

Open Source in Application Development - 10 Ways to Increase Benefits and Reduce Risk

You're using open source components in your application development initiatives. Almost everyone building custom software is. The benefit is clear: you can deliver better software in less time by reusing proven components. But there are quality, security, and licensing risks too. The difference between positive and negative results comes down to how well you're managing your use of open source. The ten tips below will help you effectively manage open source for maximum ROI.

I. Understand where you are

- Analyze your consumption to understand what and where components are being downloaded.
- Identify problematic components currently being used in development projects
- Classify existing projects based on business importance to establish the role of OSS within the enterprise's existing software portfolio.

II. Analyze your key production applications for security vulnerabilities and licensing issues

- Examine the complete bill of materials for your applications, not just first level dependencies. Flawed components may be hidden deep within your applications
- Analyze new and existing applications – it's never too late to find and fix issues.

III. Establish an open source governance program

- Include guidance on quality, security and licensing – the three major risk areas to consider when using open source software components.
- Design a measurable policy that works for application development as well as legal, risk management, and security.

IV. Start with a pilot program

- There is no need to boil the ocean—this adds risk and expense. Start with a few groups of developers and a few key applications.
- Develop a plan to move from pilot program to department to enterprise-wide based on specific success criteria.
- Ensure policies are both enforceable and non-punitive to ensure acceptance and adoption.

V. Evaluate open source components before using them in development

- Determine if the component meets your requirements for quality, security and licensing.
- Analyze the component's dependencies to discover hidden security or license issues.
- Ensure you have trusted source for each component, such as the Central Repository.

VI. Standardize on a common set of open source components

- Lower maintenance costs by reducing the number of components that need to be supported.
- Limit the number of components that need to be evaluated.

VII. Establish a policy of service and support

- Determine the level of support required and identify the resources required. For some projects, community-based support will be fine.
- Identify critical components that require commercial service and support contracts with binding SLAs.

VIII. Build open source management into your software development process

- Provide developers the tools they need to standardize on a set of thoughtfully chosen components free of license or security defects.
- Ensure your policies enable development, not disrupt it.
- Ensure projects do not include flawed components by using automated analysis during the development process.

IX. Continuously monitor your production applications to learn of newly discovered defects

- Maintain a record of the bills of materials for your applications so that you always know what components were included.
- Ensure you have a mechanism to know when a component is updated for critical reasons such as a fix to a security or stability problem.

X. Establish mechanisms to monitor the effectiveness of your open source governance program

- Monitor open source consumption from outside sources, such as the Central Repository and vendor sites.
- Determine if potential problems exist by identifying internal teams that may not be adhering to your OSS policies.
- Audit your existing applications to ensure the included components meet your quality, security, and licensing guidelines.
- Analyze software delivered by subcontractors or software suppliers to ensure it meets your requirements.

Learn how you gain visibility and control of your open source usage with Sonatype Insight. Sonatype Insight helps you build better software faster without unnecessary quality, security, or licensing risks and without disrupting or delaying your development process. Learn more at www.sonatype.com/insight.