GITHUB, GITLAB, BITBUCKET & AZURE DEVOPS: WHAT'S THE DIFFERENCE?



Version control is the backbone of any software development pipeline, enabling multiple users to simultaneously develop the same code base. It also acts as a single source of truth with a perfectly kept history of all the codebase changes.

There are different types of version control mechanisms in the market; the primary ones are Mercurial and Git. While both are solid version control methods, Git has gained wider adaptation in the industry due to its extensive feature set and configuration options. These configuration options enable Git to be easily integrated into projects at any scale.
One of the most important decisions any software development team needs to undertake is choosing a proper version control platform. This is a decision that should be finalized before starting the development, as migrating to a new platform during an active development can be a complex and time-consuming process. It will also negatively impact the complete development process.

In this post, we will discuss how to go about selecting a git-based version control platform and the current leading version control platforms.

(This article is part of our DevOps Guide. Use the right-hand menu to navigate.)

Choosing a version control platform

You have to consider multiple factors when selecting a version control platform. The primary factor

is the requirements of the development team, which trumps all the other factors.

Even if the platform is the industry standard, if it does not satisfy particular requirements, it is always better to go with a product that is aligned with the user requirements.

After you've considered the user requirements, you can begin to consider the following factors.

Additional features/integrations

Version control platforms have evolved beyond simple source control functionality and offer a wide range of features from robust collaboration features to automation and even complete build pipelines. These extra features will become even more valuable with most organizations moving to DevOps methods.

Pricing

As with any organization, CapEx or OpEx directly impacts their bottom line. Therefore, pricing plays a major role in the final decision when introducing a new platform or tool.

If your organization is unable to afford a particular platform, that platform will be out of the selection process, no matter its features.

User experience/ease of use

This comes down to the usability of the platform, from configurations and setup process to flexibility offered when creating different workflows will be factors that determine the user experience of the platform. Even if a platform offers an excellent feature set, if it comes with complex configuration and usage requirements, users will prefer to go with a simpler platform.

Additionally, if the development team has previous experience in a particular tool, it will be a good idea to give preference to that tool as it will improve the platform adoption experience with fewer complications.

Support/community

All paid options come with some kind of official support with different payment options offering different levels of support. On the other hand, free services are limited to community-based support. As a general rule, the higher the cost, the greater the support will be.

User community plays an important part as they can be a wealth of knowledge regardless of if the tool is paid or free. Platform maturity also plays a vital role here as mature platforms will have a larger user base leading to greater community-based resources.

Summary

At their core, GitHub, GitLab, Bitbucket, and Azure DevOps are comparable when it comes to core version controlling functionality. Their true differentiating factors are the additional features and integrations offered by each platform that aid in the software development life cycle.

In the next sections, we will dive into each of these platforms to get a better understanding of their feature set.

What is GitHub?



GitHub by Github Inc was founded in 2007 and now has grown to become the largest version control platform for hobbyist, open source, and enterprise projects with over 65 million developers with 72% adaptation in Fortune 50 companies.

Microsoft purchased GitHub in 2018, and now it is under the Microsoft services umbrella with the full backing of Microsoft. With Microsoft's renewed focus and investments on open source platforms, users can confidently use GitHub for any open source project. This has made the previously paid features like private repositories into free services.

GitHub is a fully online platform without any option to run a self-hosted coming under the enterprise tier. It exclusively supports Git and offers a feature set that spans collaborative coding, automation, and CI/CD to project management.

GitHub Features

- Unlimited Public and Private repositories with unlimited collaborators for both free and paid tiers
- GitHub Codepaces to create a secure and high-performance development environment backed by VSCode
- GitHub Actions automates workflows directly within GitHub repositories — and teams can go further by adding the Control-M integration with GitHub Actions to schedule, monitor, and govern Actions workflows alongside enterprise data and application pipelines.
- GitHub Pages to directly create a website from a GitHub repository
- Dedicated security feature for both user and repository management from 2FA to code scanning, dependency vulnerability alerts
- SAML, LDAP support for the enterprise
- Wide range of client apps (GitHub CLI, Desktop, Mobile)
- Inbuilt project management features with features such as GitHub Projects, GitHub Issues
- Hundreds of apps via GitHub Marketplace to extend and integrate with third-party services

What is GitLab?

GitLab introduces itself as a complete DevOps platform designed to power modern DevOps processes. It comes under two versions:



- GitLab Community Edition
- The GitLab Enterprise Edition, which offers free and paid versions of GitLab services

GitLab aims to provide a single application to manage the DevOps lifecycle. This includes features

that range from project planning, code management, packaging to deployment and monitoring.

GitLab has a comparatively smaller community and a limited set of third-party integration options compared to GitHub, which can be complex for new users. The primary caveat of GitLab is that most advanced features are only available for paid users, and ultimate versions can get quite expensive.

GitLab Features

- Single platform to power a complete DevOps pipeline
- Built-in CI/CD and AutoDevOps services to build, test and deploy applications
- GitLab ChatOps to integrate CI/CD to services like slack and be managed by slack commands
- Azure DevOps offers deep pipeline capabilities and source control integration — and teams can integrate Azure DevOps with enterprise orchestration via Control-M to schedule, monitor, and govern ADO pipelines alongside broader data and application workflows.
- Security scanning including support for scanning containers and the ability to implement network policies
- License Compliance to directly validate if the project dependencies are compliant with project requirements
- Dynamic Application Security Testing (DAST) to analyze running web applications

What is BitBucket?



BitBucket by Atlassian started its life as a Mercurial based platform. Over time, however, Mercurial support has depreciated and BitBucket has shifted to a complete Git-based platform.

BitBucket is offered as:

- A cloud solution via Bitbucket Cloud
- A self-managed solution via Bitbucket Data Center. The Data Center option is an enterprise offering targeted at providing complete control over the development environment.

BitBucket offers a free version up to five users with a relatively cheap premium option compared to other platforms. As a product managed by Atlassian, BitBucket shines when interacting with other popular Atlassian products such as Jira, Confluence, Trello, Opsgenie, etc.
This makes Bitbucket the go-to choice if the organization relies on other Atlassian products.

BitBucket Features

- Unlimited free private repositories for free users
- Direct integrations with Atlassian products that can be used to facilitate all aspects of the software development lifecycle (SDLC)
- Atlassian Open DevOps approach and the Marketplace provides the ability to integrate with multiple third-party services and platforms
- BitBucket Pipelines, an integrated CI/CD service that can be used to create automated workflows
- Sourcetree client for Windows and Mac to manage Git repositories
- Integrated security practices from pull request scanning to the ability to integrate Snyk vulnerability scanner allows organizations to easily facilitate DevSecOps

- Advanced user management with support for SSO, JIT, and OAuth 2.0
- Built-in analytics tools to get actionable data and insights from all supported Atlassian products

What is Azure DevOps?

Azure DevOps is a complete DevOps platform based on the Microsoft Azure cloud platform, with the version control coming under the Azure Repos service in the DevOps suite. The Azure DevOps platform comes in two flavors:

- The Azure-powered Azure DevOps Services
- The Azure DevOps Server is the successor to the Microsoft Team Foundation Server (MTFS) and offers an offline self-management option

The cloud-based Azure DevOps service offers the complete platform excluding test plans completely free of charge for up to five users. Visual Studio subscriptions gain full access to Azure DevOps services.

Azure DevOps will be the best option to power software development pipelines at any scale if you are engrossed in the Microsoft ecosystem or reliant on Azure for your application.

Features of Azure DevOps

- Direct integration with Azure cloud platform and other Microsoft service offerings
- A full feature-rich platform for free users that includes unlimited private repositories
- Azure DevOps Server Express to implement Azure DevOps offline for individual developers or teams up to 5 users completely free of charge
- Over 1,000 integrations are available via the Extensions Marketplace to complement and extend DevOps functionality
- Azure Boards, Pipelines, Repo, Test Plans, Artifacts to power all stages of an SDLC
- Support for any Git client
- Semantic code search

Summing up version control

Choosing a platform is not as simple as selecting the best version control tool as all these platforms have evolved beyond the simple version control functionality to offering complete DevOps platforms that can power SDLC at any scale.
If you are in the Microsoft ecosystem, go with Azure DevOps, and if you rely on Atlassian products, Bitbucket will be the way to go. GitLab's heavy focus on a completely centralized DevOps platform makes it the ideal solution for someone who is looking for a single solution.
Finally, GitHub, with its maturity and feature set, is the best well-rounded solution that can be used to power any software development project.

Related reading

- BMC DevOps Blog
- Implementing GitOps To Deliver Cloud Native Applications

- Ignoring in Git: How To Use .gitignore Files
- The Role of Microservices in DevOps
- DevOps Metrics & KPIs