

# Paul Johnson

www.pjcj.net - paul@pjcj.net

- Technical leadership, management, QA.
- Systems analysis, architecture, design and programming.
- Design and implementation of commercial testing tools.
- Native English speaker, with some German and French.

## Interests

- The production of, and the processes which lead to high quality software.
- Fostering a culture which emphasises quality throughout the entire software process.
- I contribute to open source projects, especially Perl and CPAN modules.
- I enjoy leadership, building proficient and motivated teams and helping others learn.

## Perl

- I have been an active member of the Perl community for many years.
- Contributed numerous patches to the perl core, core modules and CPAN modules, including a core patch to reduce memory usage by up to 2.5% which was recently released.
- Various modules on CPAN.
- Presented papers at seven European Perl Conferences. Won the best paper award at the 2005 conference.

## Devel::Cover

- I am the author of Devel::Cover, the code coverage module for Perl.
- The module is used for CPANCover and as a part of the Phalanx QA project for Perl.
- Other users include UBS, Morgan Stanley, SpamAssassin and Bricolage.

## Experience

- I have a broad range of experience and a flexible attitude gained from founding and working in a startup company and also from working in large companies.
- Understanding of software development processes and ability to manage teams to ensure the production of high quality software delivered on schedule and to a budget.
- Balancing of business requirements, user expectations and technical best practices.
- Research into new technologies and techniques and investigation of whether they could be beneficial to the business. Determination of how best they should be deployed and devising of strategies for their use.
- Experienced software architect and developer.
- Devised, driven or managed the QA process in a number of companies.
- Very familiar with the technical aspects of software testing and QA having written commercial code coverage tools, test harnesses and the Perl coverage module Devel::Cover.
- Comfortable working with both technical and business teams. Used to making presentations to both groups.

## October 2001 - : UBS

*Swiss bank*

- Integration of an externally developed cash management product into the standard bank environment.
  - Architecture and design of interfaces to 14 subsystems, including Oracle, MQSeries, WebSphere, logging, packaging and monitoring.

- Implementation of these interfaces.
- Design and implementation of tools using Oracle, Perl, Catalyst, Java and Ajax.
- Design and implementation of high availability solution on backup server.
- Automated build and packaging of entire application.
- Development of automated testing.
- Creation of monitoring, reporting and support tools.
- Functional and monitoring improvements have lead to a decrease in production downtime of over 90%.
- The working language is German, and I am able to effectively perform my work although I cannot claim to be fluent or even particularly proficient in German.

Previously:

- Responsibility for integration and production testing of a large project unifying credit systems and involving numerous internal groups and external companies.
  - Ensuring that software was fit to go into production.
  - Coordinating the activities of the various QA teams. Providing support and advice as required.
  - Making process improvement recommendations to the development and production teams and assisting in their implementation.
  - Setting up automated testing suites using Perl and commercial packages.
  - Introducing monitoring and reporting tools for the various applications under test.
  - Ensuring that installation processes were complete and accurate.
- The introduction of automated testing, monitoring and reporting tools, coupled with process improvements, lead to an increase in the availability of the testing systems from an estimated 75% to over 95%.

### **July 2000 - August 2001 : Olsen & Associates**

*Research into and supply of high frequency financial data*

- Formalised the QA process, updated and extended the tests and standardised and documented the entire test system.
- Worked closely with the quality engineers to ensure the integrity of the data and the correctness of the software.
- Designed, implemented, maintained and extended Perl and C++ libraries and wrote interfaces between them.

### **May 2000 - June 2000 : contracted to Ordnance Survey**

*Britain's national mapping agency*

- Debugged and extended the suite of Perl programs that was the backend to the National GPS Network.
- Put in place a testing system. Analysed and reported upon the entire project.

### **1992 - 2000 : TransEDA Ltd**

*Provider of code coverage tools for hardware design languages*

- I was a founder of this company, which had a successful IPO on the London Stock Exchange. TransEDA was the first company to bring code coverage analysis to the EDA market.
- Responsibility for all aspects of product development, including management, planning, specification, design, implementation, testing, documentation and maintenance.
- Worked closely with customers, field engineers and support staff, managed sub-contractors and conducted recruitment interviews.
- Designed and implemented large sub-systems of various EDA products, including:
  - Code coverage tools
  - A VHDL front end to synthesis tools
  - Synthesis algorithms
  - A technology mapper
- These were all OO designs written in C++ and Perl on Unix.
- Created QA strategy and infrastructure and built or selected the tools necessary to ensure high quality products. Provided training and support in this area as required.
- Designed, implemented, and maintained the build, test and licence generation systems. These were written entirely in Perl and worked across multiple operating systems.

- Responsible for the introduction of Perl into the company, first for internal programs and then for products, for which it became the major development language. As this happened I conducted many internal training sessions for the company engineers.
- Designed and implemented GUIs in C++/X/Motif and Perl/Tk.
- Ported software products between various versions of Unix, including Solaris, SunOS, HP/UX, AIX, Ultrix and Linux, and to other Operating Systems including VMS, DOS and Windows.
- Wrote user manuals for early products. Proof read later manuals and promotional material.

## July 2004 - November 2004 : contracted to Elixent Ltd

### *Semiconductor IP targeting Reconfigurable Algorithm Processors*

- Created algorithms mapping high level operations to configurations of the processing array.
- Implemented algorithms in a proprietary HDL, specifically a modified booth multiplier algorithm.
- Wrote tests, documentation and various utilities.

## Other Experience

- A web site (<http://www.kw.com/>) for Keller Williams Realty. I was responsible for the dynamic parts of the site including the parts which allow individual agents to manage their own sites. The site runs using Perl, mod\_perl, Template Toolkit and Class::DBI with an Oracle backend. The site currently supports over 60,000 agents and their individual web sites.
- A web site (now closed down) providing on-line quotes for CD replication and allowing all of the company administration to be performed via a web interface. I designed and created the site using Perl, apache and mod\_perl on the server, and dynamically generated JavaScript and DHTML on the client.
- A program to help improve the reading and spelling skills of dyslexic children. It is written in Perl/Tk and uses sonic and visual clues to present phrases to children, which they then type in. It runs on Linux and Windows and is currently being used in a very successful teaching programme. My paper on this topic won the best paper award at the European Perl Conference in Braga in 2005.

## 1990 - 1992 : Roke Manor Research

### *Research into CAD tools for VLSI design*

- Responsible for the development and implementation of logic synthesis algorithms, and for the development and optimisation of low-level modules to facilitate this.

## Qualifications

- **M.Eng.** (Software Engineering) Class II(1) Hons. Imperial College of Science and Technology, London.
- **3 A levels** (Computer Science, Mathematics and Physics), and **11 O levels** including English Language.

## Technical

- OO design and programming in Perl, C++ and other languages, the use of debuggers, memory checkers, profilers, code management tools, shells, Unix utilities, build systems and other development tools.
- Proficient in many dialects of Unix, and familiar with a number of other Operating Systems.
- CGI, Apache, mod\_perl, JavaScript, Dojo, jQuery, Ajax, CSS and other web technologies.
- Always keen to learn new technologies and improve my skills.