

HIGH-PERFORMING DEVELOPMENT TEAMS USE AUTOMATED PERFORMANCE TESTING



High-performing development teams are always looking for ways to improve. So you may be asking yourself, "How can I help my teams improve application performance and reliability?" One way is with [BMC AMI DevX Performance Test](#), a tool that helps development teams find and fix defects faster and improve the performance of their applications.

Traditionally, code changes weren't tested until near the end of a project—that is too late. Testing of changes to your application code needs to be done earlier in the development cycle and more frequently. Automating test cases allows issues to be resolved closer to when they are created, when they are easier and less costly to find and fix.

How does a team "shift left" testing so that it is done earlier in the development cycle? Let's say you need to test your order processing application. To start, individual test cases are recorded for a specific function. For example, one test case tests displaying a customer's order history, another provides details of a selected order, and another tests change details of the order. These can be run individually while making changes in your code. The recorded scripts can also be used in [BMC AMI DevX Total Test](#) to provide expanded testing scenarios.

Combining individual scripts into an automated testing vehicle (ATV) provides greater capabilities. An ATV is a project environment that contains several individual test cases that collectively address a defined regression or performance testing requirement. Adding the required testing elements to an ATV playlist simplifies the management of that test project. The three individual test cases

mentioned earlier, taken together, address the wider scope requirements of testing the order process.

If [BMC AMI Strobe](#) and/or BMC AMI iStrobe are available, you can obtain application performance results. The performance vehicle will automatically attempt to execute a Strobe measurement session and profile report. If you do not have Strobe or iStrobe installed or available, no reporting will occur.

While the ATV automation on its own is valuable, even greater value comes when you combine running your ATV playlists with a continuous integration and continuous delivery (CI/CD) pipeline like Jenkins, GitHub Actions or Azure DevOps. This automation can be orchestrated to run any time code is changed or promoted.

Automated testing gives your teams significant gains in speed, quality, and consistency versus time-consuming manual testing. While it may require a higher initial cost, once in place, automated testing allows developers the time to focus on more important things, like development that supports your business requirements, with the payoff that you can complete development on time and with higher quality.

Listen to Kevin and colleagues discuss the benefits of shift left and how to get started with automated testing in the podcast, "[Ten Steps to Effective Mainframe Testing](#)."