How Version Control System (VCS) is different from SCM?

■ Introduction

• Version Control System (VCS) and Source Code Management (SCM) are closely related but not the same.

• VCS is a subset of SCM. While VCS tracks changes in code, SCM covers a broader set of processes including collaboration, workflow, and release management.

• Both are essential for modern software development teams.

■ Version Control System (VCS)

• A Version Control System tracks changes made to files over time.

• It allows developers to revert back to earlier versions when needed.

• Supports branching and merging for parallel development.

• Examples: Git, Mercurial, Subversion (SVN).

• Real-time Example: Using Git to create branches for new features, then merging them into the main branch after testing.

■ Source Code Management (SCM)

• SCM is a broader concept that includes VCS as one of its components.

• It manages not only code versions, but also team collaboration, workflows, and integrations with CI/CD pipelines.

• SCM helps enforce coding standards, review processes, and release management.

• Examples: GitHub, GitLab, Bitbucket (which provide SCM on top of Git).

• Real-time Example: GitHub Actions automating build, test, and deployment along with version control.

■ Comparison: VCS vs SCM

Aspect	VCS	SCM
Scope	Tracks code changes	Manages code + collaboration + workflow
Focus	Commit history, branching, merging	Team productivity, release management
Tools	Git, SVN, Mercurial	GitHub, GitLab, Bitbucket
Collaboration	Basic (via branches and merges)	Advanced (issues, pull requests, CI/CD)
Real-time Example	Developer commits locally	GitHub triggers CI/CD pipeline on commit

■ Conclusion

• VCS is primarily about tracking changes and managing versions of code.

• SCM is a broader discipline that includes VCS but extends to collaboration, workflow, and automation.

• In simple terms: All VCS are part of SCM, but not all SCM is just VCS.