SVN vs Git - Detailed Comparison

1. Command Comparison: SVN vs Git

Operation	SVN Command	Git Command	Description
Checkout/Clone	svn checkout <repo_url>	git clone <repo_url>	Fetch repository
View Status	svn status	git status	Check changes
Add Files	svn add <file>	git add <file>	Add files to version control
Commit Changes	svn commit -m 'msg'	git commit -m 'msg'	Commit with message
Update/Pull	svn update	git pull	Update working copy or pull chan
Revert Changes	svn revert <file>	git checkout -- <file>	Discard changes
View History	svn log	git log	View commit history
Create Branch	svn copy <url>/trunk <url>/branches/xyz	git branch xyz	Create new branch
Switch Branch	svn switch <url>/branches/xyz	git checkout xyz	Switch to branch
Merge Branch	svn merge <url>/branches/xyz	git merge xyz	Merge changes from a branch
Delete Branch	svn delete <url>/branches/xyz	git branch -d xyz	Delete branch
Resolve Conflicts	svn resolved <file>	git add <file>	Resolve conflicts manually
View Remote Repos	N/A	git remote -v	View remote URLs
Tagging	svn copy trunk to tags/v1.0	git tag v1.0	Create a tag

2. SVN (UI) vs Git (CLI/UI) Comparison

Operation	SVN (UI)	Git (CLI/UI)	Notes
Checkout/Clone	Right-click > SVN Checkout	git clone or UI	SVN gets working copy, Git gets full repo
Commit	Right-click > Commit	git commit or UI	Git commits locally, needs push
Update/Pull	Right-click > Update	git pull	Pull changes from remote
Add Files	Right-click > Add	git add or UI	Stage files before commit
Check Status	Overlay icons	git status or UI	Check file status
View Logs	Right-click > Show Log	git log or UI	See commit history

SVN vs Git - Detailed Comparison

Branching	Right-click > Branch	git branch / UI	Lightweight in Git
Merging	Right-click > Merge	git merge or UI	More visual in Git tools
Conflict Resolution	Manual + Dialog	Git merge tools	Git has advanced tools
Tagging	Right-click > Tag	git tag or UI	Both support tagging
Push Changes	Auto with commit	git push	Git separates commit and push
Offline Access	Limited	Full access	Git supports full offline work
Cloud Integration	Manual setup	GitHub, GitLab, etc.	Better in Git
Learning Curve	Easy (UI)	Medium (CLI), Easy (UI)	Git UIs help learning
Performance	Slower branching	Fast operations	Git optimized for scale

3. Summary

- SVN (via UI) is user-friendly and centralized, suitable for simpler, controlled workflows.

- Git (CLI/UI) is distributed, supports offline work, fast branching, and integrates easily with modern tools.

- Git is better for collaborative and scalable projects, while SVN may be suitable for legacy systems or strict control needs.