

ODHAD, KTORÝ POČÍTA SO VŠETKÝM?

Ak chceme nepriateľa poraziť, musíme ho najprv dobre spoznať.

Peter Krátky

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

Abstrakt. Táto esej pojednáva o procese vytvárania presných časových odhadov pri plánovaní projektu. Na začiatku sa venujem tomu, prečo je presný odhad pri plánovaní dôležitý a aké problémy najčastejšie spôsobujú prekročenie odhadov. Ďalej uvádzam najrozšírenejšiu techniku detailného priblíženia projektu ako základ pre vytváranie odhadov. Oboznamujem čitateľa so základnými skupinami metód pre vytváranie odhadov a porovnávam ich najmä z hľadiska toho, ako sa v nich prejavuje prítomnosť, resp. neprítomnosť ľudského faktoru. V súvislosti s touto časťou upozorňujem na dôležitosť zodpovedného a čestného prístupu pri vytváraní odhadov. Na záver odpovedám na otázku, ktorú metódu je vhodné zvoliť v plánovacom procese vzhľadom na charakter a prostredie projektu.

Kľúčové slová: plánovanie projektu, časové odhady, problémy pri odhadovaní, rozpis práce, metódy odhadovania

Termíny sú na to, aby sa (ne)porušovali

Túto jar nám začali zatepľovať bytovku. Pôvodný termín ukončenia prác bol stanovený na tridsiateho júna. Samozrejme, ako to chodí vo väčšine projektov, aj v tomto sa termín dokončenia niekoľkokrát presúval a posledné zvyšky lešenia odvážali až koncom septembra. Lepší termín by snáď predpovedala aj veštica.

Spomínaná firma chcela lákať zákazníkov na rýchlu prácu, doplatila však na to, že si stanovila nereálny odhad a stratila na nájomnom za lešenie a pokute zákazníkovi za veľké meškanie. Môže niečo podobné nastať aj v softvérovom priemysle? Aj softvérová firma sa

musí trápiť s problémami, ktoré výrazne ovplyvňujú jej časový plán. Dokonca si dovoľím tvrdiť, že faktory, s ktorými treba rátať pri tvorbe plánu v softvérových projektoch sú omnoho sofistikovanejšie. V hre sú veľké peniaze a spokojnosť zákazníka, a preto by plán mal byť dokonale premyslený.

Aký odhad, taký plán

Čo vlastne spôsobuje porušovanie plánovania v softvérovom projekte? Skôr, než sa pokúsime zodpovedať túto otázku, priblížme si, čo je to plán projektu. Plán v skratke pozostáva z úloh, ktoré sa budú vykonávať, časových odhadov trvania týchto úloh a kedy budú hotové. Je teda logické, že ak zle odhadneme už čas trvania úlohy, ovplyvní to termíny úloh, a taktiež i celý plán. Rôzne prieskumy ukazujú, že odhady zlyhávajú na tom, že pri plánovaní sa neráta s nečakanými udalosťami, prehliadajú sa úlohy, ktoré na prvý pohľad nie je vidno, zle sa vymedzia zdroje, často sa menia požiadavky klientov, či požiadavky klientov nie sú jasne komunikované [4].

Myslím si, že na to, aby sme sa stretli s týmito problémami, netreba pracovať ani vo firme. Každý z nás sa s nimi stretol už len pri školskom projekte. Určite poznáme to, keď si stanovíme dve hodinky na jednoduchú domácu úlohu, ale v tom nás zmrazí neposlušný vývojový nástroj alebo prostredie a jeho inštaláciou nakoniec strávime pol dňa. Ďalej sa stáva, že napohľad jednoduchá úloha skrýva v sebe problém, ktorý sme nevideli na prvý pohľad a jeho riešenie trvá dlhšie ako odhad celej úlohy. Asi tým najväčším problémom, ktorý by som spomedzi všetkých zdôraznil, sú podľa mňa zle odkomunikované požiadavky. Keď si myslíme, že rozumieme, čo zadávateľ úlohy od nás chcel, ale v skutočnosti sme zle pochopili jeho myšlienkový pochod, sme nútení prerábať úlohu odznova. Takto sa odhad hneď niekoľko násobne prekročí.

Vyhovárať sa na tieto problémy však nie je na mieste a podľa mňa by s nimi mal dobrý odhad rátať. Ako sa teda v praxi odhady vytvárajú, tak aby v nich bolo obsiahnuté všetko, čo môže spôsobovať ich nenaplnenie?

Nazeranie až do útrov projektu

Aby sme vedeli, čo náš môže stretnúť, je potrebné venovať dostatočnú pozornosť detailom toho, čo nás čaká. Hovorí sa, že ak chceme poraziť nepriateľa, musíme ho dokonale poznať. Preto aj dobrý odhad vyžaduje podrobnú analýzu toho, čo treba vykonať a aké zdroje sú k dispozícii. Toto je podľa mňa základ, na ktorom buď odhad stojí alebo padá.

Všeobecne používaná technika na podrobnú analýzu projektu nielen v softvérovom inžinierstve sa volá rozklad rozpis práce (angl. Work Breakdown Structure). Pomocou nej sa vytvára hierarchický rozklad produktu, ktorý odkrýva detailné súčiastky produktu alebo sa vytvára hierarchický rozklad činností, ktorý podrobne vymenováva úlohy na projekte. Ide o dva prístupy, ktoré by sa nemali prelínať, ale by sa malo postupovať podľa zvolenej alternatívy. Zdenko Staníček, špecialista na projektové riadenie z Masarykovej Univerzity v Brne, odporúča voľbu prvej alternatívy [5]. Ak sa jedná o softvérový projekt, ktorého cieľom je realizácia nejakého softvérového systému, tak si myslím, že je prirodzenejšia voľba vytvoriť rozklad produktu na súčiastky. Prispieva k tomu fakt, že softvérový systém sa dá jednoducho predstaviť ako skladačka viacerých komponentov

a často sa tak aj znázorňuje. Na druhej strane projekt charakteru údržby alebo opravy chýb v systéme by bolo vhodné rozmeniť na drobné cez činnosti.

Niet pochýb o tom, že takto detailne rozčlenený projekt nám uľahčí minimálne odhadovanie tým, že menším častiam je možné ľahšie odhadnúť trvanie ako celku. Okrem toho, klesá riziko, že sme niečo zabudli započítať do odhadu, miesta, ktoré môžu byť problematické sa zviditeľnia a lepšie sa ukáže aj množstvo potrebných zdrojov.

Čo máme k dispozícii na odhadovanie?

Je odhadovaná dĺžka trvania úlohy a následne termín dokončenia len čisto pocit v kostiach autora odhadu? Alebo existuje postup, ktorý by umožnil tieto odhady vystavať na nejakých objektívnejších základoch? Štandardne sa vždy postupuje podľa nejakých metód, ktoré vhodným spôsobom odôvodnia odhad. Existuje veľká skupina metód, ktoré využívajú čisto objektívne nástroje ako matematiku a strojové učenie na priamy výpočet hodnôt. Ide o odhadovanie založené na formálnych modeloch. Samozrejme aj spomínané „pocity v kostiach“ majú stále svoje nezastupiteľné miesto a v kombinácii s matematickou štatistikou sa pretavili do ďalšej veľkej skupiny odhadov založených na posúdeniach expertov. Toto delenie navrhli dvaja priekopníci v odhadovaní nákladov projektu – profesor Magne Jørgensen a profesor Barry Boehm [3].

Pre priblíženie, najznámejšou formálnou metódou je model COCOMO, ktorý navrhol Barry Boehm už v osemdesiatych rokoch a v obmenách sa používa dodnes. Vznikol podrobnou analýzou viacerých projektov a na základe nej boli určené kľúčové atribúty projektu, ktoré majú najväčší vplyv na vytváranie odhadu. Kvantifikované atribúty spolu s odhadom veľkosti projektu potom vstupujú ako parametre do modelu a algoritmicky sa z nich odvádza výsledok [6].

Zástupcom druhej skupiny je metóda Wideband Delphi. Je vytvorený rozmanitý tím odborníkov, ktorého členovia samostatne vytvárajú odhady dekomponovaným častiam projektu. Tie sú na spoločnom stretnutí prezentované a diskutované, pokiaľ sa nenájde konsenzus [6].

Ľudský faktor v odhadoch

V softvérovom inžinierstve vznikajú veľké polemiky o tom, ktorá skupina odhadov dáva lepšie výsledky, ktorá je flexibilnejšia, či ktorá je menej nákladná. Pokúsme sa na chvíľu detailnejšie nazrieť do tejto problematiky. Hovorí sa, že ľudské skúsenosti sú jedinečné a nedajú sa nahradiť len tak ľahko nejakým systémom. Magne Jørgensen na margo tohto pripája, že formálne metódy nedokážu zahrnúť do odhadov rôzne špecifické situácie, ktoré vo firme nastávajú, ako napríklad konflikty lídrov, atď [3]. Súhlasím s tým, že v takýchto situáciách je práve ľudský faktor nenahraditeľný. Avšak okrem tohto prípadu by som si dovoľil povedať, že formálne modely sa dokážu vyrovnáť tým expertným. Ako povedal náš profesor matematiky, štatistika nepustí. Som presvedčený o tom, že matematická pravdepodobnosť a štatistika je skutočne mocný nástroj. Ak máme k dispozícii veľkú množinu údajov o predchádzajúcich a podobných projektoch, tak si myslím, že údaje, ktoré sa z nich dajú získať sú také kvalitné, že odhady na ich základe budú naozaj pomerne presné. Okrem toho, nie je to len štatistika, ktorá sa v modeloch

4 Peter Krátky

používa. Preniklo do nich aj strojové učenie, vďaka ktorému model stále zlepšuje svoje reakcie na známe situácie.

Ako problém expertných metód uvádza Magne Jørgensen ten, že odhady odborníkov môžu byť skreslené psychologickými aspektmi [3]. Podľa mňa je v podvedomí niektorých ľudí zakorenené, že ak je vo vyhliadke veľa problémov, tak pred nimi strkajú hlavu do piesku. Možno si nechcú pripustiť to, že by mohli v ich projekte takéto problémy nastať a ich hrdosť im nedovolí s nimi rátať. Alebo majú obavy šéfovi priniesť nie veľmi optimistický odhad, preto sa rôzne problémy zamlčia a dúfa sa, že nenastanú. Toto je, myslím si, žiaľ bežná prax nielen pri odhadovaní v softvérovom projekte. Veď aj zateplňovacia firma spomenutá na začiatku by zákazku nedostala, keby povedala svoj reálny odhad.

Tu sa dostávame k bodu, kedy na jednej strane je ľudský faktor stimulátorom presného odhadu a na druhej strane jeho inhibítorom. Je samozrejmé, že v plánovacom procese sú úprimné odhady absolútne nevyhnutné. A to aj napriek tomu, že prinášajú možno až bolestivú pravdu. Nečestné odhady môžu vyústiť do toho, že vedenie schváli projekt, ktorý by inak zamietlo a stratí na tom nemalé financie.

Čo sa týka nákladov, tak sa dá povedať, že formálne modely vyžadujú menšie ľudské úsilie. Avšak ani tie by bez ľudských zdrojov nebolo možné prevádzkovať, keďže niekto musí rozumne určiť všetky atribúty, ktoré projekt charakterizujú a do modelu vstupujú a zbierať údaje o ukončených projektoch.

Iná firma, iný mrav

Bolo by trúfalé pokúšať sa jednoznačne odpovedať na otázku, ktorá skupina metód je lepšia, pretože každá má svoje špecifiká a vo firmách s rôznym charakterom sa hodí iná. Jedna z vedúcich osobností v softvérovom inžinierstve, Dr. Dennis J. Frailey, odporúča použitie formálnych metód ako COCOMO vo veľkých organizáciách a odhadovanie na základe vyjadrenia expertov v malých firmách [1]. Takéto rozdelenie je podľa mňa logické, vzhľadom na to, že veľké firmy majú uskutočnených veľa projektov, preto majú aj veľké množstvo údajov, z ktorých sa dá ťažiť. Naopak v malých firmách je proces odhadovania odkázaný na skúsenosti ľudí z praxe, ktorí sa už s podobnými projektmi stretli v minulosti. Napriek tomu však zapojením viacerých ľudí z firmy do odhadovacieho procesu dosiahneme ich väčšiu informovanosť o projekte [1]. Majú tak väčší prehľad o súvislostiach a možno prinesú aj nové dobré nápady.

V každom prípade, najvýhodnejšie by bolo spojiť kladné vlastnosti oboch skupín v nejakej rozumnej kombinácii. Toto riešenie by bolo ideálne vzhľadom na presnosť odhadovania a Magne Jørgensen ho dokonca aj odporúča [2]. Avšak po finančnej stránke by bolo určite nákladné a menšie firmy by si ho nemohli dovoliť.

Zhrnutie na záver

Základom dobrého plánu projektu je dobrý odhad. Na to, aby sme mohli odhad vytvárať, je potrebné dokonale poznať projekt. To nám umožní pri vytváraní odhadu rátať s problematickými situáciami. Je treba mať na pamäti, že problémy nemožno prehliadať a neúprimné odhady spôsobia len škodu. Na samotnú realizáciu odhadovania sú vyvinuté

metódy, ktoré buď využívajú algoritmy na ich výpočet alebo sú založené na posúdeniach odborníkov. Pri výbere je potrebné zvážiť pozitíva aj negatíva jednotlivých metód. Nedá sa tvrdiť že jedna je určite lepšia ako druhá, dá sa však zvoliť najlepšia alternatíva vzhľadom na dostupné prostriedky a charakter projektu, pričom ideálne je spojiť pozitíva oboch metód.

Použitá literatúra

1. Frailey, D.J.: Experience Teaching Barry Boehm's Techniques in Industrial and Academic Settings. In: *Software Engineering Education and Training* (2006), 219-225.
2. Jørgensen, M.: Practical guidelines for expert-judgment-based software effort estimation. *IEEE Software*, Vol. 22, No. 3 (2005), 57- 63.
3. Jørgensen, M., Boehm, B., Rifkin, S.: Software Development Effort Estimation: Formal Models or Expert Judgment? *IEEE Software*, Vol. 26, No. 2 (2009), 14-19.
4. Jørgensen, M., Moløkken-Østvold, K.: Reasons for Software Effort Estimation Error: Impact of Respondent Role, Information Collection Approach, and Data Analysis Method. *IEEE Transactions on Software Engineering*, Vol. 30, No. 12 (2004), 993-1007.
5. Staníček, Z., Hajkr, J.: Řízení projektů zavádění IS do organizací TUTORIAL. In *DATAKON 2005*, T. Hruška (Ed.), Brno (2005), 1-25.
6. Stellman, A., Greene, J.: *Applied Software Project Management*. O'Reilly Media, 2005. 338 s. ISBN 978-059600948. 34-51.

Annotation

Estimation that counts on everything?

This essay discusses the process of creating accurate time estimations of the project plan. In the beginning I concentrate on the necessity of the accurate time estimations and issues resulting in the overrun of the estimation. Next I introduce wide-spread technique to get a detailed look of the project as a basis of the time estimations creation. I acquaint the reader with the cardinal groups of the methods to create time estimations and I compare them particularly in the point of view how they are influenced by presence resp. absence of the human factor. In connection with this I underline the importance of liability and fairness in creating estimations. Finally I answer the question of which method is the best choice in the time estimation process with respect to character and environment of the project.