

OPEN-SOURCE VS. KOMERČNÉ NÁSTROJE PRE RIADENIE SOFTVÉROVÝCH PROJEKTOV

Existuje operačný systém, ktorý je matematicky korektný. Čo to však dokazuje o našom svete?

Milan Freml

Slovenská technická univerzita
Fakulta informatiky a informačných technológií
Ilkovičova 3, 842 16 Bratislava
xfreml [zavináč] stuba [.] sk

Abstrakt. Moja odborná esej je ohraničená priestorom tém podporných prostriedkov pre riadenie softvérového projektu, so zameraním na využitie ľudských zdrojov a sledovanie úloh. Obsahuje úvod do problematiky riadenia softvérového projektu. Snaží sa zistiť, aké možnosti pre samotnú podporu riadenia a plánovania, prípadne sledovania úloh existujú. Je ich niekoľko, ako napríklad Ganttova alebo sieťová schéma. Skúma, ktoré vlastnosti sú pri podporných prostriedkoch softvérovými inžiniermi vyžadované, respektíve želateľné. V esejí sú priblížené niektoré komerčné a open-source nástroje pre podporu riadenia softvérového projektu, snažím sa o ich objektívne porovnanie vzhľadom na skupinu niektorých funkcií. Esaj ďalej prináša autorov názor na otázku potrebnosti komplexných nástrojov pre plánovanie pri menších projektoch.

Kľúčové slová: softvérový projekt, plánovanie, podporný nástroj, riadenie, sledovanie úloh, ľudské zdroje, open-source

Úvod do problematiky

Softvérové inžinierstvo je ako celá oblasť informatiky pomerne mladou disciplínou. S rapidným rozvojom informatizácie spoločnosti však prišli nové požiadavky na softvér. Tento sa stáva stále zložitejším a vyžaduje neustále zvyšovanie nárokov na softvérových inžinierov. V záujme zachovania vysokej kvality softvéru sa na konci 20. storočia začali

miesto jednotlivcov presadzovať tímy ľudí, ktoré softvér vytvárajú. Proces vytvárania produktu v tíme sme nazvali projektom a čoskoro vznikla potreba tieto projekty istým spôsobom riadiť. Preto z nutnosti vznikla nová disciplína – manažment softvérových projektov.

Problémy, ktoré so sebou táto disciplína prináša, môže ilustrovať nasledujúci príbeh: *”Byl jednou jeden podnik a v něm čtyři zaměstnanci, kteří se jmenovali Každý, Někdo, Kdokoliv a Nikdo. Jednoho dne bylo třeba splnit důležitý úkol a Každý si byl jist, že to Někdo udělá. Mohl to udělat Kdokoliv, ale Nikdo to neudělal. Někdo se rozzlobil, protože to přece byla práce pro Každého. Každý si myslel, že by to mohl udělat Kdokoliv, ale Nikdo si neuvědomil, že to Každý neudělá. Nakonec Každý obviňoval Někoho, že Nikdo neudělal to, co mohl udělat Kdokoliv”* [5].

Riadenie softvérového projektu je proces, ktorý pozostáva najmä z tvorby plánov na rôznej úrovni granularity, pridelovanie úloh jednotlivým výkonným členom softvérového tímu a následnou kontrolou plnenia plánu. Príkladom jednotlivých plánov môže byť plán zodpovedností, plán rozsahu projektu, plán rizík.

Plány sú pre projekt mantinelmi, ohraničujú priestor jeho stavov do želateľnej roviny a stanovujú hraničné termíny. Plánovanie v softvérovom projekte (a projektoch všeobecne) však nie je mladou disciplínou a ukazuje sa, že tie projekty, kde nebol vopred určený plán boli málokedy úspešné [3].

Ďalším problémom, ktorému musí projektový manažér čeliť je odhadovanie v plánovaní. Jednou z hlavných príčin pre neúspešnosť projektov je totiž nedostatok času [2]. Ten môže byť spôsobený nedostatočným plánovaním, nedodržiavaním plánu, alebo prílišným optimizmom manažéra plánovania, jednoducho zlým odhadom pri plánovaní. Následkom zlého alebo žiadneho plánovania bývajú zmeškané termíny, prekročenie rozpočtu alebo zastavenie celého projektu.

Projektový manažér má niekoľko možností, ako odhadnúť časovú, nákladovú, prípadne inú náročnosť:

- odhad na základe analógie s podobným projektom, ktorý mal analogické podmienky tvorby, času, ľudských aj materiálnych zdrojov.
- posúdením experta – jednotlivé hraničné termíny určí expert, ktorý má skutočne mnoho relevantných skúseností s predošlými projektami.
- odhad na základe modelu – trvanie akcií určí niektorý z modelov, napr. Cocomo alebo Putnam. Modely dávajú individuálne odhady na základe určitých netriviálnych vlastností projektu, neustále sa zlepšujú a aktualizujú.
- iné – objavujú sa výskumy, snažiace sa vytvoriť umelé agenty, ktoré by pomohli pri rozhodovaní.

Keď sa vytvoria odhady trvania jednotlivých aktivít v projekte, je dôležité vytvoriť zodpovedajúci časový rozvrh, teda plán postupu prác. Z tohto by mal byť jasný naplánovaný čas trvania aktivít, prípadne aj miera zodpovednosti členov tímu za splnenie

konkrétnych úloh a vhodným doplnkom je aj stav úlohy v časovom priebehu. Takto je možné jednoducho monitorovať splnenie úloh.

Časové rozvrhy majú niekoľko typov, medzi najčastejšie používané patrí úsečkový graf (*Ganttov diagram*) a sieťový graf.

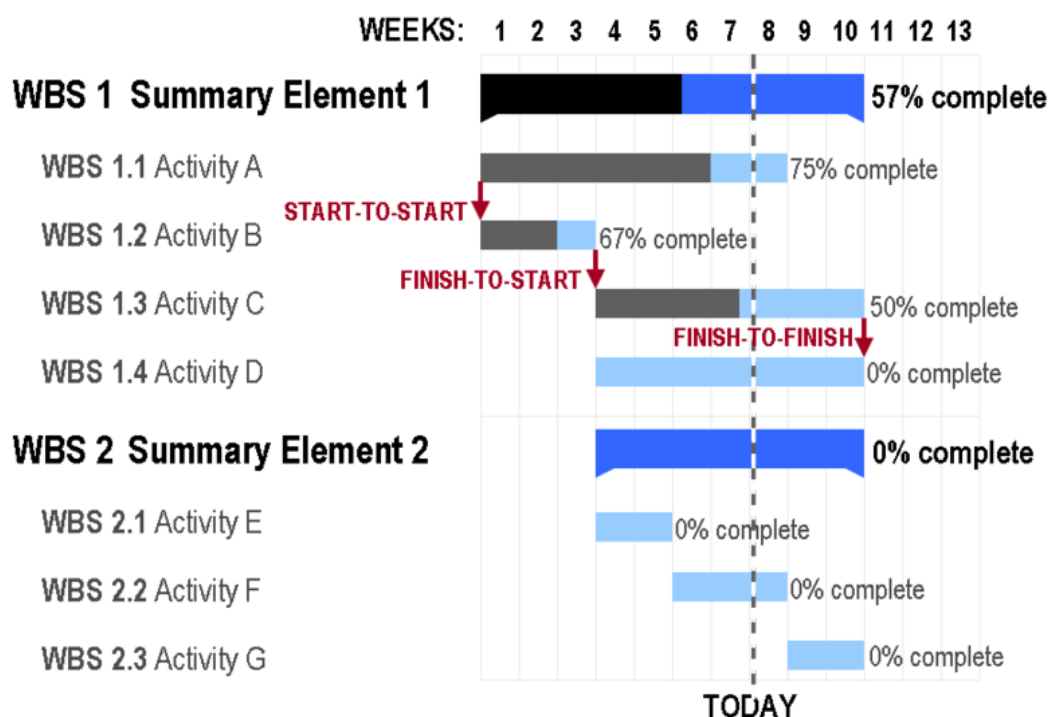
Pre jednoznačné určenie miery zodpovednosti je možné použiť *maticu zodpovednosti*. Matica zodpovednosti je jednoduchá tabuľka, ktorej stĺpce reprezentujú úlohu v projekte a riadky osoby, ktoré sa na nich zúčastňujú. Jednoduchým číslom potom možno reprezentovať zodpovednosť každej zúčastnenej osoby na úlohe.

Časové rozvrhy

V tejto časti sa oboznámime s dvomi typmi časových rozvrhov – Ganttov diagram a Sieťový graf.

Ganttov diagram

Historicky bol prvýkrát použitý Poliakom Adamom Adamieckim, ktorý však svoje výsledky nepublikoval. Všeobecne sa za vynálezcu považuje Henry Gantt, ktorý svoj časový rozvrh navrhol v rokoch 1910 [1].



Obr. 1. Ganttov diagram¹.

¹ Zdroj [png]: <http://upload.wikimedia.org/wikipedia/commons/e/e2/GanttChartAnatomy.png>

Ganttov diagram je úsečkový graf ilustrujúci projektový plán. Zobrazuje začiatok a koniec činností v procese tvorby softvérového projektu. Dĺžka úsečky znázorňuje dobu trvania jednotlivých činností. Dajú sa v ňom zobrazovať aj závislosti (vzťahy) medzi činnosťami – napr. niektoré úlohy nemôžu začať pred ukončením iných, ďalšie sú nezávislé a tak môžu byť riešené súčasne. Okrem toho je možné doň zaniest aj percentuálne naplnenie jednotlivých aktivít a vertikálnu čiaru znázorňujúcu „DNES“. Toto umožňuje sledovať skutočný priebeh projektu a kontrolovať plnenie plánu.

Veľkou výhodou Ganttových diagramov je ľahká pochopiteľnosť a jednoduchosť zmeny. Takýto diagram je na obr. 1. na predchádzajúcej strane.

Napriek častému a pomerne jednoduchému použitiu má táto schéma aj nevýhody. Pri väčších projektoch, ktoré majú rádovo stovky úloh totiž jeho rozmery presahujú veľkosť papiera, prípadne monitoru a celkový pohľad na projekt sa tak stáva značne neprehľadným. Vtedy je zrejme rozumné uvažovať o dekompozícii diagramu na jednotlivé etapy projektu, čím zmenšíme jeho rozmery a zlepšíme pochopiteľnosť. Ak chceme zobrazovať viacero závislostí jednej činnosti na iných, stáva sa nezrozumiteľným.

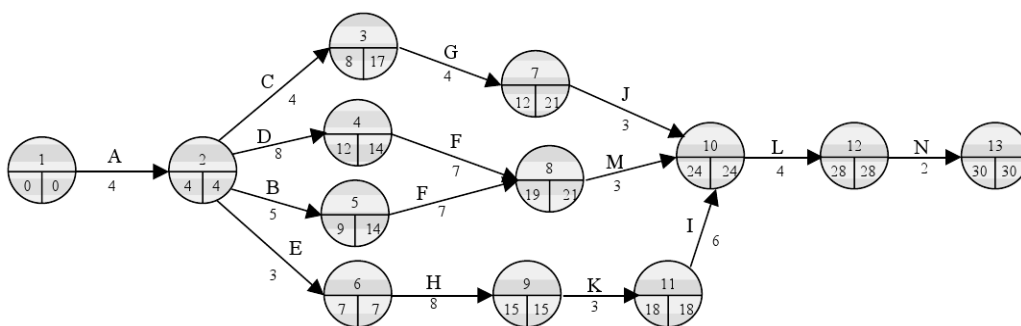
V neposlednom rade pomocou Ganttovej schémy nie je možné zobrazovať objem prác, potrebných na dokončenie úlohy, ale len čas. Na to sa však dajú použiť iné metódy, ako napríklad sledovanie vytváraných dokumentov a kódu.

Sieťový graf

Sieťové grafy delíme na dva typy:

1. *Hranovo orientované* – činnosti projektu sú vyjadrené hranami grafu, vrcholy znamenajú mílniky. Tento typ grafu je na obr. 2.
2. *Uzlovo orientované* – činnosti sú vyjadrené uzlami grafu.

V časových harmonogramoch rozpoznávame aj *CPM – Critical Path Method*. CPM určuje činnosti, ktorých oneskorenie by spôsobilo meškanie celého projektu.



Obr. 2. Sieťový graf - hranovo orientovaný. Prebrané z [4].

Podporné nástroje

Čas, potrebný na vytvorenie úsečkových a sieťových grafov sa však s narastajúcou komplexnosťou vytváraného softvéru predlžuje. Softvéroví inžinieri by preto chceli niektoré tieto činnosti automatizovať alebo zjednodušiť, preto vzniklo množstvo podporných nástrojov pre manažment projektov.

Softvéroví inžinieri však potrebujú aj nástroje pre samotné projektové činnosti, ako je podpora spolupráce tímu, správa verzií kódu a generovanie reportov a štatistických výstupov. Tieto je vhodné integrovať do rovnakého rozhrania ako podporu riadenia, aby sa všetky činnosti dali vykonávať z jedného miesta.

Podporné nástroje sa dajú rozdeliť na viacero typov. Na základe robustnosti na komplexné a jednoduché, na základe architektúry na klient-server a samostatné aplikácie, na základe vývojovej paradigmy na inkrementálne a agilné a v neposlednom rade na open-source a proprietárne nástroje. V nasledujúcej časti sa budem venovať vybraným komplexnejším podporným nástrojom na riadenie softvérového projektu, výhodám a nevýhodám konkrétnych možností.

Komplexné nástroje

Komplexné nástroje väčšinou ponúkajú vlastnosti, ako je sledovanie zmien v projekte, podpora plánovania na rôznych úrovniach projektu, prepojenie na systém verziovania dokumentov a kódu, podporu rozhodovania a komunikácie s tímom.

Proprietárne systémy

Proprietárne systémy patria do skupiny nástrojov s uzavretým zdrojovým kódom, skôr komerčného rázu. Predávajú sa teda primárne za účelom zisku a nie je možné ich ďalej rozširovať. Medzi zástupcov takýchto systémov pre podporu riadenia patrí *Microsoft Office Project Server* a *Workspace.com*, aj keď svoje implementácie majú aj firmy IBM, Oracle a iné.

Microsoft Office Project Server

je riešenie od firmy Microsoft. Tvorí serverovú časť a webové rozhranie, spolupracuje s klientskou aplikáciou MS Project, architektúra je teda klient-server a umožňuje spoluprácu tímu cez web. Podporuje riadenie softvérového projektu na všetkých úrovniach, umožňuje plánovanie, rozdeľovanie úloh, generovanie reportov, dá sa prepojiť so systémom verziovania. Pre študentov existuje verzia zdarma, ktorá je voľne prístupná z MSDNAA. Jeho veľkou nevýhodou je obmedzenie na platformu MS Windows. Napriek námietke, že 90% počítačov beží pod MS Windows, diskriminácia menších a preferencia monopolov nikdy nič pozitívne neprinesla.

Workspace.com

je webová aplikácia dostupná na adrese www.workspace.com. Zlepšuje spoluprácu cez možnosť pridať komentár ku akejkoľvek časti projektu, podporuje systém verziovania, plánovanie a generovanie Ganttových diagramov, riadenie ľudských zdrojov, bugtracking a testovanie. Veľkou výhodou je automatizovaná tvorba vyčerpávajúcich štatistík.

Zlepšuje tak kontrolu nad postupom projektu. Navyiac umožňuje importovať plány vytvorené v MS Project.

Open-source nástroje

Open-source technológie vnášajú do priestoru podporných prostriedkov pre riadenie softvérových projektov ideu, že aj za málo finančných prostriedkov je možné získať výborný produkt (ak si dovoľíme zanedbať cenu energií, prípadne hostingu, žiadne). Ale je to naozaj tak?

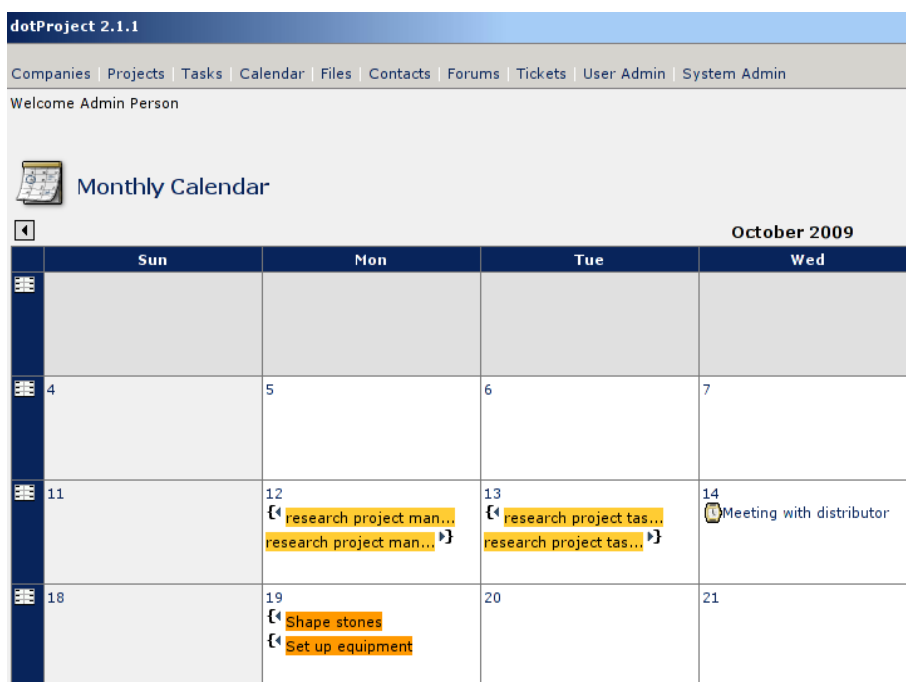
Open-source nástroje sú väčšinou webového typu a medzi výborné aplikácie patrí rozhodne *dotProject* a *]Project-open[*.

DotProject

je webová aplikácia založená na PHP a MySQL. Je značne rozsiahla a podporuje spravovanie ľudských zdrojov, vytváranie Ganttových diagramov, kalendár, generovanie reportov, fórum, verziovanie kódu a systém tiketov. Postup jednotlivých častí projektu sa hodnotí percentuálne. Nevýhodou je, že percentuálne hodnotenie stavu spracovania úlohy nie vždy vyjadruje koľko času a úsilia bude ešte potrebné do úlohy vložiť. Rozhranie kalendára *dotProjectu* je na obr. 3.

]Project-open[

Veľmi robustná aplikácia umožňujúca riadiť celý biznis. Podporuje teda celý rad riadiacich funkcií, výrobu Ganttových diagramov, manažment ľudských zdrojov, financií, vlastnú wiki pre zjednodušenie spolupráce. Má modulárnu architektúru, čo znamená jednoduchú rozšíriteľnosť (ako je možné vidieť aj z počtu funkcií). Jeho veľká komplexnosť je zrejme asi aj najväčšou nevýhodou, keďže členovia tímu musia stráviť dlhší čas, aby sa naučili s takto robustným systémom efektívne pracovať.



Obr. 3. Rozhranie kalendára nástroja dotProject.

Výsledok porovnania

Túto časť by sme mohli začať citátom, ktorý naznačuje výsledky porovnania: „*The fact is, there aren't just two sides to any issue, there's almost always a range of responses, and "it depends" is almost always the right answer in any big question.*“ (Linus Torvalds)

Ako sme ukázali, open-source nástroje sú čo do funkcionality plne konkurencieschopné komerčným produktom. Hlavnou nevýhodou komerčných systémov je vo väčšine prípadov cena, ktorú za podporný prostriedok zaplatíme. Tu je na mieste sa spýtať, či sa vložená investícia vráti. Takýto podporný prostriedok však nepoužijeme iba na jeden projekt a súčasťou investície býva aj zaručená technická podpora, prípadne zaškolenie pracovníkov.

Open-source nástroje sú zadarmo, avšak podpora je často diskutabilná, väčšinou vo forme mailing-listu, fóra, prípadne wiki. Pri otvorenom zdrojovom kóde a predpoklade, že so systémom pracujú profesionáli v oblasti IT je možné, že chybu budú vedieť odstrániť na základe odporúčaní, prípadne modifikovaním kódu.

V konečnom dôsledku teda o výbere podporného prostriedku rozhodnú priority projektového manažéra, či dá prednosť technickej podpore alebo finančným prostriedkom. Osobne by som dal prednosť open-source nástroju pred komerčnou verziou, keďže finančné prostriedky sú u mňa na prvom mieste. Pre názorné porovnanie funkcií jednotlivých systémov som vytvoril tabuľku porovnania vlastností (tab. 1).

Tab. 1. Porovnanie niektorých vlastností vybraných podporných prostriedkov.

Názov prostriedku	Ganttov diagram	Prideľovanie úloh	Generovanie reportov	Správa verzií kódu	Wiki (alebo fórum)
MS Project Server	Áno	Áno	Áno	Áno	Áno
Workspace.com	Áno	Áno	Áno	Áno	Áno
dotProject	Áno	Áno	Áno	Áno	Áno
]Project-open[Áno	Áno	Áno	Áno	Áno
Názov prostriedku	Architektúra	Obmedzenie na platformu	Licencia		
MS Project Server	Klient-server	Áno	Proprietárna		
Workspace.com	Web-aplikácia	Nie	Proprietárna		
dotProject	Web-aplikácia	Nie	Open-source		
]Project-open[Web-aplikácia	Nie	Open-source		

Aký podporný prostriedok k menšiemu projektu?

Pre úspešnosť projektu je plánovanie a sledovanie priebehu projektu kľúčové. Použitie podporných prostriedkov by teda malo pomôcť projektovému manažérovi ušetriť čas pri tvorbe a kontrole plánov a centralizované rozhranie pre projekt pomôže celému tímu ušetriť prácu aj čas pri revíziách kódu, jeho tvorbe a pri písaní dokumentácie.

Čo však v prípade tímov menších, rádovo 5-7 ľudí, menších projektoch, kde nie je toľko riadiacich úkonov, úlohy sa dajú rozdeliť pomerne transparentne a ich závislosti netvorí nesúvislý priestor?

Keďže softvéroví inžinieri sú leniví, podporný prostriedok pre riadenie by ich mal odbremeniť od istej formy administratívnej práce, ktorá v závislosti od typu osobnosti nie každému sadne. Je namieste otázka, či nestrávia zbytočne veľa času zháňaním podporného prostriedku, jeho inštaláciou a oboznamovaním sa s novým systémom, ktorý nemusí mať triviálne ovládanie. Niektoré úkony systému by pritom zvládol aj obyčajný tabuľkový editor a zdieľanie súborov na sieti.

Preto je vždy nutné zvoliť systém ekvivalentne k náročnosti projektu a jeho riadenia. Pre menšie projekty postačujú podporné prostriedky jednoduchšieho typu, ako napríklad *Trac* alebo *Redmine*.

Trac

Síce nedisponuje robustnými nástrojmi pre plánovanie, ale systém tiketov je dostatočne prehľadný a postačujúci ak nepotrebujeme sledovať vysoký počet náročných úloh, ktoré sú v čase nekonzistentne rozložené. Tento systém používa dostatočne veľký počet projektov, napr. Jet Propulsion Laboratory NASA používa systém *Trac* na manažovanie rôznych projektov ďalekého aj blízkeho vesmíru.

Redmine

Redmine je projekt postavený na technológii Ruby a Ruby on Rails. Je dostatočne flexibilný, poskytuje Ganttové diagramy, kalendár. Pre podporu tímovej spolupráce dodáva jednoduchú wiki. Veľkou výhodou je podpora rôznych systémov pre verziovanie softvéru, ako je svn, cvs, git... Tým pádom nie sú tímy obmedzené vo výbere takéhoto systému. Má jednoduché rozhranie, ktoré je preložené do mnohých jazykov (vrátane slovenčiny).

Tím bez podporného prostriedku

Napriek tomu, že softvéroví inžinieri sú leniví (a možno práve preto), stále sa nájdu také softvérové tímy, ktoré všetko plánujú a riadia „ručne“. Plány píšú do tabuľkového editora a v tom lepšom prípade ich zdieľajú cez web, úlohy pridelujú telefonicky, prípadne elektronickou poštou.

Lenivosťou učiť sa pracovať s podporným prostriedkom si však vyrábajú komunikačné, riadiace a plánovacie problémy. V prvom rade musia pracovať s viacerými nástrojmi, čo ich oberá o čas, ak je potrebné tie isté údaje použiť v rôznom type riadiacej činnosti. Ďalšie signifikantné časové oneskorenie je spôsobené samotným tvorením šablón pre plány, diagramy a podobne.

„Ručný prístup“ je taktiež netransparentný pre zúčastnené strany a vyžaduje si pomerne viac úsilia, ako vkladanie údajov do na to pripraveného nástroja. Taktiež je sťažená možná archivácia pre budúcu analýzu a tvorba štatistík a reportov je prakticky nemožná.

Myslíme si, že s jednoduchým podporným prostriedkom, ktorý tieto problémy rieši za nás by mohlo viacero projektov viesť do úspešného konca v skoršom termíne.

Záver

V odbornej eseji sa venujem téme podporných prostriedkov pre riadenie softvérového projektu. Najprv sme si priblížili problémovú oblasť, teda čo je potrebné pre riadenie projektu, prečo plánujeme a hlavne akými spôsobmi. Použitie Ganttových diagramov alebo sieťových grafov je vhodné pre grafickú reprezentáciu časových rozvrhov, avšak oba spôsoby majú aj svoje nevýhody.

Podporné prostriedky by mali riadenie uľahčovať, mali by uľahčovať tvorbu časových rozvrhov, tvorbu plánov. Pre tvorbu softvérového projektu je dobré, ak umožňujú aj správu samotného kódu, ako je verziovanie a uľahčujú komunikáciu v tíme napr. cez wiki alebo fórum.

Prostriedky, ktoré takéto doplnky poskytujú sú komplexnejšie a porovnaním zástupcov zo sveta komerčných a open-source nástrojov som dospel k záveru, že čo do funkcionality sú zhodné. Rozdiely sú len v technickej podpore a cene, teda výber závisí na týchto prioritách.

V malých tímoch však komplexnosť nie je nutnosťou, ba naopak. Vyžaduje sa skôr jednoduchší prístup, ktorý nebude nútiť členov tímu učiť sa narábať s komplikovaným systémom.

Použitá literatúra

1. Blokdijk, G.: *Project Management 100 Success Secrets*. Emereo Pty Ltd (November 19, 2007). ISBN 0980459907.
2. Brooks, F.P.: *The Mythical Man-Month: Essays on Software Engineering*. Addison-Wesley, 1995. ISBN 0-201-83595-9.
3. Humphrey, Watts S.: Why Big Software Projects Fail: The 12 Key Questions. *The Journal of Defense Software Engineering*, Marec 2005
4. *Projektové riadenie* [online]. 25. September 2003 [cit. 10. 10. 2009]. Dostupné na internete: <http://web.tuke.sk/sjf-icav/stranky/obrazky2/e-texty/pr_cvicenia.pdf>
5. Staníček, Z., Hajkr, J.: *Řízení projektů zavádění IS do organizací*. Tutorial Datakon 2005.

Annotation

Open-source vs. commercial tools for managing software projects

My essay is bound by scope of support tools for managing software projects, with emphasis on human resources utilization and task monitoring. It contents introduction into management of software project. It seeks, which options exist for actual support of managing, planning and task monitoring. There is a number of them, for example Gantt chart or Network graph. It researches, which features of support software are demanded or desired from software engineers. Essay touches some commercial and open-source tools for support of managing software projects. Author is trying to compare them objectively, considering group of chosen features. Furthermore, essay brings author's opinion on necessity of using complex tools in smaller projects.