

Git: Standard pro verzování kódu

Git je nástroj, který sleduje změny v souborech a umožňuje se kdykoliv vrátit k jakékoliv předchozí verzi projektu. Na rozdíl od starších systémů (jako SVN) je Git **distribuovaný**, což znamená, že každý vývojář má v počítači kompletní kopii celé historie projektu, nikoliv jen aktuální verzi.

1. Základní stavy souborů v Gitu

Práce v Gitu probíhá ve třech hlavních oblastech (sekcích):

- **Working Directory (Pracovní adresář):** Místo, kde soubory fyzicky upravujete.
- **Staging Area (Index):** „Přípravná“, kam přidáváte změny, které chcete zahrnout do příštího uložení.
- **Repository (Adresář .git):** Místo, kde Git trvale ukládá historii jako snímky (snapshots).

2. Klíčové koncepty

Snímky (Snapshots) namísto rozdílů

Většina systémů ukládá změny jako seznam rozdílů (diffs). Git uvažuje o datech jako o **sérii snímků** celého souborového systému. Pokud se soubor nezměnil, Git nevytváří novou kopii, ale pouze odkaz na předchozí identický soubor.

Větvení (Branching)

Větvení je „vražedná funkce“ (killer feature) Gitu. Je extrémně rychlé a lehké. Větev je v podstatě jen pohyblivý ukazatel na konkrétní commit. To umožňuje:

- Vyvíjet novou funkci v izolaci (**Feature Branch**).
- Opravovat chybu bez ovlivnění hlavní verze kódu.
- Experimentovat bez rizika rozbití projektu.

3. Základní workflow v terminálu

Běžný den vývojáře s Gitem vypadá takto:

1. `**git clone <url>:**` Stáhne existující repozitář z internetu.
2. `**git add <soubor>:**` Přesune změny z pracovního adresáře do `**Staging`

Area**.

3. ****git commit -m "popis":**** Vytvoří snímek změn a uloží jej do lokální historie.

4. ****git push:**** Odešle lokální commity do vzdáleného repozitáře (např. na GitHub).

5. ****git pull:**** Stáhne a sloučí změny od kolegů ze vzdáleného repozitáře do vašeho počítače.

4. Git vs. GitHub / GitLab

Častým omylem je zaměňování Gitu s GitHubem.

- **Git:** Je samotný nástroj (software), který běží u vás v počítači.
- **GitHub / GitLab / Bitbucket:** Jsou webové služby (hostingy), které uchovávají vaše [vzdálené repozitáře](#) a přidávají funkce jako diskuse, správa úkolů a grafické rozhraní.

5. Proč je Git nepostradatelný?

- **Zpětná kompatibilita:** Pokud nová funkce rozbije aplikaci, návrat k funkční verzi je otázkou vteřin.
- **Kolaborace:** Umožňuje automatické slučování kódu (Merge) od různých lidí. Pokud dva lidé upraví stejný řádek, Git ohlásí **konflikt**, který musí lidé vyřešit.
- **Integrita:** Každý soubor a commit je v Gitu identifikován pomocí **SHA-1 hashe**. Není možné změnit obsah souboru, aniž by o tom Git věděl.

Související články:

- [Repozitář: Kde se kód ukládá](#)
- [CI/CD: Automatizace na základě Gitu](#)
- [Linux Kernel: Projekt, pro který Git vznikl](#)

Tagy: it git vcs development programming software-engineering

From:

<https://www.serviceit.cz/> - IT ENCYKLOPEDIE

Permanent link:

<https://www.serviceit.cz/doku.php?id=it:sw:git>

Last update: **2026/01/02 20:28**

