David Dudson Software Developer (Metaprogramming)

Hire me remotely or in Cairns, Australia from March 2018

WIP: Please be aware that this site is still unfinished. It is largely unstyled, although all content is present

PDF Version

About Me

I am a cross paradigm developer who primarily deals with metaprogramming. I plan to work on tooling and/or language design in the future.

Professional Developement and OSS

Conferences

I try to attend as many conferences as I can. It's always interesting to see what the community is up to.

Kiwi Pycon

Category: Python

Times attended: 2

It is regarded as one of the most social conferences and has one of the best communities. I have attended Kiwi Pycon in Dunedin and Christchurch. I gave a lightning talk on GitPython scripting in Dunedin.

Kiwicon

Category: Information Security

Times attended: 1

Kiwicon X was an amazing conference to go to. It is a huge conference about everything from physical security, hacking, damage reduction etc.

New Zealand Javascript conference (NZJScon)

Category: Javascript

Times attended: 1

I went to NZJS con hoping someone would talk about Web ASM. Although I ended up being drawn in by GraphQL and hybrid single page/server rendered apps.

Linux Conference Australia (LCA)

Category: Linux

Times attended: 0

I will be attending LCA 2018 in January. I value both Libre/OSS and Linux highly so it will be great to attend.

Open Source

Community

IT Professionals New Zealand (ITPNZ)

A project rots from the head talk
: It was the inaugural meeting for the new Palmerston North based branch of ITPNZ

Palmerston North Software Developement meetup group.

Lunch & Pub meets.
: Every so often the group have lunch together. This is a great way to network and learn about the interesting work others in the community are doing.

Videos
: Around once a month the group have a video night. Usually the talk is about some fairly generic topic such as UX, ethics, project management etc.

Talks
: Occasionally a member of the group will give a presentation. I have been to the ones of interest. Unfortunately I could not attend Jonathan Giles talk. (Oracle Employee). I have also given a talk on Scala at these events.

Workshops
: I gave a Scala workshop following my talk, teaching the basics of scala development.

Technical

Operating System

Linux

I prefer Unix based systems for general developement. I would use OSX if necessary also

Distros:

NixOS
: I've recently started using NixOS, it has the complexity of Arch so it takes a while to setup. However, once setup it is amazing, being able to have multiple versions of the same OS without bloat is great. I've had to rollback a few times too, its completely painless.

Ubuntu (Gnome)
: Ubuntu is my "old faithful" os. When I want a bit more stability I use ubuntu. Although I still tend to stick with nightly releases. I really like Gnome and use it on NixOS and ubuntu. Unfortunately there's too much bloat in Gnome these days, it may cause me to switch.

Centos
: At PSE we target RHEL, thus Centos is a prime candidate when testing. All of the infrastructure is setup for Centos, thus setting up a development on any OS other then Windows or Centos becomes a bit of a mission.

Version Control

Git

I prefer a git-flow based, rebase -no-ff merge strategy. This keeps a linear history. I've written plenty of internal tooling/scripts dealing with Git.

IDE

Profilers

Yourkit

I would recommend Yourkit to anyone, Yourkit makes optimizing code a breeze.

VisualVM

I stopped using VisualVM after we purchased Yourkit. Yourkit really is far superior here.

Text editors

Atom

Visual Studio Code

gedit

Vim

Build tools

Gradle

SBT

Maven

NPM

Gulp

Webpack

Career

Current:

Overview:

Company: Electric Memory IT (EMIT)}

Contracted To: Process Systems Enterprise (PSE)}

Time Employed: ~2 years}

There are 3 EMIT employees, all living in Palmerston North, New Zealand. We work for PSE which is based in London. PSE + EMIT have around 29 developers and 200ish employees.

I am working on the gPROMS family of integrated modelling environments (IME's), designed for various engineers who design incredibly complex systems. The kernel is written in C++ and Fortran, with the UI written in Java. gPROMS is primarily a procedural/declarative language that is used to run simulations/optimizations etc. Thus the IME's act much like an IDE would for a Software Developer.

PSE's developement process is rather loose. Specific teams can choose to work in whatever way they see fit, whether that is agile, waterfall or some other idealogy. Code review is mandatory for all code. Unit tests and occasionally integration tests are used. Some teams also used acceptence tests. All code changes get verified by the QA team. Releases are anywhere from 6 -> 12 months.

Languages:

Java

Almost all of the application is written in Java. It is a 20+ year old swing application with nearly 400,000 lines of java code

Scala

The newest module of our application is written in Scala. It is effectively an immutable data model representation of our language.

Groovy

Our UI's gradle build system is rather complex and heavily made use of gradles plugin infrastructure. The plugins are written in groovy.

Python

Most of the dev scripts I wrote used python. I used a gitpython frequently to programatically interact with my repository. I was also briefly helping a team using Django to build a performance monitoring website.

C++

My experience with C++ at EMIT is mostly read only. As part of the NLE it was necessary to port a chunk of C++ to Scala. PSE has ~350,000 lines of C++ manage

FORTRAN

Our solvers are written in FORTRAN, there was one bug that my boss guided me to fix, where a 1000 line function was returning in multiple places before jumping into the correct clean up section. It is very interesting using such an archaic procedural language even briefly.

Libraries and Frameworks:

ANTLR (2 and 4)

One of my main tasks for NLE was porting the mixed Antlr2/C++ codebase to Antlr4/Scala. Due to performance reasons, I implemented custom char/token streams, tokens etc. and basically micromanaged the parser to optimize performance.

JGraph

JGraph is our flowsheeting library which is overdue for replacement. JGraph discontinued development nearly 10 years ago. The developers working on it write a Javascript library called MxGraph. Being so old, JGraphs internals are shocking to work with, raw types everywhere. It is also rather horrible to extend and manipulate.

YFiles 2.x & 3.x

YFiles is a modern graphing library much like JGraph. On our flowsheets we require all connections between models to be orthogonal. JGraph was not powerful enough for this, so we use YFiles instead. It would be great to port the rest of our JGraph code to YFiles as it is much easier to work with.

Ansible

I've built multiple Ansible scripts for various uses. One for Gerrit/Zuul, one for a performance monitoring site.

Others

- Swing
- Corba
- GitPython
- Django
- Puppet

Projects:

New Language Engine (NLE)

The purpose of this project was primarily to remove a large chunk of the remaining CORBA code in our codebase. We wanted to make our parser and data models backend agnostic so that in the future we may be able to port the C++ code from CORBA -> ANTLR + Scala Native. The use of Scala was mostly due to case classes and an very performant collections library. Since scala is an immutable first language it is an ideal choice for a threadsafe data model. Since the project has to be highly performant, every was analysed and benchmarked by JMH and Yourkit java profiler.

Maven -> Gradle Migration

Our build system is rather complex, between CORBA and ANTLR generation steps, multiple jvm languages being used, and

multiple repositories. The build was designed to be modular via Gradle plugins. Composite builds are also active so we can test and develop our core platform with any of our various products.

Flowsheet routing

Our users were generating models in other applications, with 2000+ models and connections between them. The routing took place on the EDT and thus blocked the user from working while it happened. For user generated projects this could happen of first project load and trigger up to 8 hour GUI freezes. Of course this is not acceptable, so using a combination of CompletableFutures and a custom thread pool. Routing was moved off the EDT. The largest problem was undo. We had asynchronous edits that had to be managed within Swings very synchronous undo system.

Past

Dick Smith Electronics

Role: Salesman}

Time Employed: 1 year

Countdown Supermarkets

Role: Butchers Assistant/Checkout Operator}

Time Employed: 5 years

Paradise Valley Berry Farm

Role: Raspberry harvester}

Time Employed: 1 summer

Education

Massey University

Qualification: Bachelor of Information Science. Computer Science Major

Status: Compulsories complete, finishing electives via part time study.

Western Institute of Technology Taranaki (WITT)

Qualification: Diploma of Creative Technologies. Animation Major

Status: Incomplete

New Plymouth Boys' High School (NPBHS)

Qualification: NCEA lvl 3

Status: Complete

Highlands Intermediate

Lepperton Primary School

References

Please ask me for contact information of my references. I wish to not make it available publicly for privacy reasons. Also, when contacting my references, please take note of their timezones.

Duane Griffin

Relationship: Cofounder of Electric Memory IT

Location: Palmerston North, New Zealand

Timezone: New Zealand Standard Time (NZST - UTC+13)

Email: ???

Phone: ???

Duane is my current boss. He has worked for PSE for 14+ years. Currently heads on the NLE project. Ask Duane any technical questions about my Scala knowledge and/or involvement in NLE.

Luke Usherwood

Relationship: Cofounder of Electric Memory IT

Location: Hungary

Timezone: Central European Time (CET - UTC+1)

Email: ???

Phone: ???

Luke worked for a short time in New Zealand when I first started. He has since moved back to Hungary. Ask Luke any technical questions about the flowsheeting work I have done. He also has worked on NLE and I send a lot of code reviews his way.

Andrew Bevan

Relationship: Head of Software Development, Process Systems Enterprise

Location: London, England

Timezone: Greenwich Mean Time (GMT - UTC+0)

Email: ???

Phone: ???

Andrew is the big boss. If you wish to know any non-technical information about how I have worked with PSE, feel free to ask him.

Steven Carpenter

Relationship: Ex Manager of Dick Smith Palmerston North

Location: Palmerston North, New Zealand

Timezone: New Zealand Standard Time (NZST - UTC+13)

Email: ???

Phone: ???

Steven (Steve) was my boss at Dick Smith before I started working as a Software Developer. If you would like to know anything about my retail experience, including customer interactions, feel free to ask Steve.