JOHN BOKMA

Senior Perl Developer located in the Netherlands
Email: john@castleamber.com
Skype: john-bokma

GitHub: https://github.com/john-bokma/
Homepage: http://castleamber.com/
Personal Site: http://johnbokma.com/
Last Updated: March 26, 2023

SUMMARY

I am a freelance Senior Perl Developer with over 26 years' experience, including exposure to web scraping, data wrangling, Git, MySQL, NGINX, Apache HTTP Server, Python, XSLT, XML, RelaxNG, HTML, and CSS.

I'm an active proponent of Modern Perl. I like writing technical documentation and unit tests; both have saved my customers and I a lot of time over the years. I prefer to reuse tested code as much as possible, hence I often start a project with researching available solutions on CPAN.

My personal development projects consist of several in-house tools, including a static website and blog generator using XML as input format. Currently, I am working on a new version using Markdown for input instead. I also wrote a small static microblog generator, which is available on GitHub in both a Perl and a Python version.

SKILLS

Perl

Web Scraping Over the years many projects I've been working on involved crawling, caching, parsing, and storing the desired result in a database. I like the challenges such projects offer, like making the web crawlers robust and be able to deal with the intricacies of the Internet and its data. Regarding the latter, this is why I prefer strict parsing of data, with many checks and extensive logging using Log::Log4perl and good test coverage. This way changes to the HTML of scraped web pages, which often occurs in my experience, are detected early and collecting invalid data or missing additional data might be avoided.

Parsing Besides parsing HTML I've also experience with parsing XML, several custom formats, and domain-specific languages (DSLs).

Data Wrangling Another task that often is assigned to me is the conversion of data from one format to another; including data cleaning and verification. For example, modifying the text output of a legacy application in such a way that the new output is suitable to be printed on redesigned labels. Often the input format has to be reverse engineered and documented; a challenge I like.

Testing I prefer to use extensive testing of the custom modules and scripts I write. My favorite module is Test::Most because the module provides the most commonly used testing functions with a single line, avoiding a lot of boiler-plate.

Systems Administration

My main Perl development environment is currently Ubuntu 19.04 Disco Dingo running in a virtual machine on MacOS. I also use two different VPS providers for hosting my websites. I do the administration of each of those systems, e.g. updates, firewall rules, and other security related tasks, installing and configuring software e.g. Apache HTTP server, NGINX, Postfix, Dovecot, OpenSSH, and MySQL.

Documentation

I like to write technical documentation. I prefer to use markup languages like Org Mode (Emacs), Markdown, and LaTeX. The use of plain text makes version control and searching from the command line easily. Plain text also makes it very easy to generate code from the documentation and vice versa. Where required, PDF versions can be generated using tools like Emacs, Pandoc, and pdflatex.

Remote Work

I exclusively work remotely. Over the past years, I have worked with customers in Japan, USA, The Netherlands, and Canada. Working from home provides me with a productive environment with a minimal number of interruptions. I use the Internet to stay in touch with peers and my craft. I prefer to communicate using email. I have experience with encrypted email using GnuPG, SSH with public key encryption, Git, Subversion, and Github.

Other

I would be happy to discuss my experience with and exposure to: Python, MySQL, XML, XSLT, XSL-FO, RelaxNG, Apache HTTP Server, NGINX, HTML, CSS, Search Engine Optimization (SEO).

EXPERIENCE AS SENIOR PERL DEVELOPER

Peppels.net

January 2019 - present; remote, part-time

Skills used: Modern Perl, Template::Toolkit, JavaScript, jQuery, MySQL, HTML, wkhtmltopdf, Sass, CSS3, git, make.

Role overview:

- Maintenance of existing code base.
- Turning graphic designs into responsive HTML and CSS3.
- Complete redesign of the existing login page for both mobile and desktop.
- Adding additional PDF printing support.
- Adding support for portfolio pages for both mobile and desktop.
- Testing and adjusting mobile design.

SF Metrics

August 2015 - present; remote, part-time

Skills used: Modern Perl, LWP::UserAgent, Log::Log4perl, Test::Most, Text::Diff, XML::Parser, DBI, Config::Tiny, Try::Tiny, Path::Tiny, Plack, JSON, git, MySQL, GitHub, Emacs Org Mode, Markdown, NGINX, SEO.

Role overview:

- Developed a program to import a "JSON per line" database dump into a MySQL database. I reverse engineered the file format, designed the 70+ MySQL database tables, and added logic to avoid duplicate code.
- Developed a web crawler that downloads XML data, verifies this data, gathers some statistics, and stores desired data and statistics in a MySQL database. I am also reponsible for writing all the technical documentation, specifications, and example SQL queries. (ongoing)
- Developed several tools for analyzing search engine result pages (SERPs) (ongoing)
- Porting Perl programs to the Go programming language. (ongoing)
- Developed modules for storing data in Amazon's S3.

Interesting challenges: Because the web crawler program downloads data from the Internet it must be able to handle timeouts, partial downloads, etc. Moreover the XML parser must be able to distinguish between errors that can be safely ignored because they won't affect the desired data much, and fatal errors. As a single crawl can gather a lot of data, hence the MySQL database must be designed to deal with querying tens of millions of rows of data in mind.

The database dump program required several refactoring steps to avoid code repetition. Great care had to be taken to verify the types of JSON data and to verify all IDs had the correct type and were matching up in the original dump.

RedSocks - Malicious Threat Detection

February 2013 - December 2015; remote, part-time

Skills used: Modern Perl, LWP::UserAgent, Log::Log4perl, HTML::TreeBuilder, Test::Most, Test::More, Test::Output, JSON::XS, DBI, Try::Tiny, XML::Parser, XML::Writer, Email::Sender, Text::CSV_XS, MaxMind::DB::Reader, Dancer2, MySQL, RelaxNG, XSLT, Apache Ant, git, VirtualBox, NGINX, cron, Emacs Org Mode, OpenSSL, iptables, HTML, CSS, GNU Privacy Guard (GPG).

Role overview:

- Developed several web crawling Perl programs for gathering security related data, each crawler exporting downloaded data as a CSV file.
- Developed a Perl program that downloads security related data from several sources, verifies, cleans, normalizes the data, and export it as an XML file.
- Developed a Perl program that post-processes the exported XML data to remove duplicate entries and overlapping IP ranges using a binary search algorithm.

- Developed a Perl program that parses log files (Log4Perl) generated by aforementioned programs, creates reports, and emails those reports.
- Developed a Dancer2 web application to manually enter IP ranges and related security data and store this information into a MySQL database.
- Configured virtual machines for each of the Perl programs, including designing firewall rules, and allowing access via `rsync` and HTTPS in a secure manner where required. Documented the entire process of configuration and installation in great detail.
- Wrote documentation on how to create certificates to allow client side certificate authentication in NGINX (HTTPS), and configuring NGINX accordingly.
- Wrote extensive tests for all developed custom Perl modules and scripts.
- Designed file formats (XML, CSV). Documented each format in Org Mode format (Emacs).
- Designed Relax NG schemata to validate XML files.
- Wrote XSLT to transform XML output to older formats in order to support client programs that require an older format prior to being safely updated.
- Wrote extensive technical documentation in Org Mode format (Emacs).

Interesting challenges: Because the generated XML file is used in security related applications great care had to be taken to verify the integrity of the downloaded data and make sure that malformed data wouldn't end up in the final XML file. Because the generated data is used in programs written and maintained by other programmers, clear specifications and well-written documentation was mandatory. In order to keep this documentation under version control Emacs' Org Mode was used to write in plain text and convert it to, for example, PDF. Due to the asynchronous nature of the various programs care had to be taken to avoid race conditions.

EDUCATION

TH "Rijswijk", Rijswijk, The Netherlands Computer Science, BSc; 1991.

Utrecht University, Utrecht, The Netherlands Design of Algorithms, MSc; 1994. Thesis: "Constraint Satisfaction with Learning Automata."

COURSES

DelftX: FP101x Introduction to Functional Programming Functional Programming, Haskell, Monads; final grade: 98%; December 2014; course info; verified certificate (PDF).

Introduction to Big Data with Apache Spark Apache Spark, Python, Vagrant; final grade: 100%; July 2015; course info ; verified certificate (PDF).

Scalable Machine Learning Apache Spark, Python, Vagrant; final grade: 100%; August 2015; course info ; verified certificate (PDF).

GITHUB

tumblelog A static HTML5 microblog generator with support for RSS and JSON feeds written in Perl. A Python version with identical functionality is also available in the same repository: tumblelog.

resume-pandoc LaTeX resume template for Pandoc based on a LaTeX resume by Jason R. Blevins. The template can be used to create either a LaTeX or PDF file given a Markdown file as input. The template was used to create this resume. Repository: resume-pandoc.

MISCELLANEOUS

Developing Web Applications with Apache, MySQL, memcached, and Perl Technical Editor; ISBN-13: 978-0470414644; 2009.

Modern Perl I did some voluntarily reviewing, and am listed in the Credits section. Electronic versions.

A Retrospective Study of Polyallergy as A Marker of Non-Epileptic Seizures in the Epilepsy Monitoring Unit I provided the Perl program used in this study; article in Psychosomatics 55(6); June 2014; abstract.

Optimization of molecular methods and statistical procedures for forensic fingerprinting of microbial soil communities I assisted with Perl programming; article in International Research Journal of Microbiology (IRJM); ISSN: 2141-5463; Vol. 3(11) pp. 363-372; November 2012; full length research paper (PDF)

Languages Dutch (native), English (fluent).

My resume (PDF)