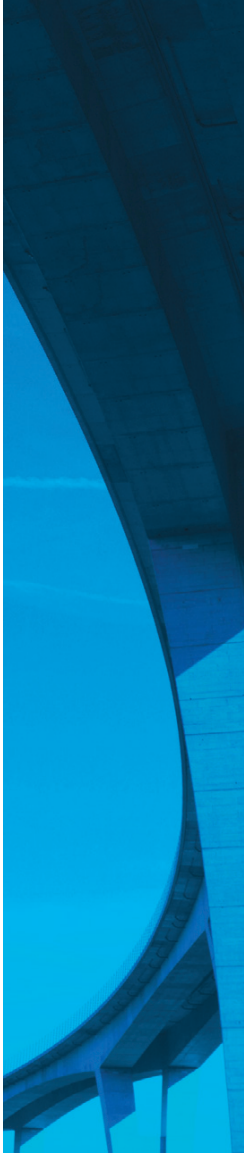
BLACKDUCK



FOUR STEPS TO CREATING AN EFFECTIVE OPEN SOURCE POLICY

by

Greg Olson

Senior Director, Open Source Management Practice

"Companies must have a policy for procuring OSS, deciding which applications will be supported by OSS, and identifying the intellectual property risk or supportability risk associated with using OSS. Once a policy is in place, then there must be a governance process to enforce it."

– Laurie Wurster, Research Director, Gartner

INTRODUCTION

WHY USE OPEN SOURCE SOFTWARE?

At this point in the evolution of the software industry, it has become difficult, if not impossible, to create any significant body of software without using at least some open source software (OSS). Open source has permeated our platforms and components, whether they are built in-house, provided by platform vendors, acquired from commercial software vendors or downloaded from the Internet. In addition to the prevalence of freely available code, there are compelling reasons to make use of open source software, including:

- The best-in-class software in some areas is OSS
- Products must interoperate with other OSS, e.g. Linux
- Customers favor, and sometimes even require OSS
- The ability to customize externally sourced software
- Lower cost alternatives to traditional commercial packages
- Faster time-to-market by avoiding development and testing of new code
- Lower development costs by using free, already debugged code
- Lower code maintenance costs by utilizing community maintenance
- Existing code base already contains significant OSS
- OSS came with a corporate acquisition

While most software managers are aware of the legal risks (e.g., license compliance with commercial strategies and additional code used, monitoring the use of code, etc.) and the operational risks (e.g., compatibility requirements, maintenance and support, integration concerns, among others) of using open source, the benefits far outweigh these concerns.

As such, creating an open source software policy is a key strategic imperative for organizations in the software industry.

WHAT IS AN OSS MANAGEMENT POLICY?

A policy is a set of rules and guidelines for using and managing OSS in your organization. To be effective, it must cover all the essential aspects of managing OSS, yet it must be succinct and easily understood; otherwise nobody will read it, much less follow it. At Black Duck Consulting, our rule of thumb is that policies should be five pages or less, and reflect the way software is developed and delivered in your company, specifically. General rules are not likely to apply directly to your requirements and processes, so the results are likely to be approximate and inefficient.

THE POLICY DEVELOPMENT PROCESS

Creating an open source policy is not a trivial matter. OSS management spans multiple areas of responsibility, and the requirements and points-of-view of each of these areas must be included and reconciled if the policy is to be successful. As such, Black Duck Consulting has identified four specific steps to creating an effective open source policy:

1. **Identify key stakeholders**
2. **Obtain an organizational commitment**
3. **Draft the policy**
4. **Review and approve the policy**

STEP 1: IDENTIFY KEY STAKEHOLDERS

In most organizations the important stakeholders represent the following functions:

- **Software architecture:** the role that specifies which elements are included in a software project
- **Software development:** the engineers who build the software
- **QA and/or release management:** those responsible for checking the quality and contents of project releases
- **Legal:** those responsible for upstream and downstream agreements and license compatibility evaluation
- **Product or line of business management:** the role responsible for the success of the software

Organizations with sensitive data may also have a security stakeholder responsible for the security of software entering the organization and being released.

STEP 2: OBTAIN AN ORGANIZATIONAL COMMITMENT

The single most important success factor is securing a commitment from the necessary stakeholders to contribute the time and effort to completing a policy. This may should obvious, but more than once we've heard executives say, "I would rather see my dentist than work on policy."

Policy development teams are much more effective when they work from a shared base of understanding. In order to solidify a commitment from stakeholders, it is wise to collect and disseminate information about your organization's use and plans for OSS. Such documents may include:

- Existing policies and processes related to OSS
- Inventories of OSS currently used within the organization
- Existing license compliance requirements and procedures
- Upstream and downstream agreements, and business relationships that involve OSS
- New initiatives that might involve the use of OSS

We have found that policy development teams are typically more effective when they work from a clearly articulated statement of the company's strategy for using OSS. Usually this is a statement of the benefits the company wishes to realize and how it intends to ensure them. If your company does not have an OSS strategy statement, that would be a good place for the policy development team to start.

STEP 3: DRAFT THE POLICY

The OSS policy is typically developed in a series of interactive meetings with participation of the relevant stakeholders in the organization. This is where the trade-offs inherent in any policy must be discussed and resolved in a way that best meets the organization's needs. Some common trade-offs that occur in the multiple elements of an OSS policy include:

- Controlling risk vs. development productivity
- Broad, simple rules vs. specific, more complex rules
- Self-checking vs. independent checking
- Use of judgment calls vs. detailed prescriptions

Many companies have found that using a facilitator with extensive experience in OSS policy and its operational implications can speed the working sessions to quicker results.

We recommend that open source policies include the following eight elements:

1) Program administration and management

These first crucial decisions determine who will be responsible for the policy itself and who will oversee the OSS management program. These could be individuals or a permanent version of the policy development team, usually called something like the Open Source Management Board.

Most companies define some additional roles, as well. The concept of an OSS component owner is very useful when defining policy for code management, support and maintenance and community interaction. The review and approval process typically requires a decision authority, usually called something like the *Open Source Review Board*.

It is good practice to state whether the policy is confidential or shareable, and how it will be published. Since training is a critical success factor for OSS management, it is advisable to define a training policy as well.

Remember: no policy goes unchanged for long. Most companies will go through several rapid revisions as they gain experience during the period of initial implementation. After that, it is a good idea to establish a periodic review and fine-tuning of the policy and OSS management processes.

2) Discovery, acquisition and evaluation

The greatest point of leverage occurs at the beginning of the OSS acquisition cycle. Efficiently finding and properly evaluating new software packages at this stage helps avoid future problems and risks. An OSS policy typically recommends certain sources of code and establishes evaluation criteria for the applicable classes of use within the organization.

One of the most cost effective recommendations for sourcing OSS code is to evaluate components already used in-house. This kind of "shared use" delivers economies through lower overall complexity, support and maintenance and shared expertise. Companies may also choose to favor certain known, reliable distributors, distributions of OSS or original project sites rather than mirrors or redistributors.

Establishing clear OSS evaluation criteria will generally prevent engineers from wasting time by evaluating software that will not be approved for use. Typical criteria include elements such as:

- Architectural compatibility
- Component modification needs
- License compatibility
- Code quality
- Code stability and maturity
- Quality and completeness of documentation
- Security evaluation
- Availability of support
- Activity level of the community or health of commercial support vendor
- Project maturity and its originating community
- IP risk evaluation

Companies with sensitive data often specify security procedures and "sandbox" environments for evaluation of downloaded software in order to protect the integrity of their corporate network environment.

Another important dimension of OSS evaluation is the type of use anticipated. Criteria for internal software development tools are likely to differ from criteria for components included in a product distributed to customers or deployed in a mission critical infrastructure system. Engineers will make better choices when use types and their associated criteria are well-defined.

OSS license compatibility is one of the most complicated aspects of OSS evaluation. It is unreasonable to ask software engineers to interpret license documents, and it is also impractical to send every OSS candidate to the legal department for review. Many organizations have developed lists of common OSS licenses with indications of their suitability for each type of use applicable to the organization. As each new OSS license is evaluated by the legal department, its compatibility with each type of use is added to the list. As the license list grows, the approach allows engineers to quickly decide whether an OSS component is worth further evaluation, and the workload on the legal department diminishes.

3) Review and approval

No process can be considered reliable unless it is checked. The policy for review and approval specifies how an OSS component evaluation is reviewed and who may approve it for the specified use. Typically, a policy establishes an OSS review board consisting of key stakeholders such as architecture, software development, product management and legal to approve new requests. In many cases, though, a simpler approval cycle may be established for a new release version of an already approved component or reuse of an already approved component. These "fast track" opportunities typically reduce the overhead of OSS management and provide a strong incentive to developers to reuse OSS and work within the policy guidelines.

4) Software procurement

When a company acquires OSS embedded in the delivery of a third-party supplier, it is subject to the same license compliance requirements and many of the same operational risks as when it downloads the OSS directly. It is extremely common for OSS to enter an organization through commercial software suppliers or contract development organizations. An OSS policy should provide procurement guidance by requiring suppliers to report each OSS element embedded in their deliverables, whether it has been modified, and its license and license compliance terms. For software vendors that redistribute the delivered code, the policy may require a warranty and indemnification sufficient to protect the organization or specify code scanning to verify the contents and compliance terms.

5) Code and documentation management

This policy element is all about managing the operational risks that come with OSS. It is simple enough to manage a few OSS downloads through the lifecycle, but companies that use hundreds, or even thousands, of OSS components in multiple versions deployed in multiple configurations must manage this complexity carefully to be efficient and minimize problems.

The policy should specify that the original source code, build files, documentation and license declaration for each OSS component must be archived. These elements should also be archived for OSS downloaded in binary form. OSS licenses sometimes change, and this is the only way to protect your organization's right to continue using the component under the original license.

All internal modifications must also be tracked, because compliance requirements are sometimes different for modified OSS. Most organizations also require that OSS archives be kept up-to-date with all internal bug fixes and enhancements so other projects can reuse these assets.

The policy should also require tracking so that reports can easily be generated to identify all uses of a given OSS component (for addressing vulnerability reports or bug fixes) and to identify all OSS used in a given application or system (for producing lists of compliance requirements for a distribution).

6) Support and maintenance

OSS from communities is typically provided without warranty and under a "self-service" support model. Most OSS policies identify a responsible party for tracking security vulnerabilities and bugs, notifying other users of the component within the organization and applying fixes, as necessary. This role is typically called a *component owner* or *code owner* within an organization. Where commercial support is purchased for an OSS component, the component owner is typically the support contact for the organization.

7) License Compliance

For organizations that distribute software containing OSS or offer network-delivered services using software containing OSS, ensuring compliance with the relevant OSS licenses is a critical element of OSS management. The compliance requirements of most OSS licenses are triggered only on distribution, or, for a few OSS licenses such as the AGPL, on delivery of a network service using the code.

The OSS policy of most organizations that distribute software requires an audit of each release to verify that no undocumented or mis-documented OSS is included. As a best practice, a policy should specify that these audits be performed well in advance of each scheduled release to prevent potentially extensive delays caused by a requirement to replace an incompatibly licensed OSS component at the last minute.

Such an audit typically produces a list of OSS components and compliance requirements for the release. The policy should also clarify responsibilities for checking and executing the compliance requirements, which may include:

- Code notice compliance
- Documentation notice compliance
- Splash or about-screen compliance
- Contract addenda and terms compliance
- Source code provisioning compliance

In cases where your organization delivers software to customers that will further redistribute, the policy should also assign the responsibility to produce a list of OSS components and their licenses.

8) Community Participation

For OSS components acquired from communities, the only source of support is typically through community participation. An OSS policy should specify the kinds of community participation permitted (or required) and the standards and controls for these activities. The possible levels of participation include:

- No community participation
- Participation only through a commercial intermediary
- Participation from personal account with no organizational attribution
- Participation with organizational attribution
- Presentation at conferences
- Contribution of bug fixes
- Contribution of documentation
- Contribution of new functionality
- Creation of a new OSS project

The company's strategy for using OSS and its business goals should dictate the kinds of participation allowed by the policy.

STEP 4: REVIEW AND APPROVE THE POLICY

The result of the working sessions is a draft policy document that must be circulated among the stakeholders for review and approval. Most organizations are able to develop a final policy document after two or three revisions.

IMPLEMENTATION

Once the OSS Policy document is completed, the next step is to implement the policy through a set of processes. Depending on the size and distribution of your organization, these processes may be supported by manual document-based procedures or automated workflow systems. In either case, good processes facilitate both efficient software development and effective OSS management, making it easy to "do the right thing." Your processes must also contain adequate checks to make sure that the OSS policy is consistently followed.

A key success factor for OSS management is training for all participants in the policy and processes. Even the best-intentioned individuals cannot follow rules and processes they don't know and understand.

SUMMARY AND CONCLUSIONS

Any organization that uses more than a few OSS components will benefit from establishing a formal OSS policy. Typical benefits include:

- **Increased development velocity and flexibility**
- **Cost savings:** higher quality sourcing reduces costly problems down the road
- **Reduced incompatibilities:** a managed OSS code base reduces duplication and incompatibilities
- **Reduced support needs:** well-managed support reduces new problems and eliminates duplicated support activities
- **Overhead savings:** license compliance can be assured with minimal overhead
- **Enhanced support:** quality customer support and IP reporting become possible
- **Increased IP asset value:** provable license compliance increases the value of the company's IP assets

ABOUT BLACK DUCK CONSULTING

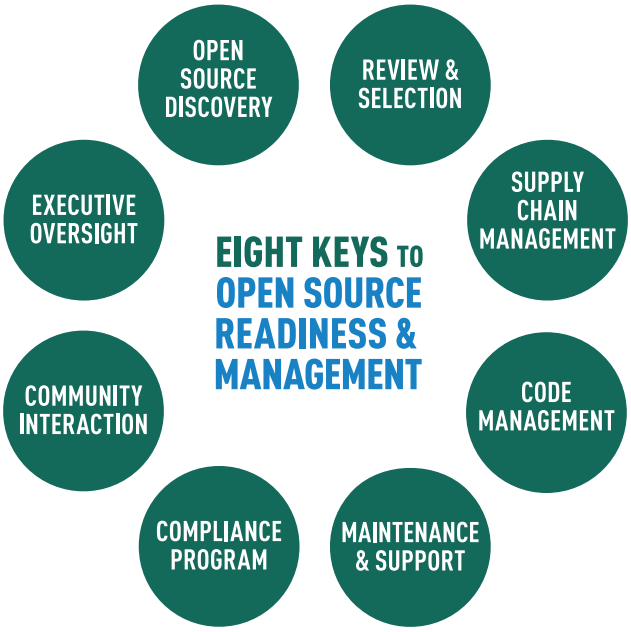
The world's leading companies turn to Black Duck Consulting to assess and evaluate how open source technologies and methods can help them achieve long-term business results. Black Duck consultants are experts in open source strategy and governance, as well as community engagement models, applying open source development processes in corporate IT, and the necessary policies and processes required for effective implementation. With hundreds of engagements assisting companies ranging from start-ups to the world's largest corporations over the last decade, the Black Duck Consulting team creates innovative open source strategies for proven business success.

OPEN SOURCE MANAGEMENT ASSESSMENT SERVICE

For enterprises that have not yet implemented an open source management program, Black Duck Consulting offers a quick and easy way to learn about industry best practices and assess organizational readiness and governance maturity. The easy-to-use, Black Duck Open Source Management Assessment (OSMA), begins with a free self-guided client survey and includes a phone consultation. The assessment is designed to help clients accelerate the implementation of a governance program by weeks or months.

The OSMA evaluates your organization with respect to the eight keys of open source readiness and management:

1. Open Source Discovery
2. Review and Selection
3. Supply Chain Management
4. Code Management
5. Maintenance and Support
6. Compliance Program
7. Community Interaction
8. Executive Oversight



WHY DO AN ASSESSMENT?

- It's a quick, easy, free and confidential way to benchmark your organization against industry best practices for open source code management.
- Discover the approaches employed by industry leaders to get the most from open source while managing the risks.
- Accelerate your compliance program development schedule by weeks or months by leveraging the experience and know-how of industry experts.

CONTACT

To learn more, please contact:
sales@blackducksoftware.com
or call +1 650.493.3800

Additional information is available at:
www.blackducksoftware.com/consulting/

ABOUT BLACK DUCK

Offering award-winning software and consulting, Black Duck is the partner of choice for open source software adoption, governance and management. Enterprises of every size depend on Black Duck to harness the power of open source technologies and methods. As part of the greater OSS community, Black Duck connects developers to comprehensive OSS resources through **Ohloh.net**, and to the latest commentary from industry experts through the **Open Source Delivers** blog. Black Duck also hosts the **Open Source Think Tank**, an international event where thought leaders collaborate on the future of open source. Black Duck is headquartered near Boston and has offices in San Mateo, St. Louis, London, Paris, Frankfurt, Hong Kong, Tokyo, Seoul and Beijing. For more information about how to leverage open source to deliver faster innovation, greater creativity and improved efficiency, visit **www.blackducksoftware.com** and follow us at @black_duck_sw.

To learn more, please contact:

UNITED KINGDOM & IRELAND
info-uk@blackducksoftware.com
or call +44 20 3290 0770

Additional information is available at:
www.blackducksoftware.com

DACH
info-germany@blackducksoftware.com
or call +49 (69) 67733-196

Additional information is available at:
www.blackducksoftware.de

FRANCE
info-france@blackducksoftware.com
or call +33 9.70.46.81.60

Additional information is available at:
www.blackducksoftware.fr

WP-OSPOL-A4EUR-1013

BLACKDUCK