

Podporné prostriedky pre riadenie a koordináciu distribuovaných tímov

LUBOŠ OMELINA

*Slovenská technická univerzita
Fakulta informatiky a informačných technológií
Ilkovičova 3, 842 16 Bratislava
bubo[závináč]kmit[bodka]sk*

Abstrakt. Distribuované projekty a projektové tímy sa stávajú štandardom pri vývoji veľkých softvérových systémov. Koordinácia a riadenie projektových tímov je veľmi komplexná úloha. Jej zložitosť rastie so všeobecnosťou softvérových systémov a distribúciou projektových tímov. Práve koordinácia medzi členmi distribuovaného tímu je často kritickou časťou pri riešení softvérového projektu a prináša so sebou viacero špecifických problémov. Preto sa najmä v distribuovaných softvérových projektoch používajú nástroje a podporné prostriedky, ktoré pomáhajú pri riadení, zlepšujú koordináciu a tým zefektívňujú proces tvorby softvéru v týchto podmienkach. Táto práca pojednáva o problémoch v koordinácii, ktoré vznikajú v distribuovaných tímoch, ponúka ich prehľad a rozoberá ich obmedzenie pomocou podporných prostriedkov, ktoré sú v súčasnosti používané na riadenie a koordináciu distribuovaných tímov.

Úvod

V tejto eseji sú rozobrané problémy spojené s procesom riadenia softvérového projektu, proces manažovania práce distribuovaných tímov a využitie podporných prostriedkov pre riadenie týchto procesov. V práci som sa zamerlal hlavne na koordináciu distribuovaných projektov a projektových tímov a problematiku spojenú s distribuovaným vývojom softvéru.

Proces riadenia vývoja softvéru, a manažovanie nie len distribuovaných tímov je široká problematika, preto v tejto eseji budú rozobrané len jej hlavné nedostatky a problémy, aby čitateľ získal najmä všeobecný prehľad o tejto problematike. Problémy spojené s koordináciou v distribuovaných tímoch je možné obmedziť použitím podporných prostriedkov, preto som uviedol aj základný prehľad o funkcionalite a vlastnostiach týchto prostriedkov.

Čo je iné v distribuovanom vývoji?

V tradičnom vývoji softvéru v tímoch, kde členovia tímu pracujú spolu existuje mnoho spôsobov ako koordinovať prácu na projekte. Prácu zjednodušuje aj to, že komunikácia medzi členmi tímu je intenzívnejšia a frekventovanejšia ako pri distribuovanom vývoji. Každý člen tímu má prehľad o tom čo robia ostatní a neustále spolu vytvárajú jednotný pohľad na problém a zároveň aj jeho riešenie.

Základný problém distribuovaného vývoja je to že veľa mechanizmov, ktorých funkcia je koordinovať prácu v klasickom tíme na malom priestore v distribuovanom prístupe chýbajú alebo nie je možné ich realizovať. Vzdialenosť medzi členmi tímu do značnej miery ovplyvňuje ich schopnosť kolaboratívnej práce. Umiestnenie celého tímu do spoločného priestoru má veľký efekt na rýchlosť a efektívnosť vývoja. Vzdialenosť medzi členmi tímu zvyšuje širokú rozmanitosť prekážok v koordinácii. Distribuované projekty sú rôznorodé a od ich odlišností závisí aj rôznorodosť problémov vznikajúcich pri koordinácii. Nasledujúce miesta sú identifikované ako najpravdepodobnejšie pre vznik problému [2] :

Menej komunikácie, menej efektívnej komunikácie. Ľudia komunikujú oveľa menej so ľuďmi ktorý sú vzdialený. Rovnako aj v tímoch ktoré sú distribuované ľudia komunikujú menej. Vo väčšine prípadoch sila komunikácie priamo úmerne rastie ak sa vzdialenosť medzi ľuďmi znižuje. Existuje viacero dôvodov pre túto odlišnosť od klasického vývoja v tíme na malom priestore. Do značnej miery to ovplyvňuje menší výber možností ako komunikovať so stúpajúcou vzdialenosťou. Pri distribuovaných tímoch ktoré oddeľuje značná geografická vzdialenosť k týmto dôvodom pribúdajú aj rôzne kultúrne a jazykové rozdiely.

Nedostatočná informovanosť. Pretože pri distribuovanom vývoji si ľudia udržiavajú relatívne malú spoločnú bázu informácií o stave projektu, majú tendenciu strácať prehľad o tom čo robia vzdialený členovia tímu zo dňa na deň. To vzniká aj ako dôsledok menšej komunikácie čo už bolo spomenuté. Táto nedostatočná informovanosť robí aj zložitejšie nadväzovanie kontaktov s ostatnými členmi tímu, pretože dostupnosť jednotlivých členov nie je neobmedzená.

Nekompatibilita. Ak spolu pracujú v tíme ľudia ktorý sú od seba vzdialený často jednotliví členovia používajú rôzne vývojové nástroje, rôzne procesy, praktiky a zvyky pri vývoji. Vzdialenosť medzi členmi im neumožňuje sa úplne zjednotiť vo všetkých požadovaných faktoroch. Tieto rozdiely navyše môžu byť navzájom nekompatibilné a nezlučiteľné.

V distribuovanom vývoji môže vznikáť mnoho ďalších problémov ako napríklad duplikovanie práce alebo nekompatibilita vytvorených častí projektu. Väčšina týchto problémov vzniká ako dôsledok slabej koordinácie a slabej vzájomnej informovanosti.

Vynaložením úsilia na vytvorenie lepšej koordinácie, zvýšením frekvencie a intenzity vzájomnej komunikácie je možné týmto problémom predísť, alebo ich obmedziť. K tomuto účelu bolo vytvorených množstvo nástrojov a podporných technológií, ktorých požadované vlastnosti sa môžu líšiť v závislosti od konkrétneho tímu alebo konkrétneho softvérového projektu.

Podporné prostredia a prostriedky

Softvérové inžinierstvo je už dlhší čas zamerané na vyvíjanie a nasadenie softvérových prostriedkov ktoré podporujú prácu v tíme a pomáhajú pri vývoji veľkých softvérových systémov. Pri vývoji týchto systémov sa stalo štandardom využívanie nástrojov na manažment verzií a sledovanie zmien. Tieto nástroje sa stále častejšie integrujú do podporných riadiacich prostriedkov a nástrojov na koordináciu v tíme a jedná sa o jedno a to isté prostredie alebo prostriedok. Základom pre distribuovaný vývoj softvéru sú práve tieto podporné prostriedky, ktorých používanie má dopad aj na výslednú kvalitu vyvíjaného produktu.

Podporné prostriedky používané pri distribuovanom vývoji sa často používajú aj pri spoločnom (lokálnom) vývoji na jednom mieste (napríklad vo firmách) najmä kvôli možnosti ich rozšírenia a ich neskoršieho zapojenia do distribuovaného vývoja. Taktiež aj pretože napriek tomu, že sú primárne určené na distribuovaný vývoj, zvyšujú efektivitu aj v lokálnych tímoch. Ich použitie je výhodné aj pretože centralizujú dáta pohybujúce sa v procesoch vývoja a tým umožňujú ich uchovanie a neskoršiu analýzu. Centralizovanie dát zjednodušuje prácu s nimi a ich zálohovanie ako projektového KNOW-HOW. Z tohto hľadiska je možné empiricky rozdeliť nástroje používané pri distribuovanom vývoji do dvoch skupín. Do prvej skupiny patria prostriedky využívané na komunikáciu a koordináciu pri vývoji projektu a do druhej skupiny patria projekty na uchovávanie informácií a projektovej histórie.

Prostredia a prostriedky na komunikáciu a koordináciu.

Ako som uviedol vyššie, jeden z hlavných problémov pri distribuovanom vývoji je koordinácia medzi členmi tímu. Tým vzniká nevedomosť členov tímu o tom čo práve ostatný robia, s akými problémami pracujú a aké je tempo ich produkcie. Práve preto sa robí výskum v tejto oblasti a vykonáva sa aj množstvo experimentov priamo v reálnych tímoch vo firmách. Do prostriedkov určených na koordináciu a riadenie sa pridávajú neformálne komunikačné prostriedky ako je chat alebo VoIP. Táto komunikácia je určená predovšetkým na diskusiu o projekte ale aj nepracovné využitie medzi členmi tímu. Niektoré projekty používajú na komunikáciu diskusné fórum alebo mailing listy.

Kolaboratívny prístup k riešeniu projektu je v dnešnej dobe populárny a je jeden z najpoužívanejších hlavne v distribuovanom prístupe. Výskumné projekty ktoré sa zaoberajú kolaboratívnym vývojom umožňujú používateľom spoločný prístup k zdrojovým kódom v tom istom čase. Napríklad projekt Palantir umožňuje vývojárom vizualizovať informácie o tom kto kedy mení nejaký artefakt v ich vlastnom prostredí.

Vyhodnocuje tieto zmeny a predpovedá radikálnosť ich dopadu na projekt. Iné projekty ponúkajú kolaboratívne príslušenstvo pre špecifickú fázu životného cyklu projektu. Prostriedky sú založené na nekonvenčnom prostredí vrátane kolaboratívneho návrhu funkcionality a požiadaviek na projekt. Tým je možné napríklad jednoducho informovať investorov o zmenách v požiadavkách. Ďalej existujú podporné prostriedky špeciálne pre agilné plánovanie a agilný vývoj softvéru, prostriedky pre sledovanie vykonávania práce na projekte a vizualizovanie jednotlivých procedúr špecifických pre daný projekt, a prostriedky na distribuovanú asynchrónnu inšpekciu veľkých projektov.

Prostredia a prostriedky na uchovávanie dát a projektovej histórie.

Pri klasickom softvérovom projekte vzniká pomerne bohatá projektová história a množstvo dát spojených s projektom ktoré treba uchovávať pre správu prípadného produktu ktorý vznikol na základe projektu. Taktiež treba uchovávať jednotlivé verzie zdrojových kódov, používateľských príručiek a ostatných dokumentov.

Mnoho vyvíjaných komerčných prostredí a prostriedkov na uchovávanie dát sú navrhované tak, aby ich bolo možné ich použiť na čo najväčšiu škálu projektov. V oblasti uchovávaní dát však majú vedúce postavenie OpenSource technológie ako napríklad SVN alebo CVS. Svojou kompatibilitou s vývojovými nástrojmi umožňujú ich široké použitie na celom svete. Nástroje podporujú manažment verzií a sledovanie zmien a v niektorých prípadoch aj mailing listy. Gutwin a spol. [1] poukázal na to ako OpenSource vývojári používajú tieto prostriedky na udržiavanie publikujúcich informácií pre širokú verejnosť a špecifických informácií napríklad pre expertov. Moderné podporné prostriedky na uchovávanie dát majú širokú funkcionality. Prostriedky podporujú napríklad automatické vyhľadávanie súvislostí medzi údajmi, jednotlivými artefaktmi v zdrojovom kóde a ich vizualizáciu. Dokážu analyzovať a vybrať začiatkový bod pre študovanie zdrojových kódov. Analyzujú použité technológie v projekte. Napríklad nástroj Mylyn počíta stupeň významnosti určitých artefaktov v projekte, používa obsah zameraný na úlohy a projektovú históriu, a obsahuje filter ktorým je možné identifikovať použiteľnosť artefaktov. Iné projekty sa zameriavajú na hľadanie súvislostí medzi ľuďmi namiesto artefaktov. Takýmto spôsobom je možné vizualizovať vzťahy medzi vývojármi odhadovať ich zúčastnenosť na projekte. Na takúto analýzu sa používajú metódy z oblasti analýz sociálnych sietí. Výsledky takýchto analýz pomáhajú používateľom identifikovať role a funkcie jednotlivých ľudí v projekte. Tieto informácie majú vyšší význam v vysoko roz distribuovaných projektoch, kde ľudia nemajú neustály prehľad o postaveniach ostatných členov tímu, ale aj v projektoch na ktorých sa zúčastňuje viac tímov a je potrebná komunikácia medzi viacerými tímami.

Napriek rozdeleniu týchto prostriedkov podľa použitia, si myslím, že často môže ísť o jeden a ten istý prostriedok alebo skupinu prostriedkov. Mnoho prostriedkov určených na komunikáciu a koordináciu ukladá priebeh ich použitia a neskôr tieto informácie môžu byť použité ako spätná väzba pri riadení projektu a určovaní slabých

komunikačných a koordinačných miest. Prepojenie prostriedkov sa robí aj pretože komunikácia medzi členmi tímu je zameraná na informácie týkajúce sa projektu a teda aj údajov, ktoré využívajú prostriedky určené na uchovávanie. V softvérovom inžinierstve je týchto dát viac ako v iných odvetviach aj pretože samotný subjekt vývoja a produkcie je uložený ako informácia.

Moderné prístupy ku koordinácii

Na priblíženie moderných prístupov ku koordinácii priblížim niektoré prístupy. Prvý prístup spočíva vo vytvorení *virtuálnych lokálnych tímov*. Spočíva vo vytvorení robustného virtuálneho prostredia pre vývojárov do, ktorého sa umiestnia len ľudia, ktorý spolu potrebujú koordináciu a vzájomnú komunikáciu. Robustné prostredie umožňuje komunikáciu prirodzenou formou ako je napríklad reč, alebo obraz. Preto je dôležité dobre identifikovať kto potrebuje s kým komunikovať. Tento prístup je pre vývojárov zatiaľ ojedinelý avšak dosť preferovaný. Druhý prístup počas projektu *kontinuálne využíva projektovú históriu*. Ak máme zozbierané informácie o minulých projektoch a znalosť ako identifikovať jednotlivých postupoch a ľuďoch, vieme odhadnúť ako by mala koordinácia medzi členmi tímu fungovať. Na základe minulých projektov vieme vhodne vybrať technológie najlepšie pre nový projekt. Podobný prístup je v *rozširovaní projektovej histórie*. Projektová história je založená na údajoch zozbieraných v predošliých projektoch. Mnoho riešení a problémov spojených s koordináciou je špecifických pre daný projekt, avšak použitá metóda môže byť použiteľná aj v neskorších projektoch. Preto je potrebné aby už v čase realizácie prvého projektu sa identifikovali údaje a postupy, ktoré budú potenciálne použiteľné v budúcich projektoch. Malou zmenou predošlého prístupu je prístup vo forme vytvárania *projektovej histórie a využívania infraštruktúry kolaboratívnych nástrojov*. Pri zbieraní poznatkov do projektovej histórie o tom ako riešiť problémy koordinácie sa zapoja aj priamo nezainteresované strany. Avšak ľudia od ktorých sa zbierajú informácie by mali byť ľudia zainteresovaný do konkrétneho projektu.

Mnoho moderných prístupov ku koordinácii je vo výskumnej fáze a sú použité na konkrétne tímy čo môže viesť k zdanlivej úspešnosti. Úspešnosť je však závislá aj od konkrétnych ľudí zapojených do projektu. Dôkladné prešetrenie procesov a koordinácie často nie je postačujúci ukazovateľ, ktorý by spoľahlivo hodnotil jednotlivé prístupy. Jediný spoľahlivo osvedčený ukazovateľ, ktorý zohľadňuje všetky faktory vplývajúce na koordináciu je opakovane overená úspešnosť pri použití jednotlivých prístupov v praxi.

Záver

Kľúčom k úspešnému distribuovanému vývoju je úspešné vyriešenie problémov spojených s koordináciou a vzdialenosťou. Rovnako je potrebné dobre manažovať všetky činnosti v tíme závisiace od koordinácie. Na to je potrebné porozumieť metódam na vyriešenie týchto problémov a vybrať tú správnu.

Existuje množstvo individuálnych riešení, ako je použitie podporných prostriedkov, praktík a metód. Podporné prostriedky sú hlavnou súčasťou distribuovaného vývoja softvéru. Ich použitie rieši viacero problémov v koordinácii a komunikácii. Táto esej mala za cieľ priblížiť celú túto oblasť a oboznámiť čitateľa s trendom používaným v dnešnej dobe. Ak realizujeme distribuovaný projekt do budúcnosti musíme starostlivo navrhnuť architektúru koordinácie na strane každého vývojára.

Oblasť riadenia a koordinácie v distribuovaných tímoch je málo rozvinutá a málo preskúmaná. Existujú metodiky ako riešiť určité typy problémov avšak vzniká potreba nových teórií a prístupov, ktoré dokážu zvýšiť efektívnosť koordinácie v distribuovaných tímoch. Je potrebné aby tieto prístupy išli s aktuálnym trendom a nestali sa zastaralými.

Použitá literatúra

1. Gutwin C., Penner R, Schneider K: *Group awareness in distributed software development*, in Proceedings of the 2004 ACM conference on Computer supported cooperative work. 2004 ACM Press:Chicago, Illinois, USA.
2. James D. Herbsleb: *Global Software Engineering: The Future of Socio-technical Coordination*, FOSE'07 (2007) Future of Software Engineering
3. Rebeca E.Grinter: *Using a Configuration Management Tool to Coordinate Software Development*, COOS 95 Milpitas CA USA, ACM Press

Annotation

Support technologies for control and coordination distributed teams.

Distributed projects and project teams are becoming the standard for large software systems development. Coordination and control of project teams is a very complex task. The complexity of such task increases with the complexity of large software systems and distribution of project teams. Coordination among members in a distributed team is often a critical part in solving software projects, and brings more specific problems. This is why we use tools to support technologies especially in distributed software projects. These can help us control and improve coordination and make effective process of developing software in such ambient. This essay deals with problems in coordination that are in distributed teams and reviews their summary. I conclude their reduction with support technologies, which are currently used for control and coordination of distributed teams.