Transforming software development and delivery
White paper
May 2008

Rational. software

Improve IT governance and compliance with static analysis.

IBM Rational Software Analyzer software

Contents

Introduction

Increased regulatory requirements and growing corporate scrutiny are driving software development organizations to adopt more vigorous IT governance and compliance policies. Plus, new security threats are emerging every day, which means that it's more important than ever to make sure applications undergo a thorough security review before they're deployed. These pressures are pushing companies to conduct more internal software audits than ever before, which can be challenging, given limited resources. In addition, there's a growing need to get products to the marketplace quickly while simultaneously cutting costs, minimizing development resources and reducing outsourcing. These factors, combined with the increasing complexity of many software applications and today's accelerated development schedules, can make it difficult to maintain code quality. That's why it's critical to implement governance and compliance as early as possible in the software development lifecycle (SDLC) and make sure consistent code review processes are followed throughout the organization.

IBM Rational® Software Analyzer is a comprehensive, automated static analysis solution that can help you detect—and in many cases correct—coding problems to help improve overall code quality early in the SDLC. Rational Software Analyzer delivers capabilities to help increase development team productivity; provide management insight into development projects and governance processes; automate and centralize code reviews through integration at the build stage; reduce development, quality assurance (QA) and testing costs by detecting source code bugs earlier; and decrease time to market. In addition, IBM Rational Software Analyzer provides a solid foundation that can help your teams focus on addressing the needs of the business with the rapid delivery of high-quality software. Plus, by using an automated static analysis solution, you can help ensure that your organization is adhering to governance and compliance processes earlier in the SDLC, thereby avoiding the high cost of fixing problems later in the cycle.

Highlights

This paper looks at why it's important to implement governance and compliance processes early, and it discusses the principles and benefits of static analysis. It also shows how the IBM Rational Software Analyzer application can help you enforce IT governance and compliance to help improve the quality of your software.

Using the Rational Software Analyzer application to assist in IT governance and compliance

It's becoming more and more challenging to meet IT governance and compliance objectives, as companies take on more and more software projects.

Meeting IT governance and compliance objectives is becoming more challenging, putting companies at unnecessary risk. Existing teams and resources cannot scale to volume as new projects begin and acquisition activity increases. For the CIO, vice president of engineering or IT, project manager of operations (PMO), director of engineering and development, and other individuals or teams responsible for overseeing and executing on a corporate IT or governance strategy, it becomes increasingly challenging to enforce a code review process down to the development organization. With increased efforts to outsource parts of this function, having the ability to enforce IT governance policies to unknown developers becomes a tremendous challenge.

Following are some of the pain points:

Pain points include corporate policies that require more code reviews, security risk concerns, and increasing numbers of internal software audits.

- *Corporate policy is instituted such that code review must be conducted to identify any open source references introduced or in use.*
- *Because of security risk concerns, code for certain application projects that reside outside the firewall must go through a thorough security review to identify changes or verify whether new code must be added before final deployment to production.*
- *The number of internal software application audits is increasing.*

Highlights

Software defects cost companies in the United States billions of dollars a year, but nearly one-third of these costs can be avoided by implementing effective software testing earlier in the software development lifecycle.

The cost of fixing software defects can be as much as 30 times higher after the software is released.

The high cost of software defects

Studies have shown that software defects cost the U.S. economy billions of dollars a year. And these estimates don't include the potential business costs associated with software reliability and security problems—costs incurred by damage to brand and customer loyalty, loss of revenue and marketshare, and even (in safety-critical systems) loss of life. But studies have also shown that nearly one-third of these costs could have been avoided through better software testing earlier in the SDLC. In fact, the longer you wait, the more it costs. Figure 1 shows the typical application lifecycle comprising four phases: design and architecture, implementation, QA testing, and operation. And figure 2 shows the relative cost of code fixes at various stages of the application lifecycle—as much as 30 times higher after release than during the design phase

Product development processes are considered effective when they yield the desired results for product quality and customer satisfaction. Therefore, to meet your quality goals, it can be helpful to monitor and measure the effectiveness of your processes through an automated static analysis solution.

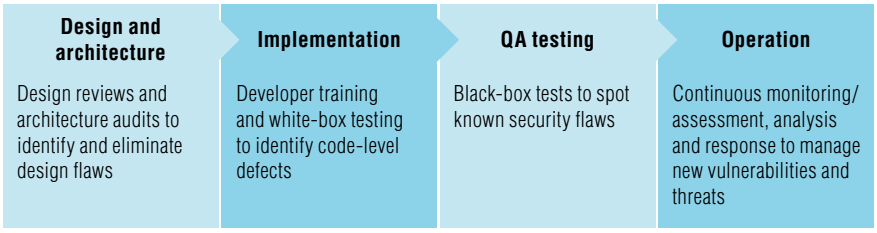

Figure 1: An end-to-end view of an application lifecycle

Design and architecture	Implementation	Integration testing	Customer beta test	Postproduct release
1X*	5X	10X	15X	30X

*X is a normalized unit of cost and can be expressed in terms of person-hours, dollars, etc.

Figure 2: Cost of defect fixes throughout the software lifecycle

Highlights

Analyzing software without actually executing the programs built from it is know as static analysis.

Static analysis tools apply a set of rules to the code that helps identify source code defects.

Rational Software Analyzer can help developers review code and identify bugs early in the development lifecycle.

Why static analysis is essential to good code development

Static code analysis refers to analyzing software without actually executing the programs built from that software. This is in contrast to dynamic analysis, which is performed on an executing program. In most cases, static analysis is performed on a version of the source code. However, in some cases, the analysis is performed on a form of the object code. Static analysis usually refers to an analysis performed by an automated tool; human analysis more commonly refers to program understanding or program comprehension.

Source code–level defects are introduced in the implementation phase. Because the average developer has little or no training in code review development practices, implementation and coding errors are common sources of software defects. Static analysis tools, sometimes referred to as white-box tools because of the visibility they provide into code-level information, apply a set of rules to the code that helps identify possible source code defects without actually executing the program. Static analysis can therefore help you identify and eradicate flaws before your applications are deployed, which usually results in a less costly remediation process.

Tangible benefits from the Rational Software Analyzer application

Rational Software Analyzer is an extensible software development application that enables developers to perform software code review and bug identification early in the development cycle. It is particularly valuable in industries where the cost of software defects and rework is high, such as transportation, aerospace and defense, and medical equipment. Rational Software Analyzer is designed to give you a high return on your investment by:

- ***Helping to improve application quality.*** *By enabling you to identify defects in the earlier stages of the SDLC, Rational Software Analyzer can help reduce the number of software defects found in the field, potentially resulting in reduced product maintenance costs and increased customer satisfaction.*
- ***Decreasing development and defect remediation costs.*** *By finding bugs earlier, you can potentially reduce your development, build, support, testing and quality management costs.*

Highlights

Using the Rational Software Analyzer application can help increase developer productivity and help speed products to market.

- ***Increasing developer productivity.*** *By automating your static analysis processes, you can potentially reduce the amount of time and resources your developers dedicate to defect management and product development. Developers can therefore focus more time on coding and product futures, which can help lower overall operating costs for the company.*
- ***Helping to speed products to the marketplace.*** *By enabling you to detect bugs earlier in the SDLC, Rational Software Analyzer can help streamline your product release cycles.*

The application includes programming rules that can help increase style consistency, reduce errors and improve application performance.

Features programming rules to enforce best practices

Rational Software Analyzer includes programming rules that can help you review code using development best practices, helping to increase style consistency, reduce errors and improve application performance. The application features:

- *More than 550 Java™ rules covering code review, architectural discovery and deep analysis.*
- *More than 130 C/C++ code review rules.*
- *More than 40 Java software metrics rules.*
- *Code quick fixes for many rules, enabling you to select an automatic fix for select code errors.*

Rational Software Analyzer easily integrates with Eclipse environments, which can help speed the solution's adoption.

Integrates quickly and easily into existing Eclipse environments to accelerate time to value

Rational Software Analyzer software integrates easily with Eclipse environments, which can help decrease the amount of staff training required and speed the solution's adoption. In addition, by working with a framework that you already have in place, you can help simplify utilization for your development team as well as reduce desktop application clutter. Rational Software Analyzer supports the Microsoft® Windows® 2003, XP and Vista operating systems, along with the Red Hat and SUSE Linux® Enterprise Server platforms. The application has been translated into multiple common languages and can be quickly installed using the installation manager tool.

Highlights

The Rational Software Analyzer API allows you to centrally manage legacy and third-party technologies.

Integration with IBM Rational Build Forge software enables you to leverage a comprehensive software code scan solution that centralizes and automates code quality analysis as part of the build process.

The enterprise edition of Rational Software Analyzer provides powerful reporting features, customizable rule configurations and a rich set of out-of-the-box reports.

Provides a central point from which to integrate and run other analysis tools

The Rational Software Analyzer application features an extensible, enterprise-class framework that enables you to develop customizable and consistent workflows, helping to simplify static analysis utilization. Through the Rational Software Analyzer application programming interface (API), you can centrally manage legacy and third-party technologies. The application therefore enables developers to execute multiple scan rules and tools—and even multiple forms of analysis on different program languages—simultaneously from a common framework, which can help increase productivity. Plus, Rational Software Analyzer includes selectable and customizable rule sets that enable a collaborative enterprise approach to development best practices.

Provides a centralized, automated software code scan solution through integration with IBM Rational Build Forge software

Through its smooth integration with IBM Rational Build Forge® software, Rational Software Analyzer provides a comprehensive software code scan solution that centralizes and automates code quality analysis as part of the build process. The application's automated code scans help enforce utilization of static analysis to potentially reduce the risk that software bugs will enter the marketplace. And Rational Software Analyzer enables developers to run remote code scans from within the Rational Build Forge environment, extending the benefit of the application and alleviating the need to have local software installed. The application ships with a Rational Build Forge adapter and includes command-line interface (CLI) support.

Helps improve project visibility, making it easy to comply with corporate programming governance and compliance mandates

The IBM Rational Software Analyzer Enterprise Edition application provides powerful reporting features that give your management team a high-level view of software quality and compliance-related issues. Plus, rule configurations are customizable, which makes it easy for you to align corporate governance requirements with programming guidelines. A rich set of out-of-the-box reports

and metrics is available in HTML, Adobe® PDF or customizable data export formats. The IBM Rational Software Analyzer Developer Edition application includes individual reports, while the the enterprise edition offers centralized report views. The enterprise edition also features an automated code review that's independent of individual or location, helping your team adhere to corporate IT compliance standards. And Rational Software Analyzer Enterprise Edition results are accessible from the Rational Build Forge log files, making them available for auditors to verify and review.

Conclusion

Rational Software Analyzer software can help you effectively implement and manage your IT governance and compliance objectives. The application integrates with other build automation tools, such as Rational Build Forge, and incorporates automated and centralized code review, making it an essential part of your existing prebuild processes. Plus, Rational Software Analyzer gives your management team the insight it needs to help ensure that your organization adheres to its governance and compliance policies, and to help improve code quality. With Rational Software Analyzer, software analysis can become an integral part of your SDLC, helping to ensure that all code is reviewed, regardless of the source or individual developers' activities.

For more information

To learn more about how IBM Rational Software Analyzer software can help you implement IT governance and compliance earlier in the SLDC, contact your IBM representative or IBM Business Partner, or visit:

ibm.com/software/awdtools/swanalyzer

IBM Corporation
Software Group
Route 100
Somers, NY 10589
U.S.A.

Produced in the United States of America
05-08

RAW14025-USEN-00