

Monitorovanie softvérového projektu

Princípy a metriky

JÁN KRAUSKO

*Slovenská technická univerzita
Fakulta informatiky a informačných technológií
Ilkovičova 3, 842 16 Bratislava*

krausko02@student.fiit.stuba.sk

Abstrakt. Snaha vyvíjať softvér rýchlo a kvalitne je zložitá úloha pre každý softvérový tím. Priebežné monitorovanie stavu a postup softvérového projektu nielenže pomáha manažérom kontrolovať vývoj projektu, ale tiež môže upozorniť na vznikajúce komplikácie a problémy a tak im predchádzať. Cieľom tejto eseje je predstaviť princípy monitorovania softvérového projektu, jeho vplyv na riadenie a ďalšie plánovanie a poukázať na rozdiely v prístupe k monitorovaniu v závislosti od veľkosti projektu. Tiež sa pokúsim priblížiť niektoré techniky a zamyslieť sa nad ich použiteľnosťou.

Čo je monitorovanie softvérového projektu?

Pojem monitorovanie, alebo sledovanie, je hodnotenie priebehu projektu počas jeho realizácie. Ale aby sme sa nemýlili, nejde len o odpoveď na otázku: „Robíme dobrú prácu?“ Zmysel monitorovania v softvérovom projekte je principiálne tiež snaha o skvalitnenie riadenia. Nespočíva teda len v obyčajnom sledovaní postupu projektu či len v jeho kontrole, ale naopak, jeho význam rozširuje. Neobmedzuje sa totiž na spätnú kontrolu procesu, ktorý už bol vykonaný, či cieľov, ktoré boli alebo neboli splnené, ale na preverovanie prípravy procesov či koncipovania cieľov. Takýto prístup predurčuje aj orientáciu monitorovania nie do minulosti, ale do budúcnosti. Aktívny je už na začiatku celého procesu prípravy riadenia, všetkých relevantných aktivít a jeho výsledky vedú k zefektívneniu ďalších plánov a riadenia projektu.

Manažment softvérového projektu či už väčšieho alebo menšieho rozsahu, je vždy netriviálny problém. Je v záujme každého projektového manažéra a samozrejme aj v záujme úspešného ukončenia projektu, aby bol od začiatku do konca pod kontrolou. Cieľom tohto dokumentu je priblížiť základné princípy a význam monitorovania softvérového projektu, predstaviť niektoré používané techniky a zamyslieť sa nad ich vplyvom na ďalšie riadenie a plánovanie softvérového projektu.

Monitorovanie a veľkosť projektu

Ako som už v úvode uviedol, monitorovanie a priebežná kontrola ovplyvňuje viacero projektových činností ako sú [1]:

- *revízia vykonanej práce* - väčšinou sa týka celého tímu a je užitočná na pravidelné zisťovanie stavu projektu a jeho postupu.
- *revízia ďalšieho napredovania* - vykonáva sa najmä pri stredných a väčších projektoch (mesačne alebo štvrťročne), kde sa zainteresované strany informujú o stave projektu a jeho napredovaní.
- *revízia projektového plánu* - vykonáva sa ak sa monitorovaním identifikuje potreba významnej zmeny v niektorej z častí projektu (harmonogram, rozpočet projektu,...). Najčastejšie sa takáto potreba ukáže po ukončení fázy životného cyklu projektu.
- *manažment zmien* - rieši zmeny, ktoré v priebehu projektu nastanú a sú identifikované sledovaním projektu, alebo môžu byť iniciované zákazníkom (zmeny požiadaviek).
- *a hodnotenie výsledného produktu* - všetci do projektu zainteresovaní musia rozumieť kontextu nasadenia produktu aby nakoniec nedošlo k odlišnostiam s prvotným plánom.

Sila závislosti medzi monitorovaním a jednotlivými činnosťami a miera samotnej potreby monitorovania závisí od potrieb konkrétneho projektu. Tabuľka 1. ponúka pohľad na rozdiely v jednotlivých činnostiach podľa veľkosti projektu.

Činnosti	Malý projekt	Stredný projekt	Veľký projekt
Priebežné monitorovanie postupu	Ako základ pre monitorovanie a kontrolu stačí použiť súhrn vykonaných činností (project summary)	Projekt manažér použije ako základ pre monitorovanie projektový plán a každý člen tímu, jemu alebo tím lídrovi, týždenne podáva správu o svojom postupe	
Revízia prác	Mailovou komunikáciou alebo na bežnom stretnutí	Na týždennom koordinačnom stretnutí, kde sa preberie prehľad hotových prác, miera rizík a aktuálne otázky a problémy, ktoré treba riešiť	Na pravidelných spoločných stretnutiach jednotlivých menších sub-tímov, kde sa preberie doterajší postup projektu, vzniká o tom súhrnné správa
Revízia smeru postupu prác a cieľov	Mailovou komunikáciou alebo na bežnom stretnutí	Na mesačnom stretnutí s senior manažérom a zákazníkom	
Manažment zmien	Týka sa jedného alebo dvoch ľudí	Týka sa zástupcu projektu, zástupcov zákazníka a zástupcov manažmentu	
Revízia plánu	Po dohode manažmentu	Významné zmeny potrebujú prerokovať a odsúhlasiť všetci zainteresovaní, ktorí prijímali originálny plán.	
Hodnotenie výsledku práce	Na bežnom stretnutí, dvomi alebo tromi ľuďmi	Technickou kontrolou alebo inšpekciou	Technickou kontrolou a inšpekciou, zameranie na kvalitu

Tab. 1. Rozdiely v činnostiach v závislosti od veľkosti projektu podľa [1].

Sledovanie postupu softvérového projektu a metriky

Aby sme vedeli odpovedať na otázku postupu projektu jednoducho porovnáme súčasný stav so zaznamenaným stavom z nedávnej minulosti. Napríklad, včera mala budova jedno poschodie, dnes už dve.

Pri softvérovom projekte sa ale naskytajú viaceré otázky:

- Aké charakteristiky projektu reprezentujúce jeho stav zaznamenávať?
- Ako vyhodnocovať tieto charakteristiky aby sme mohli presne určiť stav projektu?

Jednoznačná odpoveď asi neexistuje. Avšak dodržaním niekoľkých základných doporučení [2] si môžeme monitorovanie projektu uľahčiť.

- *Projektový plán* – je asi základ každého projektu. Obsahuje časové i finančné odhady a umožňuje v jednotlivých svojich bodoch určiť odchýlky aktuálneho stavu projektu a prijať prípadné opatrenia.
- *Začiatok a koniec každej činnosti* – jednotlivé činnosti projektu delím na menšie kým viem určiť ich začiatok a koniec, to umožní identifikovať a sledovať kritické miesta projektu a jednotlivé väzby medzi činnosťami.
- *Zdroje pre každú činnosť* – ak sú v projekte definované zdroje pre jednotlivé činnosti naskytá sa možnosť ich porovnávania so skutočne spotrebovanými zdrojmi. Do určitej miery to umožní predpovedať ďalší postup projektu.
- *Výsledky činnosti* – Výsledkom každej etapy v životnom cykle softvéru sú výstupy, ktorých sledovanie ponúkajú softvérové prostriedky manažment kvality¹ a meranie v softvérovom projekte [3].

Dobre, toto všetko už máme, ale ako exaktne určiť stav projektu? Čo vlastne monitorovať? V názve tejto kapitoly som použil aj ďalšie záhadné slovo: *metrika*. Áno, metrika a meranie v softvérovom projekte existuje a je veľmi dobrý nástroj pre orientáciu a určenie stavu v ktorom sa projekt nachádza. Merať, porovnávať a vyhodnocovať môžeme naozaj veľa vecí [4] [4], napríklad: počet riadkov kódu, počet opravených chýb, počet hodín strávených pracovníkom v kancelárii, počet prístupov do CVS², počet špecifikácií či množstvo elektronickej pošty. Všetky tieto metriky sa v praxi často používajú, pretože je jednoduché ich zmerať. Zároveň si treba uvedomiť, že sledujeme len vykonanú prácu a výpovedná hodnota týchto metrik môže byť skreslená najmä psychologickým efektom, ktorý má na ľudí samotné meranie. Začne sa pracovať na dosiahnutí lepších výsledkov v meraní a nie na dosiahnutí určeného cieľa. Napríklad počet riadkov kódu nehovorí nič o jeho efektívite, ale o jeho „riedkosti“. Odpracovaný čas zamestnanca tak isto nemusí odzrkadľovať reálny čas naozaj strávený prácou na projekte a podobne.

Postup softvérového projektu, a teda aj jednotlivé metriky je taktiež dôležité vhodne zobrazovať. Jednak pre potreby riadenia samotného tímu, ale aj pre prezentáciu zákazníčkovi. Niektoré zo spomínaných metrik sa pokúsim priblížiť viac.

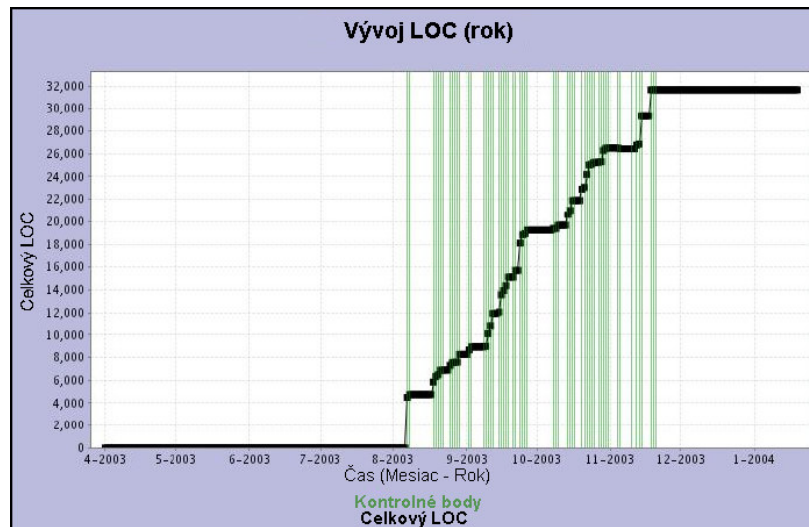
¹ Viac informácií nájdete napríklad na stránkach Software Quality Assurance Forums, <http://www.qaforums.com/>.

² CVS - Concurrent Versions System, je systém na správu a udržiavanie histórie súborov, <http://www.nongnu.org/cvs/>.

Počítanie riadkov kódu

Počet riadkov kódu (LOC – lines of code) je vďaka svojej jednoduchosti obľúbenou metrikou a to aj napriek svojim nevýhodám. Používa sa hlavne ako ukazovateľ produktivity práce a postupu projektu, ale môže tiež naznačiť koľko času ešte potrebujeme na dokončenie projektu [6]. Výrazné rozdiely zistené sledovaním LOC a jeho porovnaním s hodnotami predošlého obdobia môže manažmentu naznačiť problém a včas iniciovať príslušné protiopatrenie. Takéto monitorovanie postupu projektu nesie so sebou aj komplikácie. Či už ide o spomínaný efekt, kedy sa ľudia snažia o dobré výsledky pri meraní a cieľ projektu ide bokom, ale aj istú nespravodlivosť tejto metódy. Občas sa programátorovi, ktorý optimalizuje svoj kód stáva, že za pracovný deň viac riadkov zmaže ako pridal. Kód je optimalizovaný, práca úspešne dokončená, programátor sa teší, ale podľa ukazovateľa LOC vlastne nepracoval. Preto je samozrejmé, že výkyvy treba zaznamenávať aj s popisom odoviedajúcej práce na kóde a pri vyhodnocovaní údajov tento fakt zohľadňovať.

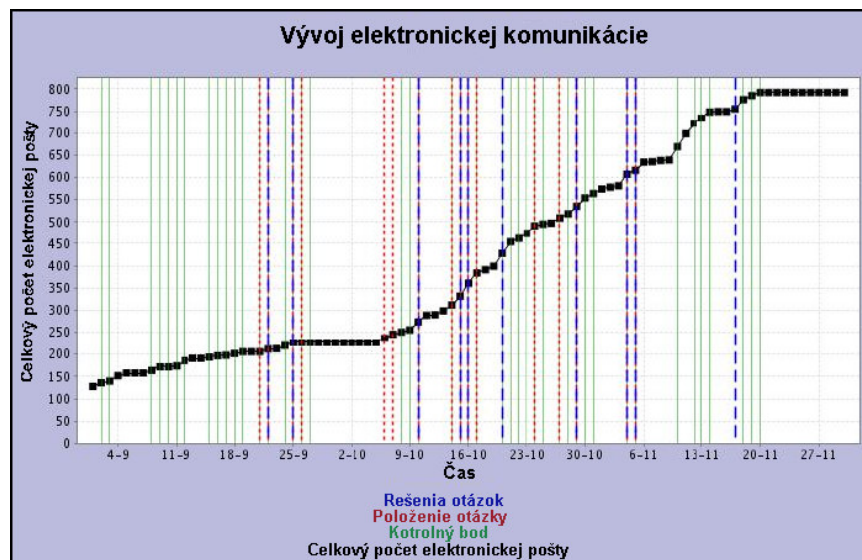
Používanie LOC ako jediného ukazovateľa nebude postačovať a určite to nie je tá najlepšia cesta k monitorovaniu softvérového projektu. Avšak v množine viacerých ukazovateľov si určite svoje miesto nájde. LOC nám indikuje pokles alebo rast produktivity práce, ale ostáva na manažérovi určiť a vyhodnotiť príčiny týchto javov a prijať stanovisko. Obr. 1. ukazuje príklad znázornenia ročného vývoja LOC pomocou grafu podľa [4].



Obr. 1. Príklad vývoja metriky LOC počas roka v softvérovom projekte podľa [4].

Elektronická komunikácia medzi vývojármi

Obr. 2. ilustruje zobrazenie ďalšej metriky, elektronickej komunikácie medzi vývojármi, počas softvérového projektu. Je na ňom vidieť celkové množstvo elektronickej pošty (čierna čiara), ale aj časy od polozenia otázky (červená bodkovaná čiara), ktorá sa behom riešenia projektu vyskytla, až po jej vyriešenie (modrá prerušovaná čiara).

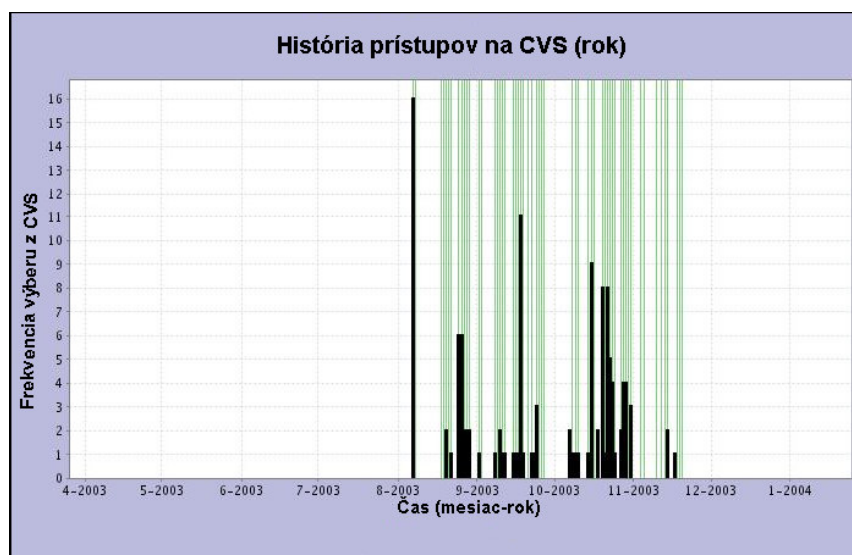


Obr. 2. Príklad histórie elektronickej komunikácie v softvérovom projekte podľa [4].

Z tohto grafu môžeme vidieť komunikáciu vývojárov o problémoch a objavených chybách projektu. Ak by chyba bola objavená ale vývojári by o nej nekomunikovali, môže to naznačovať komunikačný, prípadne iný problém medzi nimi. Komunikačné problémy často spôsobujú znižovanie produktivity práce a spoľahlivosť výsledného softvéru [7].

Počet prístupov do CVS

Obr.3. ilustruje frekvenciu práce s CVS od času v softvérovom projekte. Keď niekto súbor v CVS aktualizuje, ostatní vývojári ho zvyčajne z CVS vyberú a aktualizujú si ten svoj na lokálnom disku. Málo výberov z CVS signalizuje, že súbor, ktorý bol do CVS pridaný, resp. bol aktualizovaný nie je príliš dôležitý, alebo ostatní vývojári o tom nevedeli. To môže informovať manažéra o tom či sa vývojári medzi sebou o zmenách informujú alebo je v tíme problém s komunikáciou.



Obr. 3. Hustota prístupov na CVS počas roku v softvérovom projekte. Príklad podľa [4]. Čierne čiary znamenajú výber z CVS (check-out) a zelené značia čas kedy bol v CVS nejaký súbor aktualizovaný (check-in).

Záver

Pre projektového manažéra je nepopierateľne veľmi dôležité aby poznal stav a postup projektu. V softvérových projektoch je to ale ťažšie, pretože výsledky práce nie sú hmatateľné. Mojim cieľom bolo predstaviť problematiku merania a monitorovania v softvérovom projekte a naznačiť okrem princípov a metrík aj vplyv na riadenie a ďalšie plánovanie projektu.

Použitá literatúra

1. Department of Information Resources. *Process for Project Monitoring and Control* [online] March, 2000, V1.0 Initial Release. Dostupné z <<http://www.dir.state.tx.us/eod/qa/monitor/index.htm>>.
2. BIELIKOVÁ, M.: *Softvérové inžinierstvo: Princípy a manažment*. Bratislava: STU, 2000.
3. MAH C. M., PUTMAN H. L. Software by the numbers: An aerial view of the software metrics landscape. *AMERICAN PROGRAMMER* [online] Vol. 10, No. 11. Dostupné z <<http://www.qsm.com/aerialview.html>>.
4. OHIRA, Masao et al. Empirical Project Monitor: Automatic Data Collection and Analysis toward Software Process Improvement. [online] Február, 2004. Dostupné z <<http://www.empirical.jp/English/research/archive/8/8.pdf>>.
5. LINDEN Joop. Monitoring Progress in Software Development. [online] Júl, 2003. The Haagse Hogeschool. Dostupné z <<http://www.empirical.jp/English/research/archive/8/8.pdf>>.
6. ARMOUR, P.G. The business of software: Beware of Counting LOC. *Communications of the ACM*, [online] 2004, Vol. 47, No. 3 21-24. Dostupné z <<http://portal.acm.org/citation.cfm?id=971635>>.
7. DUTOIT, H. A, BRUEGGE, B. Communication Metrics for Software Development. *IEEE Transactions on Software Engineering* [online] 1998, Vol.24, No.8, pp.615–628. Dostupné z <<http://portal.acm.org/citation.cfm?id=253286>>

Annotation

Monitoring progress of software project : Principles and metrics.

The effort to develop this software quickly and usefully is difficult task for every team. Continuous monitoring of status and progress of software project helps management to control the growth of the project as well as can remind the different types of problems and helps to avoid them. Purpose of this essay is to introduce the principles of software project monitoring and it's influence on operation and further planning and show the differences in approach of monitoring depended to upon size of the project. I will try to discuss closer some of the techniques and think about their usability.