

TEN ČI ONEN, KTORÝ SYSTÉM NA SPRÁVU VERZIÍ ZVOLIŤ?

Dvakrát meraj a raz rež.

Maroš Ubreži

Slovenská technická univerzita
Fakulta informatiky a informačných technológií
Ilkovičova 3, 842 16 Bratislava
maros.ubrezi[zavináč]gmail[.]com

Abstrakt. Podporné prostriedky sú v súčasnosti neoddeliteľnou časťou procesu vývoja nového softvéru. Jedným z najdôležitejších je systém na správu verzií. Vďaka nemu môže na projekte organizovane pracovať súčasne viaceró ľudí. Lenže ako si zvoliť ten najvhodnejší pre náš projekt? Najprv je potrebné vybrať si verziovací prístup. Nikto vám nikdy nepovie, že jeden prístup je zaručene lepší ako druhý. Výber vždy závisí od špecifik projektu. Na základe vlastných skúseností a názoru odborníkov diskutujem o základných potrebách firemného, školského a komunitného tímu. Identifikujem prístup, ktorý ich najlepšie podporuje. Následne je potrebné si zvoliť konkrétny nástroj spadajúci do vybranej kategórie. Zamýšľam sa nad otázkami, ktorých odpovede usmerňujú a pomáhajú pri výbere toho najvhodnejšieho. Prečítaním eseje zistíte, že kedy považujem za vhodné použiť centralizovaný a kedy distribuovaný prístup. A hlavne dôvody prečo to tordím. Pri výbere konkrétneho nástroja sa budete môcť ríadiť zostavenou skupinou otázok.

Kľúčové slová: verzia, distribuované systémy pre správu verzií, centralizované systémy pre správu verzií, výber konkrétneho nástroja

Úvod

V súčasnom procese vývoja softvéru zohrávajú dôležitú úlohu podporné prostriedky. Bez nich sa žiaden moderný vývojový tím nezaobíde. Postupom času ako narastali požiadavky zákazníkov na výsledný produkt, narastala aj veľkosť tímov a počet riadkov zdrojového

2 Maroš Ubreži

kódu. Pridávaním novej funkcionality prestával byť ľahko udržiavateľný a vyskytla sa otázka ako vyvíjať softvér v tíme tak, aby sme v ňom nestrácali prehľad.

Do procesu vývoja softvéru sa zaviedlo používanie verzií, ktoré priniesli požadovanú organizovanosť. Postupne začali vznikať prvé systémy slúžiace na ich správu. V súčasnosti ich však existuje mnoho. Ako si teda zvoliť vhodný systém pre správu verzií?

Pred výberom konkrétneho nástroja je potrebné si správne zvoliť verziovací prístup. Autori Brian de Alwis a Jonathan Sillito v článku [1] vyjadrujú názor, že centralizované prestávajú byť vyhovujúce pre určité typy tímov a odporúčajú presun na decentralizované systémy. Ktorý z prístupov si zvoliť ak máme firemný, školský alebo komunitný tím?

A ako si zvoliť konkrétny nástroj? Konrad Hinsien, Konstantin Läufer a George K. Thiruvathukal vo svojom článku [2] uviedli otázky, ktoré majú pomôcť pri voľbe tohto nástroja. Sú to tie správne? Postačujú nám?

Strýčko Google pomôže

Čo by vás ako prvé napadlo keby vám niekto povedal, že by bolo dobré keby ste pri riešení použili systém na správu verzií? Alebo viete, že niečo takého existuje, bolo by dobré to použiť, no nikdy predtým ste s niečím takým nepracovali?

Každý uvedomelý používateľ internetu začne hľadať informácie o tomto probléme či už pomocou vyhľadávania Google alebo niektorého iného. Z vlastnej skúsenosti viem, že každý sa skôr či neskôr dopracuje k rozdeleniu systémov pre správu verzií. Existujú dve kategórie určené na základe prístupu. Centralizované a distribuované.

Preštuduje si materiály, ktoré mu o nich prezradí ich architektúru, typ repozitáru spôsob manažovania konfliktov a mnohé ďalšie informácie. No sú tieto informácie dostačujúce pre výber konkrétneho nástroja vhodného na verziovanie projektu? Dokázali by ste sa vy rozhodnúť bez pridania informácií o vašom tíme? Lebo ja nie.

Centralizovaný vs Distribuovaný prístup

Základom je správne si zvoliť kategóriu. Následne je možné si vybrať z nástrojov, ktoré patria do danej kategórie. Predpokladám, že všetky základné informácie o typoch systémov na správu verzií sme si našťudovali (Tab.1.). Ako teda zvoliť ten správny pre určitý typ tímu?

Tab. 1. Základné porovnanie verziovacích prístupov [3].

Prístup	Repozitár	Implicitne všetky práva na repozitár	Jednoduché zlučovanie vetiev	Offline operácie
Centralizovaný	centrálny	nie	nie	nie
Distribuovaný	lokálny	áno	áno	áno

Firemný tím

Čo si predstavujem pod pojmom firemný tím? Skupinu pracovníkov, ktorí všetci sídlia v jednej budove, firma má vlastný server a všetci majú k dispozícii pripojenie na internet. Taká bežná slovenská firma.

Na základe vlastných skúseností viem, že firemnému tímu úplne postačuje centrálny repozitár, pretože všetky zmeny ktoré používateľ vykoná je potrebné premietnuť hneď do testovacej vetvy na serveri. Lokálny repozitár je v tomto prípade zbytočný.

Predpokladáme, že vo firme máme pripojenie na internet, teda nie sú potrebné ani offline operácie.

Zlučovanie vetiev nie je častou operáciou. Testovacia vetva sa s produkčnou zlučuje len v prípade ak sú všetky časti novej verzie otestované.

Práca vývojára nevyžaduje mať implicitný prístup na celý repozitár, ale len na tú časť, na ktorú vyvíja novú funkcionálnosť. V prípade ak áno, tak mu ju zamestnanec ktorý má na starosti správu verziovacieho nástroja ihneď prideliť.

Po prejení hlavných vlastností je hneď jasné, že v tomto prípade je vhodné zvoliť centralizovaný prístup.

Školský tím

Pri vývoji v školskom tíme považujem za veľmi dôležité mať lokálny repozitár a s ním spojené offline operácie. Ako študent ktorý býva na internáte zvyknem na víkendy cestovať domov. Aj tento čas sa dá využiť na riešenie tímového projektu. Keďže však nemám v autobuse k dispozícii internet, tak vykonané zmeny commitujem do lokálneho repozitáru. Neskôr keď prídem domov, kde mám internetové pripojenie, posuniem zmeny z lokálneho repozitára do centrálného.

Hlavne z týchto dôvodov by som zvolil distribuovaný prístup, ten mi to všetko umožňuje. Centralizovaný nemá zmysel v tomto prípade ani uvažovať, keďže by som musel mať neustále k dispozícii pripojenie na internet.

Open source komunita

Vhodnosťou použitia systému pre správu verzií v prostredí open source komunit The Perl Foundation, OpenOffice, Python a NetBSD sa zaoberá článok [1]. Autori vychádzajú z emailovej komunikácie predstaviteľov jednotlivých komunit identifikujú prečo je vhodné prejsť z centralizovaného na distribuovaný systém pre správu verzií.

Komunity požadujú:

- implicitné poskytnutie prístupu ku všetkým častiam repozitára,
- atomické commity,
- jednoduché zlučovanie vetiev,
- zvýšenú podporu experimentálnych zmien,
- podporu offline operácií.

Považujem to za opodstatnené požiadavky, keďže najmä v prípade veľkej komunity s niekoľko tisíc členmi môže chvíľu trvať, kým by jednému členovi nastavili prístupové práva na časť repozitára v ktorej chce robiť zmeny.

4 Maroš Ubreži

Atomické commity by zabezpečili, že nepríde k poškodeniu súboru pri posielaní zmien.

Každý používateľ vyvíja novú funkcionálnu vetvu a zlučovanie s niekoľko tisíc používateľskými vetvami by bolo v centralizovanom systéme podstatne zložitejšie.

Podpora offline operácií je dôležitá, veď už nikto z nás nechce byť závislý od internetového pripojenia keď chce vykonávať zmeny v repozitári.

Ktorý nástroj to bude?

Samotná voľba verziovacieho prístupu nám nestačí. V každej kategórii sa nachádza hneď niekoľko konkrétnych nástrojov. Ak sme si už zvolili správny prístup pre našu situáciu, tak zostáva posledný krok – výber konkrétneho nástroja. Autori Konrad Hinsien, Konstantin Läufer a George K. Thiruvathukal vo svojom článku [2] uvádzajú štyri základné otázky, ktoré by si mal každý zodpovedať aby si zvolil ten najvhodnejší systém pre správu verzií.

1. Poznám niekoho kto má dostatočné skúsenosti s nástrojom na to aby mi vedel pomôcť?
2. Viem získať distribúciu ktorá sa bude dať nainštalovať na každý z mojich počítačov?
3. Dá sa systém na správu verzií dobre integrovať s mnou používanými vývojovými prostriedkami (IDE)?
4. Je mi jeho dokumentácia zrozumiteľná?

Sú to skutočne tie najvhodnejšie otázky?

Otázka č.1

Otázka č.1 možno nie je najvhodnejšia. Lebo aj keď poznám niekoho kto má dostatočné skúsenosti s daným nástrojom, neznamená to hneď že mi pomôže. Ten človek môže byť zaneprázdnený a nebude mať na mňa čas, alebo nebude mať vôbec chuť mi to vysvetľovať. Ja by som ju skôr zmenil na otázku: Mám s tým nástrojom dostatočné skúsenosti?

Otázka č.2

Otázka č.2 je veľmi vhodná. Nemá totiž zmysel vôbec uvažovať nad niektorým nástrojom keď ho nebudem vedieť nasadiť na svoj počítač. Dôvod môže byť rôzny. Napríklad ja mám iný operačný systém ako ten na ktorý je určený.

Otázka č.3

Podstatná otázka. Nechcem totiž zbytočne používať ďalší program. Čo možno najviac vecí by som rád ovládal z môjho IDE.

Otázka č.4

Pri používaní každého nástroja nastane situácia, že si neviem poradiť a potrebujem získať návod ako sa dá vyriešiť. V tom prípade prichádza na rad dokumentácia. A zrozumiteľnosť dokumentácie mi môže podstatne skrátiť čas na nájdenie riešenia.

Ja by som k týmto otázkam pridal ešte ďalšiu a to: Poskytuje výrobca voľnú alebo platenú licenciu? Je to z toho dôvodu, že nemám záujem zbytočne platiť za licenciu jedného nástroja ak existuje nejaký iný, ktorý mi poskytne rovnako vyhovujúcu funkcionálnosť. Myslím, že nikto z nás.

Môj zoznam otázok

Namiesto pôvodného zoznamu navrhujem používať nasledovné otázky:

1. Mám s daným nástrojom dostatočné skúsenosti?
2. Viem získať distribúciu ktorá sa bude dať nainštalovať na každý z mojich počítačov?
3. Dá sa systém na správu verzií dobre integrovať s mnou používanými vývojovými prostriedkami (IDE)?
4. Je mi jeho dokumentácia zrozumiteľná?
5. Poskytuje výrobca voľnú alebo platenú licenciu?

Postupným odpovedaním na jednotlivé otázky pre každý z nástrojov vo vybranej kategórii sa zúži počet vyhovujúcich adeptov. V prípade ak ich zostalo viacero, tak je jedno ktorý z nich si zvolím. A ak nezostal žiaden, tak musím znížiť svoje požiadavky tým, že otázky ktoré považujem za menej dôležité vynechám.

Záver

Nikto vám nikdy nemôže povedať, aby ste vo svojom projekte použili distribuovaný, prípadne centralizovaný prístup správy verzií, bez toho aby bližšie poznal vaše požiadavky. Nie je možné tvrdiť, že distribuovaný je lepší lebo je to novší prístup.

V tejto eseji som sa snažil opísať aké požiadavky môžu mať na systém pre správu verzií rôzne typy tímov. A na základe ich špecifických požiadaviek určiť najvhodnejší prístup. V budúcnosti by zrejme bolo možné pozrieť sa na túto problematiku aj pre iné rozdelenie tímov

Ďalej som rozoberal problém výberu konkrétneho nástroja využívajúceho zvolený prístup. Diskutoval vhodnosť otázok, ktoré by mali napomáhať pri jeho voľbe. Hlavnou myšlienkou bolo pomôcť a načrtnúť spôsob výberu systému pre správu verzií od voľby prístupu až po konkrétny nástroj.

Použitá literatúra

1. de Alwis, B., Sillito, J.: Why are software projects moving from centralized to decentralized version control systems?, chase, pp.36-39, 2009 ICSE Workshop on Cooperative and Human Aspects on Software Engineering, 2009

2. Hinsien, K., Läufer, K., Thiruvathukal, G.K.: Essential Tools: Version Control Systems, Computing in Science and Engineering, vol. 11, no. 6, pp. 84-91, Nov./Dec. 2009
3. Jotov, V.: An investigation on the approaches for version control systems, Proceedings of the 9th International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing, 2008

Annotation

One or the other, which way of version control to choose?

Supportive measures are now an integral part of the process of developing new software. One of the most important is a version control system. Thanks to him, more people can work organized on a project at the same time. But how to choose the most suitable for our project? First, you need to choose versioning approach. Nobody will ever tell you that one approach is certainly better than the other. The choice always depends on the specifics of project. Based on my own experience and opinion of experts, I discuss the basic needs of business, school and community team. I identify the approach that best supports them. Then it is necessary to choose a particular tool from selected category. I think about questions whose answers provide guidance and help in choosing the most suitable. Reading this essay you will find out that when it is appropriate to use a centralized approach and when distributed. And also reasons why I claim this. When you will be choosing a particular tool you can use the answers from questions.