**Discussion Topics:**

Research version control tools/software (remember GitHub?), and find at least two sites that provide tools for version control.

1. Do both tools provide the same features? If not, what is the difference?
2. Any advantages/disadvantages mentioned?
3. If you were making the decision, which would you select and why?
4. Include the URL of each site.

Version control is an important aspect of creating code. It helps maintain and track code updates. Many tools are available to improve this process, including various features. I found two websites with version control tools: Apache Subversion and Azure DevOps.

With Azure, there are features like Azure Boards, Pipelines, Test Plans, GitHub Advanced Security for Azure DevOps, GitHub Copilot for Azure DevOps, Repos, Artifacts, and Managed DevOps Pools (Microsoft, 2024). Azure Boards means teams can track work with various things like Kanban boards, backlogs, dashboards, and custom reports (Microsoft, 2024). Other customization options include the dashboard and workflows (Microsoft, 2024). It can also connect with extensions and GitHub (Microsoft, 2024). Azure Pipelines are cloud-hosted pipelines for different operating systems, making building and deploying code quicker (Microsoft, 2024). Test Plans help plan a testing solution (Microsoft, 2024). Azure Artifacts allows packages to be created, hosted, and shared with a team (Microsoft, 2024). Some disadvantages are that Azure DevOps is often locked in with Microsoft, so other integrations may cause issues (Nerella, 2024). Azure DevOps is also expensive and complex (Nerella, 2024).

Apache Subversion is an open-source project under Apache. There are many features within Apache Subversion. Some of those include it having most CVS features; directories are versioned; copying, deleting, and renaming are versioned; free-form versioned metadata; atomic commits; branching and tagging are cheap; merge tracking; file locking; symbolic links can be versioned; executable flag is preserved; Apache network server option, with WebDAV/DeltaV protocol; standalone server option; parseable output; localized messages; interactive conflict resolution; repository read-only mirroring; and much more (Apache, n.d.). Some disadvantages include a complex branding model, limited performance and scalability, and only basic merging capabilities (Perforce, 2023).

Both tools provide version control, which includes branching, merge tracking, access control, and support for extensions or other compatible software. If I were to select one, I would choose Azure if I were using a Microsoft operating system, since it can more seamlessly integrate with other software.

https://subversion.apache.org

https://azure.microsoft.com/en-us/products/devops

**References**

Apache. (n.d.). *Apache Subversion*. Subversion.apache.org. Retrieved June 10, 2025, from https://subversion.apache.org

Microsoft. (2024). *Azure devops services | microsoft azure*. Azure.microsoft.com. https://azure.microsoft.com/en-us/products/devops

Nerella, H. (2024, April 18). *Azure DevOps: Insights, Advantages, and Challenges - DZone*. Dzone.com. https://dzone.com/articles/azure-devops-insights-advantages-and-challenges

Perforce. (2023, November 3). *What Is SVN (Subversion)?* Perforce Software. https://www.perforce.com/blog/vcs/what-svn

***Before you submit your thread, put your name in the subject line.***

**Assignment Requirements and Grading:**

1. An initial post of approximately 250 words is due by **Thursday, 11:59 p.m., CST**.
2. For the initial post to be considered substantive, it should be at least 250 words in length and fully cover the topics being presented. Single sentence definitions or responses will not be awarded points.
3. Submit your post by clicking on the **Assignment Link** above, then **Create Thread**. You must create a thread in order to view your peers' posts. Tip: Create your post in a Word document and then copy and paste your work into the thread.
4. A minimum of three (3) responses, **to the original threads of other students**, of 100-200 words each are due by **Sunday, 11:59 p.m., CST**.
5. To view the rubric grading criteria, click on the following link: [Discussion Board Grading Rubric](https://content.bellevue.edu/cst/csd/rubricdbv3.pdf).

**(50 points)**

Hey, Lea! You did a very nice job on your post for this week! Apache Subversion and Mercurial are examples of great version control tools. I was amazed to see how the different version control systems are available. If one does not seem to fit your needs, there is always another one that might be better suited. I also wrote about Apache Subversion and used the article from Perforce. Apache Subversion also might not be as good as other version control tools like Azure, since it is highly compatible with Windows operating systems. Mercurial does seem to fit better for Windows than Apache Subversion.

Hello, Brett! I enjoyed reading your post for this module. You did a comprehensive job explaining each version control system you found. I like the approach you took to begin the assignment. Understanding the differences between distributed and centralized version control is vital when deciding on the best implementation system. I have really enjoyed working with git. At least, I had some issues with setup, but once I got over that hurdle, I have loved using it ever since. It is so easy to sync a repository through the terminal and it works seamlessly. Plus, so much helpful information on GitHub can be helpful with projects or other coding circumstances.

Hi, Truman! I completely agree that version control tools are a necessary part of software development projects. Our software tools come with pros and cons, so I agree that understanding each of their strengths and weaknesses can make it easier to decide which is best for each circumstance. I also chose SVN as one of the possible version control tools, so it was interesting reading which parts stood out to you. I did not come across New Relic during my research, but after reading your explanation, I would have to agree that New Relic is better to use than SVN.