Implementing an Automated Testing Program

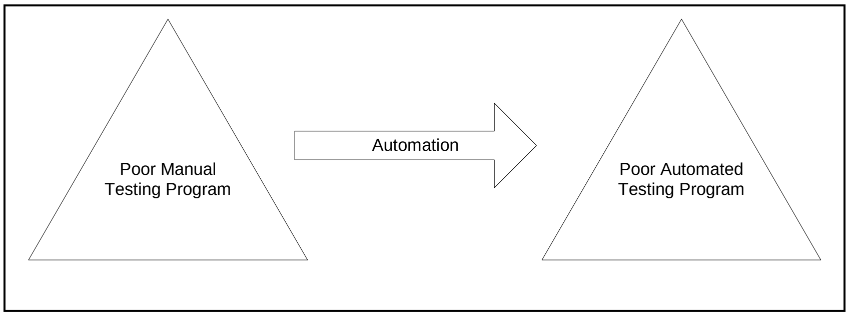
An Innosphere ‘Ritepaper’

INNOSPHERE SYSTEMS DEVELOPMENT GROUP LTD.
147 WYNDHAM ST. NORTH, SUITE 306
GUELPH, ONTARIO, CANADA, N1H 4E9
TEL: 519.766.9726

WWW.INNOSPHERE.CA

An Automated Testing Program (ATP) is an automated version of your existing manual test program. Many QA organizations attempt to automate without first assessing the effectiveness of their existing manual testing program. Automation of a poor manual program results in a poor automated program.

Figure 1.1 – Automating the Existing Manual Program



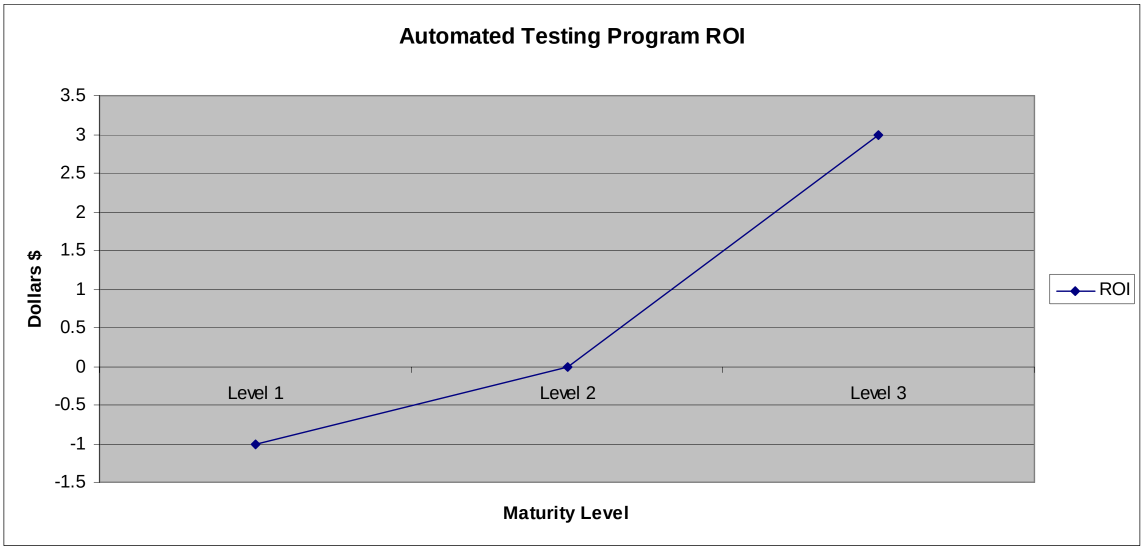
Implementing an ATP within a Level 1 organization, as defined by the Innosphere Testing Process Model (Figure 1.2), can potentially increase testing costs by up to 50% with little improvement in product quality.

Figure 1.2 Innosphere Testing Process Model (ITPM)

Level 1 Unstructured	At Level 1 processes are not documented and not consistently applied. Success of software testing projects is very much dependent on key individuals. The organization does not consistently apply QA management to the process or effectively use technology tools to manage the process. *Organizations at Level 1 are at high risk of cost overrun, schedule overrun, and of producing low quality software products.*
Level 2 Structured	At Level 2 processes are documented and applied consistently on all software testing projects. The organization applies QA management to the process and effectively uses tools to manage the process *Organizations at Level 2 are at low risk of serious cost overrun, schedule overrun, and are able to produce quality software products.*
Level 3 Competitive Advantage	At Level 3 processes are integrated into the corporate culture and are continually improved. The organization applies innovative QA management to the process and effectively uses innovative tools to improve the process. Process data is collected, measured, and analyzed. *Organizations at Level 3 are at able to achieve competitive advantage by reducing cost overrun, reducing schedule overrun, and producing superior quality products.*

Typically the breakeven point in terms of the return on investment (ROI) of the ATP will not occur until an organization has achieved Level 2 or higher (Figure 1.3). This is the point where the cost savings from improvements in testing equal the cost of implementing the ATP. Some organizations implementing an ATP may not decrease overall testing costs but will experience increased product quality that may result in increased product sales and/or a higher customer retention rate.

Figure 1.3 Automated Testing Program Return on Investment by Maturity Level



Things to Consider Before Automating

If you are considering an ATP it is likely that your current testing program is not achieving your goals. Implementing an ATP is a possible solution but should not be considered a goal within itself. There are a few things you should be aware of before you decide whether to implement an ATP

- There will be significant expenditures for the initial purchase of automated testing software licenses, training, and annual maintenance fees.
- If you want to conduct both performance and functional testing you may need to purchase more than one type of test tool or multiple modules of one suite.
- In the beginning, overall costs will increase as the current work of testing will need to be conducted while the additional work of developing the ATP is simultaneously carried out.
- Implementing an ATP requires that a documented test process be in place, in addition to a documented software development process.

- Using the simple capture and playback functionality of automated testing software adds little value to an automated testing program. An effective ATP requires programmers that can develop test scripts using their knowledge of the tool's scripting language.

- It rarely makes sense to automate everything, so you will always have a need for manual testing.

- Test scripts need to be maintained and updated as the application under test is modified. Maintenance activities can be a significant component of the ATP

- To maximize the effectiveness of the test tool, testability needs to be built into the code of the product under test. This requires a cooperative effort between the test engineers and the product developers.

- Design standards need to be developed for writing test scripts in order to achieve maintainability and reusability.

- Developing an ATP for a sophisticated product may take up to 2 years to complete.

Prerequisites for Automating

The first step in determining if an ATP is appropriate it to define the goals of your testing program. Once those goals are defined you can determine if an ATP is appropriate for your organization. Developing an ATP requires a significant amount of resources in terms of cost and time. Skilled personnel are required as well as a long term commitment from upper management.

A well documented and proven test process is critical to the success of the ATP. This includes formal documented test cases following a particular standard and a system to manage the test cycle. If your existing process is not formalized this may be where to focus your effort. Attempting to automate a poor or non existent test process is ineffectual.

In addition to the test process, the overall software development process must be at an adequate maturity level in order to facilitate an ATP. If the current process is ad hoc and not standardized then the implementation of an ATP will likely fail. The more mature your process is, the greater the potential ROI for the ATP.

If the above prerequisites have not been met you need to work towards achieving them or reconsidering your decision to automate. Perhaps improving your existing manual testing process may be the most beneficial next step.

Figure 1.4 – Automated Testing Program (ATP) Prerequisites

- Defined goals
- Commitment from upper management
- Adequate budget
- Availability of skilled resources
- Documented software development process
- Documented test process and test cases
- Test management system

Benefits of Automating

If your organization has the necessary prerequisites (Figure 1.4) then the following benefits may be achieved.

- Improved product quality.

- Reduction of test cycle time or an increase in the number tests that can to be conducted during the existing test cycle.

- Improved performance testing ability. Performance test tools can simulate the simultaneous actions of thousands users using the product under test. These tests are not feasible with manual testing

- Automated test scripts allow for repeatability of tests. A manual tester may execute a test case in a slightly different manner each time. This can make it difficult to reproduce an error reported during testing.

- Automated test tools automatically generate the results of test case execution and provide reports for analysis

- Automated test scripts can be initiated and then be left unattended for hours. This allows testing to be conducted 24 hours a day without the need for additional personnel

- Automating mundane and repetitive tests allows manual testers to focus on more sophisticated tests.

- Automation can allow the same test to be run on different configurations simultaneously.

- In addition to testing, automated tools can support other areas of the Software Development Lifecycle such as requirements management, design, and programming.

Selecting the Correct Test Tool

Once the necessary prerequisites are in place and you have made the decision to implement an ATP the first step is to review the available test tools. This can take a significant amount of time and effort depending on your goals and the number of tools required.

The only way to truly evaluate a tool is to apply it in your software development environment. Most automated software vendors offer evaluation copies allowing you to test the tools on your products and determine if they meet your requirements. Every automated tool is different. Some tools may not work with certain applications or components of an application so it is critical to ensure that the test tool can support your products.

Once you have acquired the desired tool or tools the next step is to determine the best method to introduce the tool to your organization and align it with your overall ATP implementation plan.

Deciding What to Automate

What product to automate

Some products may be more appropriate for automation than others. Take into consideration the complexity of the product and its lifecycle stage. Complex products will increase the automation effort and products under going significant future enhancement will require additional efforts in maintaining automated scripts. An ideal product for initial automation may be an established product in the maintenance phase that will not be undergoing significant development.

What test cases to automate

Decisions must also be made regarding what test cases are best to automate. It is never possible or economical to automate every test case. If a test case is only going to be used once it may not be practical to automate it. More complex scenario test cases can take a great deal of time to automate and may be more amenable to manual testing. The best choice is test cases that are going to be executed repeatedly over the long run as they will generate a large ROI.

Planning for Implementation

An implementation plan is required in order to increase the probability of early success and organizational buy-in. If the first automation project is a success it will create momentum within the organization.

Determining how to implement the ATP is an important step. It will be necessary to plan for an increase in the number of available resources or an increase the length of the test cycle to accommodate the additional workload. Furthermore, regular testing activities must be conducted concurrently with the development and implementation of the ATP.

If existing personnel do not have adequate programming skills it may be necessary to hire qualified personnel if educational upgrading is not an option. If possible test engineers should receive formal training on the test tool prior to implementation. Hiring an external consultant can expedite the learning process for existing personnel. Whatever training strategy is decided upon should be documented and budgeted for in the implementation plan.

Strategies for Implementation

There are several strategies for implementing the test tool. One possibility is to introduce automation in small increments across all products or projects. This could involve using the automated tool for environmental setup to generate data to expedite current manual testing. The test tool may be introduced by generating a limited number of test scripts for build verification or regression testing. Alternatively full automation could be attempted on a low risk pilot project.

Designing the Automated Test Program

All the necessary scripts to be developed should be documented. The automated test scripts will only be as good as the test cases that they were based on so it is necessary to have test cases that have been developed by the product's subject matter experts.

In addition the relationships and dependencies of the test scripts need to be identified and documented. Any given test script may have preconditions that require one or more test scripts to have been previously executed. Those previously executed test scripts may also have had multiple preconditions. In a large complex product managing the interrelationships of test scripts can be a very demanding task.

Development of test scripts need to follow development standards that apply to the programming language being used. Test scripts developed without standards may be difficult to reuse or maintain.

As the test scripts are developed they need to be placed under configuration management and based lined in the same way test plans and other process documentation is.

An important, and often overlooked task, is the testing of test scripts. Just like any other programmed code test scripts may contain errors. Once testing has been completed, they are ready to be executed in the actual testing cycle.

Executing the Automated Test Program

The test tool must be installed and setup in an adequate test environment. A development architecture similar to that for the software development project must be setup. The execution of both automated and manual test cases will need to be coordinated using your test management system. Your test tool will generally have some type of reporting mechanism to indicate if the test script passed or failed upon completion.

Deciphering Test Results

A failed test script does not necessarily indicate that a defect exists in the product. The result could be a false negative caused by a modification to the code that was not accounted for in the automated script. The code modification may have been authorized or unauthorized, but either way a breakdown in process has occurred. To manage this, any changes made to requirements, design, and code must be under configuration management to ensure that test scripts get updated prior to the initiation of the test cycle. If the test scripts do not get updated they may fail even though defects do not exist. If extensive changes have occurred it may not even be possible to update the affected scripts and testing will need to be delayed or manual testing will need to be used.

Similarly, a passed script does not necessarily indicate that a defect does not exist. The result could be a false positive, meaning the test script failed to detect an existing defect. To avoid this problem it is critical to thoroughly test all test scripts.

At the end of the test cycle a post mortem should be conducted to evaluate the effectiveness of the initial automation attempt. A review of the initial automation attempt will provide valuable lessons. Results may indicate that the entire automated test program and all the scripts need to be revised or even completely redesigned. This is not a negative result but simply part of the learning process necessary to develop an effective ATP.

Summary

- Implementing an ATP is a significant task and should not be entered into without some analysis and planning.
- The necessary prerequisites (goals, commitment, budget, skilled resources, processes, and management) should be established prior to implementing the ATP.
- Selecting the appropriate tool or tools is critical and significant resources must be allocated to this task.
- It is important to identify the appropriate products and test cases to automate.
- An implementation plan is required in order to execute the ATP effectively.
- The results of the executed test scripts will need to be analyzed.
- The ATP should be reviewed on a regular basis so that improvements can be made. In the long run the ATP can become an effective addition to an organization's quality culture.