



Schneider Electric Improves Product Quality While Saving Over 2,500 Engineering Hours with Coverity

“We run the analysis from a centralized team and send out an email one week later announcing the results are available for the developers to review. If there is ever a delay in getting this information out to the developers, they come to us and seek it out. Not a single developer did this in the past. Now we have developers demanding Coverity.”

– Frank Klosek, Qualimetry and Senior Technical Manager

Business Benefits

Developer Efficiency and Time to Market

Engineering hours to fix a critical or high defect have been reduced from 2 days to 15 minutes per defect: a savings of 2,520 developer hours to-date

Product Quality

100% of P1 defects are now identified and resolved prior to product release

Customer Satisfaction

Coverity directly contributed to a reduction in the number of customer support calls over time

Business Overview and Challenge

Schneider Electric (Schneider) is a global leader in energy management, developing solutions to make energy safe, reliable, efficient, and productive from plant to plug. In 1999, Schneider’s Automation and Control software division started to develop a new automation system called Unity, a solution to automate manufacturing control systems, machine equipment and related applications. Unity is comprised of two parts: a Programmable Logical Controller (PLC) to automate the machine and UnityPro, a custom-built software program to operate the controllers and enable application design within an integrated development environment (IDE). In 2003 Unity was launched into the market, selling to Systems Integrators, OEM partners, end users and manufacturers in a variety of industries, from chemicals to automotive.

Unity is the largest software development project within Schneider Electric, initially built with about 200 developers and now totaling almost three million lines of code. It is also the most complex code base due to the interdependencies within the code and integration with other components in the software supply chain. In light of this, Schneider has zero tolerance for risks to product quality. As an ongoing measurement of quality,

Schneider's customers and integration partners report defects identified in the field back to Schneider through the submission of open problem requests. Schneider monitors defect rates and trending over time, and with every new release of Unity aims to decrease the number of incoming problem requests, a direct correlation to increasing quality.

In 2006, Schneider began experiencing an increase in support calls via problem report submissions due to software issues at customer sites. Some of Schneider's largest customers began complaining about an increase in software defects, and Schneider took notice. This was a material issue that posed a serious risk to Schneider's brand and customer satisfaction, one which required serious and immediate action. Schneider's CEO and executive team instituted a quality program for Unity, with a specific emphasis in the software development organization. Schneider deemed it necessary to look not only at its existing software development processes, but to also evaluate tools that can be embedded in the development process to improve code quality, and therefore product quality, at customer sites to reduce support calls and accelerate time to market with new releases.

According to Frank Klosek, Qualimetry and Senior Technical Manager of Schneider's End User department, "Providing our customers with a high quality product and experience is of the utmost importance to Schneider Electric. We are continually looking for process improvements and tools which will contribute to a decrease in support calls from customer sites over time, and with every new product release we are looking for a downward trend in submitted problem reports compared to previous releases."

Schneider was already familiar with static analysis solutions, as a portion of the software development organization was using PC-Lint, a low-cost tool in the market. Unfortunately, developer adoption was minimal and consequently developers did not use the tool or monitor its results. In fact, Schneider experienced a critical software failure resulting in a product crash at a major automobile manufacturer, a defect that was reported in PC-Lint before the crash at the customer site, and consequently one which could have been avoided if the tool was used. Schneider believed in the value of static analysis, but needed a tool that would be accepted and adopted by the development organization to minimize costly defects from entering the field.

Solution Evaluation

Mr. Klosek read about Coverity in a trade publication and was interested in purchasing the solution, but since this was a high profile quality initiative within Schneider, any purchase decision must be based upon solid and proven business benefits for all impacted functions within the company; therefore, Schneider underwent a full evaluation.

Schneider's static analysis evaluation included Coverity, Klocwork, and PolySpace. Because UnityPro was a large, complex application with two and a half million lines of code at the time of evaluation, it required a static analysis product with a robust analysis engine, as well as easy installation, adaptability to the build environment, and the ability to run it without any code modifications. Web access, including the presentation of defects and the ability to create components and mappings to developers, was also important. All of these factors were critical to developer adoption, so Mr. Klosek involved ten key developers in the evaluation and gave the tool to them to use,

on their own code and within their environment. To be fully accepted, the solution had to be adopted by developer demand, so Mr. Klosek believed the best way to achieve this was to simply hand the tool to the developers and let them evaluate each on their own. The developers needed to see the benefit and not consider it a burden.

PolySpace was eliminated during the first phase of the evaluation, as it could not scale past 200,000 lines of code, well shy of Schneider's line of code count. Schneider ran both Coverity and Klocwork on the full line of code set. During the evaluation, the developers ran into three key problems with Klocwork's solution: 1) it was hard to execute given the complexity of the code base; 2) frequent crashes put the product's stability to handle the current code base or further increases to the line of code count in question; and 3) the false positive rate was much higher compared to Coverity, translating into workload and wasted developer hours.

In addition to lower false positives, the depth of the analysis engine was much greater with Coverity. The solution was easily installed and within one day the developers were scanning their code and tuning the tool. The developers in the evaluation unanimously voted for Coverity as their static analysis solution of choice, easily convincing management to make the investment.

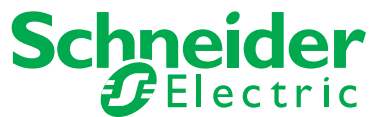
Coverity Deployment and Benefits Realized

Coverity Static Analysis has been deployed for three years across all of the development teams within Schneider's software division for UnityPro, projects which today total almost three million lines of code.

Schneider tracks defects and analyzes code with 34 checkers, classified into three priorities based upon criticality: P1, P2, and P3. P1 checkers, for example, find defects such as buffer overflows and memory leaks; critical and high priority defects that could result in a system crash. P1 bugs now need to be 100 percent resolved by launch date. Before Coverity these types of defects could have gone undetected in the development cycle and let into the field. After Coverity, these defects are now found and resolved prior to being released in the field. The development teams have validated that these are hard to find defects that were therefore missed in code reviews and other testing, and have reinforced the value that Coverity provides to help catch the defects early in the cycle, saving time and cost while increasing quality.

Since deploying Coverity, Schneider has reported the following results:

- The first run in 2007 resulted in 4,000 defects identified (400 P1 and 3,600 P2 defects).
- Since then, approximately 3,000 additional defects have been identified, inspected, and fixed, 6% of which were classified as critical or high in priority.
- Engineering hours to fix a critical or high defect have been reduced from 2 days to 15 minutes per defect: a savings of 2,520 developer hours to-date.
- Schneider recouped its cost of investment within 10-12 months of initial deployment.
- The number of customer support calls have been decreasing over time—an indication of product quality improvement.*



In addition, Coverity has helped to improve developer coding style. Developers get feedback as to where they are susceptible to making coding errors, and when they identify a trend they fix it to improve over time. All of these factors have led to full developer acceptance. The development teams have embraced it instead of treating it like a policing tool, and have come to rely on Coverity as part of their daily work. According to Mr. Klosek, "We run the analysis from a centralized team and send out an email one week later announcing the results are available for the developers to review. If there is ever a delay in getting this information out to the developers, they come to us and seek it out. Not a single developer did this in the past. Now we have developers demanding Coverity."

Conclusion

Schneider Electric has adopted Coverity to improve product quality and software integrity while reducing development costs and re-focusing resources on innovation, benefits which have been realized within the development organization, across the company, and supported at the highest level within Schneider Electric management.

**Note: the reduction in support calls is a result of several elements, including Coverity.*

About Coverity

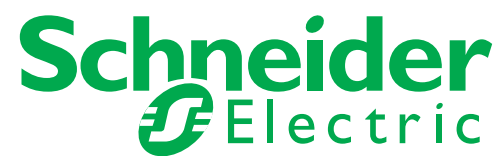
Coverity is the trusted standard for companies that have a zero tolerance policy for software failure, problems, and security breaches.

Coverity's award-winning portfolio of software integrity products enables customers to prevent software problems throughout the application lifecycle

Coverity Inc. Headquarters

Coverity Inc. Headquarters
185 Berry Street, Suite 1600
San Francisco, CA 94107 USA
www.coverity.com
sales@coverity.com
(800) 873-8193

© 2010 Coverity, Inc. All rights reserved.



Coverity Inc. Headquarters

Coverity Inc. Headquarters
185 Berry Street, Suite 1600
San Francisco, CA 94107 USA
www.coverity.com
sales@coverity.com
(800) 873-8193