A Practical Guide to Implementing a Third-Party Compliance Program for Open Source and Commercially Licensed Software Components

Helping enterprise software companies prevent lawsuits, audits, and loss of ownership

www.EntenteSoftware.com
Introduction

In order to decrease time to market and leverage valuable engineering resources, many software companies include third-party open source and commercial software (“Components”) in their products.

Prudent use of Components can provide substantial benefits for software companies. However, unmanaged use can result in audits, law suits, stop- ships, and possible loss of ownership of software products.

Open Source as a New Concern

Prior to the rapid growth of open source software, Component management was focused on compliance with commercial vendors. For commercial Components, the key concern is compliance with the terms under which the Components are licensed to avoid unexpected royalty and legal settlement expenses.

Open source Component management presents a different set of business concerns. Although open source Components are typically obtained without cost, there are a wide variety of license terms under which the Components are made available. Unmanaged use of open source Components can have unexpected results, including a requirement that your proprietary software be made publically available in source code form, at no cost.

Although the business issues involved in the management of commercial and open source Components are different, the business processes are fundamentally the same and both require a small but consistent investment in people, education, process and tools.

Personnel Requirements

Component management requires the assignment of existing personnel to incremental roles:

Compliance Program Director (CPD): This individual sets policy for Component compliance across the company, working cross functionally with in-house Engineering, Product Management, Legal and Finance. It is important that the CPD be motivated and capable of grasping key issues related to Component licensing and compliance.

Product Compliance Managers (PCMs): These individuals implement the CPD-established policy within their respective products groups. The ideal PCMs are engineering team leads, high skill level QA analysts, or technically savvy product managers. Regardless of the group that a PCM is selected from, they must have the respect and cooperation of their engineering product group.

Legal Advisor: This individual, typically an intellectual property or contract attorney, interprets commercial and open source licenses and agreements that are coordinated by the CPD. In order to assure consistency in the presentation of issues, the Legal Advisor works primarily with the CPD rather than individual PCMs. The ideal Legal Advisor should be an attorney with experience in software licensing. Experience with open source licensing is a plus, but this is currently a rare skill, typically obtained “on the job”.

Cross-Functional Support: Although additional individuals from Finance, Support, Marketing, etc. will be involved in certain Component licensing issues and decisions, their levels of involvement will be minor, and they will typically be associated with processes already in place for approvals in other functional areas. No special training is required in these roles.
Education

The personnel investment in Component compliance must be supplemented with basic education. Since there is no curriculum for Component management, independent efforts are required, especially by the CPD, to grasp key issues. A basic understanding can be obtained by reviewing the company’s existing Component license agreements with the assistance of the Legal Advisor.

The licensing portion of commercial Component agreements is typically in the “License Grants” section. License grants vary considerably by vendor, by your use of source or binary versions of Components, and by the extent to which you will incorporate, modify and distribute the vendor product. Negotiated commercial agreements, or “custom agreements”, may contain additional license terms and pricing information in the form of Exhibits after the signature page of the agreement.

For open source licensing, a variety of resources are available including books, Frequently Asked Questions documents, and various websites (see Appendix A). In the end, it is the combination of these sources, plus consultations with Engineering and the Legal Advisor, that is required for informed decision making.

Process

The general process required to get the full value out of your investment in Components is straightforward: PCMs gather and review pertinent information from the engineering team and forwards to the CPD; the CPD adds details and perspective based on a portfolio view of all Components in the company and works with the Legal Advisor and other cross-functional individuals to obtain an “approved” or “unapproved” recommendation for the proposed Components. A sample process chart for the review of Components is available in Appendix B.

Timeliness Counts

A key element to success is early visibility to commercial and open source components that will be included in a company’s products. Early access gives the CPD the opportunity to review the use case in detail with the PCM or engineering team, and to present a succinct use case to the Legal Advisor. Ideally, this is done prior to any substantial investment in the Components by Engineering.

Commercial components that are licensed via custom agreements are a special case – these agreements must be addressed especially early in the review cycle to avoid the risk of an essential component being unavailable. Conversely, open source technology that is licensed via permissive licenses can be reviewed later in the review cycle without substantial risk. The best practice is to have multiple reviews and a final cut off date for inclusion of inbound technologies in a product.
The toolset used to track and manage Components is both straightforward and crucial. In the absence of a good toolset, Component reviews and tracking are done ad hoc, typically with inconsistent information provided in a variety of formats and maintained in multiple locations with different update schedules. Information maintained in this fashion is of little value when making portfolio-wide Component decisions or when a company encounters licensing issues.

To resolve these issues, two key elements are required in the toolset: (1) a Component information form that documents key characteristics and use case information related to the proposed Components, and (2) a broadly accessible, database-driven repository that tracks, supplements and displays this information in a company-wide portfolio view.

Component Information Form

Providing the information below, typically by the engineering team, starts the review process for a particular Component. Although full provision of the information will take up to 15 minutes of engineering time, experience has shown that early access and review of this information will result in considerable downstream time savings by having key Component issues dealt with early in the review process rather than as a gating item to the product release schedule.

The exact format of the form used to gather this information is arbitrary. The form you create is designed to start the review process rather than to answer all the questions that will come up throughout the review process. Listed below are key items to be included in early information gathering.

- Initial review date and name of individual providing the information
- Name of Components and associated in-house product (include version, release and patch)
- Name of Product Compliance Manager (PCM) for the in-house product and next product release date
- Name of Components provider and type of Components (i.e., Commercial or Open Source)
- Web addresses of Components homepage and download page
- Commercial License Information
  - Attach the agreement AND the redistributables list
  - Verify that only listed redistributables are redistributed
  - Provide backup showing that the Component and redistributables are covered by the named license
- Open Source License Information
  - Name of the open source license (include version of the license (e.g. LGPL 2.1)
  - Provide a link, located on the open source project page, to the actual license
  - If applicable, include a link to GPL exceptions associated with the open source license
- Use Case Information
  - General: what parts of the Components (source and binary) will be included in the in-house product?
  - Modification: will the code be modified, and if so, how?
  - Link Type: static link, dynamic link or inline embed?
  - Platform Type: how will the in-house product be delivered (physical, download, SaaS, cloud)
  - What other in-house product groups or outside companies will use the associated component?
Database-Driven Repository

Your repository of Components must be sufficiently flexible to allow the CPD, PCM, Legal Advisor, and engineering team to easily input key information, and to retrieve it in meaningful ways. Individual spreadsheets, Word documents and file lists will not accomplish this. In order to avoid the liabilities associated with Components use, and to experience the upside, you must invest in a repository that will allow both individual product and portfolio views of your Components.

Although there is a temptation to write an in-house application to address these issues, nearly a decade of consulting in this area has shown that companies spend far more time and money developing and maintaining such an application than is merited by the value provided – a straightforward, hosted application such as Entente™ IPCompass® will do the tracking and let you invest your time in the quality of the data.

Regardless of your choice to buy or build the tracking application, it is essential that the application provide the following functionality:

Programmatic Tracking: The application must have the capability to programatically track multiple levels of Component embeds so as to accurately determine the content of any product or bundle, even if the product or bundle is put together as a series of tiered products and Components. Without this capability, every occurrence of a Component in a product must be individually input and tracked. It is essential that each Component be input only once, at the appropriate detail level, and that the application manage the various relationships without further input from the user. This allows a building-block approach that tracks various SDKs and other design components that are shared across multiple products or among multiple engineering groups. In the absence of such functionality, the engineering efforts required to sustain the program will increase substantially and key license relationships will likely be missed, thus introducing risk.

Role-based Views: The application must also provide consistent views across the company’s products that include key information, grouped by role. Key roles include Compliance Program Director, Product Compliance Manager, Legal Advisor and Financial Reviewer. Depending on the company and its products, other role-based views, such as a corporate architect, may be required. Each view consolidates key information in a fashion relevant to the individuals doing the review. Role-based input screens ensure that the information being captured is both accurate and all-inclusive.

License-centric & Product-centric Views: Managing Component compliance and costs requires various views for Component license rights, expiration dates, payment due dates, and other licensing information. These views assist in controlling costs and in preventing the unwanted consequences of out-of-compliance Component use. The product-centric view provides a portfolio view of Components that is most utilized in the company’s products, and hence most likely to provide leverage, require infrastructure investment, or benefit from contract renegotiation for volume price agreements.
Implementing a Component Compliance Program at Your Company

While the benefits and liabilities related to Components are well documented, there remain substantial differences in opinion regarding the appropriate efforts and resources to manage these issues. Typically companies that have experienced the business disruptions associated with audits, law suits and stop-ships are very favorably disposed toward investments in this area, while others who have not yet experienced these hardships are less committed.

Successful implementation of a compliance program has three key characteristics: (1) Executive Support, (2) Balanced Investment, and (3) a “Go Forward Clean” approach.

Executive Support: In order to successfully implement even a modest compliance program, it is essential that you have the support of the head of Engineering and an attorney at your company who reviews license agreements, preferably technology licenses. If, after outlining the benefits and liabilities of a compliance program, you do not have the support of these individuals, your probability of success will be low.

Balanced Approach: The largest personnel investment in a software company is its engineering team. Your approach to implementing a compliance program must take this into consideration. The program that you put in place must obtain product and licensing information while utilizing the minimum required Engineering resource, and utilizing that resource in a structured and predictable fashion that is in line with the development methodology and processes in place within your engineering organization. The compliance effort must accommodate Engineering, not the other way around.

“Go Forward Clean” Approach: Many companies have a large number of published software releases. Although a comprehensive approach to verify the Components present in all releases, new and old, could be argued for, the practical approach is to obtain information only for the current and strategic releases of each of your products, and then to “go forward clean” by rigorously following a standard process with each new release. In this fashion, the compliance effort is integrated into the product release process and focused on minimizing cost while protecting the value of new releases.

Summary

By utilizing the practices outlined in this report, companies will obtain the substantial benefits provided by Components while minimizing the associated risks. This practical approach leverages existing staff, and protects the substantial engineering investment of the company. A successfully deployed Component compliance program will reduce costs, reduce time to market, and provide an environment free of the disruptions caused by late-stage engineering rework, law suits, audits and stop-ship threats.

More to Come

Future topics related to Component management will be published by Entente Software in the future, including “Planning vs. Scanning” and “How to Substantially Reduce Commercial Component Costs”.

Contact

For more information, please contact Entente Software at info@ententesoftware.com, or visit us online at www.ententesoftware.com.
Appendix A – Open Source Educational Resources

**Note:** although there is a substantial amount of information available on the web with regards to open source licensing, much of it is simply incorrect. Entente Software recommends that the PCD have a good understanding of open source licensing based on the information below, but have the Legal Advisor review each unique license / use case combination.

**Books:**

**Frequently Asked Questions (FAQ):**
- Open source license FAQ: CPL, EPL, GNU, MPL, NPL, etc.

**Websites:**
- Debian License Information
- Fedora License Information
- FOSSBazaar (fossbazaar.org): License Compliance
- Open Source Initiative List of Licenses
- Wikipedia (wikipedia.org): Open Source, AGPL, Apache, BSD, EPL, GPL, LGPL, MIT, MPL, etc.
Appendix B – Process Flow for Approval/Rejection of Components

(1) Engineering completes the inbound technology form, or equivalent.

(2, 3) Engineering and Product Compliance Manager (PCM) iteratively review forms for accuracy and completeness

(4) The Compliance Program Director (CPD) reviews the form and adds additional information based on knowledge of the company’s entire inbound technology portfolio

(5) The CPD provides a preliminary “Approved” or “Unapproved”. Depending on company policy, a preliminary approval may allow engineering to make a minor investment in the inbound technology. An “Unapproved” at this point is typically due to an unacceptable license or vendor.

(6, 7) Commercial and Open Source agreements take different approval paths. For instance, Finance is generally not part of an Open Source review

(8) Legal Advisor approval is primarily determined by the rights granted in the license and the specific use case

(9) Finance approval is primarily determined by the cost to license the inbound technology, and the extent to which that cost can be tolerated as a percentage of the sale cost of the software product

(10) Final CPD approval is dependent on a variety of items. Company policy may require additional reviews from Marketing, Support, or an architecture committee. It is the responsibility of the CPD to cross-functionally guide the organization through the various steps