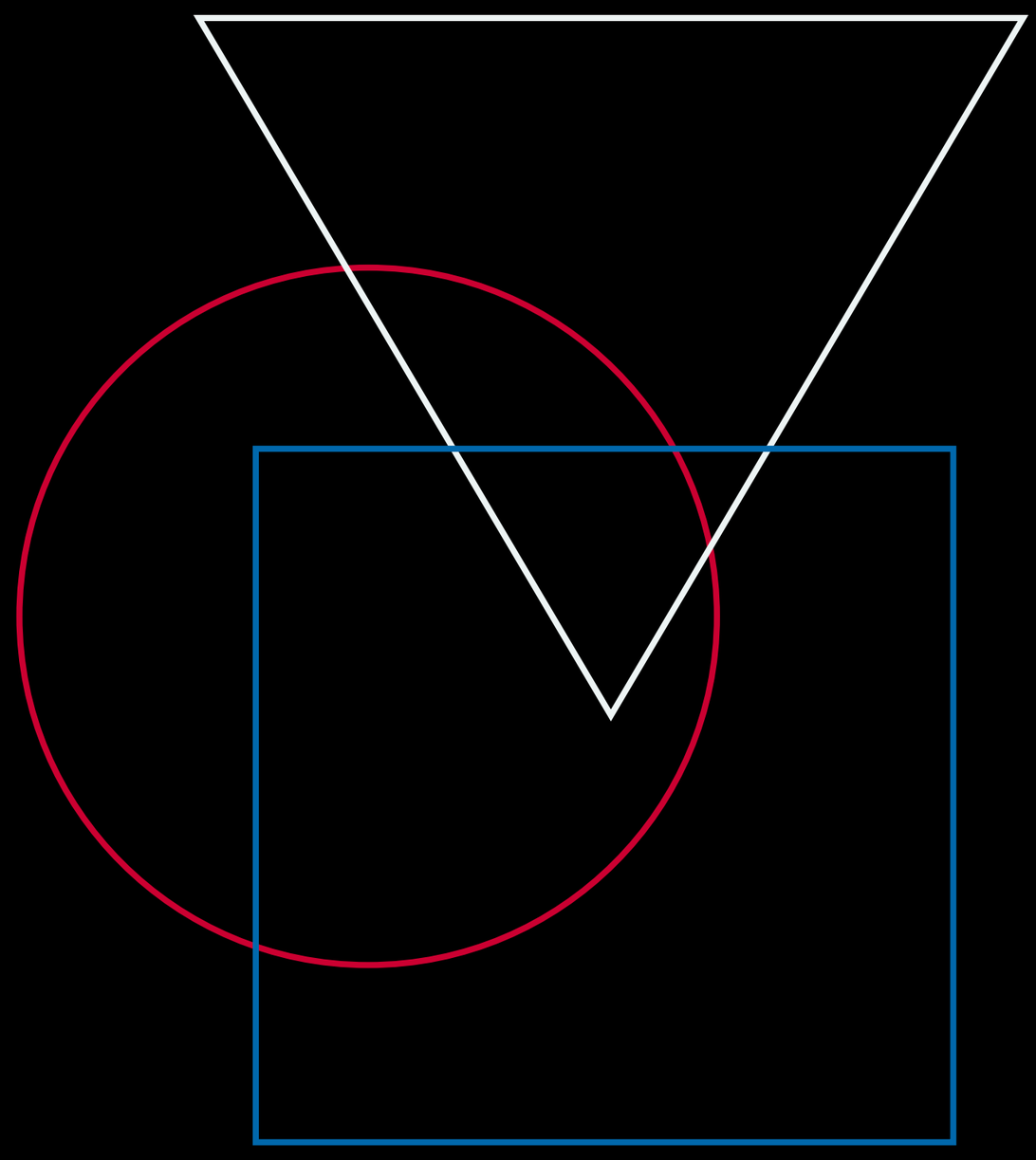
Performance Testing Whitepaper



By Optimus QA

Index

Introduction

This document is meant as a whitepaper in preparation of the creation of an issue for your project. The issue to be created will ask for a change in direction with regards to the creation of a performance testing (PT) capability within your organization. In this document, challenges with the creation of the PT capability are highlighted and an alternative is offered.

The current status of the work on the PT capability: The team has documented performance test tool requirements and have vetted IBM Rational Performance Tester to determine if this tool would support the PT work for your organization. Discussions on how such a tool should be deployed and what knowledge levels need to be present to create an effective PT capability,
brought up significant concerns with the change of success at your organization.

The reasons are multiple and are addressed in this document. Recognizing the issue is the first step, providing guidance that will help to chart an alternative is the second step and this document intends to do this.

What is Performance Testing?

Performance testing is a type of testing intended to determine the responsiveness, throughput, reliability, and/or scalability of a system under a given workload.

Performance testing is commonly conducted to accomplish the following:

- Assess production readiness
- Evaluate against performance criteria
- Compare performance characteristics of multiple systems or system configurations
- Find the source of performance problems
- Support system tuning
- Find throughput levels

At the highest level, performance testing is almost always conducted to address one or more risks related to expense, opportunity costs, continuity, and/or corporate reputation. Some more specific reasons for conducting performance testing include:

Assessing release readiness by:

- Enabling you to predict or estimate the performance characteristics of an application in production and evaluate whether or not to address performance concerns based on those predictions. These predictions are also valuable to the stakeholders who make decisions about whether an application is ready for release or capable of handling future growth, or whether it requires a performance improvement/hardware upgrade prior to release.
- Providing data indicating the likelihood of user dissatisfaction with the performance characteristics of the system.
- Providing data to aid in the prediction of revenue losses or damaged brand credibility due to scalability or stability issues, or due to users being dissatisfied with application response time.

Assessing infrastructure adequacy by:

- Evaluating the adequacy of current capacity.
- Determining the acceptability of stability.
- Determining the capacity of the application's infrastructure, as well as determining the future resources required to deliver acceptable application performance.
- Comparing different system configurations to determine which works best for both the application and the business.
- Verifying that the application exhibits the desired performance characteristics, within budgeted resource utilization constraints.

Assessing adequacy of developed software performance by:

- Determining the application's desired performance characteristics before and after changes to the software.
- Providing comparisons between the application's current and desired performance characteristics.

Improving the efficiency of performance tuning by:

- Analyzing the behavior of the application at various load levels.
- Identifying bottlenecks in the application.

- Providing information related to the speed, scalability, and stability of a product prior to production release, thus enabling you to make informed decisions about whether and when to tune the system.

Challenges with Building a Performance Testing Capability In-house

Building a PT capability in-house has its challenges, maintaining this capability proves to be a challenge nobody really counted on.

The key challenges are:

- Economic
- Resource
- Knowledge/Experience
- Education/Training
- Continuity
- Effectiveness
- Planning PT

Economic Challenge

Let's not mince words about the economics of PT: It is very expensive!
The following categories of expenses have been identified:

Performance Testing Tools

Commercial PT tools are very expensive because most often
the tool vendors will charge for:

- the actual performance testing tool,
- the individual network protocols that need to be supported and
- the amount of users (virtual users or VUs) that need to be supported.

Other Tools

To do proper PT, you have to monitor your system usage. Monitoring tools that will allow you to do that often come at an extra expense. An assumption caveat here: many people assume that because their environment contains production monitoring they can reuse this for PT. This is a wrong assumption as the production monitoring tools are often not installed on test systems and typically do not yield the depth of information that a performance tester would need.

- Network simulation tools:
  It is common to simulate network conditions in initial PT. These tools often save the organization a lot of money and headaches. Companies like Shunra offer these devices that can be configured to simulate WANs and other network configurations.

- Analysis Tools:
  When collecting a large amount of test and monitoring results, it is often

needed to deploy other analysis tools to be able to mine the data. A very common tool to use is Excel, but more advanced statistical tools and related storage products might be needed.

- Miscellaneous Tools:
  Performance testers can require tools like protocol analyzers, log file parsers, databases, a variety of smaller performance tools for specialized load generation on for instance web services, SMTP servers, authentication mechanisms etc. Many of these tools are either low or no cost.

Hardware

- Load Generating hardware:
  System load is created by simulating large amount of users (VUs) from load generators. These generators are servers class machines that typically would simulate up to 500,750 users. At your organization, we could easily count on needing 46 of those server class machines. It is advised to not run these generators as virtual machines as timing problems and unpredictability in load generation timing causes unpredictable and unrepeatable results.

- Master Workstation:
  This machine is the conductor of the performance test execution; it typically consists of a powerful workstation or a medium server class machine. Often such a machine is colocated with the load generating hardware and remotely accessed by the performance testers.

- Performance Tester workstations:
  The testers will develop their scripts and test their scripts from their own workstations. These workstations need to be top of the line as the system demand on them is high. Also many performance testers have multiple machines on their desks as some of their machines might be tied up during test runs. It is not uncommon for performance testers to have 3 or more machines on their desks and several more they can remotely access.

- Repository/Database Machine:
  Most performance testing tools need a centralized repository for scripts, test results and archive. Since usage of this repository is particularly heavy during test runs (all the result data streaming in) and the volume of monitoring and test results is very high (easily many Gigabytes at the end of a small PT exercise), it is advised to target a physical server for this work.

Human Resources

- Performance Testing Architect/Lead:
  Very experienced, resourced with a varied technical and testing background who sets the direction for PT, manages the work, communicates with the stakeholders and is responsible for reporting.

- Performance Testing Analyst:
  Analyses the user work load in order to define the work load that will be simulated. This resource has a strong Business Analysis focus and is versed in creating user load models.

- Performance Testing Scripter:
  The scripter is a master of the performance testing tool and is able to translate the work from the Analyst into working performance testing simulations.

- Technical support personnel:
  Support Personnel are typically needed at system setup and to start and stop monitoring sessions on the different platforms.

- Technical Platform specialist:
  In case of found issues on a specific platforms or whee reviewing the monitor logs, the specialist plays a very important consulting role.

- Network Specialist:
  This specialist is typically engaged to perform monitoring on the network while a performance test is running or to help troubleshoot network issues.

- Security Specialist:
  This specialist typically has a consulting and reviewing role and will be consulted on PT triggered security issues.

- Application Specialist:
  The application specialist is consulted throughout the performance testing cycle, from analysis to result evaluation.

- Development Resources:
  Most often the lead developer of the application under test will be involved in the result evaluation sessions.

Test Environments

- Copy of Production:
  Most performance testers will demand an exact copy of the production environment to run their tests in. This is obviously a costly proposition. The reasoning is that without it no accurate predictions can be made about the viability of the application in production.

- Almost copy of production (less power but with all the components):
  An often viable alternative is to scale down the capacity of the system but still maintain all the moving parts so that potential negative interaction can be evaluated. Such an environment would not allow you to make accurate predictions.

- Dedicated PT environments:
  These would be environments isolated from other testing but not necessarily be “complete” environments. The isolation provides independence and will

allow the PT team to work in parallel with other testing work. This is important as PT can be very disruptive and even destructive.

- Shared Environments:
  The least preferred option and would typically only serve during PT script development. Test runs in such an environment need to be scheduled during off hours and will, therefore, incur overtime costs and scheduling issues.

Past experiences have shown that a minimum cost for a PT exercise is about $15,000 for an Architect/Analyst and a scripter, all other resources are deemed free, medium costs are around $40,000 to $60,000 and large projects start with $100,000 and go up from there.

Resource Challenge

PT is a highly specialized discipline that requires a wide range of skills. One of my fellow performance testing friends calls it: "I need to be medium level on anything."

Areas of expertise that are required for PT:

- Tool Knowledge
- Scripting/Coding Knowledge
- Development experience
- Network Protocol/Technology Knowledge
- Failure type awareness (knowing where stuff fails)
- Performance testing methodology
- Deep database knowledge
- OS Knowledge (Windows, Linux, Unix, Solaris, Mainframe etc.)
- Authentication Mechanisms
- User load analysis
- Load modeling
- Statistics
- Reporting
- Scope negotiation
- Testing methodology
- Test Data usage/generation

The problem in the market is that people who have developed scripting skills on a particular tool are considered "Performance Testers". These people perform well below expectation but with the same high cost. Companies who have experienced this type of PT are typically very disappointed about the lack of value obtained.

The roles were described above, but I'll add some more detail here:

Performance Testing Architect/Lead :

Very experienced, resourced with a varied technical and testing background who sets the direction for PT, manages the work, mentors team members, interfaces with other technical resources in the company, facilitates review sessions, facilitates root cause analysis.

Performance Testing Architect/Lead :

Very experienced, resourced with a varied technical and testing background who sets the direction for PT, manages the work, mentors team members, interfaces with other technical resources in the company, facilitates review sessions, facilitates root cause analysis, communicates with the stakeholders and is responsible for reporting. This resource is typically not found in abundance and many of those are either very gainfully employed or operate as highly paid consultants. A $100K per annum price ticket is a starting point. It is possible to homegrow such a resource, but your raw material already needs to be a person possessing many of the above mentioned skills. Even with those skills, count on at least 6 months ramp up, provided the resource is mentored. It could easily take up to a year if this resource is not mentored. Another key requirement for such a resource that this person needs to be rabidly interested in the field and commit her/himself to continuous study and education. It is challenging but for the right person, a very fulfilling job.

Performance Testing Analyst:

Analyses the user workload in order to define the workload that will be simulated. This resource has a strong Business Analysis focus and is versed in creating user load models. Out of all the roles this one is almost the easiest to fill. Most BAs, with some guidance, will be able to pick this up fairly rapidly. I have trained BAs and System Analysts to do this work and within a couple of weeks they can be productive. A trained BA would be in the $60K to $85K range.

Performance Testing Scripter:

The scripter is a master of the performance testing tool and is able to translate the work from the analyst into working performance testing simulations. Scripters are a category apart. It is very easy to learn the basics of a performance testing tool, but very quickly the learning curve hits a very steep incline and demands attention, additional learning, experimentation and persistence. The core concepts of performance testing scripting are not terribly difficult to understand but they are a bit different from functional test scripting. Many new scripters operate without any goals or objectives, just the mere fact that they can make something work and ramp up the user load often seems to be enough. This results in nonsense tests like logging in 2000 users at exactly the same time where there is no perceivable reason for doing that. A trained scripter would be in the $50K to $75K range.

Availability in the Market

Performance Testing Architect/Lead

Very Low and do come at a high price

PT Analyst

Not abundantly available, but relatively easily trained in-house

PT Scripter

Some availability, but often bill themselves as the full package deal.

Inhouse Development Potential

Performance Testing Architect/Lead

Hard. Trainable resources will already be key resources in the organization.

PT Analyst

Relatively easily trained in-house, provided you can provide mentoring.

PT Scripter

Scripters are fairly easily trained in the tool but need strong mentoring to develop.

Skill Maintenance Issues

PT is very cyclical, it is, therefore, very hard to keep dedicated resources busy with PT. With big breaks between PT exercises, it has been observed by me that PT quality and intensity tends to slip to a lower common denominator.

Given the long and steep learning curve for PT, PT suffers more significantly from frequent resources swaps than other testing disciplines.

PT has significant similarities to security testing in that it requires a tremendous amount of study and experimentation to become very good at it. However, both disciplines also have in common that they are cyclical in nature.

Knowledge/Experience

The heavy knowledge and experience requirements, as highlighted above in the previous chapter, can usually be mitigated by assembling larger teams in which the combined knowledge covers all areas needed.

But, in general, you can say that effective performance testing groups are populated with experienced people, to the point that the effective performance testers I am familiar with are all over 40 with a minimum of 15 years in diverse technical and testing roles.

Education/Training

Ongoing education and training is part and parcel of a performance testing engineer's life. The performance testing community in North America is small but intensely committed to learning and sharing of information. Small, but popular, workshops like the Workshop on Performance and Reliability (WOPR) are already in its 9th year with no sign of flagging interest.

Interestingly enough, there are very few toolagnostic performance testing courses available. I personally know of only 2 courses (one of which I teach). Product related performance test tool courses are available but would typically require people to travel to big US cities. (New York tends to be one of the course hubs for performance testing.)

Continuity

As mentioned before, it is very easy to get rusty in performance testing. Even more so than in other testing disciplines, this is due to the fact that it is complex and success is typically dependent on many variables and variations. With interruptions of months in between PT engagements it is very easy to lose the edge. Another aspect is that you will see some “drift” happening in organizations that have deployed PT for a while. Particularly if this PT has been quite effective in finding issues before they became problems. A drifting away from the need to do performance testing is typically very gradual and innocuous. A good and nasty production problem is all that is needed to regain focus.

Effectiveness

In performance testing, it is very possible to be very busy and burn through a lot of budget without adding any significant value to the process. This is a problem that can have multiple reasons:

1. Red herring testing: Pursuing goals that do not make sense and add no value.

2. Too much time and budget: Not focused enough, time gets filled anyway.

3. Too little budget: Too focused on just a couple of scenarios leaving very little time for exploration and experimentation.

Planning PT

Planning for PT is challenging for multiple reasons:

1. The need for PT is not readily understood.

2. Because of the lack of understanding, PT is often (over) simplified.

3. PT is highly cyclical, that means one month you are very busy and the next 6 months, there is no demand for performance testing.

4. PT can be extremely expensive, particularly, if planned as a one off big bang performance test, adjusting the planning to spread out the activities runs into incomprehension and lack of commitment from the resources that should be part of thtis.

Typical estimates for PT (assuming all tools and hardware is present and your organization's technical support is not billable) for different size projects are:

- Small Project (34 Usage Scenarios): $15,000 to $25,000 – 2 weeks
- Medium Sized Project (up to 10 Scenarios): $25,000 $90,000 – 46 weeks
- Large Projects: $90,000 to $500,000 26 Months

Specific Challenges for Your Organization

Your organization has several unique and not so unique challenges when contemplating building a performance testing capability.

1. Limit on the amount of resources that can be hired.

2. Timeliness of the capability: Building a capability, training people and building up the experience required, could easily take 6 - 9 months with further improvements to be pursued for another year.

3. Experienced performance test resources are scarce and demand a premium. Current salary bandwidths at your organization fall short of what these resources would expect. It is not uncommon for experienced performance test leads/testers to make well over $100,000 per annum.

4. Performance testing was never seen as a necessity.

5. Performance testing would require significant funding on top of the funding that projects would require. To build a successful practice, your organization should count on having to fund 2 resources for 1.5 to 2 years.

Alternative Approaches

There are, of course several alternatives or combinations of alternatives:

1. Do nothing, ignore performance testing.

2. Build an awareness of performance issue into the development process and evaluate performance on an ongoing basis with simple (free) tools and checks..

3. Outsource the complete PT process.

4. Build the capability to do simple performance testing and outsource the big efforts.

5. Depend on the COTS vendor to tell you that the system performs.

Risks Related to the Alternatives

1. With the new complex systems that are being built, you will run into performance problems that will cause [CONSULTING CLIENT] grief.

2. Most development resources are not up to date with current technology and would need to be first. It is unlikely that this approach would mitigate all the performance risks.

3. Planning is more complex and startup is typically lengthy (and therefore expensive). It is likely that such a vendor would be limited in their scope (just do this) and therefore will be limited in the value they can add.

4. The IS QA group members could be trained in basic PT techniques with low cost or free tools. This would allow for early flagging of performance issues and they could be instrumental in the preparation of the work for the external PT vendor and with doing that, limiting the startup time such a vendor would need.

Market Offerings

I have made some phone calls to several friends and peers who run PT services companies and the offering is actually quite broad.

Here's a quick listing of some of the key characteristics of the offerings out there:

- Tool Focused:
  There are several vendors out there that operate tool suites from the cloud and offer the capability to test your website from many locations in the world. Interesting offerings, but they tend to focus on external facing websites with either fairly static pages or with very simple functionality.

- Preferred or previous contractor model:
  This is a very wellknown model for me and I know quite a few people who operate this way. This is a model where an individual has had a previous engagement at a client as a Performance Tester and this person is repeatedly asked back to do the next iteration. This is a model based solely on individual merit and that individual's experience with the organization. This model does not allow for a rampup for larger engagements.

- Big 6 Performance Testing:
  Large consulting firms who have all the capability and knowledge available. Problems with this approach is scheduling, negotiation of the work and (in my experience) subpar resources that are parachuted into the organization.

- Overseas outsourcing:
  In this model, the PT resources are not working onsite but they connect into your network. Communications issues, subpar resources are risks with this approach.

- Boutique Shops:
  Small companies (up to 10 people) that specialize in PT and typically have one or more very well known resources. These are the cream of the crop and they typically know it too. They are not as expensive as the Big 6 type companies and they are significantly more effective. They could be spread very thinly and they are keen on keeping the sales process going and being able to support current and past clients. You will find the deepest expertise with this type of company.

- Testing Companies:
  These are companies that are offering all testing activities and have included PT in their offering. The trick is to understand if they really have the capability or if this advertised capability is merely to get all the boxes checked. Testing companies will try to be more to your organization than just PT.

Variables

The following variables seem to be in play:

- Location: In-house, off-site or a combination
- Tools: Vendor brings tools, vendor rents out tools, your organization buys tools, your organization rents Tools
- Hardware: Client organization provides hardware
- Knowledge Level: Junior resources (scripters) to full bore PT engineers
- Type of relationship: Ad Hoc, Individual, Project, Subscription
- Price: From individual contracting rate to full bore projects costing several millions
- Results: From extremely disappointing, Ho Hum to valuable
- Timeliness: Long lead times to immediate availability
- Duration of engagement: 2 weeks to years

Recommendation

I would recommend that your organization develops a two pronged approach where there will be an emphasis on performance (response time checking during functional testing possibly with the help of some simple and free tools and where an external specialist is engaged to assist with the more complex back end performance evaluation and tuning. There is potential for having the internal resources be able to plan for and design the performance tests that the external specialist will need to script and execute.

I also recommend that an EOI for finding such an external specialist would contain a questionnaire addressing all the variables mentioned in the previous chapter. Following my recommendation would position your organization nicely to be able to mitigate performance risks and potentially avert usability disaster scenarios that complex systems are sadly prone to.

Thank You

By Optimus QA