

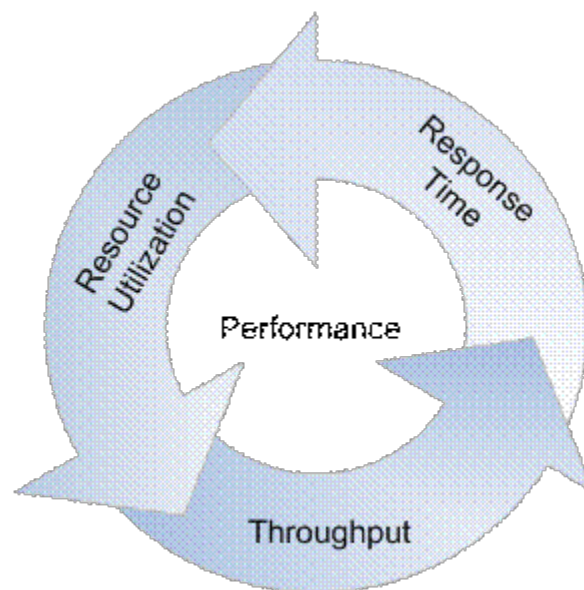
SELECTING PERFORMANCE TEST TOOLING – PART 1

How do you come to a logical and effective performance test tool set? Yes, I say tool set since quite often just one tool will not suffice.

As could be read in my previous post I am currently heavily involved in performance testing for a large Dutch retail organization which is merging two large organisations into one. For the ERP system they both use I have written a concise Performance testplan. Now the time to execute has come.

In order to execute the performance tests on the ERP system, which is both desktop application (WPF) and web-based I will need a solid tool to

1. generate load on the servers by emulating functional behaviours
2. functionally walk through the actual desktop and web-applications to measure the true application performance



Performance testing

Generating load is not the biggest challenge, quite a few tools are capable of doing that, especially since there is a website we can push the load through. The functional walk-

through is going to be a bit less simple to create, especially since I have to recreate the same scenarios twice; once for the web application and once for the desktop application.

To complicate things even more, I want the functional walk-through and thus the real measurements, to be done by the same tool. This way I can make sure the measurements are reporting on the same thing in the same way and thus can be compared to one-another to some extent.

Long list

First off in looking for the right tools I started setting up a long list, or actually, I Googled for a longlist...

This search resulted in several useful pages I checked out:

- [10 best tools for test automation](#)
- [Open source testing](#)

Just based on previous experience, tool knowledge etc. I dismissed a set of the tools and ended up with a nice set of possible tools.

Next step, which of these tools actually fit with the organisation and application landscape? I have specific needs at the moment for this tool, however I also need to keep the future needs of my customer in mind. In other words, I need a tool that is a) good for me now and b) good enough for the foreseeable future to be worth the investment (in license and time or just time, either way quite an investment).

What will the tool need to support?

- WPF/Win32/WinForms
- Java (Swing) UI's
- Oracle
- SAP GUI
- Web applications of all sizes and shapes
- RFC communications

- SOAP
- Tibco

The search has begun.

In the next installment of this series I will tell more about what the shortlist has become, what I tested on the applications and how I have come to a decision which application seems the best fit for this environment.

Source : <http://martijndevrieze.net/2013/04/11/selecting-performance-test-tooling-part-1/>