CASE STUDY

TestArchitect

Halliburton reduces end-to-end integration testing time by 99%

LogiGear

Business Needs and Challenges

Halliburton's product enhancement (PE) software technology organization develops critical internal enterprise software. The business unit's 2 major software products are developed on a Windows platform using the .NET framework. This software is extremely complex with a rich user interface, vast source code, and N-Tier architecture. Software defects have the potential to impact product quality, thus impacting Halliburton's ability to provide services to global customers. In order to minimize that risk, extensive testing is performed for every update.

The PE software technology organization had been using Agile practices for a number of years and recently implemented Continuous Integration and deployment, supporting multiple feature branches to shorten development sprints. By the end of every 2-week development sprint, the software was tested for being potentially shippable. New tests were created and executed along with any necessary system integration regression testing. Test Reports were then sent to product management and development teams for review and prioritization of bug fixes. After the bugs were addressed, the software was sent back to the team for verification testing, which could be run overnight. Major updates required running the full regression suite, which took approximately 8-10 weeks to complete. Due to the time required to perform manual testing, the anticipated improvements in the frequency of product release failed to be fully realized.

FAST FACTS

INDUSTRY:	• Energy
WEBSITE:	• www.halliburton.com
COMPANY SIZE:	• 50,000+ employees
LOCATION:	• Houston, TX
LOGIGEAR DELIVERED:	• TestArchitect™ • Software Testing Services • QA Training & Consulting
BUSINESS IMPACT:	• Over 2,000 test cases automated resulting in regression run time reduction from 8 weeks to 5 hours. Achieved endto-end integration using Microsoft's Team Manager.

QA Manager, Cheronda Bright took over the PE Software testing group shortly after the development team implemented the feature branches. In the new structure, functional enhancements were being performed in the development branch while the user experience team updated the user interface in a separate feature branch. In order for testing to keep the pace with development team's velocity, it was clear that major changes were needed. The options were to modify the regression testing strategy, cut the feature scope, extend the schedule, or speed up the velocity of testing. Due to the critical nature of the software, compromising on features or the amount of testing was immediately ruled out. This left the team with the challenge of reducing testing cycle times and maintaining quality. While test automation appeared to be the obvious solution, a number of challenges had to be overcome in order to arrive at a workable solution.

Multiple process management platforms between functional roles were hampering collaboration among Agile teams. The development team was using Microsoft's Visual Studio 2012 for source control. Scrum Masters and product managers were using TFS for user story and bug management, while the test team used Hewlett Packard's Quality Center (QC) for test management, execution, and status reporting.

The separate platforms made it difficult for teams to keep track of the latest changes which often resulted in duplicating tests that had been run in unit testing, or failing to run needed tests. The teams' effectiveness was also impacted by the lack of a single product traceability matrix for requirements, code tests, and bugs. But the biggest issue was that after initial success, automation coverage decreased with each new build due to the amount of maintenance work required to keep the automation scripts up-to-date.

The Vision

The vision for Halliburton was to implement a fully integrated development and test environment. Achieving this vision would require a significant organizational change and any technology solution employed would have to provide cross-team collaboration to maximize efficiency. Both the test team and the development team were asked to provide input on what they believed was essential to make the project a success. Reducing test time was the top requirement for both teams. The development team also wanted testing of impacted code changes executed automatically after each build. And, it was clear that both teams required better visibility and reporting.

Since the development team had already implemented Microsoft Visual Studio and TFS, switching to Microsoft Test Manager (MTM) for the testing platform appeared to provide the best solution for a unified platform. There were a number of potential benefits to be gained by replacing QC with MTM. Testers would be able to select specific test suites and create test configurations based on the project plan in TFS, and the entire Scrum team would have access to test and project plans. By creating status reports for all aspects of the project, the issues of duplicating or missing essential tests could be eliminated. Better reporting would allow traceability through the utilization of linking test cases, bugs, and user stores in TFS. Also, eliminating the QC platform would provide a significant cost reduction to the organization.

The biggest hurdle in moving to a common platform was the selection of a test automation solution. The native coded UI automation tool in MTM was not robust enough to scale automation to meet the team's needs, making it necessary to evaluate third-party tools that would integrate with MTM. The tool would also need to support test execution on N-Tier architecture and auto-deployment of the testing environment utilizing build definitions. Automation would add a level of complexity and require a new way of approaching testing for the test team. For the project to be a success, the team would need assistance in implementing the automation strategy, acquiring the necessary skills, and integrating the platforms.

Key Initiatives

- Implementing Test Automation without impacting Project Schedules
- Training the test team globally on ABT, MTM and TestArchitect™
- Migrating manual tests from QC to MTM by recreating them using ABT
- Creating detailed status reports that include images and log files on failure
- Automating and certifying manual tests before moving into production
- Clearly defining a workflow between the manual and automation teams
- Running tests unattended by auto-deploying the execution as a part of the build definition

Implementation

Two PE test managers were assigned to work with LogiGear as automation leads and to implement the strategic direction. Their initial task was to identify tests that would be the most beneficial to automate. All 3,000 test cases were audited and 600 were selected for the initial automation effort. The automation work, which included designing and creating tests using the ABT method, was outsourced to LogiGear. Leveraging LogiGear's automation expertise allowed the PE test team to continue the current pace of testing.

LogiGear started the automation effort by creating custom actions in the TestArchitect™ action library that were unique to PE software. The PE project manager provided domain expertise that helped LogiGear identify an application type of low-level action, which PE testers could then use to assemble a high caliber of business-level actions. At the same time, the PE team worked to customize MTM and TestArchitect™ so that there would be a seamless workflow between the 2 tools. The objective was to provide testers with a self-serve model that would allow the running of any combination of automated and/or manual tests as needed from within MTM.

The resulting processes made it possible for the PE manual test team to create tests using business-level actions' definitions within MTM. From business-level actions, automation engineers created and stored tests in TestArchitect™'s test repository. Testers could then select individual tests for manual testing or create a full automation test suite to be run, by using the association feature-a TestArchitect™ plug-in that installs in MTM to connect the test repositories. The integration and workflow process was tested and refined by assembling and running a number of test suite variations from MTM-first in a separate DIT environment (physical) and then in a SIT environment (virtual/physical). After certifying that all tests were successfully executed from within MTM and the test objectives were met, production (virtual) environments were configured and deployed to meet the maximum run-time of 5 hours or less, for a full suite of tests.

The final stage of the integration project involved the migration of all remaining manual tests from QC to MTM. The initial plan was for the PE team to continue using QA and migrate manual tests as time allowed. While this approach was resource-efficient, it didn't solve the visibility and reporting issues as long as tests were being executed from QC. A third-party converter tool was considered to push the data over from QC to MTM. Although the converter tool would speed test migration, it wouldn't address test design issues. Ultimately, it was decided to re-create tests using the ABT method. After considering the cost and resource requirements, test migration was outsourced to LogiGear. The experienced LogiGear team eliminated redundant test cases and quickly deployed highly efficient, reusable tests. Ultimately, the long-term success of the conversion would be dependent upon the processes and skills of the PE test team, so the decision was made to invest in test and automation training. Hands-on instruction in test design and ABT provided testers with the knowledge and correct approach for building complex tests required for the PE toolkit.

The Results

Halliburton's PE team set out to improve on Continuous Integration while achieving maximum test coverage with highly compressed release cycles that consisted of 2-week sprints and multiple feature branches. The collaboration between Halliburton and LogiGear has proven to be a winning combination for increasing testing velocity while maintaining quality. After automating over 2,000 tests, the regression suite that previously took 8 weeks could now be completed in 5 hours using a combination of 17 virtual and physical machines.

Additionally, by changing to the ABT method, the average time to create and run a single verification test was reduced from 3 hours to less than 10 minutes. The project was successful in achieving the objective of reducing the test team's cycle in order to match the velocity of testing with development. TestArchitect™, along with a common ALM platform and LogiGear automation support, made it possible to quickly create and deploy tests required to keep up with Continuous Integration, multiple feature branches, and variations. While the initial plan was for a complete handover of test automation to the PE test team, it was ultimately decided to outsource the automation engineering and execution of automated tests to LogiGear. This approach was a much more efficient use of the domain expertise of Halliburton's testers, while leveraging the efficiency and low cost of LogiGear's automation engineering staff.

Today, Halliburton and LogiGear work collaboratively to fine-tune the testing workflows, with the PE team creating all test requirements and manual test cases in-house. LogiGear's automation team works 1 sprint behind software development to create automated tests. Completed tests are certified by the Halliburton team and moved to production, where they are automatically executed from MTM.

About LogiGear

LogiGear is a leading provider of software testing services, test automation and application development and maintenance. We help organizations deliver better products while saving time and money. Since 1994, we have completed testing projects with hundreds of companies from early stage startups to Fortune 100, across a wide range of industries and technologies.

LogiGear USA - Headquarters
1730 S. Amphlett Blvd. Suite 200
San Mateo, CA 94402
Tel : +1(800) 322-0333 (Sales)
: +1(650) 572-1400 (Support)
Website : *www.logigear.com*
Email : *sales@logigear.com*