To Test or Not to Test?

By Brian Doyle, Delivery Manager
April 1, 2014

What is COTS?

Commercial Off-The-Shelf (COTS) software is pre-built software usually provided by a 3rd party vendor. COTS applications typically require configurations and / or customizations that are tailored to specific requirements of the customer for their software solution. The implementation of COTS software has become increasingly more common as part of the IT strategy within many organizations.

Below are two assumptions most organizations make when they choose to implement a COTS-based solution:

- Since COTS software is already commercially released and (we assume) vendor-tested, there is no need for the organization to test the COTS application
- When testing is considered as part of a COTS implementation, the effort should be relatively lightweight and straight-forward with few issues expected along the way

While it’s true that the implementation of COTS software does not follow the traditional software development lifecycle, the problem with the assumptions above is that the organization is not looking at the larger picture and recognizing the impact that configuration, customization and integration of the COTS software will have within the organization’s IT environment.

If an organization were to perform a proper risk analysis of the full impact of a large-scale COTS implementation, they may realize that a testing approach is required that entails effort that is on par with (or exceeds) that of a custom development project.

> “Testing the integration points is clearly required”

Types of Testing

The COTS application is assumed to be stable and to have been unit and functionally tested by the vendor. Minimal functional testing of the core product should be required. Therefore, don’t focus on retesting the features of the COTS application itself. Functional testing activities should focus on customized and enhanced areas of the COTS application in accordance with the organization’s testing methodology.

System Integration Testing

Since the COTS application is most likely communicating with other systems, testing the integration points is clearly required. Again, the goal of integration testing is not to verify the functionality of the COTS application, but to assure that the information sent and received from other applications is correct. These integration points should be identified as high-risk areas for potential defects.

As well, if the COTS system is replacing a legacy system within the organization, data migration from the existing application to the COTS application must be tested to ensure that the existing data has been correctly migrated into the COTS application. GUI- and API-based service functions should also be thoroughly tested where applicable.

> "Role-based security testing is often a good candidate for test automation"

Security (Role-based) Testing

Security access (roles / privileges) testing should be performed on the COTS application to ensure that vulnerability and accessibility concerns are addressed by performing access control and multi-privilege tests with user accounts. The most important feature of this testing is to verify the individual roles and their permissions to each function, module and unit of the COTS application. This testing is generally conducted using a test matrix with positive and negative testing being performed. Role-based security testing is often a good candidate for test automation.

Performance Testing

All high impact workflows that are critical to the business should be performance tested with realistic usage patterns. These patterns should be simulated with large volumes and loads based on realistic user distribution. Aside from addressing the stated risks identified during the risk assessment phase, performance testing also aims to achieve the following benefits:

- **Gauge the readiness of the COTS application**
  This helps to not only ensure that the COTS system will meet its stated service level agreements (SLA's), but will also help to set appropriate user expectations around system performance. An initial pre-production performance testing exercise will also establish a baseline against which to compare future performance tuning measures.
- **Assess the organizational supporting infrastructure**
  The COTS application will be dependent upon organizational infrastructure to support its performance targets. While organizational infrastructure is also responsible for supporting other applications, it should be assessed in terms of its direct support of the COTS application performance targets.
- **Identify performance tuning techniques prior to release**
  Pre-production performance analysis will allow the COTS performance testing team to understand, plan for and experiment with tuning techniques that can be used in the production environment to address system performance concerns.

User Acceptance Testing

User Acceptance Testing (UAT) is required as a final confirmation of the readiness of the COTS application and business preparation prior to implementation. During this phase of testing, it is assumed that no major issues with COTS system functionality will be identified and that the only anomalies identified will deal with usability, data content or training issues. When the business users have completed UAT, a formal signoff process is recommended to officially signal approval by the business to implement the system.

Other Types of Testing to Consider

Depending upon the type of COTS application being implemented and its purpose, consideration for other types of non-functional testing (in addition to performance testing) may be required. Listed below are additional testing activities that may be considered for COTS projects. The testing types chosen should correspond to the specific non-functional requirements and SLA's for each system; therefore, this article will not go into the details of each.

- Portability Testing
- Compliance Testing
- Failover and Recovery Testing
- Scalability Testing
- Security (Vulnerability) Testing
- Maintainability Testing

Summary

Given that the information above can apply to any number of COTS implementations, you can probably guess that the answer to this article's title question is "We should definitely test!"

With potential short term savings in mind, it may be tempting to dismiss the need for testing COTS applications - but several factors need to be considered. Take the time to analyze the COTS project in order to balance the cost of testing against the potential risk and the cost of failure.

As organizations rely more on vendor-developed products to meet their needs, a test strategy for COTS applications should be ingrained within the organization's IT methodology. Implementing a COTS application that has been vendor-tested and commercially released does not relieve the customer of the responsibility to test in order to be assured the application will meet business and user requirements.

About The Author

Brian Doyle is a Delivery Manager with Professional Quality Assurance with 15 years of experience in the IT industry, the last 12 years spent in software testing. His experience includes manual and automated testing, test planning, test coordinating, test management and quality assessments. He has worked in a variety of industry verticals including healthcare, insurance, banking and financials, gaming, transportation, public safety and public utilities. Brian has been with PQA for over 10 years and is always looking forward to his next testing challenge.