Open Source Lessons from the TODO Group

http://todogroup.org
AGENDA

- Talk Openly, Develop Openly: Why TODO? (3 minutes)
- Lessons from open source programs (25 minutes)
  - Netflix
  - Yahoo
  - Microsoft
  - Capital One
  - Box
  - Sandisk
  - Google
- Concluding Thoughts (2 minutes)
- Panel / Q&A (10 minutes)
TALK CONTRIBUTORS

- Chris Aniszczyk (Linux Foundation)
- Chris DiBona (Google)
- Nithya Ruff (SanDisk)
- Jeff McAffer (Microsoft)
- Andrew Spyker (Netflix)
- Benjamin VanEvery (Box)
- Jim Jagielski (Capital One)
- Gil Yehuda (Yahoo)
What is the TODO Group?

- TODO is an open group of companies who want to collaborate on practices, tools, and other ways to run successful and effective open source offices, projects and programs.

- TODO Group became an official Linux Foundation Collaborative Project:
  - [http://todogroup.org/blog/todo-becomes-lf-collaborative-project/](http://todogroup.org/blog/todo-becomes-lf-collaborative-project/)
  - [https://github.com/todogroup](https://github.com/todogroup)
  - [https://twitter.com/todogroup](https://twitter.com/todogroup)
We all agree:
• Open Source is vital to the way technology is developed
• Our involvement in Open Source is beneficial to our company, our engineers, and our tech-communities
• Companies need some way to manage their involvement with Open Source development
• Sharing our practices with each other helps all of us
• Sharing our practices with others helps improve the state of Open Source as a whole

We differ:
• How we manage the details (different companies have different approaches, needs, staffing levels, expertise, and risk profiles)
Netflix Open Source – “Freedom and Responsibility”

- Anyone can open source: [https://netflix.github.io/](https://netflix.github.io/)

Common help on

- Build and publishing
- Github team/contributor management
- Security scanning
- Legal guidance
- All from others who have open sourced

Monthly Internal Forum, Quarterly Meetups
Distributed leadership aligned with engineering

Big Data, Build & Delivery Tools, Common Runtime Services & Libraries, Data Persistence, Insight, Reliability & Performance, Security, User Interface

Responsible for
- Consistency & Documentation
- Functional Area Health
Netflix: Automating Open Source Tracking

Visibility into health of github repos – OSS Tracker

Clearly indicate the state of projects

OSSMETADATA file

- Active, maintainance, archived
- Transitioned Asgard, ICE, Astynax
Without ZeroToDocker

• Documented technology that we expected you to assemble

With ZeroToDocker

• Running technology that we assembled and validated
• Up and running in minutes

Not Production Ready

• Examples only
• Not run this way at Netflix
• Security, HA, monitoring, etc. ignored
Microsoft: Scaling Open Source
Microsoft: Scaling Open Source

Business models across 100s of teams
- Playbooks that help people think about the true value we deliver
- Everything else supports this

Policies and processes for 100s of teams
- Simplify, unify, minimize
- Focus on the key, opportunities and risks -- tool the heck out of them

Culture and mindset for 10,000s of engineers
- Execs get it, new kids already doing it. Help “Middle management”
- Recognition models

https://microsoft.github.io/
Microsoft: Scaling Open Source

Tools for 1000s of users and repos
- Automation, automation, automation
- If you care about it, get it in the engineering system, else forget about it
- GitHub is not enough

Data for 10,000s of projects
- Measure the heck out of it
- Project health (producer and consumer), adoption, time saved, ...

Support for 100s of communities
- Enable teams to build awesome communities (tools, metrics, CoCs, ...)
- Keep it personal

https://microsoft.github.io/
Capital One: Our Open Source Journey

- It Starts with awareness and acknowledgement: We Are All Tech Companies
- It Continues with a Culture Shift: Communication and Transparency
- It Survives with Buy-In
  - Easiest getting Developer buy-in
  - Engineering Management is harder
  - Legal and Security is hardest
- Leverage Inner-Source to complement Open Source strategy
- Education is Key: Low-level shared understanding
- Find, and empower, internal Champions
- Wash – Rinse – Repeat
Box: Starting an Open Source Program

At Box, we rely on open source software every day. That's why we give back to the open source community whenever possible, by contributing code to outside projects and sharing projects that we've developed internally.

http://opensource.box.com/
Grass Roots?
Box: Starting an Open Source Program

Good Intentions
Box: Starting an Open Source Program

Straight Outta Compton
Box: Starting an Open Source Program

Who Cares?
Box: Starting an Open Source Program

Community
SanDisk’s Open Source Strategy

Journey
- Denial
- Consumption
- Compliance
- Contribution & Engagement

OS Office
- CTO Office
- Central Coordination
- Company-wide Working Group
- To share Common issues

Communicate
Consume
Contribute
Comply
Create Competency
Collaborate
Sandisk: Open Source Lessons Learned

In Place for 2 Years, Moved from Denial to Contribution

Lessons Learnt
• Start with believers and work with sceptics through success stories and case studies
• Don’t forget the internal audience. Be relentless in internal education and evangelizing
• Make sure the business side understands why open source and is on your side
• Engineering managers hardest nut to crack; to make time for open source contributions; consumption is never a problem.
• Have Internal Open Source Days and invite your ToDo colleagues to speak at the event. Huge support for your work with management and leaders.
• Best location for this work – CTO or Engineering
First patches started over 17 years ago
License compliance and code ‘tidyness’
Thousands of patchers for decades.
Major releases ongoing: Android, Chromium, Go, Angular, Polymer, Tensorflow, etc...
Every kind of ‘governance’ ranging from
  • Punctuated Equilibrium
  • Online everything
  • Minor project releases ‘greasing’ the wheels of release (>3000 projects)
  • Platonic projects like Gerrit
  • Ongoing leadership in Git, Samba, etc..
  • Spin outs of leadership to Apache, LF/Cloud Native Foundation (k8s)

What do we think is more important for all of us to do now?
  • Licenses are starting to fail us all. How?
Yahoo: Lessons Learned #1 Improving Engineers

- Open Source Project Offices help improve your engineering talent. Engineers who interact with open source, with our help, become better engineers, write better code, work better with others.

- Why? Because the Open Source Program Office helps engineers with
  - Working with developer communities (including those from other companies, which can get tricky)
  - Copyright and license issues (something often not taught in school)
  - How to be an employee representing the company as well as an individual in the public sphere.

http://yahoo.github.io/
How do we know which projects to publish? Which will succeed?

The truth: we don't.

- We’ve published industry-changing project (like Hadoop) and projects we thought could be as big, but were not.
- You can help projects thrive by writing clear documentation, addressing issues and PRs, and working with the community in the spirit of open problem solving.
- Best bet: ask members of the target community for input and feedback before publishing code.

http://yahoo.github.io/
Yahoo: Lessons Learned #3 Strategic Benefits of OSPO

• So why run an Open Source Program Office? For the Strategic Benefits.
  • We attract and retain top talent. a.k.a. Those who seek to work on open source projects.
  • We influence important projects that we depend upon
  • We can actively manage tech investment portfolio (what we build vs. what we leverage from others)
  • We avoid legal and security trouble by having a program that responds to issues as they arise, and proactively abates them.
  • We navigate the changing technology landscapes with agility, and avoid tech debt inherent in closed source code projects.

http://yahoo.github.io/
Conclusion / Joining the TODO Group

• Sharing is caring!

• If you’re bootstrapping or running an open source program you’re welcome to join!

• Read and sign the charter!
  • [https://github.com/todogroup/charter](https://github.com/todogroup/charter)
Q&A / Thank you

http://todogroup.org