MAXIMIZE THE BENEFITS OF YOUR MAINFRAME CI/CD PIPELINE

Including the mainframe in your CI/CD pipeline is one of the most effective ways to accelerate software delivery while maintaining the reliability mainframe is known for. By pairing modern continuous integration/continuous deployment (CI/CD) techniques with mainframe capabilities, development teams can build and deploy software faster—without sacrificing quality.

It's hard—well, impossible actually—to predict the future. Whether you define the next few years as a return to normal or a whole new normal, either way, we have been changed. The world will need vastly increased reliable computing power for everything, delivered quickly, and it has to perform well from the start.

The mainframe has been the workhorse of computing for decades and has remained so right through to today. It will continue to serve business and consumer markets well into the future. No other technology has the strength and reliability to process the enormous compute demand of this age.

Because of the ongoing increase in user demand, all software delivery lifecycle pipelines must be accelerated. To meet the faster pace, mainframe delivery has to speed up while still retaining and even improving quality, and mainframe developers must adopt newer techniques like automated testing to achieve this. By pairing more modernized CI/CD techniques with the power of mainframes, developers can feel confident that software can be built and deployed reliably.

Why is the CI/CD process important for mainframe development?

A mainframe CI/CD pipeline eliminates the manual tasks that must be performed for each deployment and uses automated unit testing to prevent bugs from reaching production. This makes the mainframe CI/CD pipeline just as critical as it has always been for distributed systems.

The build-and-deploy pipeline has always been critical for non-mainframe applications, and now it is equally important for the mainframe. The huge advantage with automating mainframe application development is that it is already familiar material. Automation allows for:

- Modern version control systems
- Build management, also called a continuous integration system or CI
- Test automation
- Automated quality assurance
- Change management
- Recovery

...along with the need for an orchestration tool to tie them all together.

How do you implement mainframe DevOps successfully?

Successful mainframe DevOps implementation requires addressing both the human and technical sides of the change. The biggest obstacle most organizations face isn't the technology—it's resistance to change. Good planning and communication make the difference.

Some strategies to help with this borrowed from a blog I wrote with Atul Bhovan include:

- Assigning strong leaders
- Kickstarting your DevOps culture
- Seeking out DevOps colleagues
- Partnering with customers
- Providing education into mainframe agile techniques
- Breaking projects into smaller pieces
- Automating the DevOps pipeline
- Choosing the right tools
- Leveraging existing CI/CD pipelines
- Identifying stakeholders
- Turning vendors into partners

Why does data management matter in mainframe CI/CD?

Data management is a foundational—and often overlooked—element of a successful mainframe CI/CD pipeline. All automated testing relies on data, and that data must be available, controllable, and secure throughout the pipeline.

Test quality, application stability, customer experience, and time to market are all critical to the business, and while we focus on the applications, it is easy to overlook the data that runs through them. It's vital to ensure that a developer is working with the right data, and that it can be set up and destroyed, and then put back for the next test. All automated testing relies on that controlled, reliable data being in place.

There is also the security dimension. People still need to secure data so that someone who's testing can't access sensitive customer information. So although it is being used for testing, it must also be guaranteed secure.

What does the future of mainframe CI/CD look like?

The future of mainframe CI/CD belongs to organizations that combine modern DevOps practices with the proven reliability of mainframe infrastructure. Businesses that serve B2B and B2C markets are reassessing their infrastructure to ensure they are best prepared to face the demands of the new normal—and all of those demands, including commerce, security, speed, communication, and reliability, rely on a well-structured, bug-free CI/CD pipeline.

There is a tendency, when looking into the future, to expect that only new and innovative tools will be able to match the demands of this new age. But as many businesses are already discovering, a great deal of the horsepower needed will come from a new generation of mainframe technology, along with the techniques and talent to drive it.

BMC AMI DevX has been a leader in this industry since 1973 and we have continued to advance and innovate to ensure that the strength and reliability of mainframe remains accessible to organizations of any size. To learn more about how to maximize the benefits of your CI/CD pipeline using mainframe, download our white paper, Maximize the Benefits of Your CI/CD Pipeline by Including the Mainframe.

Frequently asked questions

What is a mainframe CI/CD pipeline?

A mainframe CI/CD pipeline is a set of automated processes that handle continuous integration and continuous deployment for mainframe applications. It applies the same build, test, and deploy automation used in distributed systems to mainframe development—reducing manual effort, enforcing consistent quality checks, and improving delivery speed.

Why should mainframe teams adopt CI/CD practices?

Mainframe teams benefit from CI/CD because it eliminates manual deployment tasks, automates unit testing, and reduces the risk of bugs reaching production. Without CI/CD, mainframe development cycles are slower and more error-prone, making it harder to meet the pace modern business demands.

What components are needed for a mainframe CI/CD pipeline?

A mainframe CI/CD pipeline typically requires a modern version control system, a continuous integration platform, test automation tooling, automated quality assurance, change management capabilities, and an orchestration tool to coordinate all components. BMC AMI DevX provides a suite of tools purpose-built for this environment.

What is the biggest challenge in implementing mainframe DevOps?

The biggest challenge in implementing mainframe DevOps is organizational resistance to change rather than a technical barrier. Successful implementation depends on strong leadership, clear communication, and a phased approach that starts by automating existing workflows before introducing new tooling.

How does data management affect mainframe CI/CD?

Data management directly affects the reliability of a mainframe CI/CD pipeline because all automated testing depends on controlled, accurate test data. Teams must be able to provision, use, and reset test data reliably—while also ensuring that sensitive data used in testing is protected and cannot be accessed or extracted inappropriately.

The views and opinions expressed in this post are those of the author and do not necessarily reflect the official position of BMC.