

**How to
become a
free software
leader in
three easy
steps**

- I. Hi!
- II. Importance of leadership
- III. On free software
- IV. What is free software?
- V. Organizational models
- VI. Break
- VII. Mixxx & Open edX
- VIII. Being too successful
- IX. How-to Guide

Hi

I'm Molly

I was once invited to an Open Source Thought Leader event hosted by Facebook.

(I didn't go.)

**Why is
leadership
important?**

We can save the world if we know
how to build things together. That's
what communities are about.
Communities are about building
things together.

Communities need leaders and
structure.

We'll get back to this later.

What is this thing called
free software?



Free software is a philosophical stance on the rights of people who interact with or own technology driven by software.

Sometimes we call individual pieces of software “free.”

Having your software be free is a moral imperative.

Opening up code bases can lead to innovation.

There are practical benefits to your code being open.

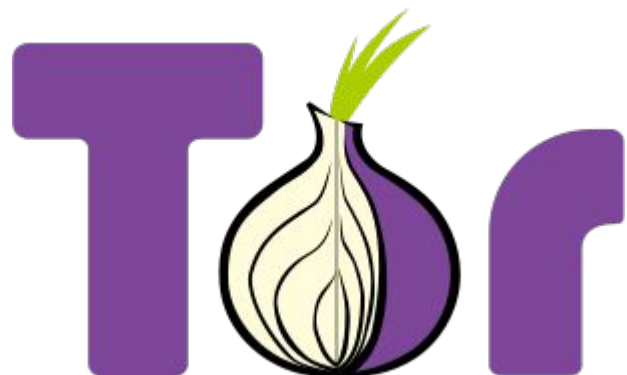
Building community allows you to experience these benefits.

Creating good free software comes
from breaking down the wall.

**“Ideas don’t
come from
people; ideas
come from
conversations.”**

-Cesar Hidalgo

Examples of free software Projects



Tor logo, CC-BY; BSD demon, All Rights Reserved; Open Hatch logo and SeaGL logo ???

Anatomy of a Free Software Project

A typical free software project



A lucky typical free software project



An ideal free software project

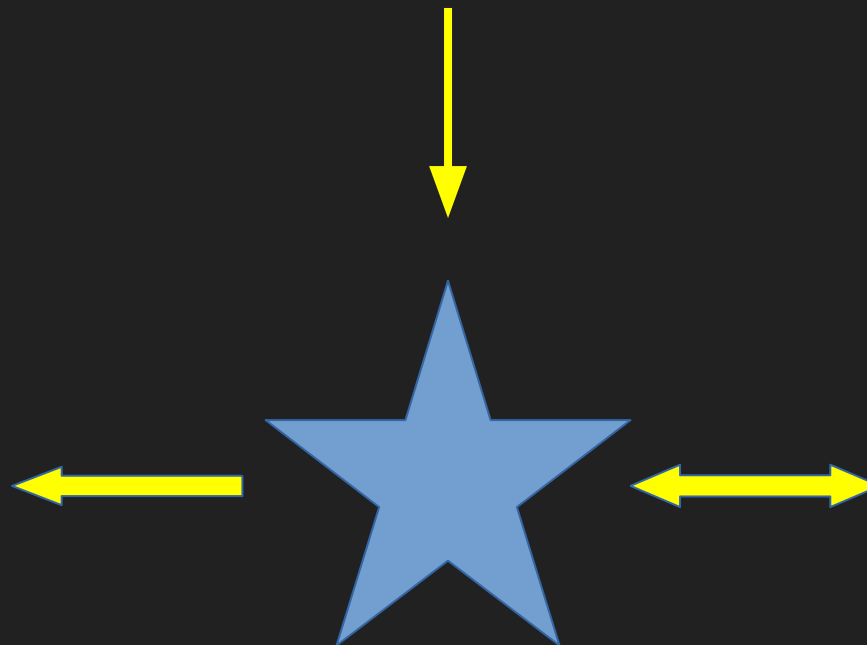


Contributors

Users



Superusers



**Successful
projects have
contributors,
super users,
and users.**

Community manager common knowledge:
Contributors are former or current users

Defining success

What are metrics we can use to define success?

Quantitative Metrics

Number of users

Number of contributors

Turnaround time on requests

Money

Qualitative Metrics

Functionality

Ability to create features

Ability to offer support

Happiness and usability

Add your own!

Some successful projects

Debian

Firefox

Git

GPG

LibreOffice

MediaWiki

Outreachy

Public Labs

R

Signal

Ubuntu

Wordpress

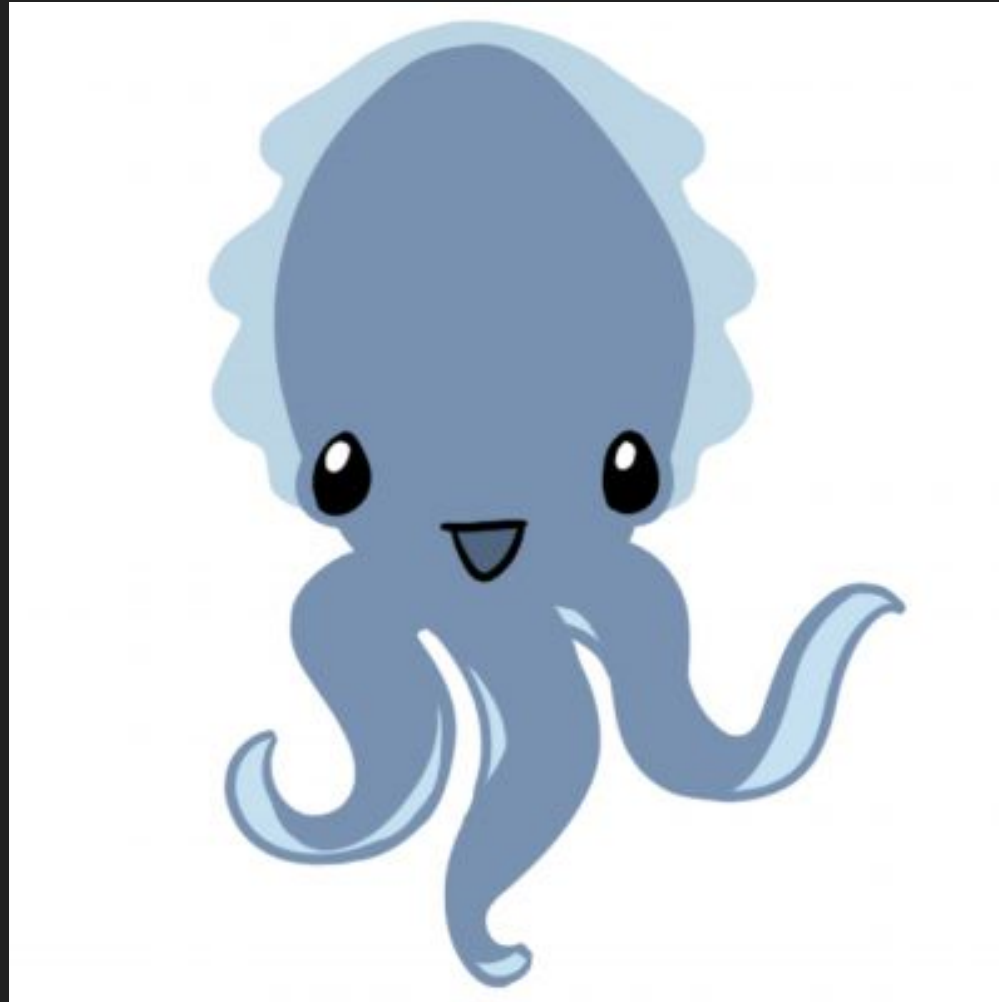
**Projects (and contributors)
are driven by
different
goals.**

job

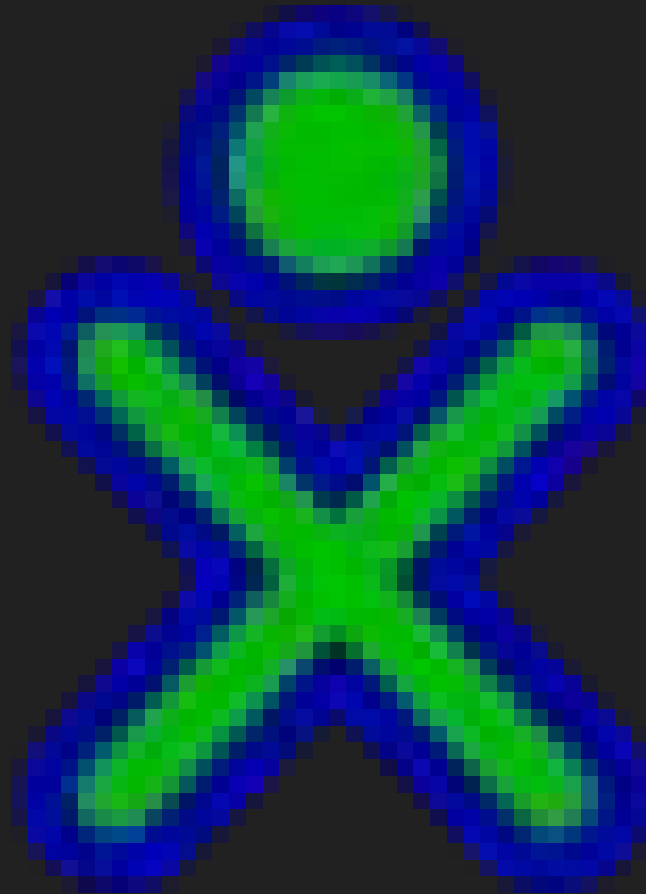


redhat.®

hobby



altruism



“doing it right”



fixing a problem
“self interest”



**There are
many
possible
organizational
models.**

An incomplete list of possible leadership structures

Benevolent Dictator
for Life

Committee (elected)

.....(appointed)

.....(earned)

Do-ocracy

Project leader (elected)

.....(appointed)

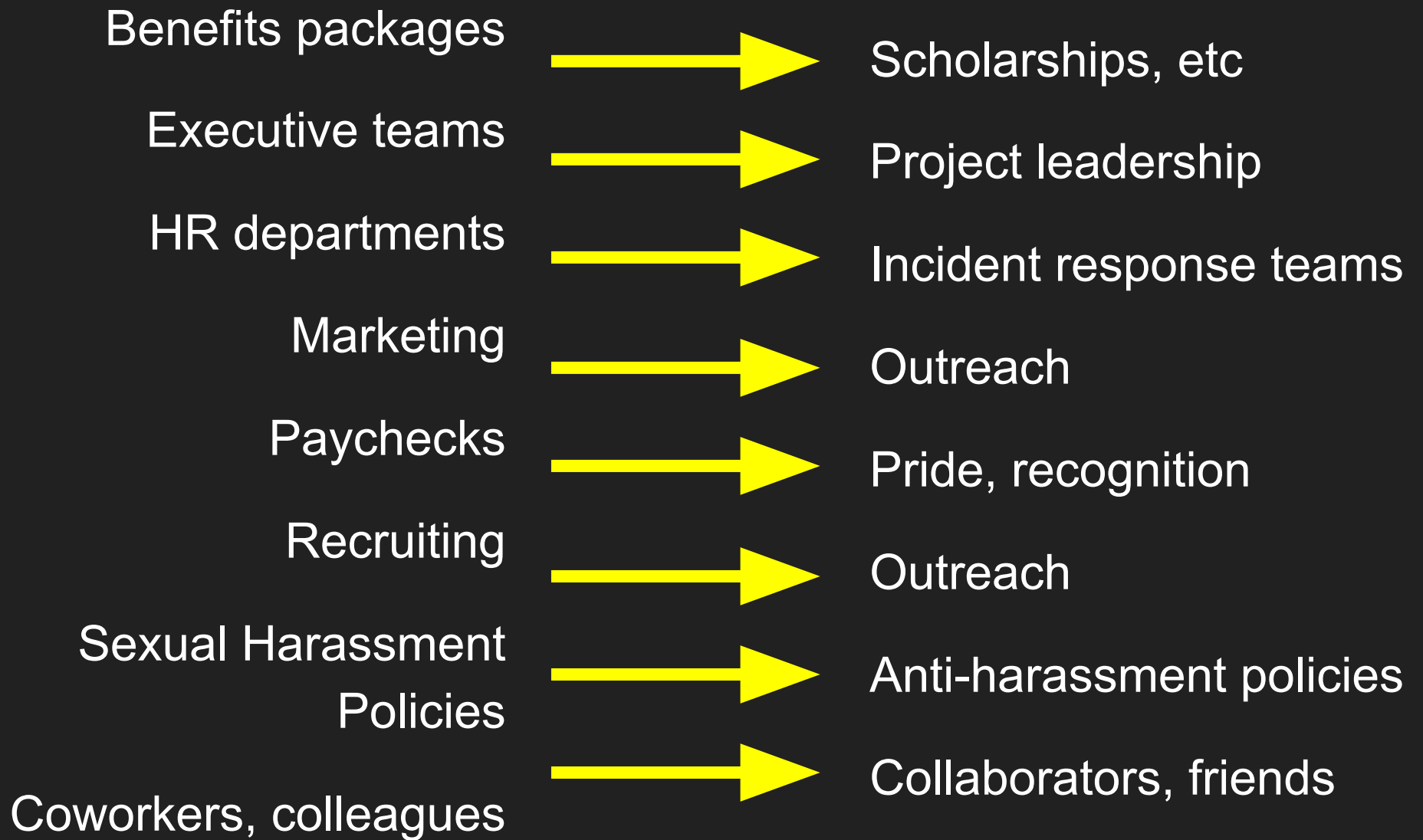
.....(earned)

Open contribution projects are not companies, but they face many of the same challenges and issues faced by companies.

Technology projects rarely view their problems as being the same as those of a company. Because of this, they try to replace the wheel with something completely different, but usually end up just making a wheel.

There's a conversation we're not going to have right now about the role forking a project has in the context of politics and power within a project.

**Some other
things people
do in
communities
and companies**



The power of community and
social belonging

Shared terminology:

- conference (happy hour, meetup, party)
- GSoC
- hackathons
- mentors
- sprints



Two Very Different Case Studies



Some details about Mixxx

2002 - Project started

2006 - First non-founding lead developer

2008 - RJ Ryan joined as a Google

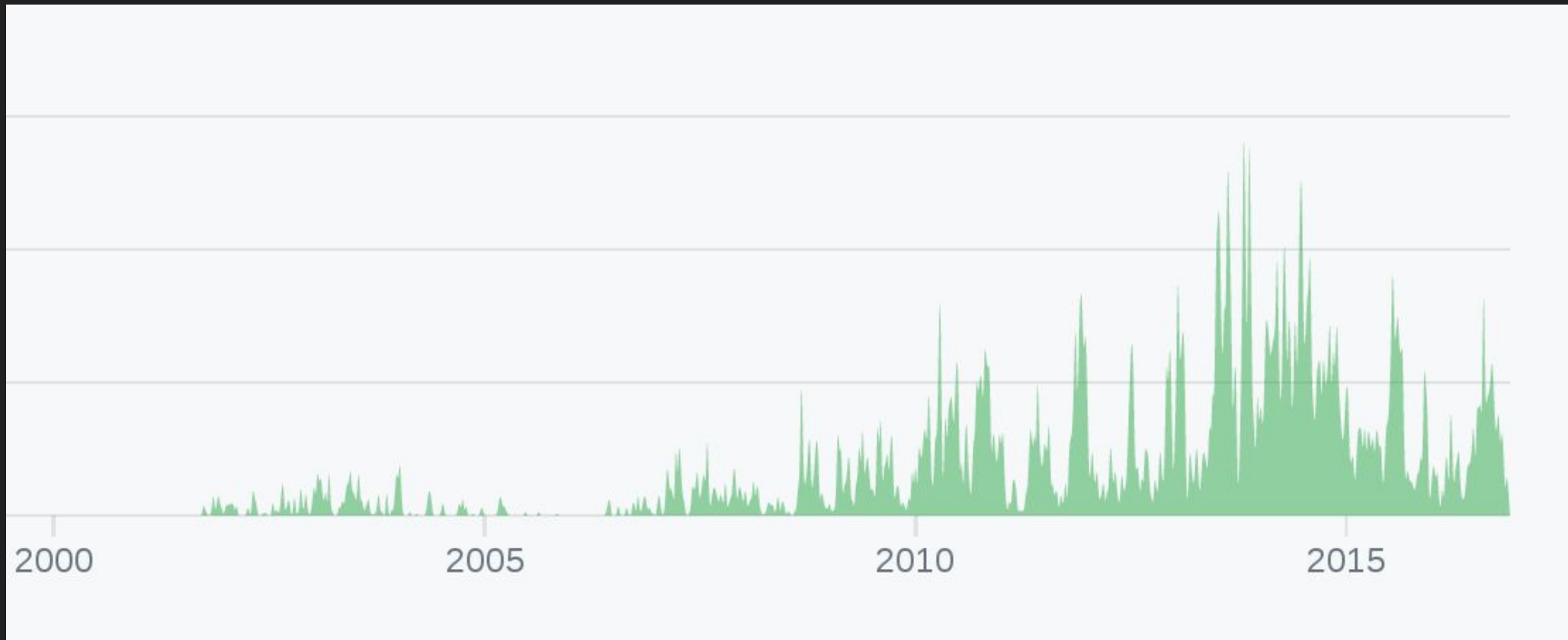
Summer of Code intern

2011 - Ryan became lead developer

2011 - Mixxx released

2015 - Mixxx 2.0 released

Mixxx is made by contributors and GSoC interns



OPENedX

Some details about Open edX

2012 - edX project announced

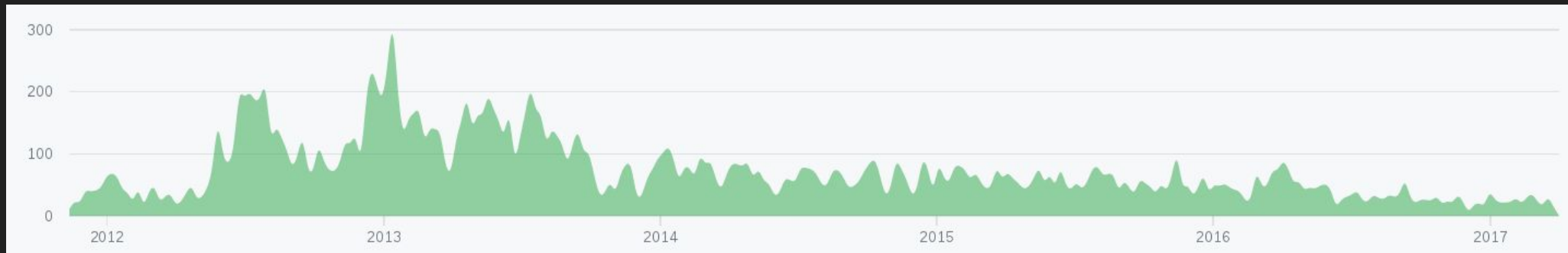
2013 - with Stanford, codebase released

2014 - first conference

2016 - XuetangX announces 5 million users

2017 - edX announces 10 million users

Open edX is made by people getting paid to work on Open edX



Mixxx is a community run and driven project.

Open edX is driven and managed by a central organization.

They have a lot of the same problems.

There are more
people who want to
help than there is
capacity.

A
Technical
Example

How a technical contribution is made

Development Cycle in a Nutshell



A contribution requires:

- Idea
- Execution
- Submission
- Review
- Merge

There just aren't enough people.

Example: Signal and Contacts



Welcome to Signal Desktop

Privacy is possible. Signal makes it easy.

[GET STARTED](#)



How to become a free software leader

Step 1. Show up

“I wanna be in the room where it happens.” - A. Burr

Know how to get there (and where
to go)

Learn the spoken (and unspoken)
rules of social behavior

How to become a free software leader

Step 2. Work

Free software contribution is not a meritocracy.

Stacking chairs and doing math.



Publicize the work you're doing.

“I ask people if they need
someone to interview.”

-Karen Sandler

Make friends and social
connections.

“It’s not luck. It’s having good social connections.”

-Dr. Rebecca Norris-Bell

How to become a free software leader

Step 3. Step up

Someone else will probably
suggest it first.

Leadership Roles

- Being the person who orders pizza
 - Core development, project maintenance, etc
 - Giving talks
 - Joining a Board
 - “Owning” a part of a project
 - Running a meeting
 - Running a meetup
-

Implicit and explicit examples of leadership:

- Big personalities (Richard Stallman)
- Docker “top core maintainers”
- Open Source Initiative Board
- Project originators who maintain power (Linus Torvalds; Guido van Rossum)

It's going to involve a lot more
chair stacking (and a lot more
math)

**These are all
the same
things you
need to do in
business.**

Free software communities come up with similar (and sometimes better) policies for taking on leadership roles.

Explicit rules and common
knowledge

People love clear paths to
advancement.

In private companies, it's a lot fuzzier.

This is actually also true for many projects.

Example: Debian and Debian Developers



Example: Debian and Debian Developers

- Checklist - required steps for Applicants
 - Step 1: Application
 - Step 2: Identification
 - Step 3: Philosophy and Procedures
 - Step 4: Tasks and Skills
 - Step 5: Recommendation
 - Step 6: Front Desk Check
 - Step 7: Debian Account Manager check and account creation
- Entry Form

Bonus round:

Why you should become a
free software leader

"[The] apparent lack of structure
too often disguised an informal,
unacknowledged and
unaccountable leadership that
was all the more pernicious
because its very existence was
denied."

-Jo Freeman

Having your way

Respect from your peers

Professional advancement

Impressing people at parties and
in airports

Your parents (and family and friends) will be proud.

It's fun.

The community benefits from
good leadership.

Questions?

Molly de Blanc
@mmillions
molly@fsf.org