Version Control Systems

The importance of version control systems for tracking changes cannot be overstated, and companies should choose the one that best suits their needs.

If you are planning to work on a big software development project that consists of technical concepts, require a collaboration of different team members and need frequent changes, then you need to use a version control system.

What is Version Control System?

- A version control system allows users to keep track of the changes in software development projects, and enable them to collaborate on those projects. Using it, the developers can work together on code and separate their tasks through branches.

- There can be several branches in a version control system, according to the number of collaborators. The branches maintain individuality as the code changes remain in a specified branch(s).

- Developers can combine the code changes when required. Further, they can view the history of changes, go back to the previous version(s) and use/manage code in the desired fashion.

Benefits of Using a Version Control System

- The main advantages of using a version control system include streamlining the development process, management of code for multiple projects and keeping a history of all changes within a code.

- A version control software saves all the changes in a repository. Hence, if the developers make a mistake, they can undo it. At the same time, they can compare the new code with a previous version(s) to resolve their grievance. This can reduce human errors and unintended consequences to a great extent. A great fit for any web development company around the globe.

- Additionally, it can be integrated with several software development tools like PaaS providers, integrated development environments (IDE) and build automation tools.

Best Version Control Systems

- There are plenty of options available in the market. Hence, I have created a list of some of best version control software to narrow the options and make things easier .

1. GitHub

- GitHub helps software teams to collaborate and maintain the entire history of code changes. You can track changes in code, turn back the clock to undo errors and share your efforts with other team members.

- It is a repository to host Git projects. For those wondering what is Git? It is an open source version control system that features local branching, multiple workflows, and convenient staging areas. Git version control is an easy to learn option and offers faster operation speed.

- 2. GitLab
-
- GitLab comes with a lot of handy features like an integrated project, a project website, etc. Using the continuous integration (CI) capabilities of GitLab, you can automatically test and deliver the code.
-
- You can access all the aspects of a project, view code, pull requests, and combine the conflict resolution.
-
-
-
-
-
-
-
- 8. Mercurial
-
- Mercurial is known for its efficiency in handling projects of all sizes. It is a free and distributed control management service that provides a simple and intuitive user interface.
-
- Developers and enterprises adore Mercurial for its backup system, search functionality, project tracking and management, data import and export, and data migration tool. It also features workflow management, history tracking, security management, access controls and more.
- 9. CVS Version Control (Concurrent Versions System)
-
- CVS is one of the oldest version control system and is a well-known tool among both commercial and open source developers. It allows you to check out the code you are planning to work on, and check-in the changes.
-
- It has the capability to handle projects with multiple branches so that teams can merge their code changes and contribute unique features to the project.
-
- Since CVS is here for a long time now, it is the most mature version control software.
- 10. Bitbucket
-
- Bitbucket is a part of the Atlassian software suite, so it can be integrated with other Atlassian services including HipChat, Jira, and Bamboo. The main features of Bitbucket are code branches, in-line commenting and discussions, and pull requests.
-
- It can be deployed on a local server, data center of the company, as well as on the cloud. Bitbucket allows you to connect with up to five users for free. This is good because you can try the platform for free before deciding to purchase.

- 3. Beanstalk
-
- Beanstalk is an ideal option for those who need to work from remote places. This software is based on browser and cloud, allowing users to code, commit, review and deploy using a browser.
-
- It can be integrated with messaging and email platforms for efficient collaborations related to codes and updates. It supports both Git and SVN and comes with built-in analytics features.
-
- For security, it leverages encryption, two-factor authentication, and password protection functionalities.

4. PerForce

- Perforce delivers the version control capabilities through its HelixCore. The HelixCore comes with a single platform for seamless team collaboration, and support for both centralized and distributed development workflows.

- It is a security solution that protects the most valuable assets. HelixCore allows you to track the code changes accurately and facilitates a complete Git ecosystem.

5. Apache Subversion

- Apache Subversion is another open source version control system, which was founded by CollabNet a couple of decades ago. Both open source arena and enterprises consider it a reliable option for valuable data.
- Key features of Subversion include inventory management, security management, history tracking, user access controls, cheap local branching, and workflow management.

6. AWS CodeCommit

- AWS CodeCommit is a managed version control system that hosts secure and scalable private Git repositories. It seamlessly connects with other products from Amazon Web Services (AWS) and hosts the code in secured AWS environments. Hence, it is a good fit for the existing users of AWS.

- AWS integration also provides access to several useful plugins from AWS partners, which helps in software development.

7. Microsoft Team Foundation Server

- Developed by Microsoft, the Team Foundation Server is an enterprise-grade tool to manage source code and other services that need versioning. It can keep track of work items to find defects, requirements, and scenarios in a project.

- It comes with several unique features like Team Build, data collection and reporting, Team Project Portal, Team Foundation Shared Services, etc.

8. Mercurial

- Mercurial is known for its efficiency in handling projects of all sizes. It is a free and distributed control management service that provides a simple and intuitive user interface.

- Developers and enterprises adore Mercurial for its backup system, search functionality, project tracking and management, data import and export, and data migration tool. It also features workflow management, history tracking, security management, access controls and more.

9. CVS Version Control (Concurrent Versions System)

- CVS is one of the oldest version control system and is a well-known tool among both commercial and open source developers. It allows you to check out the code you are planning to work on, and check-in the changes.

- It has the capability to handle projects with multiple branches so that teams can merge their code changes and contribute unique features to the project.

- Since CVS is here for a long time now, it is the most mature version control software.

10. Bitbucket

- Bitbucket is a part of the Atlassian software suite, so it can be integrated with other Atlassian services including HipChat, Jira, and Bamboo. The main features of Bitbucket are code branches, in-line commenting and discussions, and pull requests.

- It can be deployed on a local server, data center of the company, as well as on the cloud. Bitbucket allows you to connect with up to five users for free. This is good because you can try the platform for free before deciding to purchase.

•Wrapping Up

- These are the best version control systems out there that a web development company should consider using, according to the requirements, of course. While going for a VCS, you must consider the purpose, affordability, evaluation process and use cases.

- Published at DZone with permission of Vaibhav Shah. See original article here at Dzone