



PERFORMANCE TESTING

Agile Performance Testing

Abstract

When viewed from a linear perspective, the approach starts by examining the software development project as a whole. This includes finding the reasons, why stakeholders have chosen to include performance testing in the project, and the value that performance testing is expected to add to the project. The results of this examination include the team's view of the success criteria for the performance testing effort.

Once the success criteria are understood at a high level, an overall strategy is envisioned to guide the general approach to achieving those criteria by summarizing what performance testing activities are anticipated to add the most value at various points during the development life cycle. Those points may include key project deliveries, checkpoints, sprints, iterations, or weekly builds. Frequently, while the strategy is evolving, the performance specialist and/or the team will begin setting up a performance-test environment and a load-generation environment.

With a strategy in mind and the necessary environments in place, the test team draws up plans for major tests or tasks identified for imminent performance builds. When a performance build is delivered, the tasks

should be executed in priority sequence (based on all currently available information), appropriately reporting, recording, revising, reprioritizing, adding, and removing tasks and improving the application and the overall plan as the work progresses.

The process can compare two or more devices or programs in terms of parameters such as speed, data transfer rate, bandwidth, throughput, efficiency or reliability.

Performance testing can also be used as a diagnostic aid in locating communication bottlenecks. Often a system will work much better if a problem is resolved at a single point or in a single component.

Effective performance testing can quickly identify the nature or location of a software-related performance problem.

The Problem

With the ever growing need for quick access to data, and with the large volumes of data at our disposal, the need for high performance systems and applications has become increasingly vital.

The goal of performance testing is not to find bugs, but to eliminate bottlenecks and establish a baseline for future regression testing. Ideally, the software



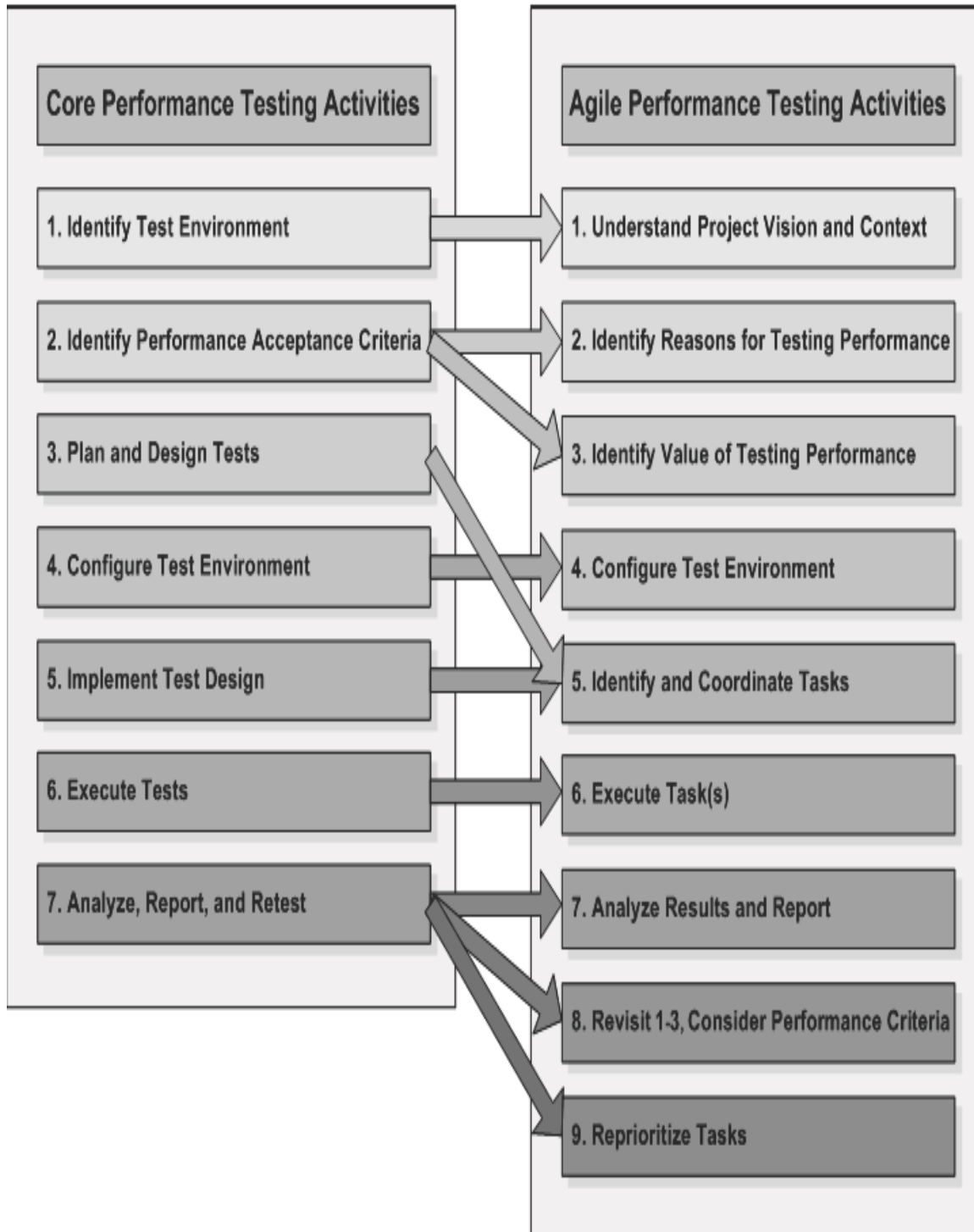
under test is already stable enough so that this process can proceed smoothly.

Valuable time can be wasted due to poorly designed hardware layout or badly developed application. Slow data transfer rate may be inherent in hardware but can also result from software-related problems, such as:

- Too many applications running at the same time
- A corrupted file in a Web browser
- A security exploit
- Heavy-handed antivirus software

Performance testing of an application under development aims to thoroughly address a multitude of performance affecting factors to result in an enhanced user experience when using an application.

Various techniques, depending on the application type, project duration, available performance testing tools and resources i.e., budget and time constraints are taken into consideration when devising the most suitable performance testing strategy.





Understanding 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

Agile Performance Testing Activities

Agile testing does not emphasize rigidly defined testing procedures, but rather focuses on testing iteratively against newly developed code until quality is achieved from an end customer's perspective. In other words, the emphasis is shifted from "testers as quality police" to something more like "entire project team working toward demonstrable quality."

Understand Project Vision and Content

This is essential to determine what performance related activities are necessary and valuable.

Identify Reasons for Testing Performance

Explicitly identifying the reasons for performance testing is critical to being able to determine what performance

related activities will add most value to the project.

Identify Value of Testing Performance

From the information obtained from the first two steps, clarify the value added through performance testing and convert that value into a conceptual performance-testing strategy.

Configure Test Environment

With a clear strategy, preparation of necessary tools and resources is the next step.

Identify and Coordinate Tasks

Performance testing professionals need to work within teams which require coordination and availability of other teams responsible and useful for input as well as ensuring smooth overlap and availability during schedules.

Execute Task(s)

It is essential to conduct tasks in 1 to 2 days segments, and seeing them through completion, with a flexible approach to including important tests which add value to performance testing.

Analyze and Report

To keep up with the iterative process, reports need to be generated and shared for valuable feedback. In case of inconclusive results, re-tests should be performed at the earliest.

Revisit 1-3, Consider Performance Criteria

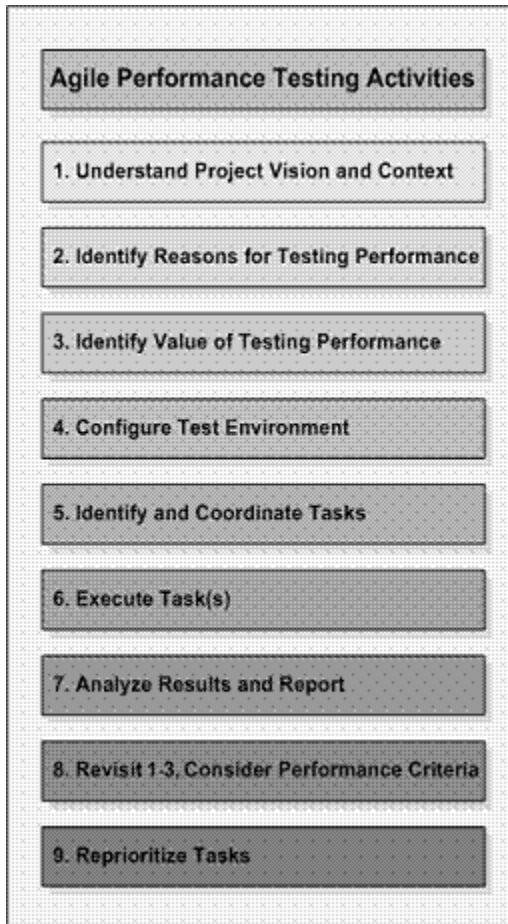
Between re-tests, it is imperative to ensure the original information (at the start of the project) has not changed. It is equally important to integrate customer



feedback and update the performance strategy accordingly.

Reprioritize Tasks

Depending on test results, new information and availability of resources etc, it is vital to reprioritize, add or delete tasks (as appropriate) and return to activity 5 (Identify and Coordinate Tasks).



Solution

Depending on the nature of the project, and the given times frames along with the availability of resources and test tools available, it is advisable to determine the most suitable performance testing strategy to adopt to successfully complete performance testing within time and budget constraints. The underlying philosophy in agile performance testing is to remain efficient by assigning different things to different teams whilst maintaining flexibility throughout the project.

Conclusion

The nature of performance testing makes it difficult to predict what type of test will add value, or even be possible. Yet, it holds an equally important place in the successful acceptance of an application which is robust, and able to deliver results. Thorough performance testing ensures that the application will deliver, without compromising quality and it is essential for every software to undergo thorough performance testing.