Athan Clark

Curriculum Vitae

✉ athan@deconfigured.com
🖉 DeConfigured

Profile

Self educated software engineer and computer scientist. Well versed - object oriented, functional, and domain-specific. Practical GNU/Linux and UNIX admin - can adopt any environment. Result-oriented, well-disciplined and tactical - skilled in complexity management and formal testing. Proficient in type safety and proofs.

Involved in advanced web development research projects - WebSockets, SVG, Animation, etc. Well versed multi-paradigm developer, with deep understandings of Functional and Object Oriented design.

Published Projects

DAG   Pioneered a new api for producing compile-time directed acyclic graphs in pure haskell. It uses an advanced form of Type-Level programming to facilitate acyclic safety.

Markup   Generalizes Html based on the style of how it's content is sourced - a simpler method for working with Html.

UrlPath   Method writing deployable urls - where the hostname is encoded in context, aiding in the port of a web application from one host to another.

Technical Skills

Programming Languages

Haskell   Haskell is the most advanced practical programming language as-of-date. I use it for my web servers, compilers, type systems, etc. - it's fast, clean, stable, and popular.

JavaScript   Node.js and it's surrounding projects have come a long way in the last 5 years - people are now paying more attention to clean composition and ideal code. I use JavaSript and Bower for my front-end development.

Ruby   Rails is everywhere, unfortunately, so it's important to know the trade.

PHP   This is another universal eye-sore that's important to know. I used to use PHP for my main web server development using Laravel, composer, and PEAR.

Python & C/C++	I have studied CPU architecture and low-level C programming (and assembly) for a while now. I only know enough to provide useful additions, not for orchestrating a codebase.

Principals

Assurance	Unit and Automated Testing with Property Establishment gives us some pseudo-certainty, but mastering Type Systems is how we get absolute correctness. Isolated Sandboxing and Reproducible Development also go hand-in-hand with this concern, Nix is a wonderful example.
Performance	Profiling is leveraged where Complexity and Asymptotic Analysis are just too difficult.
Availability	Browser-Bug Compensation and Evasion, Backward and Forward Compatibility, and Search Engine Optimization are key features for any website.
Clarity & Design	The things we create are ideas, materialized in code. But, this materialization is surjective, not bijective! It's critically important to be clear, and compensate where you're not.

Development

Administration	Installation & Configuration, Kernel Compilation, Hardware Optimization & Troubleshooting
Tools	Make & Autotools; Gulp, Grunt, Bower, NPM, PEAR, Composer & Cabal; Git, Darcs, Cvs; Vim, Atom
Networking & Basic Security	Routing & Switching, SELinux, File Permissions
Distributions	Ubuntu/Debian, Red Hat/Fedora, Gentoo, NixOS, Linux From Scratch, FreeBSD

Active Research

Ambiguous Lambda Calculus	New form of lambda calculus with Natural-number based function arity. It is designed for a flexible method of function application - perfect for arbitrary text templates.
PureScript	Purely-functional programming language that compiles to JavaScript - it is Haskell based, with some potential improvements (particularly in it's effect system).
Cryptol	Well-typed bit calculator featuring an interesting variant of dependent type theroy, where types depend on bit depth size.
Agda	A beautifully semantic dependently-typed programming language and theorem prover, similar to Haskell.

Past Work

American Financing Upgraded their legacy PHP and Ruby codebase to a unified Haskell web application - a mirror image of the previous (with semantic HTML improvements), while compensating for historic urls for proper redirection. Completed in 2 weeks.

Kifaru Outdoors Emergency web application - an iTunes clone for iPhones, but accessible from the web - to allow remote control of the song playing from the dance floor. Completed in 5 days.

Peerless Network Developed command-line applications for the Network Operations Center to maximize help-desk throughput.

References

Please contact me for references.