

Regression Testing

Article #1 in the 'QaTraq Professional Series'



QaTraq Professional Series

The 'QaTraq Professional Series' is a series of articles looking at how you can tackle some of the more critical aspects of your test process with QaTraq Professional. In this first article we examine how you can take command of your regression testing obligations.

www.testmanagement.com

PRO





Introduction

As testers we are under an obligation to carry out regression testing. This is not an obligation to be taken lightly! Failure to address this obligation has led to many products being released with defects that should have been caught. Defects, that would have been caught, had the right regression testing strategy been employed.

Sometimes though, this obligation to undertake regression testing can leave you with a bewildering list of testing tasks that ought to be completed. So where do you start with your regression testing? In fact where do you end with your regression testing? There is no getting away from the fact that it is difficult identifying which tests to re-run as regression tests and how to track those tests against all the other tests and testing that needs to be completed. How do we remove this confusion and bewilderment?

What is Regression Testing

Before we jump into showing you how QaTraq can help you, lets just review exactly what a regression test is. The Wikipedia entry for Software Testing identifies regression testing as "A regression test re-runs previous tests against the changed software to ensure that the changes made in the current software do not affect the functionality of the existing software."

So in essence we are looking to identify tests or test scripts that we have already run once. Yet we need to re-run these tests on a regular basis to ensure that any changes to the software have not affected existing areas of the software. To undertake this we are going to make use of the *Templates* and *Sets* functionality in the QaTraq Pro module. We will follow the steps outlined below to select, create and re-run existing tests and test scripts:

1. Identify existing test scripts that should be re-run
2. Create *templates* from test scripts
3. Select and group *templates* into *sets*
4. Select and deploy regression test *sets*
5. Track our regression test results

QaTraq Scripts, Templates and Sets


First though, lets take a quick look at what QaTraq *Templates* and *Sets* have to offer us. We will also look at some definitions for terms you will find used in QaTraq on a regular basis.

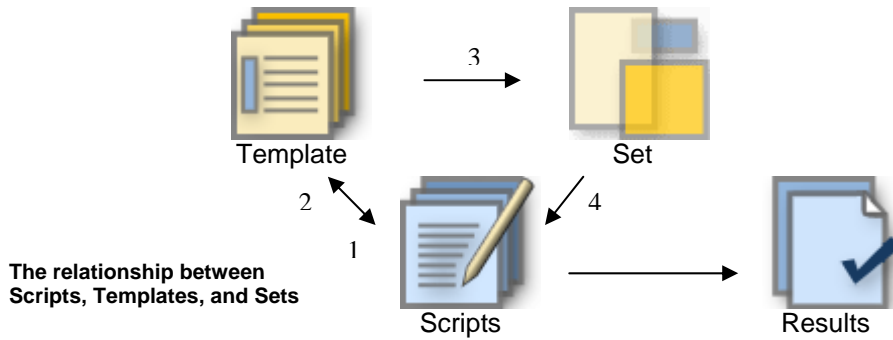
Scripts: A Test Script is a document that describes in detail how a group of related tests should be conducted. A Test Script contains information outlining the test, defines the pre-requisites and contains a number of test cases. The user enters the outline and pre-requisite text and then links the Test Script to a number of test cases that will be contained within the Test Script. Every test script has a number of associated test results; one test result record for each test case included in the test script.

Templates: A template is essentially identical to Test Scripts except that they don't have any test results associated with them. So you can develop a Template Test Script over a period of time and then, when you are ready, copy the Template to create a Test Script (which has the associated test results).

Sets: A Set is a list of references to Templates. To emphasise that point; a Set is just a number of references to a number of Templates. You can create a Set without any references to Templates and then add references at a later date. Sets then provide you with the ability to copy multiple Templates in one go to create multiple Test Scripts.

Before we look at how Sets can help us control our regression testing we should make sure that we understand the relationship between Scripts, Templates and Sets.





From this relationship diagram you can see the following:

1. Templates can be copied to create Test Scripts
(in the process creating the associated test results for the script)
2. Scripts can be copied to create a Template
(creating a Template with no test results)
3. Sets consist of references to zero or more Templates
4. A Set of many Templates can be copied to create many Test Scripts
(in the process creating the associated test results for the scripts)

So for example you might create 4 Templates. These Templates might be the basis of a set of regression tests that are run for every build of the software that is released to the test team. So you create a Set that references these 4 Templates.

When you receive a new build of the software you know you have to run these 4 regression test scripts (which are currently stored as templates). Using the Set you created, which references these 4 Templates, you include the Set under a Test Design document. This act of including the Set creates 4 Scripts, along with their associated test results, specifically for testing this new build of software.

Now we have an understanding of Templates and Sets we can start to examine how we can create our Sets of regression test scripts. Lets start then with "Identify existing test scripts that should be re-run"

1. Identify existing test scripts that should be re-run

Using the "Pro Scripts Per Build Count" report from the QaTraq we can scan an overview of the test scripts we have already executed, looking for test scripts that might make good regression test scripts.

Product Version	ver 1	ver 2	ver 3
Test Design			
Test Script			
Test Design			
Test Script 1	3 / 13	0 / 13	
Test Script 2		1 / 8	
Test Script 3			0 / 8
Test Script 4			0 / 8

Pro Scripts per Build Count

For example in the report shown on the left, we can see that we had 3 test failures out of 13 tests run in Test Script 1 (for Version 1 of the product). A re-run of that same test script in Version 2 shows that no tests failed (proving that we fixed the issues). However, we also saw a test failure with Test Script 2 and our experience may now

tell us that this is a high risk area and that perhaps we should be using this as a regression test on a regular basis. Using this "Pro Scripts Per Build Count" report and our experience of the product under test we might look for regression test script possibilities using the four criteria identified below:

- i) Test scripts which have had high failure rates (high risk areas of functionality)
- ii) Areas that are likely to be subject to significant changes

- iii) Areas with high dependencies on other modules or components
- iv) Or just simply areas that we haven't touched for some time

In the example above we might choose Test Scripts 2 and 3 as potential regression test script candidates (Test Scripts 1 and 4 might be ignored because we know the test failure showing is a low priority issue.). With some suitable candidate test scripts identified we can start to create a regression test "set" which includes these test scripts.

2. Create templates from test scripts

We have identified a number of test scripts that we think are good candidates for running as regression test scripts. Now we need to clone these test scripts, or to be more accurate in QaTraq terms, create *Templates* from these test scripts.

Using the Templates copy functionality, shown in the example on the right, we search for and then copy our existing test scripts. This action creates new templates. These templates can then be used as the basis for constructing a regression test set.



Copying Test Scripts

One key point to remember when creating these templates is to set the Approval values to 'copy latest'. This functionality is explained in more detail in the user guide and is generally used when setting up and running a test script review process. This is of no consequence for us now, as we need to make sure that all Templates have an approval value of "Copy Latest".

3. Select and group templates into sets

Next we pull these newly created Templates into a Set of regression test scripts/templates. Using the sets functionality in QaTraq we create a new Set. With a new Set created we search for, and add references to these new Templates we created in step 2 above.



A Set containing two Templates

We now have a new Set, perhaps called "Regression Test Set for Test Area Widget". This set has two references to two templates. We are now ready to deploy this set as a set of regression test scripts, as and when needed.

4. Select and deploy regression test sets

Let's say we've just received a new version of the software with several areas of new functionality included. Our first thoughts turn to regression testing the old functionality, which we knew was stable before all these new changes were made.

First lets select the regression test set that we've already created. Then we'll *include* this set in a Test Design document, creating two new test scripts with which to start our testing on the new version of software. In the example above we can see that the act of including this set allows us to select, which



Including Sets under a Test Design

Test Plan and Test Design document, we should place the new test scripts under. It also allows us specify the version of software they should apply to and other details like the tester that should run the new test scripts.

5. Track our regression test results

So with these steps we've taken this Set and quickly recreated two new test scripts that are assigned to specific testers ready for the testing to be started. From the updated Pro Scripts Per Build Count" report example shown below it is not difficult to see that we can repeat these steps each time we get a new build or version. In the example below we decided to run this regression test set every other version.

Product Version	ver 1	ver 2	ver 3	ver 4	ver 5	ver 6	ver 7	ver 8	ver 9
Test Design									
Test Script									
Test Design									
Test Script 1	3 / 13	0 / 13							
Test Script 2		1 / 8			0 / 8		0 / 8		0 / 8
Test Script 3			0 / 8		0 / 8		1 / 8		0 / 8
Test Script 4			0 / 8						

Pro Scripts per Build Count

In the example above we can see that our regression test strategy caught a test failure using Test Script 3 in Version 7 of the software. However, we were able to put this right by the time we got to the final release version, version 9. Whilst this may seem like a lot of effort just to reuse 2 test scripts on a regular basis it is easy to see that this approach scales up to using sets which might contain 10, 20 or even more templates. With this number of templates in a set you find yourself with a very powerful tool for creating and tracking regression testing.

Summary

In this article we have quickly taken you through the difficult task of managing and tracking your regression testing. From Identifying Test Scripts to Tracking Results we have seen how using QaTraq Professional Templates and Sets can simplify this once bewildering task.

About QaTraq Professional Upgrade

The QaTraq Professional Sets and Templates functionality discussed in this article is part of the QaTraq Professional application only. To install and run QaTraq Professional you will need to:

1. Install and Set-up QaTraq Professional
<http://www.testmanagement.com/download.html>
2. Purchase and Install the Professional Licence
<http://www.testmanagement.com/prouppgrade.php>

The QaTraq Pro User Guide covers functionality discussed in this article in detail:

QaTraq Pro User Guide: <http://www.testmanagement.com/prouppgrade.html>

Contact Us

If you would like to provide us with any feedback or comments on this article please contact us through our 'Contact Us' web page given in the link below:

Feedback: <http://www.testmanagement.com/contact.html>