

Non-Intrusive Software Testing – An Approach Paper

Abstract:

Employees across all ranks of the IT departments face problems when the performance of mission-critical applications is below average and this result in loss of revenue. Thus, in present times several companies are investing in effective production monitoring techniques to improve performance levels, application availability and enhance end-user experience.

Having said that, any ad-hoc approach can be pretty expensive in terms of resources overheads, integration and security requirements, and could come with some side effects. In the normal course of execution, the side effects can prove damaging for the application.

An approach that helps businesses to meet all the production issues, without any adverse side effects while the application is live would be well appreciated. The best solution lies with Non-Intrusive software testing, executed in the actual production environment. It is based on introducing monitoring probes that are non-invasive and non-intrusive. These probes gather the vital information of the application without harming the prevailing end-user experience. They aid in tracing and monitoring systems.

Read on to find out how non-intrusive testing, a breakthrough software testing, helps optimize and enhance product quality.

Challenged by Production Problems?

Improved software application availability and performance form the backbone of several core business operations today. Production managers often get to hear unfortunate stories of applications that have gone live and then failed, even though comprehensive software testing was executed throughout the Software Development Life Cycle (SDLC)

The problems come in many flavors — low performance, application design problems/bugs, bottlenecks in components, etc. Proactive monitoring of applications in the production environment can help the company handle any sudden surge in load/volume.

Probe without invading!

Proactive performance tuning and monitoring is a must for every business, since it keeps them synchronized with the enhanced business needs. They require monitors to make authoritative measurement of system consistency to ensure that the application consistently meets performance efficiency.

In the traditional method of performance monitoring and performance tuning chunks of codes were required. Along with that came a plethora of side-effects. Monitoring these side-effects can be challenging as they can be subtle and hidden. Additionally, the performance testing team must be equipped enough to determine the “trouble-causing” sections of code.

The introduction of non-intrusive probes ensures rapid time-to-resolution of any issues and can be used throughout the lifetime of the application. This methodology does away with time-consuming re-linking and recompiling of the source code.

Non-Intrusive Probes:

A Silent Worker

Non-intrusive monitors help to capture those events that do not show up as output. They get into the application and bring you the problems that are occurring. Thus, they render a more comprehensive performance testing methodology. One can carry out software testing under real-world conditions, with no added overheads. Such probes acquire and log crucial data and gives access to application performance data. Hence, they can be used aids in performance tuning.

A Safe Alternative

This form of testing is transparent to the software under the microscope. The transparent nature of software testing ensures that the “natural environs” of the application are not disturbed. Non-intrusive probe consists of several other characteristics. The source code or the application files stored in the disk need no changes to be made. Again there is no need to rebuild or recompile the application. While running in real time, the application can enable or disable probes dynamically.

System logs, application / product logs and selectively enabled and debug/informative log messages (for suspect components) are some of the aids used by this software testing methodology.

Approach to Successful Non-Intrusive Testing:

The choice of the right monitoring technique is critical to achieve the goals of production monitoring. The pressures of a production environment are many. The methodology again should integrate with the operations and must be comprehensive.

A Phased Approach

Non-intrusive approach explores to the core of the problem, and thus in a production environment, performance monitoring forms the foundation for any non-intrusive software testing assignment. A successful running of such a solution requires a complete knowledge of the application and its real-world environment. A phased approach, which offers such an ideal problem solving environment, has four phases:

System Study

Monitor Design and Development

Monitor Deployment and Validation

Data Collection and Analysis

1. System Study

In this phase, an exhaustive study of the system architecture of the application is carried out to understand the different components, their roles and their interactions.

2. Monitor Design and Development

Here, the gathered architectural knowledgebase is used to arrive at metrics required to analyze each of the identified components. In most cases, the standard performance monitors provide all the required metrics.

3. Monitor Deployment and Validation

The custom monitors are put into sanity test prior to being set up in real-time production environment.

4. Data Collection and Analysis

Here periodic data collection and analysis is performed to check any potential issues and bottlenecks. If there are any bottlenecks, recommendations are made to address them. Once the recommendation is implemented, further data collection and analysis is made to check the efficacy of the solution.

Covering the Entire Life Cycle

This approach advocates a two-pronged strategy for non-intrusive software testing. The first level focuses on conducting a comprehensive study of the engagement. Firstly, there comes the requirements collection phase, which takes the form of a formal document stating about the testing strategy. In the second phase, the deploy monitors and the collected and analyzed data gives rise to the recommended solution.

Monitoring:

Server Side Monitoring

Resources such as CPU, memory and network should be monitored at the server end for each run. The performance improvement will need to be monitored at Web Server, Application Server Database Server and the middleware.

Client Side Monitoring

For each run, the following parameters should be monitored at the client end:

Transaction Response times

Network Throughput (KB/sec)

Transactions/sec

Errors logged during the test from client side perspective.

Reporting:

A comprehensive report with the findings, analysis and the recommendations must be prepared. This report should also include the improvements or the effect of these recommendations.

Thus, it can be said that the strategy and structured methodology of non-intrusive software testing provides a realistic and implement able solution for such a scenario.

About the Author

[CSS Corp](#) provides end-to-end [Independent Software Testing](#) services on a flexible delivery model to banking, finance, insurance, manufacturing, airline and telecom industries ensuring 100% test coverage and on-time release.

Source: <http://www.tntarticles.com>