

Rezime: *Sistemi za kontrolu verzija predstavljaju alate koji pomažu u praćenju razvoja i održavanju softverskih projekata. Ovi sistemi olakšavaju višekorisnički rad pri čemu se evidentiraju izmene svih korisnika. Primenom Sistema za kontrolu verzija povećava se sigurnost podataka projekta i produktivnost zaposlenih, a održavanje projekta je olakšano. U ovom radu je prikazan razvoj sistema za kontrolu verzija, opis i osnovni principi funkcionisanja sistema.*

Ključne reči: kontrola verzija, skladište, revizija, grananje, etiketiranje

Abstract: *Version Control Systems represent tools which are helpful for tracking the process of developing software's projects and maintaining the projects. These systems allow multi-user environment and all the users' changes are logged. Using these systems improve the security of data and files, and productivity of employees. In this way maintenance of the project becomes easier. This paper shows development of version control systems, description and basic principles of functioning.*

Key words: version control, repository, revision, branching, tagging

1. UVOD

Tokom razvoja većih softverskih projekata postoji potreba za kvalitetnim praćenjem verzija kroz koje projekat prolazi. Pri tome se mora voditi računa o održavanju celog projekta kao i svih njegovih verzija. Praćenje razvoja projekta može se raditi ručno, ali je to vrlo težak i mukotrpan posao, koji je podložan greškama (gubitak izvršenih promena, nemogućnost čuvanja svih verzija, ...).

Danas se za praćenje razvoja softverskih projekata koriste pomoćni alati koji se jednim imenom nazivaju **Sistemi za kontrolu verzija** (*Version Control Systems - VCS*). Sistemi za kontrolu verzija pomažu u održavanju softverskih projekata u ažurnom stanju i prate njihov razvoj tokom vremena. Iako se ovi sistemi koriste, uglavnom, za praćenje razvoja softverskih projekata, oni se mogu koristiti i za praćenje verzija bilo koje vrste tekstualnih elektronskih podataka (word dokumenti, txt dokumenti, ...). [3]

Neki od problema, koji mogu nastati u toku razvoja projekta, su:

- Komplikovano praćenje paralelnog razvoja projekta na više radnih mesta.
- Promene u jednom delu projekta mogu prouzrokovati promene u drugom delu, a da se to nije očekivalo i predvidelo.
- Kada se izvrši neka promena i sačuva se u projektu, teško je u slučaju potrebe, vratiti se na prethodno stanje projekta.
- Na kraju razvoja projekta potrebno je spojiti različite celine razvijane na više radnih mesta, što je često težak i komplikovan posao.

Ovi problemi se uspešno prevazilaze primenom Sistema za kontrolu verzija. Danas postoji veliki broj razvijenih sistema za kontrolu verzija softvera i elektronskih podataka. Različiti proizvođači softvera nude ove sisteme kao podršku pri razvoju softverskih aplikacija. Neki od proizvođača su Microsoft, Uni Software Plus, Perforce Inc., a veliki broj ovih sistema je *open source*.

U ovom radu će biti prikazan razvoj, opis i funkcionisanje Sistema za kontrolu verzija.

2. RAZVOJ SISTEMA ZA KONTROLU VERZIJA

Početak Sistema za kontrolu verzija je bio razvoj programa *diff* i *patch*, koji su počeli da se koriste za poređenje fajlova pri projektovanju softverskih aplikacija. Program *diff* prikazuje razliku između dva fajla. Ako se program *diff* primeni na neki fajl pre i posle njegove modifikacije, rezultat će biti samo modifikovani redovi, dok će ostali biti preskočeni. Program *patch* koristi razlike nekih fajlova (uglavnom dobijenih kao proizvod programa *diff*), i te razlike pridružuje odgovarajućem fajlu. Na primer, ako se pronade razlika između fajla A i fajla B (pomoću *diff* programa), i na tu razliku i prvi od tih fajlova se primeni program *patch*, odgovarajući fajl će se dopuniti razlikama. Ovime je omogućeno da više ljudi radi na razvoju jednog projekta i da svoje promene pomoću *diff* i *patch* programa učine dostupne drugim

korisnicima. [3]

Tokom razvoja projekta, ponekad se desi da promene koje su napravljene u fajlu posle nekog vremena treba poništiti tj. projekat treba vratiti u stanje kakvo je nekada bilo. Međutim, ako se projekat održava na ovaj način jako je teško ući u trag ko je kakve promene u međuvremenu napravio, a čak i da se to ustanovi, sve promene bi morale ručno da se brišu, što je težak i dugotrajan proces podložan greškama. Rešenje tog problema je bilo u izgradnji sistema za praćenje istorije projekta tj. sistema koji bi mogao da pronađe ranije verzije projekta. Izgrađeno je više takvih sistema, a jedan od poznatijih je Revision Control System (RCS) [1]. RCS je imao veliku primenu ali su uočeni i njegovi nedostaci:

- Nemogućnost kontrole celog projekta,
- Nije dozvoljen istovremeni pristup jednom fajlu od strane više korisnika,
- Nije mrežno orjentisan - svi korisnici sistema moraju da rade na mašini na kojoj je instaliran RCS ili da organizuju transfer podataka između radnih stanica i RCS servera.

Kao rezultat prevazilaženja ovih nedostataka, 1986. godine je razvijen Concurrent Versioning System (CVS) od strane Dick Grune-a [3]. Ovaj sistem je počeo da se razvija od skupa skriptova koje RCS čine funkcionalnijim. CVS je doživeo veliku popularnost, ali se u praksi pokazalo da i on ima neke nedostatke (komplicovana promena strukture skladišta, loš oporavak od pada sistema, ...).

U periodu od 2000. do 2004. godine razvijen je sistem Subversion v1.0 [3]. Od tada njegova popularnost raste i pojavljuju se njegove novije verzije. Subversion je besplatan softver koji se može preuzeti sa različitih internet lokacija.

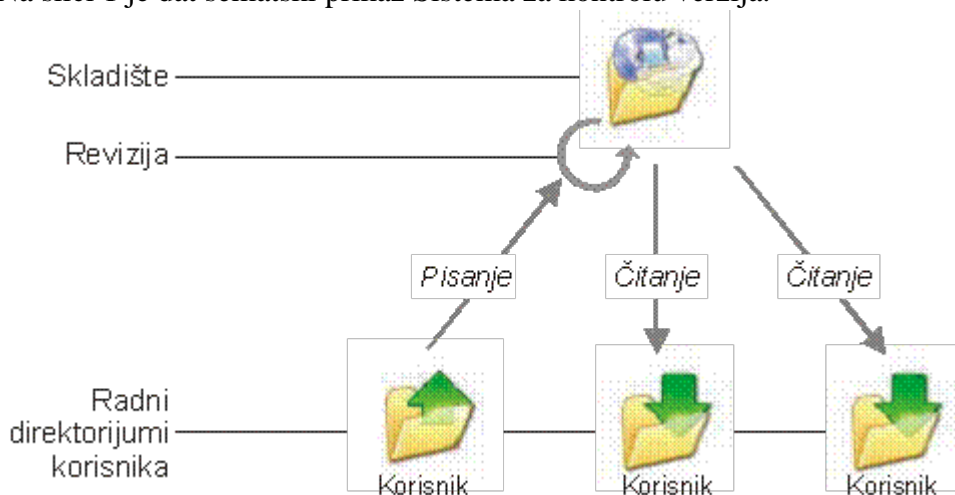
Danas postoji veliki broj sistema za kontrolu verzija kako komercijalnih, tako i onih besplatnih. Pored pomenutih sistema, koriste se i Git, Perforce, Bazaar, Visual SourceSafe, i dr. [4, 5]

3. OPIS SISTEMA

Osnovni elementi Sistema za kontrolu verzija su [2, 3]:

- Skladište (*repository*),
- Radni direktorijumi korisnika (*working directories*), i
- Revizije sistema (*revisions*).

Na slici 1 je dat šematski prikaz Sistema za kontrolu verzija.



Slika 1. Prikaz Sistema za kontrolu verzija

Skladište je centralni deo sistema za kontrolu verzija, gde se smeštaju fajlovi projekta, izvršene promene korisnika na njima i drugi sistemski podaci. Skladištu može da pristupi veći broj korisnika istovremeno sa ciljem čitanja ili izmena fajlova. Organizacija skladišta varira od sistema do sistema. Skladište može biti:

- direktorijum na serveru u koji su smešteni svi fajlovi projekta i sve njegove verzije,
- baza podataka koja sadrži zapise svih fajlova projekta tj. sve verzije projekta,
- kompleksan distribuirani sistem koji smešta fajlove projekta, tj. sve njegove verzije, na više računara.

**---- OSTATAK TEKSTA NIJE PRIKAZAN. CEO RAD MOŽETE
PREUZETI NA SAJTU WWW.MATURSKI.NET ----**

[BESPLATNI GOTOVI SEMINARSKI, DIPLOMSKI I MATURSKI TEKST](http://WWW.SEMINARSKIRAD.ORG)

RAZMENA LINKOVA - RAZMENA RADOVA

RADOVI IZ SVIH OBLASTI, POWERPOINT PREZENTACIJE I DRUGI EDUKATIVNI MATERIJALI.

**WWW.SEMINARSKIRAD.ORG
WWW.MAGISTARSKI.COM
WWW.MATURSKIRADOVI.NET**



NA NAŠIM SAJTOVIMA MOŽETE PRONAĆI SVE, BILO DA JE TO [SEMINARSKI](#), [DIPLOMSKI](#) ILI [MATURSKI](#) RAD, POWERPOINT PREZENTACIJA I DRUGI EDUKATIVNI MATERIJAL. ZA RAZLIKU OD OSTALIH MI VAM PRUŽAMO DA POGLEDATE SVAKI RAD, NJEGOV SADRŽAJ I PRVE TRI STRANE TAKO DA MOŽETE TAČNO DA ODABERETE ONO ŠTO VAM U POTPUNOSTI ODGOVARA. U BAZI SE NALAZE [GOTOVI SEMINARSKI](#), [DIPLOMSKI I MATURSKI RADOVI](#) KOJE MOŽETE SKINUTI I UZ NJIHOVU POMOĆ NAPRAVITI JEDINSTVEN I UNIKATAN RAD. AKO U [BAZI](#) NE NAĐETE RAD KOJI VAM JE POTREBAN, U SVAKOM MOMENTU MOŽETE NARUČITI DA VAM SE IZRADI NOVI, UNIKATAN SEMINARSKI ILI NEKI DRUGI RAD NA LINKU [IZRADA RADOVA](#). PITANJA I ODGOVORE MOŽETE DOBITI NA NAŠEM [FORUMU](#) ILI NA

maturskiradovi.net@gmail.com