Ian Marsman

github.com/imarsman
(289) 400-1283
ian.marsman@gmail.com

SUMMARY

I am seeking work that involves backend processing and management of data using the Go programming language. My current work includes the construction of a data intake and processing system written in Java (past 7 years) and using AWS S3 and Elasticsearch as well as several Go language projects for the past four years. I would like to shift my focus to Go more exclusively.

SKILLS

- *Languages* – Golang (4y), Java (20y), XSL (15y), Bash (14y), Javascript (20y), Ruby (20y), R (3y), HTML (25y), LaTeX (2y)
- *Technologies* – REST, GRPC, SQS, NATS, S3
- *Data Formats* – XML, JSON, YAML, Protocol Buffers, Cue – including integration with Go structs
- *Cloud* – AWS and GCP clouds – including use of AWS SDK, CLI, and GCP CLI
- *Operating Systems* – Linux (since 1995), MacOS
- *Infrastructure and Services* – Docker, AWS Fargate, Nanos unikernel
- *Tools* – Git, Github, Gitlab (including CI/CD), Nginx, Apache, ProFTPd
- *Databases* – Postgres, MySQL, SQL Server, Elasticsearch, Redis, MongoDB

WORK EXPERIENCE

Systems Programmer   XML Team, Inc.   Fall 2007-Present
                                      Remote

Work at XML Team is focussed on code to manage the incoming XML feed. Incoming documents are transformed to a standard XML schema, undergo metadata extraction, then indexing of metadata in a JSON database, and are searchable from the JSON database using permissions tied to content metadata.

Go work at XML Team and personal Go work

- *2017-Present* – Use of an API library to write a Slack bot. First Go work.
- *2017-Present* – A replacement for a legacy Perl application that downloads documents from the document feed for metadata extraction in a relational database and optional transformation and storage of downloaded files. Periodic processing and reception of documents sent in asynchronously from server is handled using Goroutines.
- *2017-Present* – A Windows service to manage the installation and running of the above legacy Perl application on client systems. Replaces a proprietary Perl tool from ActiveState. The goal was to have a comfortable and simple setup process for clients using Windows.
- *2019-Present* – In-progress replacement for existing Java feed intake and search API. Functionality includes metadata extraction from XML documents to JSON, indexing of JSON to Elasticsearch, access permission evaluation tied to document metadata and client configuration and searching and retrieval of indexed documents — with all configuration handled asynchronously via threaded code. Includes tests.
- *2020* – An HTTP service to take in XML documents, filter out duplicates based on a hash of contents, and send non-duplicates to main feed intake. Includes tests.
- *2020-Present* – An application that translates MySQL formatted SQL from MySQL Workbench and translates it to SQL readable by Postgres and SQL Server. This was done to allow maintenance of a DB schema in MySQL Workbench and distribution of the SQL schema to clients.
- *2020-Present* – A password generating library and executable used for managing .htaccess password files with bcrypt hashed passwords. For use with feed system.

- *2020* – Improvements to a Markdown generating package to add unordered lists. Used in event reporting tool listed earlier.
- *2020-Present* – An HTTP service that compiles a list of event and roster for a date for over 50 soccer leagues from per-league schedules, putting the results in Markdown and JSON format and making the summaries available via a simple API which is used by a PHP front-end.
- *2020-Present* – An ISO-8601 time library that parses periods, dates, and timestamps, converting timestamps into Go language time values. Iterative parsing was chosen over the use of a lexing library following comparison using benchmarking. This library was written to allow for flexible intake of API and document data with the goal of minimal processing overhead and flexible imput format with detection of invalid input. Speed and resource use is optimized and is very competitive with other implementations. Includes extensive tests and benchmarks.
- *2021* – A package to allow for pluggable Basic Authentication and JWT session creation and management with the ability to add in support for shared sessions with services such as Redis using function interface implementations. Includes tests and benchmarks.
- *2020-Present* – A role-based authorization package using the Casbin library. Can be added to a project to work with whatever role based model the project uses. Includes tests and benchmarks.
- *2021* – Modification of an existing benchmarking tool for testing of HTTP endpoints. Additions to existing code include introduction of random delay between requests, request per second throttling, improved help output, and improved formatting of results and output of test reports to file. Used extensively in-house when moving an application to an improved implementation.
- *2021* – A set of tests used to compare reading and writing of JSON and YAML formatted data, converting from input formats to Go structs. This includes testing and benchmarking. Also includes an evaluation of the CUE library which allows schemas to be defined for JSON and YAML data and for JSON and YAML data to be merged and simplified. CUE is designed to assist with management and merging of configurations for complex Kubernetes frameworks but the focus for this library was mostly on benchmarking.
- *2021* – A simple project to test out nanovms in Google Cloud as image VM and as cluster. See here.

EDUCATION

Computer Science Brock University 1998
- missing linear algebra and basic statistics, though I have taken linear algebra separately and taken several years of graduate level statistics

MA Psychology Brock University 1995
Focus on brain-behaviour relationships (neuropsychology).

BA Psychology University of Victoria 1993
Focus on brain-behaviour relationships (neuropsychology).

OTHER SKILLS AND INTERESTS
- French language skills through extensive undergraduate class work
- Nature photography
- Cooking and baking