6 BENEFITS OF DEPLOYMENT AUTOMATION



Automation processes have become popular among many IT organizations, but many more businesses have yet to adopt application deployment automation.

In this article, we cover the basics of understanding deployment automation and outline the top benefits of embracing deployment automation for your company. By the end, you'll see exactly why automation is essential for any dev team in any business.

WHAT IS DEPLOYMENT AUTOMATION?

Deployment automation is the use of automated processes to move your software between testing and production environments. This process is both reliable and repeatable across the software delivery lifecycle (SDLC).

The DORA State of DevOps research program, a Google-affiliated, academically rigorous investigation into the practices that drive high performance, first discovered this automation ability. Since the discovery, many deployment automation tools have been developed to drive higher software delivery and organizational performance.

Remember that deploying and releasing software aren't the same thing, but they're closely related. The biggest difference is the business rationale:

- Deploying software means you might only be pushing it to a controlled environment, for testing usually.

- Releasing software means you're making it accessible to customers.

Better deployment processes, as we'll see below, often means better releases, too. Deployment automation is particularly relevant for any organization wanting to:

- Minimize or eliminate the challenges of manual deployments
- Leverage the pace and quality of releasing software

Some of the best deployment automation tools available include:

- Jenkins
- Visual Studio
- AWS CodeDeploy

There are many deployment automation tools out there, and the benefits for an organization can vary based on your business's goals.



DEPLOYMENT AUTOMATION BENEFITS

The value of deployment automation for any business is exponential, but these are some critical benefits that deployment automation can bring to your business or organization:

- Anyone can deploy
- Faster, more efficient deployment
- Increased productivity

- Fewer errors
- More frequent releases
- Immediate feedback

Let's look at each in some detail.

Anyone can deploy

A great benefit for any business: anyone on the team can deploy the software. Understanding how to deploy or release your software doesn't take massive brainpower—but it can be a completed process in the system.

Typically, a small subset of people on a team are responsible for manual or partially automated deployments, no matter how often you need to deploy. This specific duty often falls to a single person in a given project team.

If that individual is off, ill, or in any way unavailable at short notice, the release of software can become a nightmare—you're now delayed and pushing on all sorts of other deadlines.

Deployment automation can be completed by anyone who has access and can initiate the release. This is a redundancy that any DevOps team requires.

Faster and more efficient

Deployment automation increases productivity for a business. Automated software deployments are performed in seconds, and validation can be unattended—reducing deployment from a manual half-day process to a task that you barely have to think about.

Increased productivity, focus on service offerings

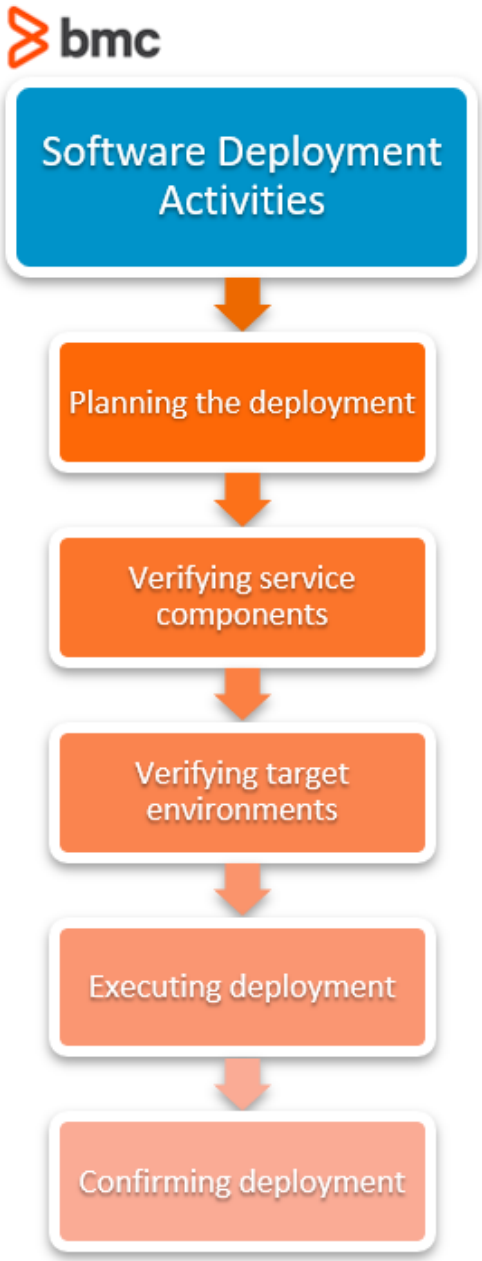
With manual processes replaced by automation, you've now freed up your dev team to focus on the business goals of improving or expanding service offerings. This freed time means your team can be more visibly productive.

Minimal errors

Deployments that are automated are much less error-prone compared to manual deployments.

Unfortunately, manual deployments involve multiple steps and can lead to human error. Essential steps in a software release can be accidentally missed, issued during a release may not be spotted, the incorrect versions of software can be shipped, or broken software ends accidentally go live.

Consider the errors that could happen at each phase of deployment:



Quick thinking and luck can minimize mistakes sometimes, but any deployment error could quickly escalate into a business issue. That's a risk most IT orgs can't take.

Once you've correctly configured your deployment automation, the process is set and will work the same way you deploy your software from the first to the last.

Release at high frequencies

Deployment automation performing a single software deployment has low overhead and can be repeated frequently—as frequently as you need. Weekly, daily, even hourly!

This frequency level is desirable for many reasons, but overall a higher frequency promotes genuinely agile software development. When your team can release a higher frequency, this brings value to the users more often and in incremental steps.

Companies that excel at software development, like Netflix and Adobe, tend to release very often. This means they can both:

- Reduce time that deployed errors are bugs are live

- Continuously release updates and service improvements

(Learn where continuous delivery fits in the CI/CD cycle.)

Immediate feedback

Because deployment automation is less error-prone and can be released at a higher frequency, you get your feedback sooner—even immediately.

Your business can collect valuable consumer feedback and integrate them into future releases for improved performance and productivity at a higher rate. This feedback is vital; it's often the difference between a product delighting its target audience or missing them entirely.

Deployment automation makes business sense

Deployment automation lets you release new features and applications more quickly and frequently while removing the need for human intervention in application deployments. If your business is ready to upgrade your software deployment systems, deployment automation offers many benefits to grow your overall efficiency.

Related reading

- BMC DevOps Blog
- Deployment Pipelines (CI/CD) in Software Engineering
- Testing Automation Explained: Why & How To Automate Testing
- What is Server Automation? Server Automation Explained
- What are Service Orchestration and Automation Platforms (SOAPs)?