# Packaging the Free Software Web

Antonio Terceiro

# About me

- Ruby maintainer
- Debian CI maintainer
- web developer
- addicted to automation

# Motivation

# Debian in the Dark Ages of Free Software

Stefano Zacchiroli, Debconf 2014

- Desktop mostly a solved problem
- Mobile devices not quite there
- But most people are using desktops and phones as dumb clients to proprietary stuff "in the cloud"

Subject: Re: **About the recent DD retirements**
From: Anthony Towns
To: debian-project@lists.debian.or

- webapps are not a solved problem in Debian
- (among others which I won't get to)
- go read that thread!

# Can Distros Make the Link? Let's Package the Customizable, Free Software Web of the Future!

Deb Nicholson and Christopher Allan Webber,
FOSDEM 2015 Distros Devroom

- Current privacy concerns
  → federation + self-hosting
- how can non-sysadmins self-host?
- "userops" (beyond "devops")

mailing list:
http://lists.mediagoblin.org/listinfo/userops

# Are distributions really boring and a solved problem?

Lucas Nussbaum,
FOSDEM 2015 Distros Devroom

Server side apps in general
are not a solved problem

# Summing up

- Server side applications are complicated
- Requires high effort from technical people
- End users can't self-host

Someone has got to do something about this …

# Ideas

# Automated configuration of packaged web applications

GSOC 2015|Outreachy project idea

# Related projects

- Sandstorm
- Yunohost
- Bitnami (proprietary)

All are either not based
on packages, or reinvent packaging

# The free software web in Debian

- $ apt-cache search web returns | wc -l
  3657
- 92 packages ship files in /etc/apache2/
  - excluding apache itself and modules
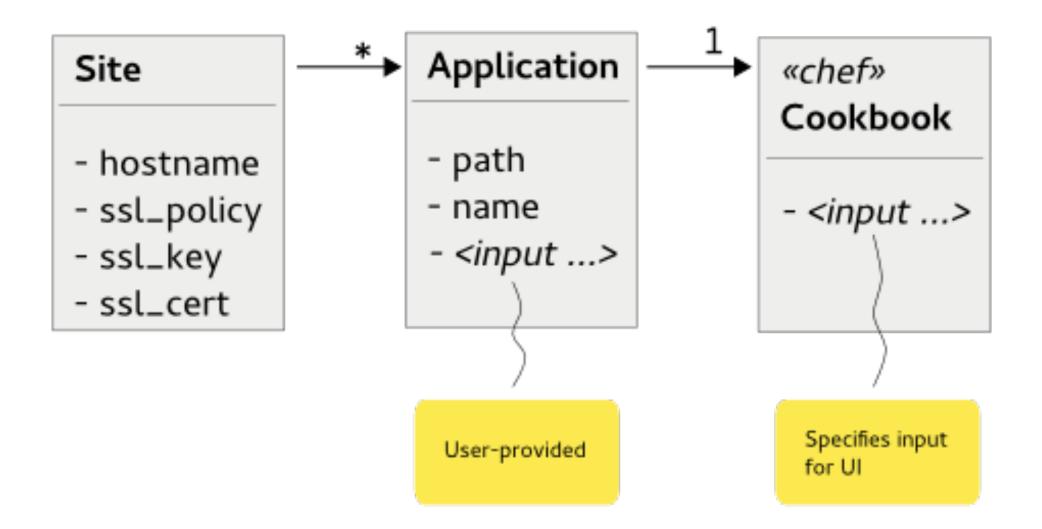  - several web apps don't do it
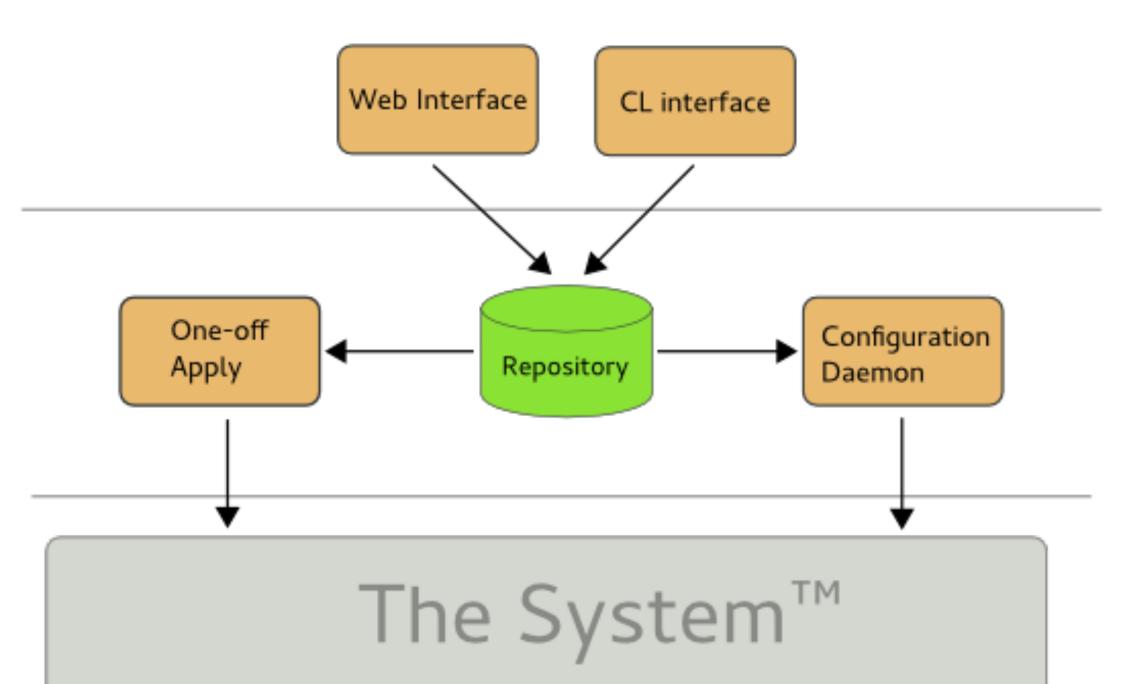    (e.g. redmine)

It seems we are quite good at packaging …

… but packages are not enough. Server-side applications still need one or more of

- cross-package configuration
- databases
- DNS setup
- email setup

# Proposed solution

- Using packages (of course)
- A configuration layer over the package manager
  - existing "devops" tools should do it
  - I'll go with chef
- A UI layer on top to empower users with no technical background to configure their own server

**Site**

- hostname
- ssl_policy
- ssl_key
- ssl_cert

\*→

**Application**

- path
- name
- *<input ...>*

User-provided

1→

*«chef»*
**Cookbook**

- *<input ...>*

Specifies input
for UI

Web Interface   CL interface

One-off Apply   Repository   Configuration Daemon

The System™

# Assumptions on packages

- Should be able to do all their specific configuration by themselves
  - not automating web server configuration is OK
- Should be able to handle their own upgrades (e.g. database schema changes)
  - **dbconfig-common** is your friend, but needs love
- Should not be crap [whishlist]

Current status: PoC

- Base system implemented
- add-site, add-app commands on CLI
- simple "static site" cookbook

https://gitlab.com/terceiro/shak

# Example

```
$ shak add-site example.com
$ shak add-app example.com static_site /
    site_type=push
$ shak apply
```

Static site deployed to
http://example.com/

Push contents via scp/rsync to
**server:/var/lib/shak/data/example.com/_/**
to update

# Next steps

# Next steps (1/2)

- web UI
- I18N
- GSOC|Outreachy: support as many packages as possible
- Improve web apps policy
- Send patches

# Next steps (2/2)

- Easy bootstrapping
  - mostly just
    `apt install shak`
- prebuilt images
  - mostly just
    `debootstrap --include=shak`
- spawn new servers in the cloud

# Ways to contribute

- Code of course
- Testing, bug reports
- Ideas on DNS/SSL/email handling
- Security audits
- Translations

https://gitlab.com/terceiro/shak