalk server.

circa 1997

**Lamont’s Department Store**

Debug **Borland C++** application for Lamont's IS department.

1995-1996

**Ameron Corp** (via CAP Gemini America)

Develop system for Ameron Corp. to provide engineering and estimation for large pipe configuration system using ParcPlace **VisualWorks** **Smalltalk** accessing **Oracle** database. Database interface via ParcPlace **ObjectLens**. Dealt with mismatch between n-dimensional OO language and 2-dimensional database … made worse by ObjectLens flattening the OO Class inheritance hierarchy. Experience enforced advantages of an Object Oriented Database for OO development, or at least something like the **TOPLink** database bridge instead of **ObjectLens**.

circa 1995

**Sierra-on-Line**

Called as expert witness for trial.

Analysis and evaluation, as an expert witness, of both **C++** code and proprietary game code (**Lisp**-like **C**) for multimedia entertainment applications vendor (**Sierra-on-Line**) in a lawsuit they had with another company.

1990’s

**Dylan OO Language**

Beta developer/tester for Harlequin's (http://www.harlequin.com) commercial version of the new programming language **Dylan** (***Dy***namic ***Lan***guage). The **Dylan** language is both object oriented like C++ and Java, and dynamic like Smalltalk and Lisp. **Dylan** is designed for building complex object oriented programs and for programs which may need to be changed dynamically as they are running, and yet deliver efficient applications via the techniques of procedural programming and typing.

## Aeronautical Engineer

**Rocketdyne** --Testing of Gemini Space Capsule attitude control thrust engines.

**Boeing** -- Develop computer models, run simulations … SST and 747 configurations to evaluate airloads.

**FORTRAN** on CDC 6600.

Compiled two documents,

* *A Unified Analysis Method for the Calculation of the Steady-State Aeroelastic Airload Distribution* – Boeing Document D6-23828TN
* SST wind tunnel tests