

Foreword by Armin Metzger

The second wave is here! I believe we are in the middle of the second wave of test automation. The first big wave clearly took place in the early 2000s, and the projects involved were initially very successful in terms of improving the effectiveness and the efficiency of test processes in some specific areas. However, in line with the Gartner cycle, the “trough of disillusionment” was quickly reached and, in my view, most projects didn’t actually reach the “plateau of productivity”.

What I observed at the time were projects that expended enormous effort over several years to work their way to a high degree of test automation. Then came technology changes such as the switch to .NET platforms, or process changes such as the switch to agile development methodology. A lot of the test automation frameworks didn’t survive those transitions. Back then I liked to give talks with provocative titles such as *Test Automation Always Fails*.

We saw two core problems: firstly, companies failed to scale isolated successes to the entire project or organization, and secondly, test automation platforms were not sufficiently flexible to absorb disruptive changes in the technology base.

It is therefore no surprise that, over time, test automation began to lose acceptance. Management aspects also play a supporting role here. In the long run, the great economic expectations of a one-time investment intended to significantly reduce regression efforts were often simply not met.

Since the middle of the second decade of the 21st Century, we see a trending new wave of test automation in large projects. Will test automation once again fall short of its expectations? I don’t think so. Both the overall test automation environment and the expectations test automation raises have changed. Test automation has now re-established itself as an indispensable factor for the success of projects in current technological scenarios. What changed?

With the introduction of agile processes, highly automated, tool-supported development has evolved significantly and has now become standard practice. Continuous integration concepts are constantly being refined into DevOps processes to create a seamless platform for the integration of automated project steps—all the way from the initial idea to final production and operation. The end-to-end automation of processes naturally forms an excellent basis for integrating test automation into the overall development process. Additionally, agile processes have helped process scaling to reach a new, higher level of importance. This development is an essential factor for the successful introduction and long-term establishment of test automation solutions.

However, a key factor in the importance (and necessity) of test automation is the current technological platform on which we operate. Disruptive technologies such as IoT (Internet of Things) and AI (artificial intelligence) are rapidly pushing their way out of their decades-old niche existence and into our products. With this comes a significant shift of priorities for the quality attributes we have to test. While 20 years ago, ninety per cent of all tests were functional tests, the importance of non-functional tests for usability, performance, IT security, and so on is slowly but surely gaining ground. The number of test cases required to assess product quality is therefore increasing rapidly, and only automated tests can effectively safeguard quality characteristics such as performance.

The development and maintenance of products takes place in increasingly short cycles. Due to the increasing variance in hardware and software configurations, entire and partial systems need to be tested in an increasing number of variants. Non-automated regression testing thus becomes an increasing burden, and it becomes more and more difficult to achieve the required test coverage while retaining an adequate level of effort.

And—fortunately—we have also learned a lot about methodology: test architectures are one of the most important factors (if not *the* most important factor) influencing quality in the maintainability of automated tests. In fact, test architectures are now so well established that the dedicated role of *test architect* is now being introduced in many organizations. This is just one example of such changes.

But beware: using the right approach and having knowledge of the pitfalls and best practices involved in introducing and maintaining test automation are key to long-term success. Introducing appropriate expertise into projects and organizations is not always easy. This is where the *Certified Tester* certification scheme—long established as an industry standard with a common glossary—can help. The *Test Automation Engineer* training and certification covered in this book are intended for advanced testers and translate the focus and factors that influence the long-term success of test

automation into a structured canon of collected expertise—for example, on the subject of test automation architectures. This book clearly shows that these skills are constantly evolving.

We are better equipped than ever and I believe we have taken a significant step forward in the field of test automation. I wish you every success and plenty of creative fun using test automation as a key factor for your professional success!

Dr. Armin Metzger

Managing Director of the *German Testing Board*, 2022