Andrew Towers

Phone:	+61 404 228 928
Email:	atowers@gmail.com
Skype:	mario.frog
Portfolio:	raffe.io
GitHub:	github.com/raffecat

I Believe

- Unexpected errors are for developers, not end-users.
- Everything should be designed for an audience and their goals.
- Code is a hand-written lowering of the program design.
- The right tool is the one that gets the job done well.
- Reliable systems are simple, not complected.

Skills & Experience

- **Back-end** node.js, ES6+, Typescript, Python, C++, microservices, idempotence, REST.
- **Front-end** HTML5, CSS, DOM, jQuery, Angular, browser quirks, IndexedDB, video.
- **Database** Postgres (SQL, joins, indexes), MongoDB (indexes, atomic ops)
- **Server** nginx, linux, debian, iptables, upstart, systemd, ssh, rsync, vim, mtr.
- **AWS** S3, CloudFront, ELB, AutoScaling, Security Groups, IAM, CloudFormation.
- **Desktop** Win32 API, DirectX, C/C++; OS X, Cocoa, Objective C; Linux, XLib.
- **Mobile** iOS, Cocoa Touch, OpenGL; web video player quirks.
- **Low-level** OS principles, bare metal, assembly language (x86,ARM)
- Data-oriented Design: CPU caches, branch prediction, game engines.
- Breaking things down into bits that can be readily understood and communicated.
- Iterating until simplicity is achieved, subject to diminishing returns.
- Interest in: declarative, distributed, failure modes, security, usability.
- Learning new tools and technologies, patterns, techniques and ideas.
- Inferring the root cause of a problem from symptoms.

Employment

2015 – now GAME TOOLKIT, CONTRACT WORK, TRAVEL

Developing a toolkit for independent game developers, making use of my low-level coding experience and familiarity with 3D APIs across Windows, Mac, Linux, iOS and WebGL. A mix of C/C++ engine, C++ and Objective C front-ends, tools and prototypes written in Lua and Javascript – using node.js and in-browser with WebGL.

Contract work on an online lunch-order website using node.js and postgres with PayPal integration – beaverstail.com.au. Angular on the front-end, served through Nginx as a static single-page app; idempotent REST APIs (with retries) to submit orders and process payment. Node.js back-end using Express. Site design provided by a 3rd-party. More details at raffe.io/beaver.html

I also spent some of this time travelling in Europe.

2012 – 2014 SMALL WORLD SOCIAL

White-label CMS product hosting per-client sites and mobile apps for corporate clients.

- Much of the bespoke CMS front-end: Angular, Bootstrap, JS, HTML, CSS.
- MongoDB schema design for atomic updates via idempotent REST APIs.
- Node.js REST APIs for CMS, schema validator, microservice glue library.
- Client-side session/retry REST wrapper with login hook.
- Drag & Drop front-end to schedule on user's calendar, Google Calendar API.
- HTML5 video player for seamless white background on Android/iOS.
- Upload service: unzip, image & video thumbnails, MOOV optimization, S3.
- REST API for incremental content delta-updates to mobile clients.
- AWS: S3, CloudFront, ELB, AutoScaling, VPC, Security Groups, IAM, etc.
- Nginx config: micro-caching generated front-end pages, proxy-cache S3.
- DevOps: dev-server to sim nginx routing and run/restart microservices.
- Front-end scripting for designers' HTML-first designs.
- Security awareness: 3rd-party penetration testers could not find a way in.

2011 – 2011 PANGUR (startup)

Python site using Jinja2 templates and a JS/jQuery front-end, using browser database features (IndexedDB, WebSQL) and SQLAlchemy with Postgres on the back-end. Included a custom forum within the site, and an in-browser document editor using browser databases.

I worked on both front-end and back-end with one other team member. I wrote the browser storage and sync-to-server parts, document editing logic, forum pages and database schema/queries, and some in-browser canvas rendering for document previews. The main site was server-side rendered; the document editor used REST APIs.

2008 – 2010 FITNESS2LIVE / McKESSON ASIA PACIFIC

Python/Twisted web app with some complex calculations to model meal planning, activity and kilojoules consumed. I worked on many of the site features; it was a monolithic server-side rendering app. Company acquired by McKesson.

2005 – 2007 TELSTRA RESEARCH LAB

C++ applications, Flash, HTML, Java; customer and technician install kits for Windows and Mac. Worked primarily on stand-alone executables that run without being installed (native Win32 APIs), a Flash layout engine with bindings to C++, integration with driver kits from 3rd-party vendors, 3G modem and home-WiFi-router configuration and control software.

2002 – 2004 BLUESTAR GAMES

Game engine design and programming, tools programming. DirectX and Win32, C++, language interpreter implementation. Also some J2ME game development for early mobile platforms.

1999 – 2002 DYTECH SOLUTIONS

Consulting firm. Multi-threaded C++ server code, Windows COM and DCOM, system architecture.

Education

1998	Bachelor of Computing (Honours 1st class)
1995 – 1997	Bachelor of Applied Computing, minor in Electronics

Referees

JEREMY VOS

Proprietor, Beaver's Tail. I met Jeremy at his coffee shop and ended up making the Beaver's Tail online lunch ordering platform. An entrepreneur at heart :)

Phone:	0433 107 829
E-Mail:	jeremy@beaverstail.com.au

TIM STEBBING

All-around good guy. I have worked with Tim on a number of projects and jobs, and he is a great communicator and star player. Tim was the tech lead at Small World Social.

Phone:	0434 746 089
E-Mail:	tim.stebbing@iflix.com

PAT OWENS

I nearly worked with Pat on a startup he was involved in, but the startup didn't pan out. Pat genuinely cares about the work he is doing and the people he is working with.

Phone:	0430 125 515
E-Mail:	pat.owens@outlook.com