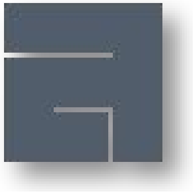
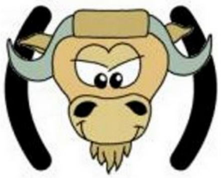


How awesome ended up with Lua and not Guile



Julien Danjou
FOSDEM 2016, Brussels, Belgium
Sat 30 Jan 2016

Hello!

I am Julien Danjou

Hacking Free Software for more 15+ years: Debian, Freedesktop, GNU Emacs, awesome, Python…

Nowadays I work for **Red Hat** on **OpenStack**.

You can follow me on Twitter: @juldanjou

1

The birth of a window manager

A long time ago, in a galaxy far, far away…

Dual screens user

> *Segmentation fault.*
>
> – FVWM

Looking for a new challenger

- Finally stumble upon *dwm*
- Start patching and managing patches for others
- Recompile to reconfigure? WTF
- Finally fork it for *jdwm*, renamed to *awesome*

This made version 1.0.

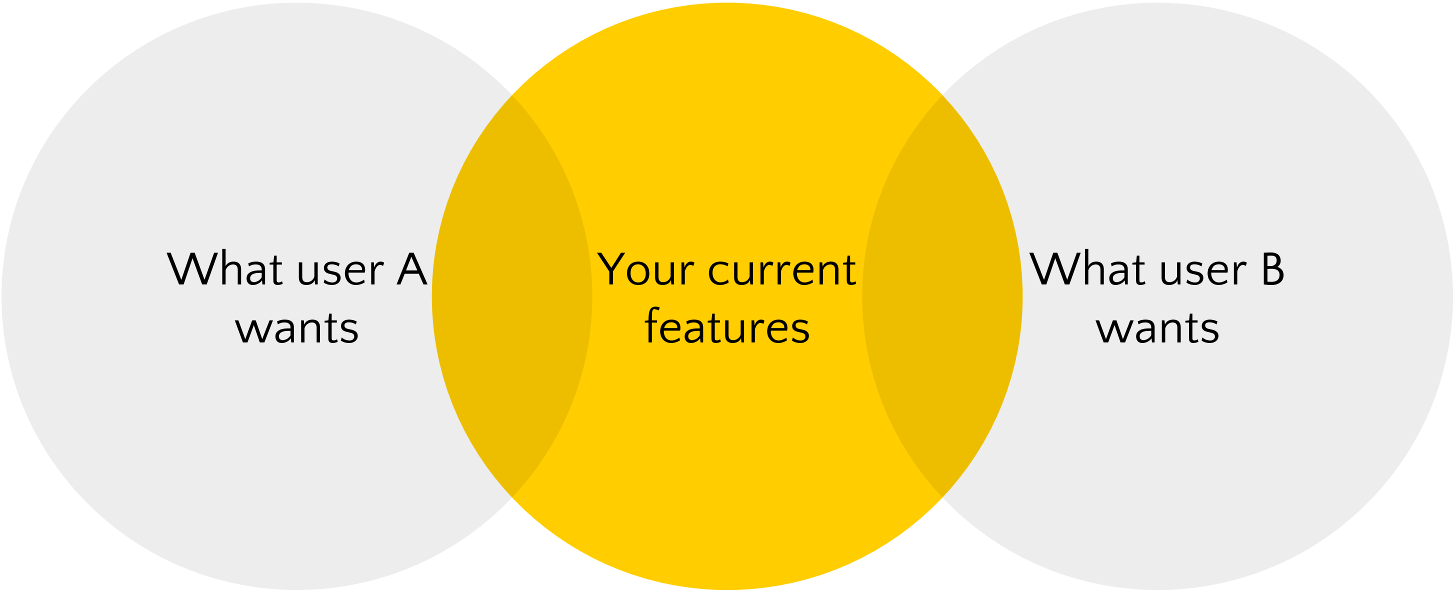
2

Can you...

Where users start complaining

> *Because dwm is customized through editing its source code, it's pointless to make binary packages of it. This keeps its userbase small and elitist. No novices asking stupid questions.*
>
> – dwm homepage

Feature creeping



I object to doing things that computers can do.

– Olin Shivers

3

Picking the tools

Where our journey really starts

Contenders

Python

Already hacked with it. Love it.

But how the hell do you embed it?

Lua

This is light, small and neat. Looks easy to embed because it's almost designed for that.

Guile

Why is there so many parentheses?

Designed for being embedded too.

Let’s try Python

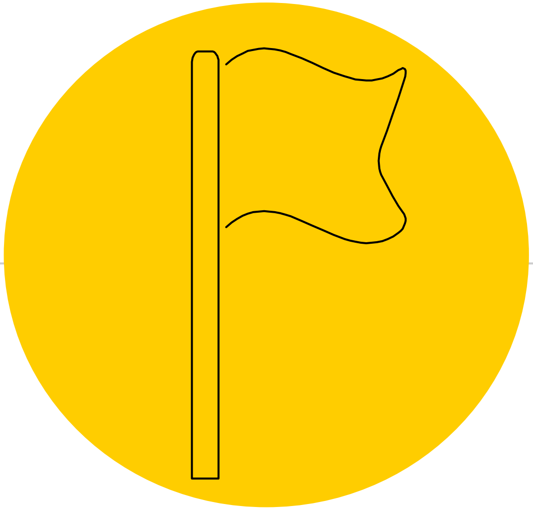
- Python 2 or Python 3?
- Embedding documentation not helping
- A few concepts I was not really familiar with, e.g. reference counting which has different methods (Py_DECREF vs Py_XDECREF ?!?)

What about Guile

- Language is scary
- Designed for being embedded
- Latest version in 2007: Guile 1.8
  - Not exciting
- Documentation scarce

Then comes Lua

- Embedding is easy
- Documentation is clear
  - Took me half a day to understand the whole language and how to embed it
- Small footprint
- Able to build a PoC in a few days

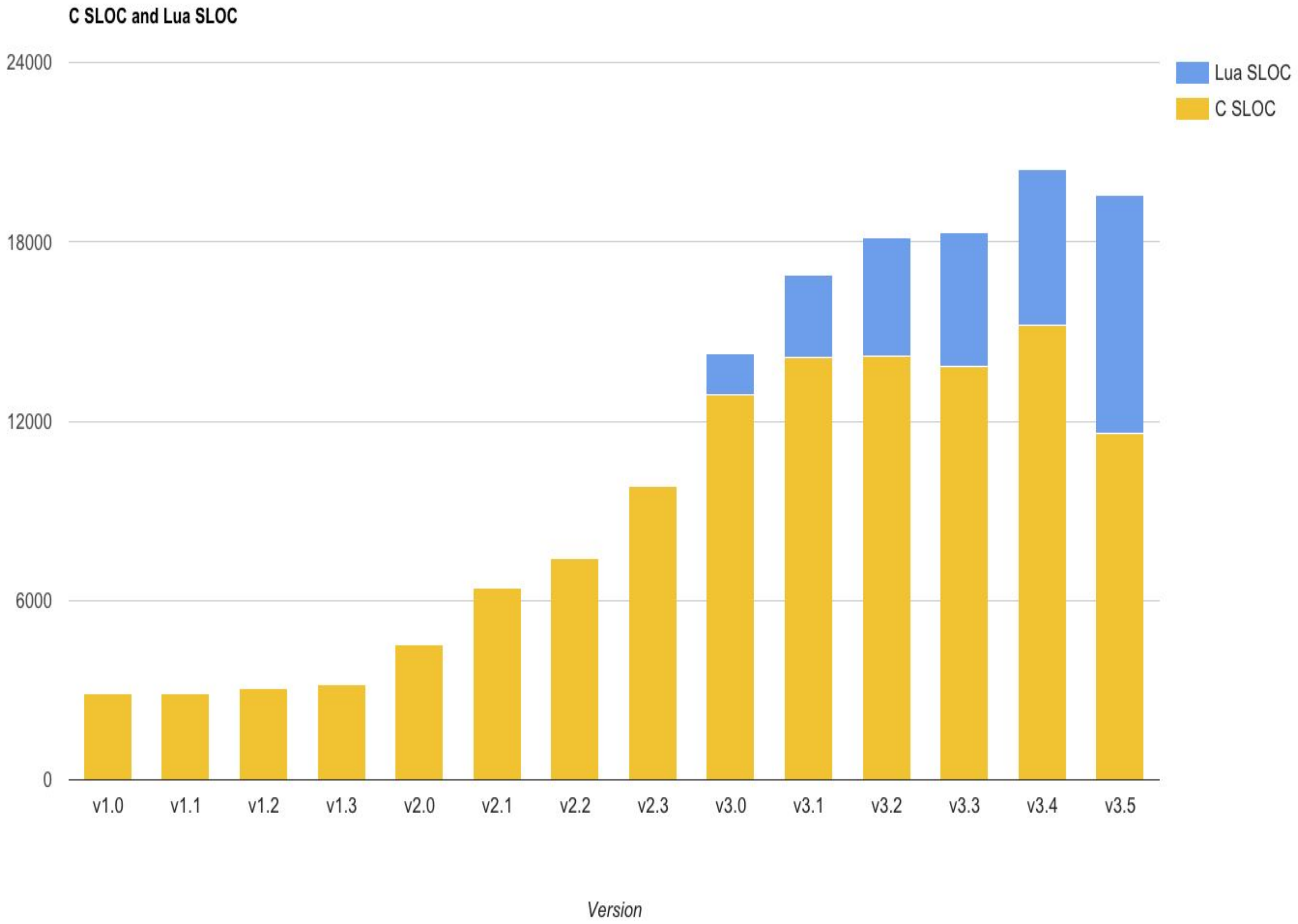


awesome 3.0

13K SLOC of C

1.3K SLOC of Lua

And is still works!



4

How it went

It’s always easier with a hindsight

Lua upsides

Fast

No other window manager offers this ratio of features, memory footprint, speed and code size, still in 2016.

Easy learning curve

Lua is easy to learn and to manipulate. REPL available, good documentation.

Community

Tell them about the time we wrote our own Pango binding and then we were able to ditch them.

> *Greenspun's Tenth Rule of Programming: any sufficiently complicated C or Fortran program contains an ad hoc informally-specified bug-ridden slow implementation of half of Common Lisp.*
>
> – Philip Greenspun

Lua downsides

Stack based C API

Embedding is simple, but building and debugging large application can be a nightmare (no gdb, no Valgrind...)

No proper paradigm

No ref-counting.

No object oriented.

Still a few design flaws in Lua 5.0/5.1

Small community

Only a few extensions and code published in 2007.

Footprint limited, so no inclusion of anything.

Guile (potential) upsides

Multiple language support

Would have love to use Emacs Lisp. But did not exist back then!

Now it supports Lua, right?

Guix

and things like that look cool.

Lispy

Should I tell something more about that?

Lessons learned

Do not embed

You never want to embed. You want to extend – at least in the long run.

Language is community

Picking a language makes you part of a community, like it or not.

YOLO

Lua is good enough for its tasks in awesome.

No regret.

Thanks!

Any **questions** ?

http://awesomewm.org

You can find me at

- https://julien.danjou.info
- @juldanjou