» The Linux Foundation

Open Sourcing Proprietary Technology Made Simple

A High Level Roadmap for Open Sourcing Your Code

January 2012

By Ibrahim Haddad (Ph.D.) and Brian Warner, The Linux Foundation

A Publication By The Linux Foundation
http://www.linuxfoundation.org

As corporate participation in open source grows, many companies are discovering the advantages of releasing formerly proprietary technologies as open source. This article introduces a basic process and checklist that may be followed when creating a new open source project from proprietary code, to ensure a successful release.

Introduction

Corporate participation in open source has reached an all time high, and continues to grow as companies realize the value of consuming and contributing to open source projects. The nature of corporate participation continues to evolve as well, as companies increasingly discover that open sourcing proprietary technologies can lead to new sources of value and stronger product ecosystems.

For companies that plan to open source proprietary code as a standalone open source project, this paper offers a high level overview of the process and provides a sample checklist that can help ensure that all tasks are properly captured and executed.

This is the fourth paper in a series by The Linux Foundation on corporate participation in open source. Previous papers on similar topics include "Understanding the Open Source Development Model", "Establishing an Open Source Software Strategy", and "A Guide to Upstreaming." These papers and others on various Linux and Open Source topics are available for free download from http://www.linuxfoundation.org/publications.

A High Level Process for Open Sourcing Proprietary Code

Open sourcing a proprietary technology involves far more than just making the source code available. There are many possible ways of building or joining communities that will use and help maintain the project, which is why it should be a well-ordered and deliberate process. Figure 1 illustrates a basic process for open sourcing proprietary technology. In the following sub-sections, we discuss the various tasks related to each of the activities shown in this figure.

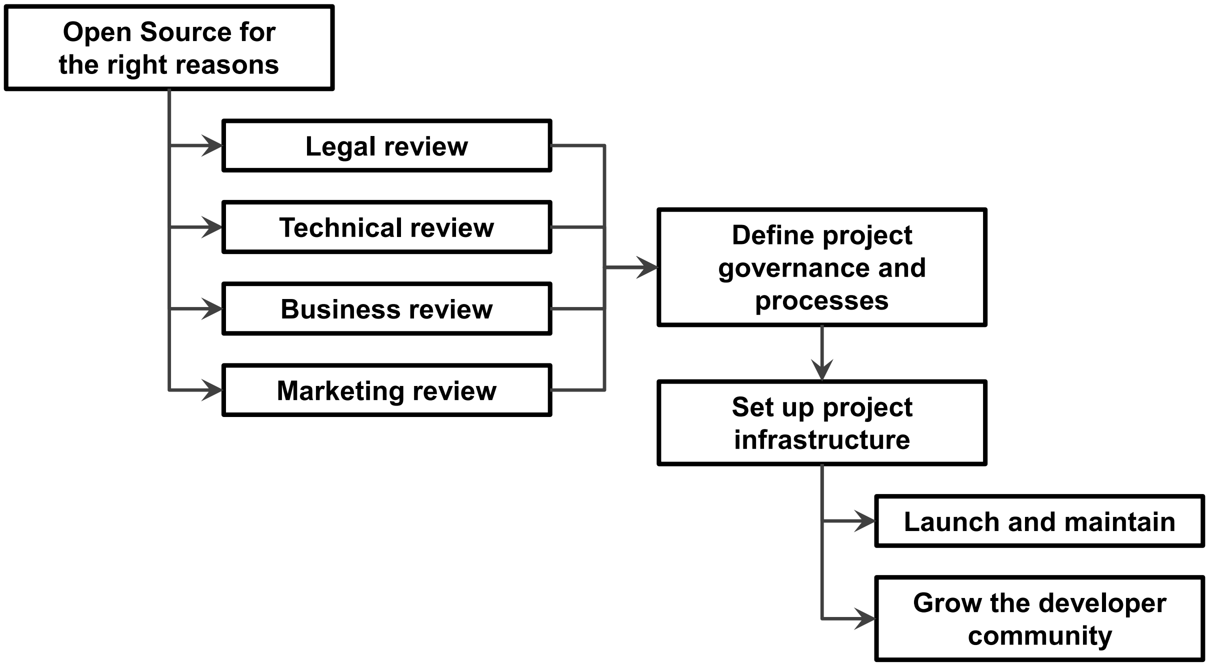
Open Source for the right reasons
Legal review
Technical review
Business review
Marketing review
Define project governance and processes
Set up project infrastructure
Launch and maintain
Grow the developer community

Figure 1: A basic, high-level model for open sourcing proprietary technology.

Open Source for the right reasons

When open sourcing proprietary technology, it is important to thoroughly evaluate the reasons for the transition, and align internal incentives and metrics accordingly. Open sourcing for the wrong reasons could have the opposite effect than is originally intended. Consider additional alternatives before creating a completely new project, such as joining an existing open source project, or collaborating with current business partners to jointly create a new open source project.

Essential tasks to cover in this early stage include:

- □ Evaluate possibility of joining an existing open source project
- □ Evaluate the company's ability to launch and maintain the project using the open source model
- □ Evaluate the likelihood that other companies may join the project from the start
- □ Evaluate success factors and set appropriate metrics for the open source project

Internal Preparations

When preparing to open source internal code, there are four main reviews that should happen concurrently: legal, technical, business and marketing.
The legal review ensures that intellectual property contained in the code will be released in a clean and orderly process. It should verify that the company has the right to release all of the code, and should include trademark due diligence and registration.

Essential tasks during the legal review include:

- □ Consider the impact of open sourcing on your company's intellectual property
- □ Ensure full compliance with open source licenses

❑ Select an Open Source license for the source code to be released
❑ Decide if you need a Contributor Agreement
❑ Decide on any trademark related considerations

The technical review verifies that the source code can function without dependencies on other internal code or development practices, and that it does not include 3rd party code the company cannot release. It should include verification of all license and copyright notices, and private code comments should be scrubbed.

❑ Remove critical dependencies on non-public components
❑ Provide documentation and use case examples
❑ Remove internal comments, references to other internal code, etc.
❑ Ensure coding style is consistent
❑ Update copyright notices in source code files
❑ Add license notice in source code files
❑ Add license text as a file in the root directory

The business review ensures that the project has sufficient funding and internal sponsorship, at least until it becomes self sustaining.

Essential tasks during the business review include:

❑ Ensure there is a corporate champion for the project
❑ Ensure there is a commitment for resources (maintainers and developers)
❑ Ensure there is a commitment for funding the open source project infrastructure

The marketing review establishes guidelines for branding. This is particularly important, as it helps to ensure a consistent message in the market.

Essential tasks during the marketing review include:

❑ Design project logo, color scheme, website, collateral, etc.
❑ Formalize branding guidelines
❑ Register social media accounts for the project (Twitter, Facebook, LinkedIn, etc.)
❑ Register domain names for the project

Define Project Governance and Processes

Governance is the process by which the project makes decisions regarding strategy, releases, direction, and development priorities. Decision making should be public and open, to help ensure that all participants are aware of changes to the project and to maintain transparency. It is important to decide early in the process what criteria must be met to participate in the project governance body.

At this stage, decisions should be formalized on how features and bugs will be tracked, how code will be submitted, and who will manage the release process.

❑ Decide on project governance
❑ Formalize processes for submitting code, patches, feature ideas, etc.
❑ Establish release management team and process

Set Up Project Infrastructure

The open source community has developed and broadly adopted a set of high quality source code management and distributed collaboration tools. When creating a new project, it is important to use these tools, as they are free for contributors and are well understood by developers.

❑ Code repository system
❑ Bug tracking system
❑ Website
❑ Mailing lists
❑ Wiki for collaborative documentation
❑ IRC channel for live discussions
❑ Automated build environment, if applicable

Launch and Maintain

Work with key business partners, and brief related open source projects to ensure strong support when the project officially launches. Ensure that the project's infrastructure, source code, governance model, license information, and initial internal development teams are all finalized prior to launch.

When the project is announced, anticipate that a significant number of open source developers will download the code and form first (and lasting) impressions about the project.

Essential tasks prior to launch:

❑ Pre-brief launch partners
❑ Ensure that all project infrastructure is running, secure, and scalable
❑ Ensure internal developers join and continually monitor IRC channel, mailing lists, etc.
❑ Release source code
❑ Follow the open source development model

Grow the Developer Community

After the project has launched, it is essential to monitor the vitality of the external community. Community building does not happen automatically. In the early stages of the project, it may be necessary to host developer events or sponsor meetups at major conferences to build momentum. In addition, it is extremely important to manage expectations and fulfill obligations for project governance and transparency.

Essential ongoing activities:

❑ Designate a community manager or community advocate
❑ Ensure any changes to direction or governance are clearly communicated
❑ Follow best practices of other similar communities
❑ Encourage and provide opportunities for face-to-face community building

Conclusion

There are many ways to successfully open source proprietary technology. This checklist provides a high level overview of the process, and can be used as a base for a more detailed internal plan of the process. While this process may seem complex, many companies have successfully followed similar procedures to bring internal code to market as an open source project. For more information on creating successful open source projects and working with open source communities, please visit The Linux Foundation.

Linux Foundation Resources

Linux Training

The Linux Foundation offers two training courses to enable organizations effectively work with open source developers:

- **LF 205: How to Participate in the Linux Community:** Working with the kernel development community is not particularly hard, but it does require an understanding of how that community works. This course is intended to bring attendees up to speed quickly on how kernel development is done and how to be a part of the process with a minimum of pain and frustration.
- **LF 271: Practical Guide to Open Source Development:** This course prepares organizations to maximize their effectiveness and shorten the time to value when participating in open source development projects. This course builds upon years of best practices and extensive experience in commercial participation in open source projects to help organizations approach the open development model in a structured and methodical manner, maximizing the likelihood of success. The course provides extensive examples from the Linux kernel community, and includes specific best practices for working with upstream.

Linux Foundation Labs

If you have a collaborative software project you need hosted at a neutral party, the Linux Foundation may be able to help. The Linux Foundation assists companies and communities by hosting collaborative software projects. The Linux Foundation provides three main services to Lab projects:

- The technical, operational and legal infrastructure so that project leaders can focus on technological innovation.
- Guidance and consulting on open source best practices gleaned from the two decades of experience of Linux and the ability to collaborate and network with the large and growing Linux Foundation community.
- By providing these services to companies and developers, the Linux Foundation provides a much needed framework for advancing and accelerating technology that allows project hosts to focus on innovation.

There are two main criteria that must be met in order for the Linux Foundation to host a lab project:

- Use of open source governance best practices including license and contribution agreement choices in keeping with the ideals of Linux
- Project must either use Linux or have the potential to enhance the Linux ecosystem

If you have a project that may fit this criteria, please contact us:

http://www.linuxfoundation.org/labs.

Open Compliance Program

The Linux Foundation's Open Compliance Program was established to boost adoption of Linux and other open source by making license compliance ever-easier to achieve, to increase awareness and understanding of open source compliance responsibilities, and to make available free resources that can help companies establish their compliance programs. The program offers comprehensive training, compliance educational materials (white papers, compliance blog, webinars), compliance tools, an online compliance community (FOSSBazaar), a best practices checklist, a rapid alert directory of company compliance officers, and SPDX^{TM}, a standard to help companies uniformly tag and report software used in their products.

Events

The Linux Foundation produces a number of technical events around the world that provide a venue to bring together developers to solve problems in a real-time environment.

Publications

The Linux Foundation produces a wide range of publications that are available for free download. These publications are divided into three categories: Open Source Compliance, Workgroups (such as Tizen, OpenMAMA, LSB, SPDX, FOSSology, etc.) and Community. The Linux Foundation publications are available from http://www.linuxfoundation.org/publications.

About the Authors

Ibrahim Haddad, Ph.D., is the Director of Technology and Alliances at The Linux Foundation and Contributing Editor for the Linux Journal.

Brian Warner is Operations Manager at the Linux Foundation.

The Linux Foundation promotes, protects, and advances Linux by providing unified resources and services needed for open source to successfully compete with closed platforms.

To learn more about The Linux Foundation, or any of our other initiatives please visit us at http://www.linuxfoundation.org/.

THE
LINUX
FOUNDATION