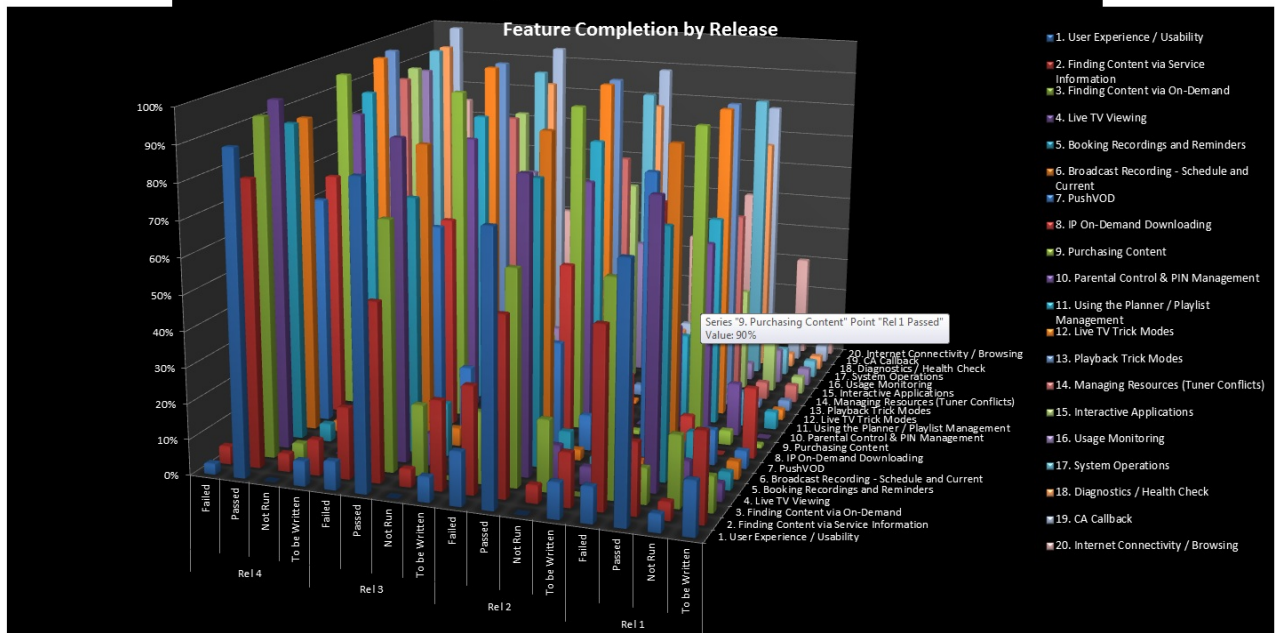
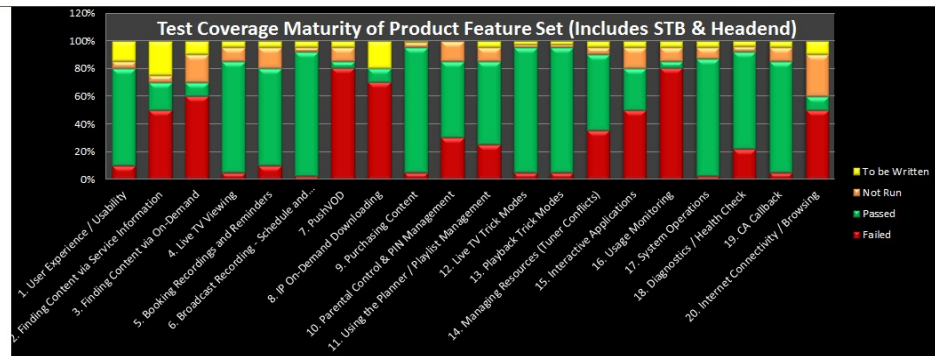


TEST AUTOMATION METRICS – WHAT DO YOU REPORT ON?

Metrics

One of the fun things of test automation is that, since you do not have to do all the tests manually, you can spend some extra time coming up with test metrics. Test metrics are tricky to do well in any situation, but in a situation where there is an abundance of metrics, such as in a test automation setup, the choice of metrics becomes the key first step. What are the metrics to look at? Code coverage? Number of tests passed vs Number of tests failed? Duration of the tests over time? Number passed now vs number passed in previous runs? Newly automated tests added since last run? You can keep going in dreaming up new metrics, but which ones will actually make sense and become representative?? And of course, how do you ensure you do not spend ages ploughing through your data to gather these metrics manually?

If you just take a test automation tool off the shelf it probably has an immense amount of options to measure on and report on, but the risk is always that you start generating reports and metrics that are not quite representative, or even worse, give a tainted view of the actual situation. So how do you make sure you don't end up with a jungle of metrics?



Audience

First thing you need to know is who is the audience of your metrics? There is a huge difference in what different levels in an organisation consider useful metrics. One manager can be mainly interested in the time spent automating versus the time won by automating; e.g. the extra time now available for testing other stuff, the stuff that matters, while a test manager might be more interested in the functional areas of the application covered and to what extent they are covered.

Type of metrics

I will not attempt to dream up the perfect metric, for every environment and situation one metric might be better than the other. It all depends on the context, the persons you are reporting to, targets of each particular business area etc.

What I do want to touch upon is the awesome power you have with metrics coming out of automation. Since your tests can run rapidly and often, there are lots of runs that can be measured. In other words, you can gather a lot of data, a lot of historical data. When reporting on metrics like amount of tests passed versus the amount failed, it generally will be a snapshot of some test run. Why limit the metric to a snapshot when you have living data at hand?

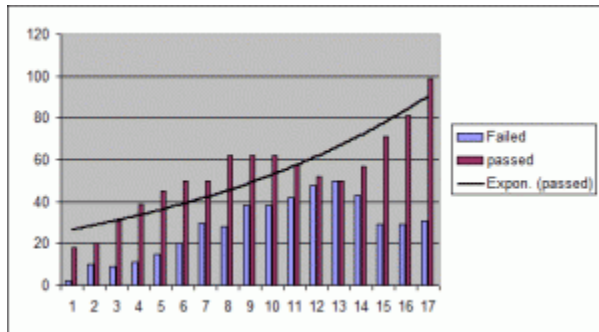
The strongest metric to show to any manager is trend lines; you need to report on the amount of tests passed vs failed or the amount of tests added to the automation suite? Need to report metrics on code coverage? All of these metrics can result in a trend line. Show the “upwards trend” and managers are generally happy without even knowing what they are looking at.

There are of course some pitfalls, the main one I have made was having a downwards sloping trend line. That seems like a bad trend, even though it can be a totally perfect trend, the sight of a trend line going down generally makes managers nervous, they expect things to always go up.

Be prepared to explain a downwards trend, cause sometimes you cannot escape a downwards or flattening trend line!

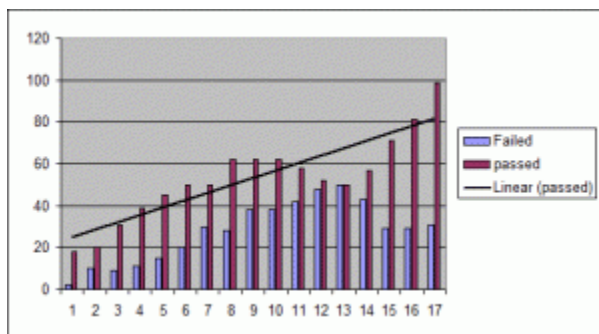
Graph examples

Below are two graphs, both with the same data, and a trend line set on the same data. The three charts however, when looking at them each tell you a slightly different story due to the style of trend line chosen for the chart.



Making the numbers seem a bit more positive than they really are by using an exponential trend line.

The exponential trend line paints a strong picture, however when using it, be prepared to explain the fact that despite the lack of growth at about two thirds of the graph, the trend is still upwards. This is a difficult story to tell.



The linear trend line gives an indication of the overall trend, when close to flat-lining you know you have a problem, when it is too steep however you also may have a problem!

The linear trend line is one usually understood well by most people, at least in my experience. It shows the gradual, overall progress being made on your metrics. Since it is a straight line, quite often questions about what happened in a “dip” period can be prevented.

Since there is an abundance in data, if you have setup your automation properly, there is also the possibility to combine data. Such as setting off the trend of passed/failed to the trend of new tests added, or even more interestingly, to new functionality added to the system under test.

Be aware!

One big warning though, when playing around with the numbers you may be tempted to make them look nicer than they are or focus on the good things. However tempting this may be, don't prettify your numbers or graphs, make sure they always paint a true story. If you manipulate the graphs, you are not only trying to fool your manager, but also yourself. Metrics should be useful for you as well as for the managers.

Source : <http://martijndevrieze.net/2012/05/30/test-automation-metrics-what-do-you-report-on/>