coverity®
A Synopsys Company

SOLUTION OVERVIEW

Coverity Software Testing Platform

The Coverity® Software Testing Platform helps teams identify, manage and remediate critical quality and security defects and improve the overall efficiency of their testing efforts, which reduces the cost, time and risk of software errors.

Smart companies understand that to reduce risk and accelerate time to market, they need to find and fix quality and security issues in their code as early in the software development lifecycle (SDLC) as possible. Market-leading organizations rely on the Coverity Software Testing Platform to help them transform their testing program from a reactive to a proactive process, and into a competitive advantage.

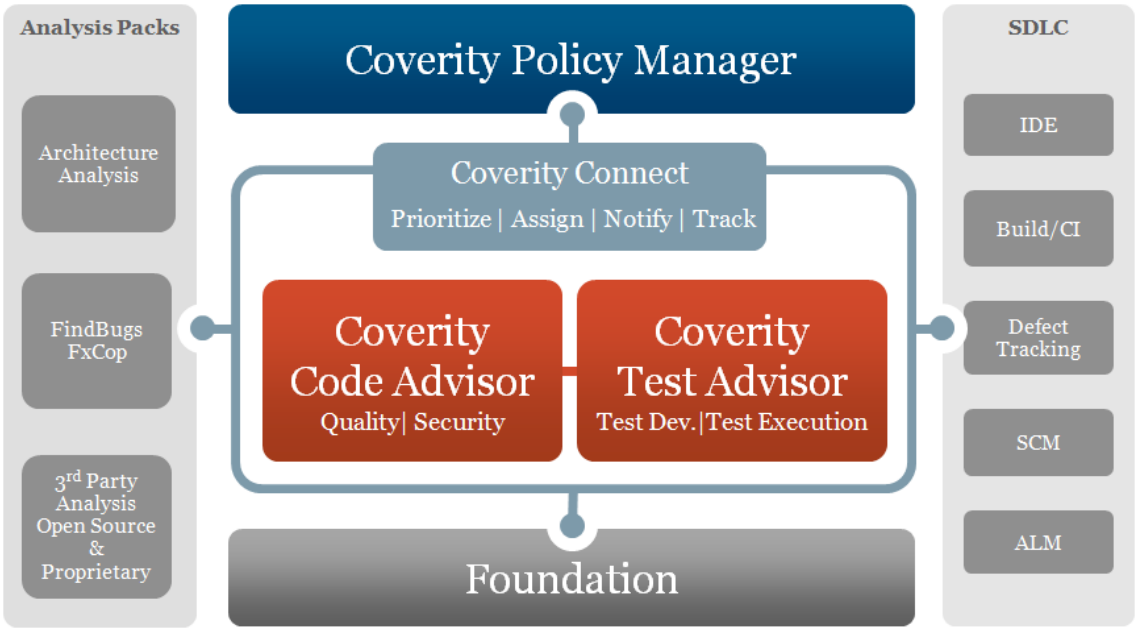
Coverity Software Testing Platform

The Coverity Software Testing Platform empowers teams to build quality and security testing into the development process at the earliest stage for fast, resilient and predictable software delivery, and shrink Quality Assurance (QA) testing cycles by enabling teams to focus automated and manual testing efforts based on the impact of change and without requiring access to the source code. This can reduce costs and risks, accelerate time-to-market, enhance customer satisfaction and increase revenues.

Key Solutions and Capabilities

Coverity Foundation

Coverity's award-winning static analysis capabilities provide the foundation for the Coverity Software Testing Platform. Based on more than a decade of research, development and analysis of more than 5 billion lines of proprietary and open source code, Coverity static analysis applies multiple patented analysis techniques to automatically test code as it is written. It can identify hard-to-spot, traditionally untestable issues with accuracy and precision. It provides full path coverage, ensuring that every line of code and every potential execution path are tested. Full interprocedural analysis enables developers to find and fix complex defects that cross function boundaries versus simple style violations or superficial



The Coverity Software Testing Platform is used by more than 1,100 customers.

BENEFITS OF SOFTWARE TESTING

- Shrink QA and security testing cycles
- Improve collaboration between QA, development and security teams
- Reduce the cost and time of troubleshooting and re-work
- Allocate more time and resources to innovation
- Minimize the number of defects that escape into the field or production
- Identify risky code changes and evaluate their impact on testing

issues. Coverity static analysis utilizes sophisticated techniques to minimize false positives and provides actionable remediation advice, including:

- Interprocedural dataflow analysis to find complex issues that cross function, file and class boundaries.
- Boolean satisfiability solvers to improve accuracy through bit-accurate analysis.
- False path pruning to understand the data dependencies in code and eliminate infeasible paths from the analysis.
- Statistical profiling to automatically detect coding patterns and learn the programmer's intent to reduce "noise" in the results – in other words, it understands what you meant to say, not what you said.
- Design pattern intelligence to understand patterns and programming idioms which are integrated into the analysis.
- Enterprise framework analyzer to augment source code analysis by providing a deep understanding of modern web applications including dependency injection, entry points and the MVC paradigm.
- White box fuzzer to automatically validate that data sanitization routines perform sufficient sanitization of untrusted data and are used in the right context.
- Change impact analysis to automatically map code and function dependencies and analyze all impacted code related to change—the changed code itself and the code impacted by a change.

The Coverity platform scales to accommodate thousands of developers in geographically distributed environments and can analyze projects in excess of 100 million lines of code with ease. Users can analyze large, complex codebases quickly by leveraging parallel analysis. This means they can scan even the most complex code bases regularly, which allows them to adopt software testing as part of a nightly or continuous build process. Through incremental analysis, developers can quickly analyze changed code or code impacted by a change, which saves valuable time.

Coverity Connect for Rapid Remediation

Coverity Connect is the collaborative issue management interface used by developers to efficiently manages all issues identified through source code analysis to resolution, within a unified workflow. Key features for automated issue management include:

- Prioritization and filtering based on criticality and impact
- Source code navigation to identify the exact path to the defect
- Impact mapping to identify every occurrence of the defect across shared code
- Automatic assignment of issues to the appropriate developer

The Coverity Software Testing Platform is extensible, enabling integration with third party analysis tools so that the results are visible in Coverity Connect. This provides a unified system for simplified code analysis and management. The extensible platform also allows rapid integration of additional components in the development environment, such as source control management systems, bug tracking systems, integrated development environments (IDEs), build systems, continuous integration systems and application lifecycle management (ALM) solutions.

Coverity Quality Advisor

Coverity Quality Advisor identifies critical quality defects in the software development workflow with accuracy and actionable remediation guidance. It enables developers to quickly and efficiently troubleshoot and fix the quality defects that matter, such as concurrency issues, API usage errors, resource leaks and performance and maintainability issues, before the code even makes its way to QA. Users can run the analysis directly from within their IDE, in conjunction with the continuous integration server or as part of the central build. Coverity Quality Advisor enables you to build quality early in the development cycle and reduce the cost, time and risk of quality issues and software failures.

In addition to the quality defects identified through Coverity analysis, you can seamlessly integrate results from additional analysis engines to manage multiple types of quality defects to resolution within a unified software testing workflow.

The following analysis packs are available for Coverity Quality Advisor:

- **Coverity Dynamic Analysis for Coverity Quality Advisor** finds concurrency issues such as race conditions, deadlocks and resource leaks in Java programs.
- **Coverity Architecture Analysis for Coverity Quality Advisor** identifies architectural flaws.
- **Analysis Integration** enables developers to manage FindBugs and FxCop defects in the same workflow as Coverity platform-found defects, providing them with a single location to surface and remediate defects.
- **Coverity Analysis Integration Toolkit** allows the integrations of third party analysis results into the Coverity platform, which means different tools can be used to look for different types of quality and security issues, while still managing all defects within a single workflow.

Coverity Security Advisor

Coverity Security Advisor helps organizations lower their risks and decrease project costs by identifying critical defects that could lead to security vulnerabilities during development. It utilizes Coverity static analysis to identify critical OWASP Top 10 and CWE Top 25 security defects in Java web applications, including SQL injection, XSS and cross site request forgery (CSRF). One of the primary reasons that legacy security tools have failed in development is high false positives, or inaccurate results. The Coverity static analysis engine was built from the ground up to address the complexity of today's modern applications, which leads to more accurate results. This means developers don't waste time chasing down false positives or false negatives. Instead they can focus their efforts on fixing real, relevant and critical issues.

For Interactive Application Security Testing (IAST), Coverity has partnered with NT OBJECTives (NTO). The results from NTOSpider can be automatically correlated with our Static Application Security Testing (SAST) findings. Results are easily integrated into the development workflow through Coverity Connect for a fully integrated Dynamic Application Security Testing (DAST) solution.

With Coverity Security Advisor, security can be effectively built into the development process, which reduces the delays and re-work costs that result from defects found late in the cycle. This can ultimately lower project costs and reduce the risk of costly, brand-damaging security breaches in the field or in production.

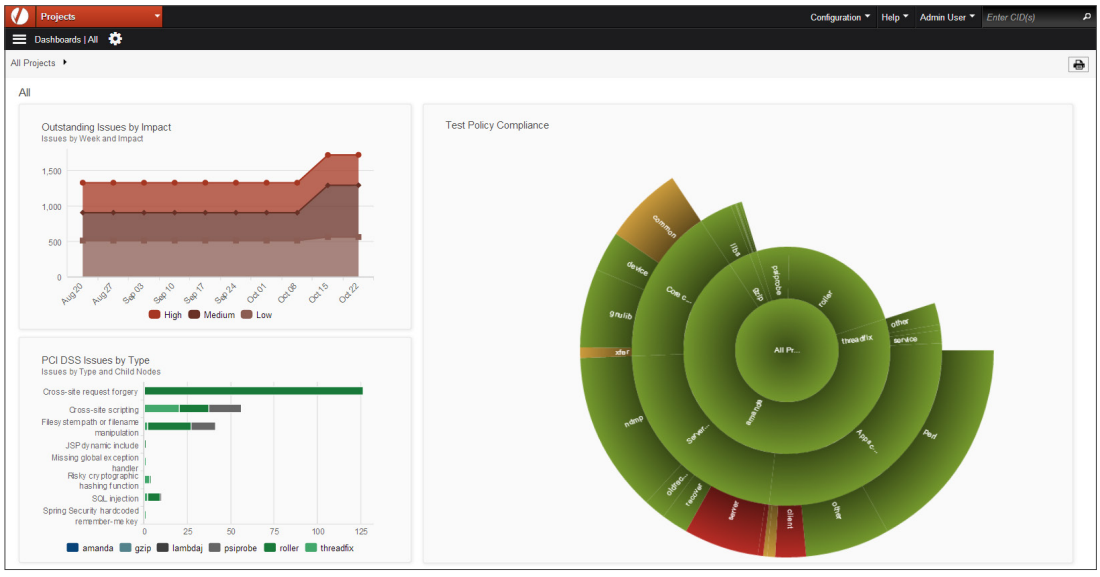
Coverity Test Advisor—Development Edition

Automated testing such as unit testing is valuable in concept but inefficient in practice. Code coverage is commonly used as the metric to determine "adequate vs. inadequate" testing in development, but can be misleading. Without the proper insight into the code, coverage is just an arbitrary number that wastes developer time and increases risk. Coverity Test Advisor – Development Edition improves the efficiency of automated testing by focusing time and resources on the most critical parts of the code and surfacing untested code violations in the developer's workflow for quick and efficient remediation.

With Coverity Test Advisor – Development Edition, development teams can specify a test policy, which defines on a fine level of granularity what code is most risky and needs to be tested thoroughly.

Coverity Test Advisor – Development Edition includes features to help test what matters most:

- **Change impact analysis** allows teams to identify the functional impact of code changes.
- **Deep code analysis** enables teams to exclude dead code segments and untestable code, as well as debugging and exception handling code from their automated testing efforts.
- **Test rule configurator** allows managers to establish and enforce consistent testing rules across projects and teams based on high-risk criteria.
- **Test prioritization** helps accelerate testing time by providing insight into which tests should be run, based on changes to the code or project priorities.
- **Test data correlation analysis** automatically correlates discrete data from multiple sources, including Coverity static analysis, test coverage and source control management tools, turning that data into actionable testing intelligence.



Achieve executive-level visibility into potential areas of risk in projects for better control.

Untested code violations are surfaced within Coverity Connect for quick and easy remediation. Insufficient testing is translated into actionable work items for developers or test engineers, and management can track progress of closing them out.

Coverity Test Advisor – QA Edition

Coverity Test Advisor – QA Edition provides Quality Assurance (QA) teams with intelligent change impact analysis for software testing. By monitoring the execution and results of automated tests, teams can prioritize tests and identify which are most critical based on changes to source code. This enables faster time to market while ensuring that the most relevant tests are executed. The result is peace of mind from knowing that your product has been thoroughly tested and issues stay resolved.

Coverity Test Advisor – QA Edition analyzes Java, C# and web applications and calculates a score for each test according to how it covers code changes. Testing time is reduced as team members focus on those tests that are relevant to the modifications they've made. Quality is improved as teams can prioritize tests where the risk is highest.

Coverity Test Advisor – QA Edition generates a functional overview of the product, correlating code changes with test footprints recorded over time. This intelligent aggregation of data allows test teams to identify gaps in coverage, preventing regression failures in the production application.

Use the Coverity Test Advisor – QA Edition dashboard to analyze risk and discover how code changes affect existing tests, while leveraging the Impact Analyzer to evaluate the impact of modifications to the test plan, eliminate redundant tests and generate scenarios that eliminate gaps in test coverage. This "what if" potential allows QA teams to model the risk of a change to the product before the testing cycle begins.

There is no need to create a special build for testing with Coverity Test Advisor – QA Edition. The recording agent monitors .NET, Java and web applications in their native runtime environment using information already found in the binary. No source code or access to the source control server is necessary and no code, applications or data are transmitted outside your network.

Coverity Policy Manager

Coverity Policy Manager allows you to define and enforce a consistent standard for code quality, security and testing across your organization. The configurable views allow you to select the development metrics and thresholds that align to your objectives, with the flexibility to modify them throughout the course of your project. You get visibility into which teams, projects or components are compliant with the standards and the traceability to examine the defects in the code which are causing the violations. This level of visibility and traceability improves decision making and the predictability of releases.

Monitor code quality and security with out-of-the-box metrics such as defect density, outstanding and resolved issue counts, outstanding defects by impact, and many others. Keep tabs on test effectiveness with metrics on test policy coverage, outstanding test policy violations and more.

Monitor the current state of quality and security, and track trends over time. Watch your defect density decrease as software testing adoption increases, with metrics that show trends in daily or monthly unique users, issues introduced, issues resolved and other key information.

QUALITY AND SECURITY DEFECTS IDENTIFIED

- Concurrency defects such as deadlocks, race conditions and misuse of lock objects
- Performance degradation problems due to memory leaks, database connection leaks, etc.
- Crash-causing errors such as null pointer dereferences, improper memory allocations and division by zero
- Incorrect program behavior caused by dead code, uninitialized variables and other errors
- Arbitrary code execution from integer and buffer overflows
- Data loss due to corruption
- Loss of integrity from misuse and mixing of data types
- Use of risky cryptographic functions
- Web application security defects such as SQL injection, XSS, path traversal and risky cryptographic functions

coverity®
A Synopsys Company

For More Information
www.coverity.com
Email: info@coverity.com

Coverity Inc. Headquarters
185 Berry Street, Suite 6500
San Francisco, CA 94107 USA

U.S. Sales: (800) 873-8193
International Sales: +1 (415) 321-5237
Email: sales@coverity.com

201407-04