Patterns and Practices for Open Source Project Success

stephen r. walli
@stephenrwalli
stephen.walli@gmail.com



Freeloaders are Necessary

stephen r. walli
@stephenrwalli
stephen.walli@gmail.com



WTFOSS*

stephen r. walli
@stephenrwalli
stephen.walli@gmail.com



^{*} Hat tip to @codepope



Community



USENIX

Linux

DECUS

Princeton

SHARE

We've shared software since we've written software Writing good software is hard work

Berkeley CSRG

Free Software Fountation

MIT Athena

ASF

Apache

Eclipse Fountation



2 Stories

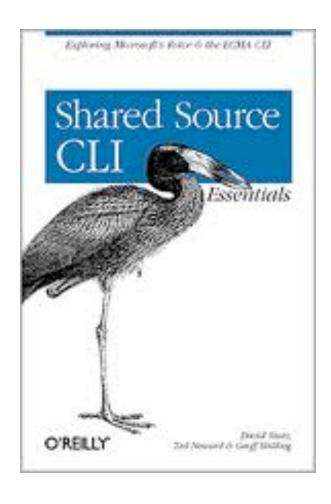


Orders

of magnitude



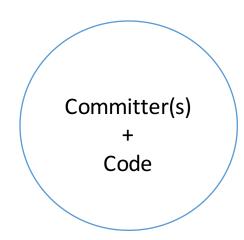
ROTOR

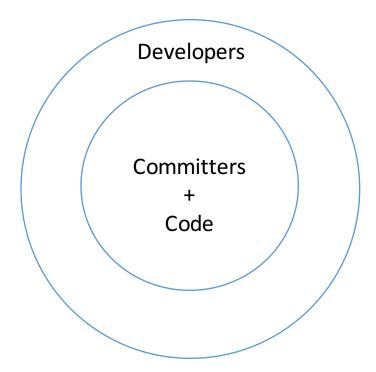


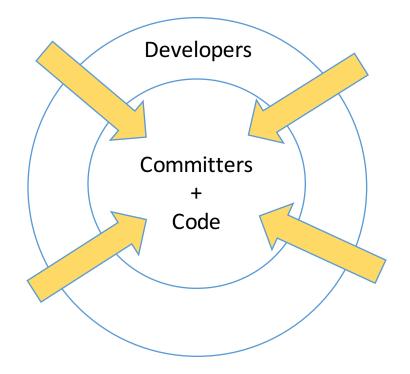
500K LoC
500K Lines-of-Test Harness
Ran on Windows, Mac OS X, FreeBSD
One script to set environment
One command to build everything
One command to test it all
Minimal documentation

24 hours later ...24 hours later again ...

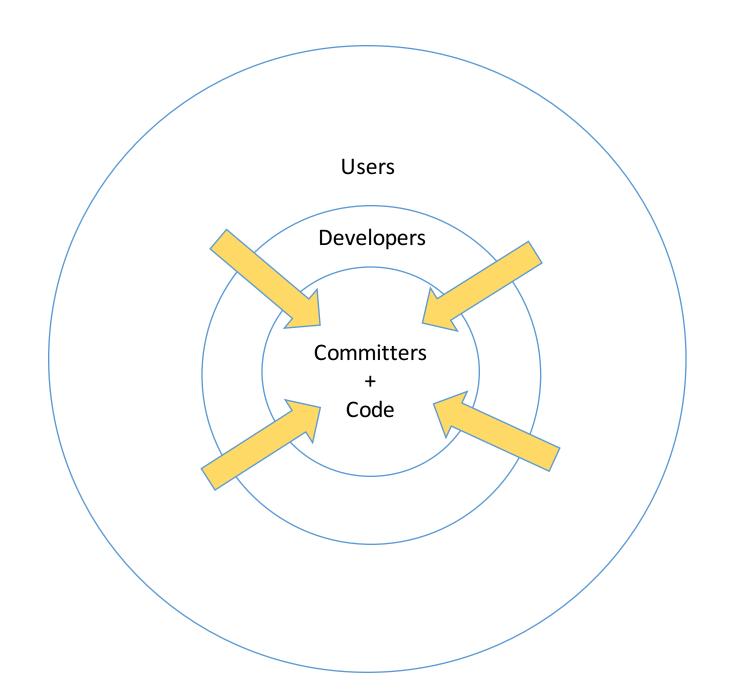




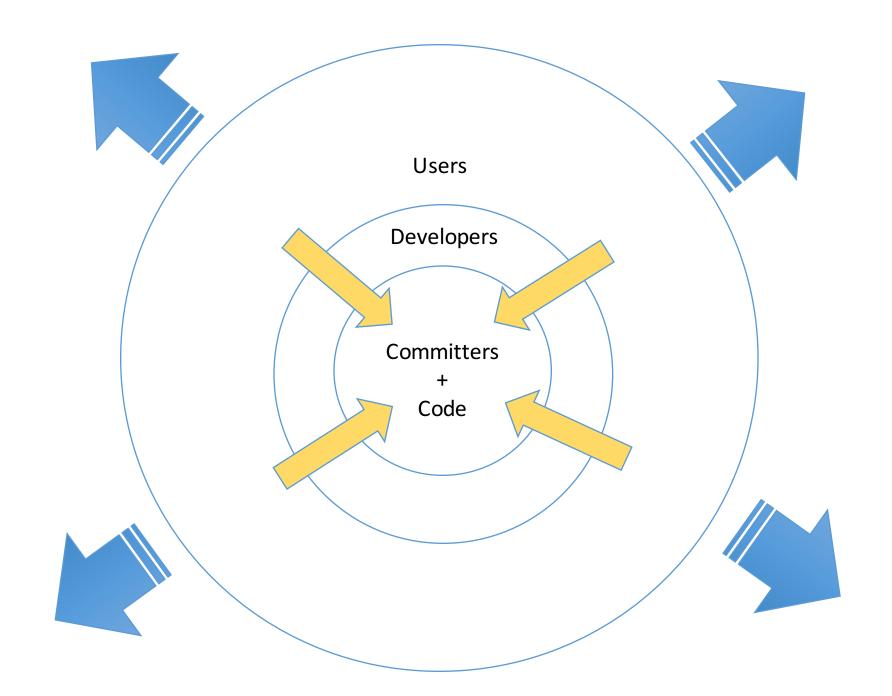














How do you increase your user base? (How do you make it easy to install/configure/use the software?)

How do you encourage developers? (How do you make it easy to build/test/experiment?)

How do you make it easy to contribute? (What do you communicate to your community)



How do you increase your user base?

(How do you make it easy to install/configure/use the software?)

How do you increase your user base?

(How do you make it easy to install/configure/use the software?)

Project Exes published Project Install Automated

Project BugTracking

Software Construction Activities



How do you increase your user base?

(How do you make it easy to install/configure/use the software?)

Project Instal BugTracking Automated published Project **Software Construction Activities** Email FAQs, Howto Forums, Community Development Activities



IANAL



A small (2 minute) diversion on licensing

- Software is protected by copyright law
- Whoever wrote the software, owns the copyright
- People often give up copyright ownership in employment agreements
- There are a few well [understood|accepted] licenses: pick one
- These licenses define the most successful collaborations in history
- If you care about making money: educate yourself* & hire a lawyer
- If you [care | worry] about patents: hire a lawyer
- If you publish your project on Github: create the *%\$ing LICENSE file Writers | Artists | Engineers | Architects do it. So should you.



How do you encourage developers?

(How do you make it easy to build/test/experiment?)



How do you encourage developers?

(How do you make it easy to build/test/experiment?)

Complete Src published Project Build Automated I

Project Test Automated I

Software Construction Activities



How do you encourage developers?

(How do you make it easy to build/test/experiment?)

Complete Src published Project Build Automated I

Project Test Automated I

Software Construction Activities

Mission Statement Contributoion Guidelines

Comms Platform

Code of Conduct

Community Development Activities



A small diversion on software engineering ...

- 2 ratios define software development
- All advances in programming languages and software engineering is attempting to beat these two ratios
- [Community|Product] scale depends on reliably delivering the known executable environment every time
- Linus's Law is about REVIEWS not BUG FIXING



How do you make it easy to contribute? (What do you communicate to your community)



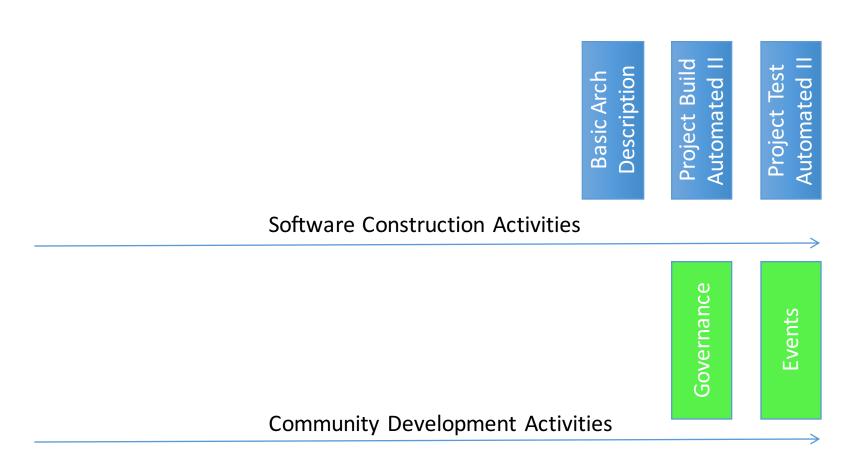
How do you make it easy to contribute? (What do you communicate to your community)

Basic Arch Description Project Build Automated II Project Test Automated II

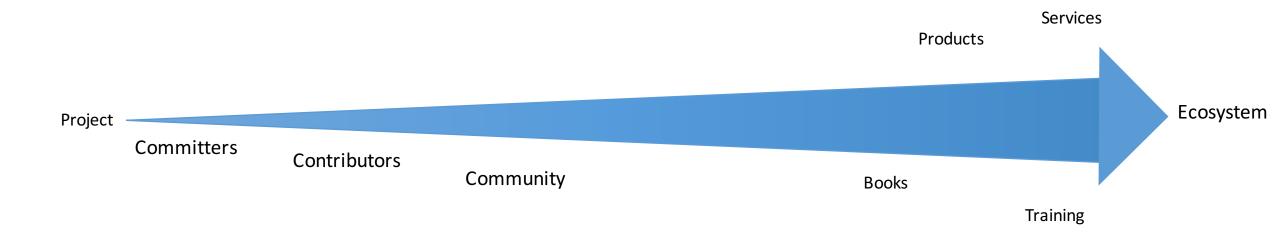
Software Construction Activities



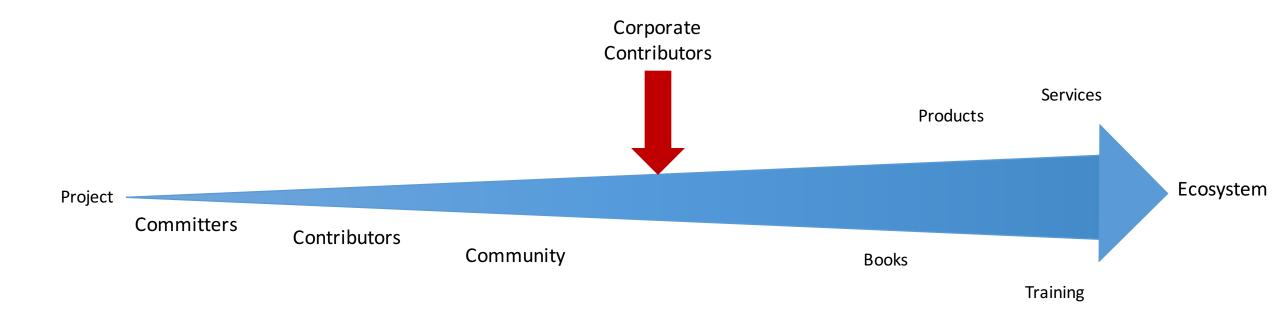
How do you make it easy to contribute? (What do you communicate to your community)













How do you make it easy to contribute? (What do you communicate to your COMMERCIAL community)

Dependencies Contributions Documented Repositories Vlanagemen Vanagemen Governance Provenance **Provenance** Committers <u>Indemnified</u> Committer **Trademark** Protected Auditted License IP Management Activities



Foundations

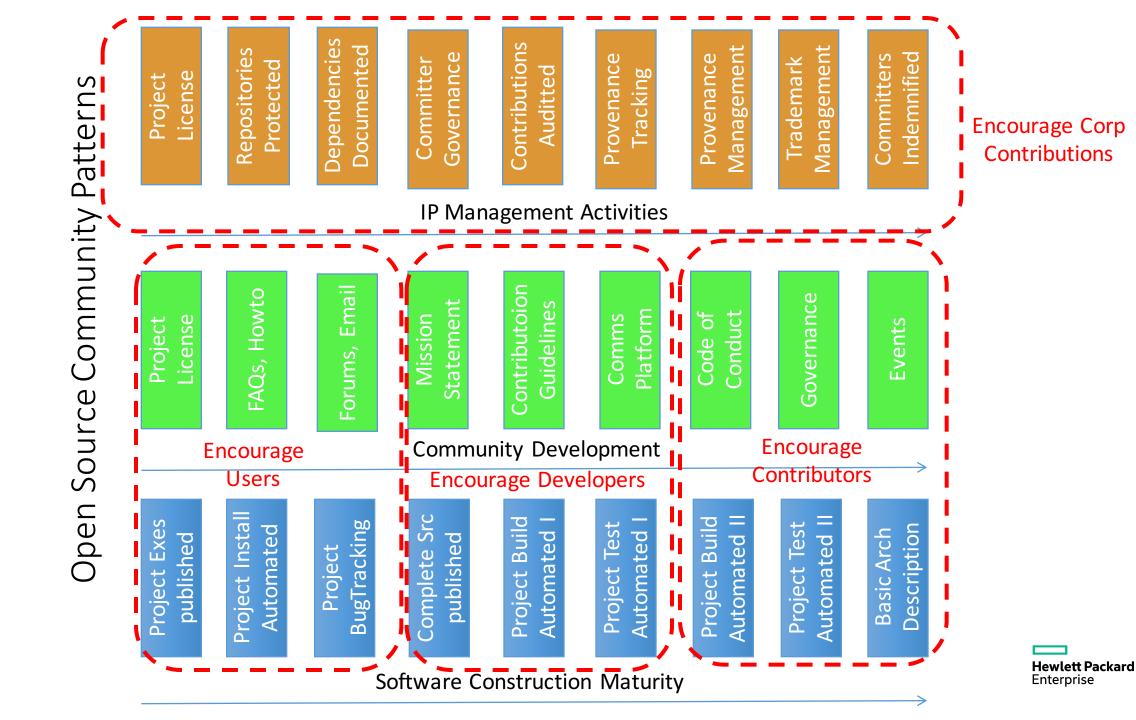
- Henrik Ingo's numbers
- Foundations provide neutral ownership and a level playing field
- Bright lines for projects versus products

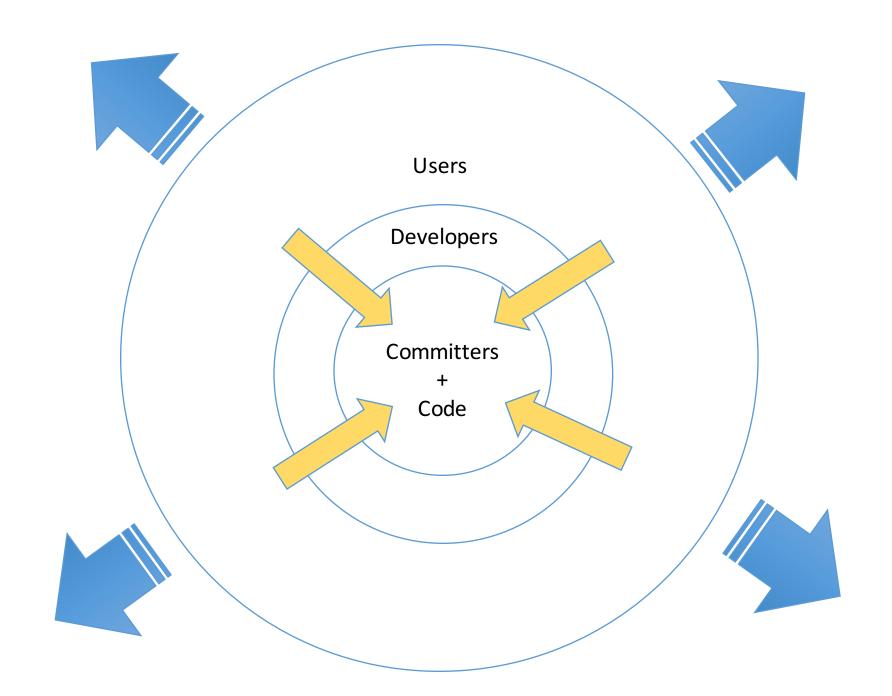


Community Practices Source Open

Dependencies Contributions Documented **Management** Repositories **Vanagement** Indemnified Sovernance Provenance Provenance Committers Trademark Committer **Protected** License Auditted **Tracking Project** IP Management Activities Email Howto Contributoion Governance Statement Guidelines Conduct Comms License Project Platform Code of Mission **Events** Forums, FAQs, Community Development Src Project Install **Automated II Automated II Project Build Project Build** Project Exes Test BugTracking Description **Automated** published **Basic Arch** Automated Automated published Project Complete Project Project Software Construction Maturity



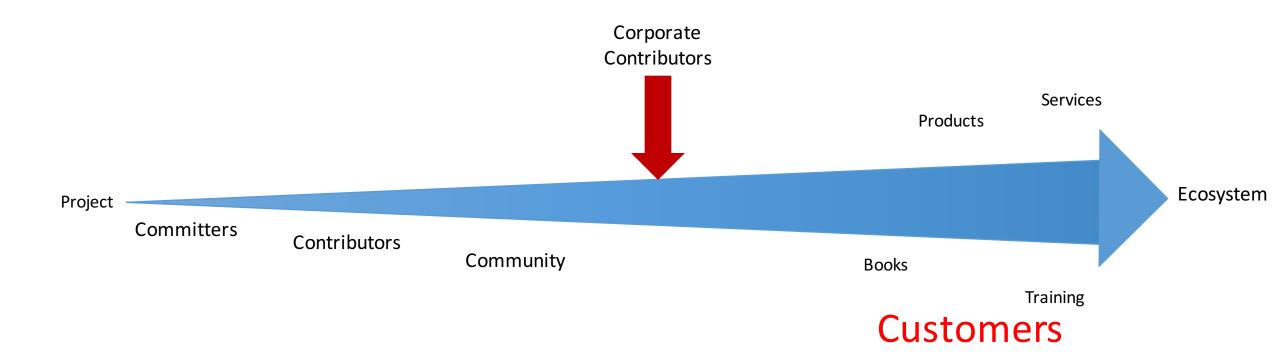




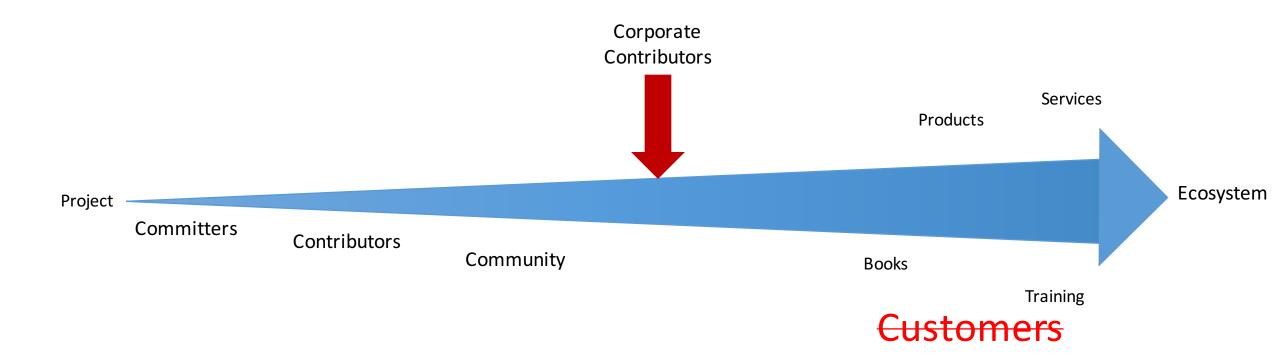


Building the Ecosystem **Products** Training Users Developers Consulting Books Committers Code **Distributions Contractors** Support











Questions & Comments

stephen r walli stephen.walli@gmail.com @stephenrwalli

http://stephesblog.blogs.com

http://opensource.com

