
The Open Sourcing of Infrastructure

SLC DevOpsDays 2017
Elizabeth K. Joseph
@pleia2



Elizabeth K. Joseph, Developer Advocate at Mesosphere

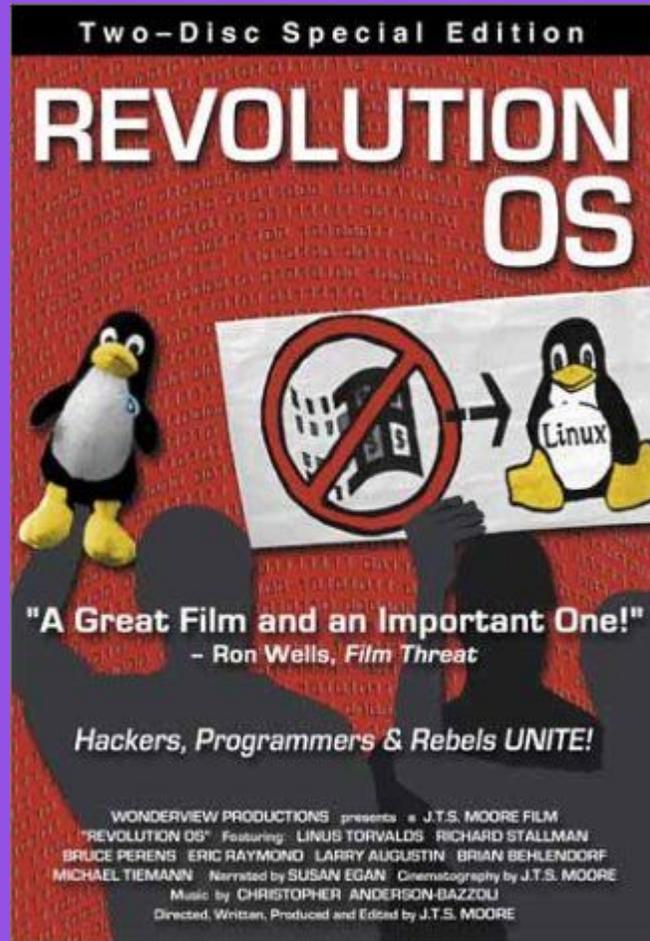
- ❑ 10+ years in Linux systems administration and engineering roles
- ❑ 15+ years working in open source communities
- ❑ Founder of [OpenSourceInfra.org](https://www.opensourceinfra.org)
- ❑ Author of [The Official Ubuntu Book](#) and [Common OpenStack Deployments](#)

The [recent] history of infrastructure

(from a highly opinionated,
open source view)

Once upon a time





Linux was an
upstart

Lots of FUD
around open
source

I liked it anyway.

So I got a junior Linux systems
administrator job!

Some of the topics during a seminar I spoke at in 2009

- What is Free/Open Source Software (FOSS)?
- How & Why Linux and FOSS can Deliver Business Results
- Managing FOSS: Thousands of Alternatives - How To Choose?
- Using Open Source Web Applications to Produce Business Results
- Managing FOSS for Business Results

Why do I want the source?

- To customize and develop the software
- To fix bugs
- To control security

Flood of changes, including

Downtime becoming [considerably more] unacceptable

Greater concern over security, ability to fix bugs

Reluctance to be locked in by a vendor

Increase in reliance upon scaling and automation

Transition from server “pets” to “cattle”

Larger focus on data (retention, speed)

Turning point: LAMP stack



Open source is now ubiquitous



Developers are using, developing on, **contributing to, and sharing** open source software!



Operations is using and developing on open source software.



When I left my ops job, I left my tools behind



Time to open source ops stuff!

Done!

Configuration management led the way:

Puppet Modules, Chef Cookbooks, Ansible Playbooks

Open application definitions:

DC/OS Universe catalog, Juju Charms

Full disk images:

Dockerhub and other container registries

Welcome to the present!

Open Sourcing Infrastructure:

PHASE 2

What were some of the reasons for going open source in the first place?

- Security
- Ability to diagnose and fix bugs without vendor intervention
- Increased control over our data and services
- Avoiding vendor lock-in

The Cloud.

Including IaaS, PaaS, SaaS...

“Most people just consume the cloud without thinking ... many users are sinking cost into infrastructure that is not theirs, and they are giving up data and information about themselves without thinking.”

Edward Snowden, OpenStack Summit, May 9, 2017

Let's think.

What is my recourse if the service vendor goes out of business, or is bought by a competitor?

Does the vendor have a history of communicating clearly and honestly with their customers about downtime, security, etc?

Will the vendor use our data in a way that I'm not comfortable with? *(or worse, isn't allowed by your own customer agreements)*

You *could* consider all these things and acknowledge them as acceptable risks.

(Many organizations do!)

((Just make sure you are actually, seriously considering them))

Or look again to Open Source!

Various infrastructure technologies are available:

- OpenStack
- Kubernetes* and Docker Swarm*
- DC/OS*
- ...more in the future!

* Can be used in the cloud or on premises



Even further into the future

(or how we're going to colonize
Mars!)

Open Source the Whole Stack

Infrastructure, configurations, tools, images

[OpenStack](#)

[OSUOSL](#)

[KDE & Gnome](#)

[Debian & Ubuntu](#)

And more at opensourceinfra.org

What do these projects get?

Contributions
from anyone,
anywhere

Vendor
independence

No lock-in

Community
ownership

Also coming up in SLC...

Tobi Knaup, CTO and Co-Founder of
Mesosphere @ DevOpsUT

Thursday, July 20, 2017
6:00 PM

<https://www.meetup.com/DevOpsUT/events/240003317/>

Questions? Feedback?

Contact me:

Elizabeth K. Joseph

Twitter: @pleia2

Email: lyz@princessleia.com