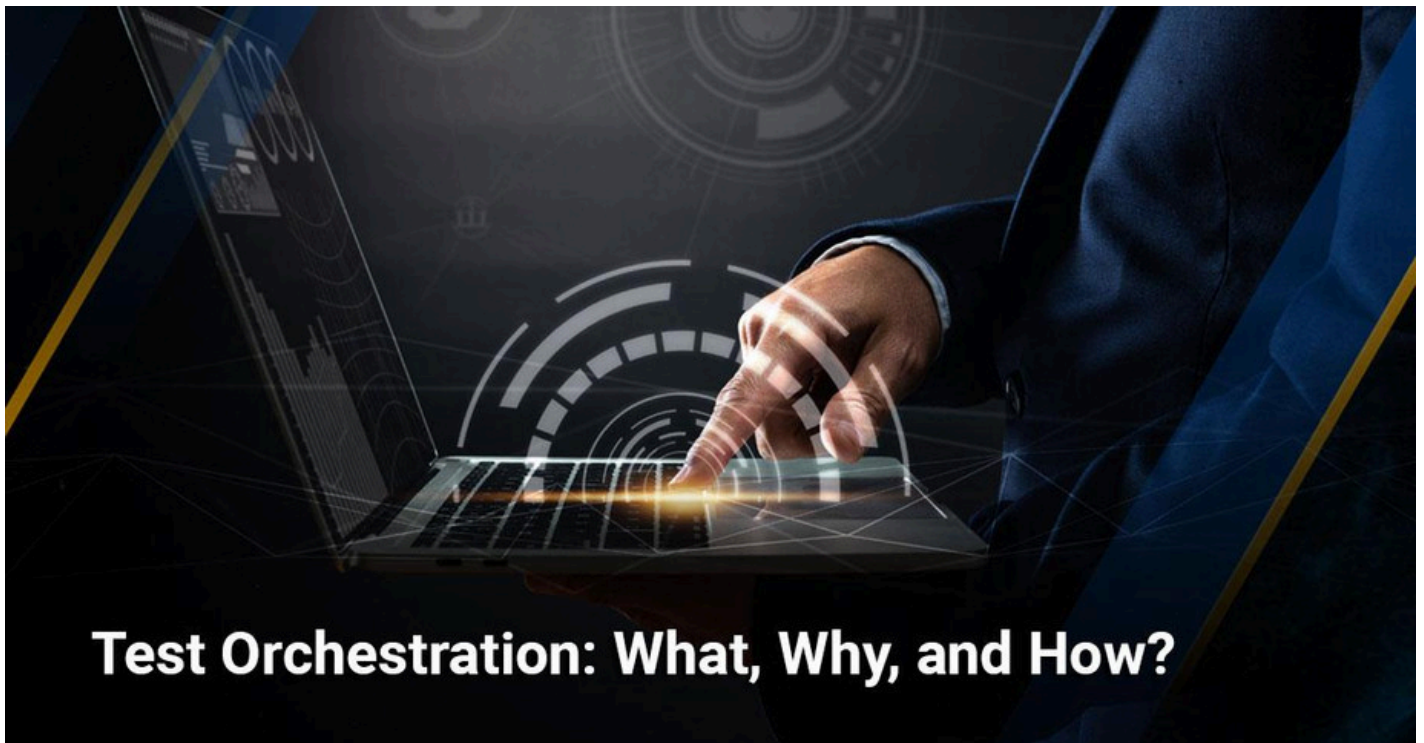




# Discuss Test Orchestration and its Role in Achieving Optimal Quality of Software



## Test Orchestration: What, Why, and How?

In the DevOps approach of developing, testing, and delivering software applications, there is a need to accelerate testing within the constraints of budget and time. This entails automating the testing process as part of the testing strategy. In many cases, continuous test automation is often considered as a discrete step in the build pipeline rather than a sequence of steps. For any continuous testing strategy, all software applications need to pass through a series of tests before they pass muster in the crucible of quality. These may include unit testing, integration testing, functional testing, smoke testing, performance testing, security testing, and others. For DevOps QA testing, each of these tests should be subjected to automation to enhance quality, reduce test time, and improve the certainty of software behavior.

However, merely automating the tests left, right, and center does not augur well when it comes to achieving an optimal quality level as well as minimizing the time and costs. In the ultimate analysis, what is needed is test orchestration. In fact, implementing test orchestration can be quite rewarding given that it accelerates [DevOps continuous testing](#) and offers a holistic view of the entire testing mechanism. Interestingly, when we talk about DevOps and the ways to enhance software quality, terms such as test orchestration and automation are thrown around randomly. This can be confusing for many stakeholders not in the know.

So, is there a difference between test automation and test orchestration? The answer is yes, as test orchestration allows DevOps teams to optimize the whole test process by scheduling the execution of automated tests in a particular sequence. It is a tool to bring about efficiency and integrates both manual and automated testing. Let us discuss test orchestration and its overarching benefits in the segments below.

## What is test orchestration?

Test orchestration is the scheduling of a set or sequence of automated test activities in a well-defined order to be executed one after the other. Here, the tests are executed in a linear order, and the decision to control the order of testing vests with the DevOps team. It aims at optimizing the test strategy at a holistic level and grabbing insights into the broader picture of software testing. In contrast, test automation is about individual activities or precise tasks to be executed using test scripts and tools. Test orchestration is a broader approach to testing with the application of greater thought. It fosters transparency and offers flexibility in testing software applications or certain features of them during the development phases. In test orchestration, multiple automated test activities can be executed as part of the test pipeline.

## Benefits of test orchestration

Test automation seeks to streamline automated testing activities and achieve optimal quality of testing. It offers a host of benefits to the DevOps teams, as discussed below:

**Increased productivity:** Test orchestration helps the testing of a software application across channels (devices, browsers, operating systems, and networks) and concerns (functionality, performance, and security, among others) through automation. This removes any arbitrariness in testing based on extraneous considerations and improves the productivity of the delivery team.

**Better coverage:** As test orchestration focuses on automation as an objective, the coverage of automation improves. With more tests being automated, the cost and timeline of delivery can be controlled or minimized.

**Easy error remediation:** Test orchestration leverages a continuous testing framework to automate testing arranged in a sequence of processes. This enables the identification and

fixing of errors early when the complexity of the software application is low. In the absence of test orchestration, the possibility of identifying errors later in the SDLC would have increased, making their remediation a difficult exercise.

**Optimized test process:** Test orchestration allows the DevOps testing team to gain control over setting up the continuous test automation strategy and test schedules. And to achieve overall efficiency, test orchestration can leverage various tools such as project management tools, automated testing tools, and DevOps tools, among others. These tools, along with a continuous testing strategy, can transform the SDLC to achieve a flexible and optimized test process.

**Faster builds:** With a range of automated tests set up to be executed in a sequence, it becomes easier to create a test pipeline. This hastened the execution of tests, leading to faster builds.

**Shortened feedback loops:** In test orchestration, whenever a certain error or concern is flagged, say security, it is only the security experts in the testing team who would be involved in verifying the results. Thus, the direct remediation of errors by the concerned people due to a shortened feedback loop reduces the test timelines significantly.

**Minimize the risk of failure:** Test orchestration is put into motion based on the risk assessment of the software application. It allows the creation of dedicated pipelines to address risks related to scalability, performance, security, availability, and robustness, among others.

## Conclusion

Test orchestration is undoubtedly the future of QA as it allows QA experts to achieve speed and quality of testing in a seamless manner. However, any successful implementation of test orchestration would need the QA team to possess the all-encompassing tools to deliver [continuous testing in DevOps](#).

## Resource

James Daniel is a software Tech enthusiastic & works at Cigniti Technologies. I'm having a great understanding of today's software testing quality that yields strong results and always happy to create valuable content & share thoughts.

**Article Source:** [medium.com](https://medium.com)