



UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY
KATEDRA INFORMATIKY

Analýza a posúdenie štandardov pre riadenie údržby softvéru

DIPLOMOVÁ PRÁCA

JÚLIUS JUHÁSZ

Vedúci práce: Mgr. Róbert Kormaňák

Čestne prehlasujem, že som túto diplomovú prácu vypracoval samostatne, s použitím literatúry a zdrojov uvedených v závere práce.

Bratislava, 7.5.2008

Ďakujem môjmu diplomovému vedúcemu Mgr. Róbertovi Kormaňákovi za odborné vedenie, cenné rady a pripomienky. Ďakujem mojim rodičom za obrovskú podporu počas môjho štúdia.

Abstrakt

JUHÁSZ, Július: Analýza a posúdenie štandardov pre riadenie údržby softvéru. [Diplomová práca]. Univerzita Komenského v Bratislave. Fakulta matematiky, fyziky a informatiky; Katedra informatiky. vedúci diplomovej práce: Mgr. Róbert Kormaňák. Stupeň odbornej kvalifikácie: Magister. Bratislava, 2008, 82 s.

Diplomová práca sa zaoberá problematikou softvérovej údržby, hlavne riadením tohto procesu. Popisuje spôsob merania udržovateľnosti softvérových aplikácií. Na základe údajov z vykonaného prieskumu analyzuje vplyv zásad riadenia údržby podľa medzinárodných štandardov na celkovú úroveň udržovateľnosti aplikácií v skúmaných softvérových spoločnostiach. Taktiež skúma vplyv iných faktorov na množstvo úsilia potrebného na údržbu aplikácií.

Kľúčové slová: softvérová údržba, životný cyklus softvéru, softvérové inžinierstvo, medzinárodné štandardy

Obsah

1	Úvod.....	9
2	Údržba v životnom cykle softvéru	11
2.1	Údržba ako etapa vývoja softvéru.....	11
2.2	Význam údržby	12
2.3	Základné typy údržby.....	13
2.3.1	Oprava.....	13
2.3.2	Prispôsobenie	13
2.3.3	Zlepšenie	14
2.3.4	Prevenca.....	14
2.3.5	Predikcia.....	14
2.4	Rozdelenie údržbových aktivít.....	14
2.5	Motivácia k prijatiu štandardov na údržbu	15
3	Identifikácia štandardov údržby.....	17
3.1	ITIL	17
3.1.1	Vývoj ITIL	18
3.1.2	Obsah ITIL	18
3.1.3	Service desk	19
3.1.4	Manažment incidentov	19
3.1.5	Manažment problémov.....	20
3.1.6	Manažment konfigurácii	21
3.1.7	Manažment zmien	21
3.1.8	Manažment verzii	26
3.2	International Organization for Standardization.....	28
3.2.1	Štruktúra ISO	28
3.2.2	ISO normy zaoberajúce sa softvérom.....	29
3.3	Medzinárodný štandard ISO/IEC 12207	30
3.3.1	Procesy na dosiahnutie dohody	32
3.3.2	Organizačné procesy	32
3.3.3	Projektové procesy	32
3.3.4	Technické procesy.....	33
3.3.5	Procesy implementácie softvéru.....	33
3.3.6	Podporné softvérové procesy	34
3.3.7	Procesy využívania skúseností a znalostí z predchádzajúcich aktivít	34
3.4	Medzinárodný štandard ISO/IEC 14764	34
3.4.1	Rozsah.....	34
3.4.2	Implementácia procesov.....	36
3.4.3	Analýza problémov a úprav	37
3.4.4	Implementácia úprav	38
3.4.5	Revízia/schválenie úpravy.....	39
3.4.6	Migrácia	39
3.4.7	Výslužba softvéru.....	40
4	Výskum údržby v praxi	42
4.1	Cieľ dotazníka	42
4.1.1	Cieľ 1 – analýza riadenia procesu údržby	42
4.1.2	Cieľ 2 – analýza vplyvu ďalších faktorov	43
4.1.3	Cieľ 3 - vytvorenie metriky udržovateľnosti.....	43

4.2	Metodika tvorby dotazníka a otázky.....	43
4.2.1	Otázky ohľadom riadenia údržby	43
4.2.2	Otázky ku analýze aplikácií	45
4.2.3	Doplňujúce otázky	46
4.3	Distribučovanie dotazníkov	46
4.4	Analýza získaného portfólia.....	48
4.5	Dôkaz rozmanitosti portfólia	48
4.6	Meranie udržovateľnosti aplikácií	49
4.6.1	Analýza metrik udržovateľnosti	49
4.6.2	Vytvorenie a overenie metriky udržovateľnosti	49
4.7	Metodika hodnotenia odpovedí.....	51
4.8	Výsledky analýzy riadenia údržby	52
4.8.1	Plány údržby.....	53
4.8.2	Analýza MR/PR	53
4.8.3	Implementovanie a testovanie	54
4.8.4	Migrácia	54
4.8.5	Manažment verzii	55
4.8.6	Manažment konfigurácií	55
4.8.7	Organizácie so štandardom	55
4.9	Vplyv ďalších faktorov na proces údržby.....	57
4.9.1	Dokumentácia	57
4.9.2	Zložitosť kódu.....	57
4.9.3	Dostupnosť vývojového a testovacieho prostredia.....	57
4.9.4	Štruktúra kódu.....	58
4.9.5	Skúsenosti tímu	58
4.9.6	Fluktuácia programátorov	58
4.9.7	Vek aplikácie.....	59
4.9.8	Analýza spoločného vplyvu.....	59
5	Zhodnotenie údržby	60
5.1	Zhodnotenie výskumu	60
5.2	Posúdenie normatívo.....	61
5.2.1	Posúdenie štandardu ISO 14764.....	61
5.2.1	Posúdenie štandardu ITIL	62
5.3	Posúdenie aktuálnej úrovne riadenia údržby.....	62
6	Záver.....	64
	Slovník.....	66
	Literatúra.....	68

Zoznam obrázkov

<i>Obrázok 1: Údržbové aktivity</i>	15
<i>Obrázok 2: Vzťahy CMDB k ostatným procesom</i>	21
<i>Obrázok 3a: schéma manažmentu zmien 1. časť</i>	24
<i>Obrázok 3b: schéma manažmentu zmien 2. časť</i>	25
<i>Obrázok 3c: schéma manažmentu zmien – štandardný problém</i>	26
<i>Obrázok 3d: schéma manažmentu zmien – urgentná zmena</i>	27
<i>Obrázok 4: štruktúra ISO</i>	29
<i>Obrázok 5: prehľad ISO/IEC noriem</i>	30
<i>Obrázok 6: Schéma skupín procesov štandardu ISO 12207</i>	31
<i>Obrázok 7: aktivity procesu údržby</i>	35
<i>Obrázok 8: rozdelenie údržbových aktivít z dotazníkov</i>	48
<i>Obrázok 9: Graf kriviek pre zložky indexu udržovateľnosti</i>	50

Zoznam tabuliek

<i>Tabuľka 1: prehľad vybraných ukazovateľov výskumu</i>	47
<i>Tabuľka 2: hodnoty metrik udržovateľnosti a odhady</i>	50
<i>Tabuľka 3: analýza procesov údržby v organizáciách I.</i>	53
<i>Tabuľka 4: analýza procesov údržby v organizáciách II.</i>	55
<i>Tabuľka 5: analýza procesov údržby v organizáciách II.</i>	57

Kapitola 1

Úvod

Scenár až príliš známy: Aplikácia slúžila obchodným potrebám spoločnosti 10 alebo 15 rokov. Počas tej doby bola mnohokrát menená, prispôbovaná a vylepšovaná. Programátori k tejto práci pristupovali s najlepšimi úmyslami, ale správne praktiky softvérového inžinierstva sa nikdy nedostali ku slovu kvôli iným okolnostiam. Teraz je aplikácia nedostatočne zmapovaná, nedostatočne štruktúrovaná, bez záloh a popisov ... jednoducho nestabilná. Stále funguje v produktívnom prostredí, ale každý pokus o zmenu je sprevádzaný neočakávanými a vážnymi vedľajšími príznakmi. Aplikácia sa však stále musí rozvíjať. Čo máme robiť ?

Neudržiavaný softvér nie je novým problémom.

Pred tridsiatimi rokmi R. Canning podľa [8] charakterizoval údržbu softvéru ako ľadovec: „*Dúfame, že to, čo práve vidíme, je všetko, čo tam je, ale ako vieme, obrovská masa potenciálnych problémov a úsilia leží pod hladinou. Na začiatku sedemdesiatych rokov bol tento ľadovec dosť veľký na to, aby potopil lietadlovú loď. Dnes môže hravo potopiť celé vojenské námorníctvo!*“

Na začiatku práce uvedieme čitateľov práve do problematiky softvérovej údržby. Spomenieme akú dôležitú úlohu zohráva údržba počas životného cyklu softvéru, predstavíme základné typy údržbových aktivít, aby sme čitateľov presvedčili, že údržba zďaleka nie je iba opravovanie chýb v programe. Priblížime, aké veľké sú náklady softvérových spoločností na údržbu aplikácii a vysvetlíme, prečo je v dnešnej dobe, ešte viac ako kedykoľvek pred tým, veľmi dôležité systematicky sa zaoberať kvalitou procesu údržby v každej spoločnosti, ktorá má záujem sa ďalej rozvíjať.

Od systematického prístupu je už len malý krôčik ku štandardom a normám. Aj spoločnosti plnej šikovných a bystrých vývojárov môže trvať roky, kým sama vytvorí a postupne zdokonalí potrebné procesy vývoja na dostatočnú úroveň. Naproti tomu medzinárodne štandardy ponúkajú týmto spoločnostiam už dnes hotové riešenia, ktoré vytvorili tímy špičkových svetových softvérových inžinierov, ich vývoj sa stále zdokonaľuje a tieto štandardy implementovali a úspešne používajú najväčší svetoví hráči, ktorým ešte viac pomohli zvýšiť zisky z predaja softvérových produktov.

Zastavme sa na tomto mieste v našej úvahe a položme si otázku či je to naozaj tak.

Je implementovanie štandardu ozaj univerzálnym riešením, ktoré zaručí okamžitý, rapídny nárast kvality, produktivity a ziskov? Oplatí sa bežnej IT spoločnosti vynaložiť (často nemalé) prostriedky a čas na prijatie normy? Aké sú tieto normy, čo ponúkajú a čo vyžadujú? Naozaj platí, že spoločnosť, ktorá riadia proces údržby svojej aplikácie pomocou medzinárodného štandardu spotrebuje menej úsilia na implementovanie novej funkcionality do aplikácie, ktorá je už v prevádzke? Pomôže týmto programátorom norma rýchlejšie nájsť a opraviť chybu, kvôli ktorej užívatelia nemôžu s aplikáciou pracovať?

Práve na tieto otázky sa pokúsime nájsť odpoveď v tejto práci. Najväčšia medzinárodná organizácia v oblasti štandardizácie pred nedávnom vydala novú verziu medzinárodnej normy, ktorá sa venuje práve údržbe softvéru. ISO však nie je jediná, ktorá má svoju verziu „najlepších návodov na riadenie údržby“.

V tejto práci dôkladne vysvetlíme ako fungujú tieto modely, čo zdôrazňujú, čo prinášajú, ale aj to, čo si musí každá spoločnosť vytvoriť sama. Takto pripravený sa potom konečne pokúsime nájsť odpovede na otázky, ktoré sme pred chvíľou položili:

Vytvoríme rozsiahly dotazník, do ktorého zapojíme široké spektrum softvérových firiem a v každej z nich do detailov zmapujeme proces údržby a získame veľké množstvo i iných informácií, ktoré nám, ako dúfame, pomôžu nájsť odpovede na čo najviac otázok.

Z takto získaných údajov sa pokúsime nájsť spôsob, ktorý nám dokáže o ľubovoľnej aplikácii povedať, nakoľko náročná je ich údržba. Ak sa nám podarí takýto spôsob nájsť, budeme pomocou neho môcť ďalej skúmať, či spoločnosti vynakladajú na údržbu skutočne menej úsilia práve vtedy, ak riadia svoj proces údržby medzinárodnou normou (alebo na základe dotazníka vieme, že ich proces je vo veľkej miere podobný týmto štandardom). To by nám dalo želanú odpoveď na dôležitú otázku, či normy naozaj prinášajú to, čo od nich očakávame.

Okrem tejto analýzy využijeme množstvo získaných údajov o spoločnostiach aj na skúmanie, či existujú iné okolnosti, ktoré nejakým spôsobom vplyvajú na udržovateľnosť aplikácií. V tejto časti výskumu sa pokúsime napríklad preveriť, či skúsenosti programátorov, vlastnosti kódu, vek aplikácie a iné aspekty ovplyvňujú náročnosť údržby v hodnotených spoločnostiach.

Kapitola 2

Údržba v životnom cykle softvéru

Táto kapitola poskytuje úvod do problematiky softvérovej údržby. Predstavíme pojem softvérovej údržby, opíšeme základné druhy údržby, jej pozíciu v životnom cykle softvéru a načrtujeme, prečo je v každej spoločnosti dôležité dôsledne sa údržbou zaoberať.

2.1 Údržba ako etapa vývoja softvéru

V softvérovom inžinierstve je evidentný všeobecný súhlas so skutočnosťou, že vývoj každého programového systému treba vykonávať systematicky. Z tohto dôvodu sa vývoj rozdeľuje na definované etapy a zavádza sa pojem životného cyklu programu. Takýto prístup prináša rozdelenie (dekompozíciu) zložitého problému na jednoduchšie a ľahšie zvládnuteľné problémy. Rozdiely v rôznych prístupoch k životnému cyklu zväčša spočívajú v členení na jednotlivé etapy a postupnosti, ktorými cez ne treba prechádzať. Podstatnou charakteristikou každého modelu životného cyklu však je, že definuje jednotlivé etapy a pre každú z nich činnosti, ktoré sa majú vykonať, rovnako ako vstupy a výstupy etapy. Takýmto spôsobom sa proces vývoja softvérových systémov stáva prehľadnejší a kontrolovateľnejší. Základné etapy vývoja softvéru sú:

- špecifikácia požiadaviek
- architektonický návrh
- podrobný návrh
- implementácia
- integrácia a testovanie
- prevádzka a údržba

Etapa prevádzky a údržby softvérového systému súvisí so zabezpečením každodennej prevádzky systému a jeho ďalšieho rozvoja. Začína sa odovzdaním (a akceptovaním) softvérového systému do produktívnej prevádzky. Zabezpečovanie každodennej prevádzky zvyčajne prechádza na používateľa

a ďalší rozvoj systému (údržba a modifikácie) zväčša zostáva na dodávateľovi (tvorcovi). Pri riešení požiadaviek na zmenu sa môže meniť špecifikácia aj návrh systému. V etape údržby sa pracuje so všetkými doteraz vytvorenými dokumentmi, tieto sa modifikujú, dopĺňajú a upravujú.

2.2 Význam údržby

Rôzne zdroje literatúry, napr. [11] uvádzajú, že údržba môže spotrebovať aj 50 – 80 % celkového úsilia organizácie, ktorá tento softvér vyvíja, navyše percento stále rastie, čím dlhšie je softvér vyvíjaný a používaný. Na mieste je teda otázka, prečo je údržba tak potrebná a prečo stojí tak veľa úsilia. Čiastočnú odpoveď ponúka vysvetlenie platné tak v deväťdesiatych rokoch, ako aj dnes: Väčšina softvéru, na ktorom dnes závisíme, je stará približne 10 – 15 rokov. Aj keď tieto programy boli vyvíjané s použitím najlepších návrhov a programovacích techník známych v tom čase (a mnohé v skutočnosti ani neboli), boli tvorené v čase, keď veľkosť programu a úložného priestoru boli principiálne záležitosť. Neskôr boli premigrované na nové platformy, upravené kvôli zmenám hardvéru i vývoju operačných systémov a vylepšované, aby dokázali uspokojiť nové potreby používateľov. To všetko bez dostatočného ohľadu na celkovú architektúru. Výsledkom je zle navrhnutá štruktúra, slabá logika, nedokonalé programovanie i chabá dokumentácia softvérových systémov, od ktorých žiadame, aby aj dnes ďalej fungovali...

Charakteristická vlastnosť *Zmien* je ich všadeprítomnosť a predstavuje základ v každej práci so softvérom. Ak budujeme počítačový systém, musíme mať na pamäti, že zmeny sú neodvratiteľné. Preto musíme vyvíjať mechanizmy na hodnotenie, kontrolovanie a realizovanie modifikácií.

Po prečítaní týchto riadkov by sme mohli protestovať a namietat', že my predsa netrávime 60% nášho času opravovaním chýb v programe, ktorý vyvíjame. Údržba softvéru však znamená oveľa viac ako len opravovanie chýb. Základné typy údržbových aktivít predstavíme v nasledujúcej kapitole.

Existuje nepriama úmera medzi nákladmi na vývoj a nákladmi na údržbu. Ak dve podobné organizácie vyvíjajú rovnaký softvér a jedna z nich sa rozhodne dať väčší dôraz na dodržiavanie zásad pre dosiahnutie vyššieho stupňa udržovateľnosti napr. tým, že prijme niektorý z osvedčených modelov riadenia údržby, jej náklady investované do fázy vývoja sa jej vrátia vo forme ušetrených prostriedkov na údržbu po dokončení softvéru a po jeho nasadení do prevádzky. Druhá organizácia sa totiž bude určite vo väčšej miere potýkať s problémami ako sú:

- ťažkosti so sledovaním, riadením vývoja softvéru, ktorý existuje vo viacerých verziách
- problémy s porozumením programov „niekoho iného“
- na údržbu sú nasadení neskúsení softvéroví inžinieri
- program, ktorý vznikol dávnejšie nepoužil moderné techniky, nie je štruktúrovaný, okomentovaný
- zmeny spôsobujú nové chyby a tie nútia k ďalším zmenám
- vo všeobecnosti nízka kvalita dokumentácie
- udržiavaný softvér nemožno vysadiť z prevádzky

Ďalším faktom, prečo netreba podceňovať význam údržby, zvlášť dnes, je ten, že stále rastie pomer prostriedkov spotrebovaných na údržbu systému ku celkovým prostriedkom vyhradených pre daný systém. Kým v sedemdesiatych rokoch vývoj spotreboval asi 65% a údržba 35%, už koncom osemdesiatych rokov to bolo naopak a na prelome storočia údržba spotrebovala nie zriedka aj 70 – 80 % všetkých prostriedkov.

V neposlednom rade si je potrebné uvedomiť, že samotný vývoj SW stráca v súčasnosti svoj pôvodný

význam z dôvodu saturácie trhov a potrieb priemyslu. Softvérové organizácie, ktoré boli pôvodne zamerané na vývoj novej funkcionality, sa musia v priebehu času preorientovať na SW organizácie zamerané na údržbu existujúcich riešení u svojich klientov.

2.3 Základné typy údržby

Jedna z definícií udržovateľnosti je: Úsilie, ktoré treba vynaložiť na ďalší vývoj a údržbu výrobku podľa meniacich sa potrieb zákazníka a aj meniaceho sa okolia. S udržovateľnosťou úzko súvisí samotná údržba softvéru. Ak je dobrá udržovateľnosť, ľahšie a rýchlejšie sa vykonáva údržba. Väčšina odbornej literatúry v súvislosti s údržbou softvérových produktov rozoznáva 4 základné typy údržby:

- opravy (corrective maintenance)
- prispôsobenia (adaptive maintenance)
- zlepšenia (perfective maintenance)
- prevencia (preventive maintenance)
- predikcia (predictive maintenance)

2.3.1 Oprava

Opravy sú vlastne modifikácie, ktoré inicializovala chyba v softvéri. Chyba sa môže týkať návrhu, logiky alebo môže ísť o chybu pri programovaní.

Chyby v návrhu vznikajú napríklad nesprávnou zmenou softvéru, po nekompletnej zmene alebo zle „odkomunikovanou“ zmenou, prípadne ak bola žiadosť o zmenu zle pochopená.

Logické chyby vyplývajú z neplatných analýz a záverov, z nesprávnej implementácie konštrukčných smerníc, chybných logických postupov alebo z nekompletného testovania dát.

Chyby pri programovaní sú zapríčinené nesprávnym realizovaním podrobného logického návrhu alebo nesprávnym použitím logiky v zdrojovom kóde. Ich príčina môže byť aj v chybách pri spracovaní údajov.

Všetky tieto chyby pôvodného softvéru, ktoré niekedy nazývané aj „bugy“, bránia programu, aby vyhovoval dohodnutej špecifikácii.

V prípade, že dôjde k zlyhaniu systému zapríčinenému takouto chybou, musia byť prijaté opatrenia, ktoré obnovia chod systému. Pod tlakom manažmentu niekedy údržbár siahne ku tzv. havarijnej náprave, známej tiež ako „patch“. „Ad hoc“ povaha tohto prístupu však dáva príležitosť vzniku mnohých problémov, ktoré môžu vyústiť v ešte výraznejšie skomplikovanie programu, či ku nepredvídateľnému a náhlemu dominovému efektu, ktorý zapríčini fatálny pád celého systému. Z pohľadu reakcie je potrebné rozlišovať:

- Opravu chýb – oprava chýb počas/mimo prevádzky softvéru bez závažných dopadov na jeho používanie
- Odstraňovanie havárií – neodkladná oprava závažných chýb, ktoré bránia prevádzke softvéru

2.3.2 Prispôsobenie

Aj keď chyby, ktoré ostali v programe nespôsobujú problémy, softvér je nútený k zmenám kvôli neustále sa meniacemu prostrediu. Prispôsobenie sa chápe ako zmena, ktorá je potrebná, aby sa systém adaptoval modifikáciám, ktoré sa týkajú prostredia, v ktorom sa daný softvérový systém nachádza.

Pojem „prostredie“ v tomto prípade znamená súhrn všetkých podmienok a vplyvov, ktoré pôsobia na systém zvonku, napríklad pravidlá obchodovania, personál, obchodná stratégia, lokálne a regionálne špecifiká, legislatívne zmeny, vládna politika, pracovná schéma alebo softvérové či hardvérové platformy. Zmena niektorej časti takéhoto prostredia si preto žiada príslušnú zmenu v softvéri.

Prispôsobenie, označované aj ako adaptívna zmena, zahŕňa aj akúkoľvek prácu, ktorá je dôsledkom presunu softvéru na odlišný hardvér alebo softvér - iný kompilátor, operačný systém alebo nový procesor.

Napríklad zvýšené nároky na výkon otvárajú možnosť ku zmene obrovského množstva riadkov zo sekvenčného kódu do paralelných prostredí.

2.3.3 Zlepšenie

Sú to zmeny softvéru, vylepšujúce produkt, resp. jeho vlastnosti, ktoré neboli zahrnuté do špecifikácie pôvodného systému. Tento pojem označuje zmeny vykonané napr. z dôvodu nárastu požiadaviek na systém zo strany používateľov alebo z potreby zvýšiť obrátkovosť alebo rýchlosť transakcií, resp. výkonnosť celého systému. Softvér, ktorý má byť úspešný, musí smerovať k postupnosti zmien, ktoré odrážajú zvyšovanie požiadaviek na neho. Vychádza to z predpokladu, že softvér, ktorý sa stane užitočným, je konfrontovaný napríklad s experimentovaním s prípadmi, ktoré sú nad rámec pôvodných možností, podľa ktorých bol vytvorený. Rozširovanie požiadaviek môže byť spôsob na skvalitnenie existujúcej funkcionality.

2.3.4 Prevencia

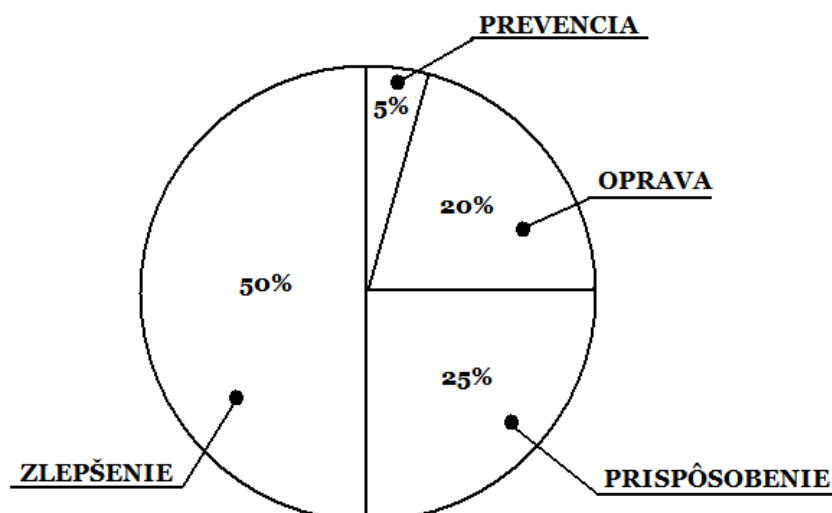
Dlhodobým efektom a dôsledkom vykonávania opráv, prispôsobovania a zlepšovania softvéru je potreba modifikovania softvéru v snahe zlepšiť ďalšiu údržbu. Preventívne zmeny sú podnikané s cieľom predísť poruchám. Takáto zmena je obvyčajne iniciovaná zvnútra organizácie, ktorá ma údržbu na starosti. Dôraz je v snahe zabezpečiť jednoduchšie pochopenie softvéru, čo v budúcnosti prispeje k ľahšej údržbe. Obvyčajne neprispievajú k zvyšovaniu základnej funkcionality. Príkladom preventívnej zmeny môže byť zálohovanie údajov, preorganizovanie kódov, optimalizácia kódov a aktualizovanie dokumentácie. Po sérii rýchlych opráv systému môže zložitosť zdrojových kódov narásť do nevládnuteľnej úrovne, ktorej jediným riešením je reorganizovanie kódov. Preventívne zásahy sú v zásade vykonávané na periodickej časovej báze.

2.3.5 Predikcia

Jedná sa o podobný typ údržby ako je prevencia – prediktívna údržba, ale nie je riadená časovými periodickými parametrami, ale konkrétnymi technickými, resp. kapacitnými ukazovateľmi – napr. dosiahnutie určitej kapacity na disku a pod. V ďalšom texte budeme prediktívnu údržbu posudzovať a analyzovať v rozsahu preventívnej údržby.

2.4 Rozdelenie údržbových aktivít

Iba 20% celkovej údržbovej činnosti pripadá na opravovanie chýb. Zostávajúcich 80% predstavuje adaptovanie existujúceho systému na zmeny vonkajšieho prostredia, práce na vylepšeniach pre používateľov a zefektívňovanie aplikácie pre budúce použitie. Pomerné percentuálne rozdelenie údržbového úsilia znázorňuje nasledujúci obrázok:



Obrázok 1: Údržbové aktivity

2.5 Motivácia k prijatiu štandardov na údržbu

V predchádzajúcich kapitolách sme sa stručne zoznámili s pojmom softvérovej údržby, objasnili sme jej význam, postavenie v procesoch životného cyklu softvéru a predstavili sme základné typy údržby. V dnešnej dobe zažívame boom firiem vyvíjajúcich softvér. Dôvody sú viac než jasné a netreba sa im bližšie venovať. Ak však spoločnosť chce obstať v tvrdom konkurenčnom boji, ktorý samozrejme v tejto sfére podnikania existuje, musí fungovať efektívne aj v oblasti údržby. Začínajúca spoločnosť musí nevyhnutne vytvoriť efektívny systém riadenia. Jedine tak totiž dokáže nemalé náklady na vývoj softvéru a ďalšie náklady spojené s prevádzkou pokryť ziskom z predaja softvéru.

Zmysel existencie každej softvérovej firmy je určený jeho podnikovou stratégiou, ktorá, okrem iného, definuje, čo je predmetom jeho podnikania, akými činnosťami sa podnik zaoberá, ako je organizovaný a riadený, aké sú jeho ciele v oblasti marketingu, obchodu, výroby, prevádzky, financií atď.

Podniková stratégia následne určuje podobu obchodných procesov, ktoré musia v podniku existovať preto, aby pomocou nich mohol podnik dosiahnuť svoje ciele určené podnikovou stratégiou.

Podnikové procesy potrebujú pre svoje zmysluplné fungovanie služby informačných a komunikačných technológií - tieto služby skrátene nazývame IT službami.

Všetky IT služby samozrejme fungujú na určitej infraštruktúre informačných a komunikačných technológií (information and communication technology = ICT), ktorá je vždy určitým spôsobom riadená - obvykle je za jej riadenie zodpovedný úsek informačných a komunikačných služieb (úsek ICT), ktorý sa stará o jednotlivé komponenty infraštruktúry.

Aj keď riadenie samotnej technickej a technologickej podstaty infraštruktúry informačných a komunikačných technológií je nevyhnutnou podmienkou správneho fungovania IT služieb, nie je podmienkou dostačujúcou - je totiž ešte potrebné určitým spôsobom riadiť aj vlastné IT služby, kvôli ktorým táto infraštruktúra existuje. Tento prístup už zvykne odborná obec označovať pojmom IT Service Management (ITSM).

Pokiaľ podnik uskutoční významné investície do drahých technológií, potom úplne oprávnené od týchto investícií očakáva ich návratnosť, ktorá sa v prípade ICT posudzuje podľa toho, aké kvalitné a

spoľahlivé služby mu tieto drahé informačné a komunikačné technológie môžu poskytnúť.

Z tohto dôvodu je nutné systematicky sa zaoberať riadením poskytovania IT služieb tak, aby výsledkom bola nákladovo optimálna dodávka IT služieb, ktorú obchodné útvary podniku budú môcť používať k tomu, aby podnik obstál v konkurenčnom prostredí a aby dosiahol zisk. Už sme vysvetlili, že práve náklady na údržbu softvéru vyvíjaného podnikom tvoria významnú časť celkových nákladov na vývoj, čo priamo súvisí s hlavnou obchodnou činnosťou softvérovej firmy.

V nasledujúcej kapitole si predstavíme najlepšie overené systematické prístupy k riadeniu vývoja softvéru a teda aj riadeniu softvérovej údržby.

Kapitola 3

Identifikácia štandardov údržby

V tejto kapitole sa budeme venovať dvom hlavným koncepciám, ktoré sa detailne zaoberajú softvérovou údržbou. Prvou z nich sú štandardy z dielne ISO a druhou je sada knižných publikácií popisujúcich spôsob riadenia IT služieb a ICT infraštruktúry známa pod skratkou ITIL.

3.1 ITIL

ITIL je skratka pre "Information Technology Infrastructure Library", v preklade "knihnica infraštruktúry informačných technológií". Referenčný rámec ITIL skutočne vznikol ako sada knižných publikácií popisujúcich spôsob riadenia IT služieb a ICT infraštruktúry na objednávku britskej vlády, ale v súčasnosti je to už medzinárodne plne akceptovaný prístup implementovaný vo všetkých priemyselných odvetviach a krajinách sveta. V súčasnej dobe sa ITIL niekedy označuje za samostatný odbor činností a podnikania, ktorý zahŕňa:

- samotnú knižnicu, ktorá vo verzii V2 obsahuje 8 titulov a vo verzii V3 obsahuje 5 nosných titulov
- súvisiace publikácie najmä vo verzii ITIL V3
- oblasť vzdelávania a certifikácie odbornej spôsobilosti
- oblasť poskytovania konzultačných služieb
- oblasť vývoja a implementácie softwarových nástrojov pre podporu procesov ITSM
- medzinárodnú platformu profesionálov a odbornej verejnosti

ITIL je rozsiahly, procesne orientovaný rámec pre oblasť IT Service Managementu, je založený na najlepších skúsenostiach (Best Practices) z praxe ITSM. To znamená, že veľa oblastí, ktoré ITIL popisuje, nepredstavuje pre ľudí z praxe nič zásadne nové alebo nič úplne neznáme. Niektoré aktivity a princípy, ktoré sú v súčasnosti už v mnohých podnikoch implementované, môžu byť

zásadám a princípom ITIL dost' podobné.

Prínos existencie knižnice ITIL spočíva okrem iného v tom, že všetky tieto skúsenosti z praxe zhrňuje do jedného uceleného a konzistentného rámca, dáva všetky ITSM procesy do vzájomných súvislostí a zavádza jednotnú a medzinárodne používanú terminológiu.

3.1.1 Vývoj ITIL

Na začiatku 80. rokov sa vo svete objavuje nový špecifický problém - narastajúca závislosť na ICT technológiách a z toho vyplývajúci rast požiadaviek na kvalitu IT služieb. Neudržateľná situácia v oblasti spoľahlivosti IT služieb viedla britskú vládu k tomu, že poverila CCTA (Central Computer and Telecommunications Agency = britská vládna agentúra) spracovaním príručky, podľa ktorej by organizácie (ako vládne, tak aj súkromné), participujúce na dodávkach IT služieb pre britskú vládu, museli záväzne postupovať. CCTA postupne vydáva 46 zväzkov zhrňujúcich najlepšie skúsenosti (Best Practices) z oblasti riadenia IT služieb a infraštruktúry pre potreby britských vládnych úradov a podnikateľských subjektov dodávajúcich IT služby vláde. Vychádza ITIL vo verzii 1.

Začiatkom 90. rokov vzniká OGC (Office of Government Commerce) zlúčením 3 britských vládnych agentúr vrátane CCTA a stáva sa autoritou pre re-edície a vydávanie ďalších publikácií ITIL, taktiež vzniká itSMF (IT Service Management Forum) a stáva sa medzinárodnou komunitou profesionálov a odbornej verejnosti z oblasti ITSM a ITIL. V tomto období preberajú rámec ITIL aj ďalšie subjekty (súkromného i verejného sektoru) a začína vydávanie prvých certifikátov odbornej spôsobilosti pre oblasť ITSM podľa ITIL

Na prelome storočia je pôvodných 46 zväzkov knižnice prepracovaných do novej podoby. Základom knižnice sú tituly Service Support a Service Delivery, ktoré obsahujú podstatnú časť z pôvodných 46 zväzkov a v podstate definujú oblasť ITSM. Sú vydávané ďalšie diely knižnice ITIL. Vzniká ITIL vo verzii 2. ITIL je už celosvetovo rozšírený a vďaka tomu je považovaný za medzinárodný štandard v oblasti ITSM. ITIL je už prakticky samostatným odborom činnosti a podnikania.

V novembri 2004 OGC zahajuje práce už na tretej aktualizácii knižnice ITIL, v decembri 2005 je zverejnená medzinárodná norma pre oblasť ITSM – ISO/IEC 20000 a v roku 2006 začína celosvetové rozšírenie normy ISO/IEC 20000. V máji 2007 vychádza tretia verzia ITIL v3.

3.1.2 Obsah ITIL

ITIL popisuje definíciu procesov potrebných pre zaistenie ITSM a zásady pre implementáciu procesov.

Definícia každého procesu zahŕňa:

- stanovenie cieľov, vstupov, výstupov a aktivít každého procesu
- stanovenie úloh a zodpovedností v danom procese
- spôsob merania kvality a účinnosti procesov
- vzájomné väzby medzi jednotlivými procesmi

ITIL vo verzii 2 obsahuje osem titulov. Dve základné a najznámejšie publikácie z knižnice ITIL sú Service Support a Service Delivery. Tieto dve knihy tvoria rámec ITSM, tzn. sú to knihy o riadení, dodávaní a podpore IT služieb. Obsahujú popis 10 základných procesov ITSM a funkcie Service Desku.

Na úrovni operatívneho (prevádzkového) riadenia prevládajú procesy charakteru každodennej, rutínnej prevádzky, tieto sú popísané v knihe Service Support, ich cieľom je každodenná podpora užívateľov a údržba IT služieb. Na úrovni taktického riadenia prevládajú procesy charakteru dlhodobého plánovania, ktoré sú popísané v knihe Service Delivery. Ich cieľom je budovanie vzťahov so zákazníkmi a dosiahnutie ich dlhodobej spokojnosti s poskytovanými IT službami.

Z tohto rozdelenia vyplýva, že našu pozornosť sústredíme hlavne na podporné procesy, teda na knihu Service Support. ITIL totiž veľmi zriedkavo používa termín „údržba“ v takom zmysle ako napríklad ISO 14764, ale úplne ekvivalentne používa termín IT service (IT služba). Pre filozofiu ITILu je príznačný pojem support, čiže podpora. Toto všeobecné označenie zahŕňa veľké množstvo procesov a aktivít. Medzi ne patrí aj manažment incidentov (Incident Management), manažment konfigurácií (Configuration Management), manažment problémov (Problem Management), manažment zmien (Change Management), manažment verzií (Release Management), Help Desk a procesy a činnosti v nich definované.

3.1.3 Service desk

ITIL má za to, že väčšina „problémov“, ktoré sa týkajú podpory a údržby začína tým, že „užívateľ má problém“ (prípadne sťažnosť alebo otázku) a potrebuje to rýchlo vyriešiť. Užívatelia teda musia vedieť, na koho sa majú v takom prípade obrátiť. ITIL hovorí, že v organizáciách má fungovať centrálné miesto, ktoré bude slúžiť práve k tomuto účelu, teda ku kontaktu s užívateľom. Ich názvy sa rôznia, najpoužívanejšie sú Help Desk, Call Centre, Service Desk alebo Customer Hot line.

Zo strategického hľadiska je Service Desk najdôležitejším miestom v organizácii. Pre zákazníka a užívateľa je Service Desk jediným oknom do organizácie. Užívatelia a zákazníci merajú úroveň poskytovaných služieb, profesionalitu a odbornosť pracovníkov organizácie ich pohľadom na fungovanie služby Service Desk a nie podľa technických aspektov a hoci aj špičkových parametrov infraštruktúry. Práve na tomto mieste, teda podnetom od užívateľov začína životný cyklus žiadosti o zmenu, resp. údržbársky zásah.

Ciele a hlavné aktivity služby Service Desk:

- zaisťovať na dennej báze aktívny kontakt medzi zákazníkmi, užívateľmi, pracovníkmi vlastnej organizácie a pracovníkmi externej podpory, čiže slúžiť ako "Single Point of Contact" pre užívateľov a zákazníkov
- zaisťovať obnovu štandardnej dodávky služby s minimálnym dopadom na zákazníkov
 - plniť úlohu 1. úrovne podpory v procese manažmentu incidentov
 - koordinovať 2. a 3. úroveň podpory v procese manažmentu incidentov
 - riadiť životný cyklus požiadaviek

3.1.4 Manažment incidentov

Cieľom manažmentu *incidentov* je obnoviť normálnu prevádzku služby a to čo najrýchlejšie, pri súčasnej minimalizácii dôsledkov výpadku služby na prevádzku (tzn. na zákazníkov a užívateľov). Vysvetlenie pojmu incident ako i mnohých ďalších pojmov nájdeme v slovníku v závere tejto práce.

Manažment incidentov je zodpovedný za včasnú detekciu incidentov, ich zaznamenávanie a riadenie ich životného cyklu. Jeho cieľom je "byť čo najrýchlejší". Manažment incidentov v zásade vôbec neskúma PREČO k incidentom dochádza, ale hľadá akékoľvek riešenie vedúce k obnoveniu služby (za hľadanie príčin incidentov je zodpovedný manažment problémov).

Vstupy:

- detaily incidentu získané (napríklad) zo Service Desku

- detaily konfigurácie z *databázy manažmentu konfigurácií (CMDB)*
- odozva incidentu z porovnania so *známymi problémami a chybami (Known Errors)*
- detaily riešenia (zo služby Service Desk)

Úlohy:

- zaznamenanie a vyšetrovanie incidentu
- klasifikácia incidentu a zahájenie pomoci
- skúmanie incidentu a diagnostikovanie
- vyriešenie incidentu a obnovenie služby
- uzavretie incidentu

Výstupy:

- *Žiadosť o zmenu (Modification request MR)*
- aktualizovaný záznam o incidente (vrátane popisu riešenia a popisu Work-arounds)
- vyriešený a uzavretý incident
- odkomunikovanie incidentu so zákazníkom
- hlásenie pre manažment (report)

3.1.5 Manažment problémov

Cieľom manažmentu problémov je zabrániť opakovaniu incidentov súvisiacich s poruchami. Dosahuje to tým, že analyzuje incidenty, snaží sa nájsť ich základnú príčinu a následne iniciuje kroky vedúce k náprave. Tým minimalizuje dopad incidentov a problémov spôsobených poruchami IT infraštruktúry. Jeho ďalšou úlohou je starať sa o znalostnú databázu a informácie v nej uložené sprístupňovať špecialistom podpory v procese manažmentu incidentov a pracovníkom Service Desku. Manažment problémov svojou činnosťou zaisťuje stabilitu infraštruktúry.

Vstupy:

- detaily incidentu z manažmentu incidentov
- detaily konfigurácie z databázy manažmentu konfigurácií (CMDB)
- akékoľvek použité zásahy typu Work-around

Úlohy:

- kontrolovať a riadiť riešenie problému
- riadenie a kontrola riešenia známych chýb (Known-errors)
- proaktívna prevencia problémov
- vytváranie záznamov o vážnych problémoch

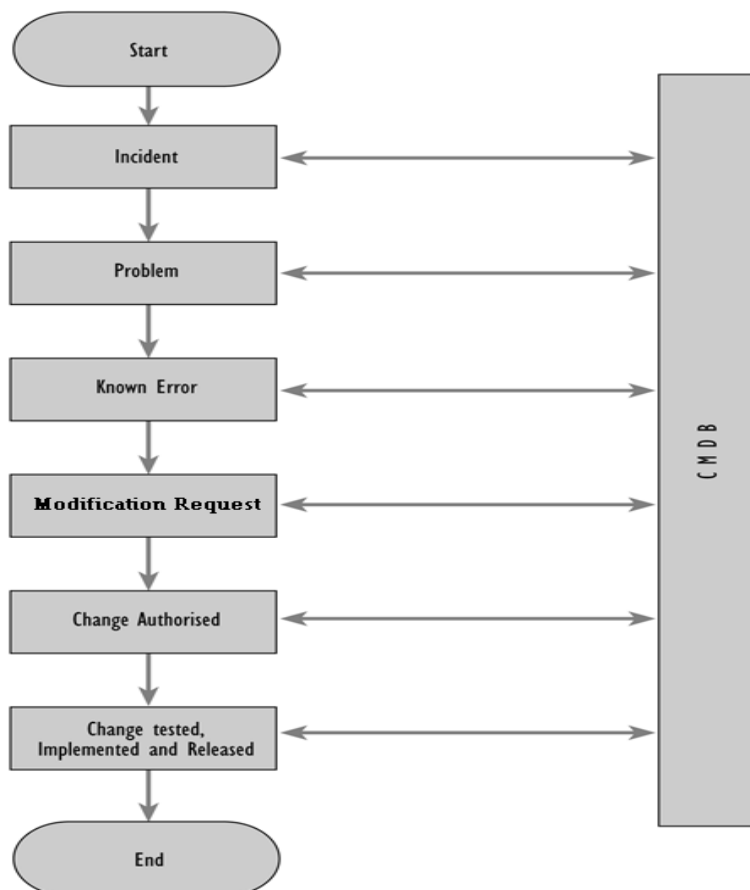
Proaktívna prevencia problémov je sústava činností, ktorými sa snažíme odhaliť nedostatky a chyby aplikácie skôr, ako vyvolajú incident a problém (súvisí s preventívnou údržbou).

Výstupy:

- Žiadosť o zmenu (MR)
- Known-error
- aktualizovaný záznam o probléme (vrátane riešenia a/alebo dostupného Work-around)
- uzavretý záznam o probléme pre vyriešené problémy
- upovedomenie manažmentu incidentov o probléme a o Known-errors

3.1.6 Manažment konfigurácii

Manažment konfigurácii poskytuje logický model infraštruktúry a služieb, identifikuje, zabezpečuje kontrolu a údržbu všetkým konfiguračným položkám a stará sa o konfiguračnú databázu CMDB. Podporuje ostatné procesy poskytovaním vierohodných informácií o konfiguračných položkách infraštruktúry a o ich dokumentácii.



Obrázok 2: Vzťahy CMDB k ostatným procesom

3.1.7 Manažment zmien

Manažment zmien zodpovedá za riadenie obehu žiadosti o zmenu (MR), za schvaľovanie a plánovanie zmien a za koordináciu implementácie zmien. Z pohľadu údržby je preto tento proces najdôležitejší. Preto sa manažmentu zmien budeme venovať detailnejšie a popíšeme schému jeho fungovania.

Vstupy:

- žiadosti o zmenu
- Databáza manažmentu konfigurácii (CMDB)

Úlohy:

- filtrovanie zmien
- riadenie zmien a vedenie procesu riešenia zmien
- riadenie komisií CAB a CAB/EC
- posudzovanie a uzatváranie zmien
- vypracovanie reportov

Výstupy:

- žiadosti o zmenu (aktualizované)
- zápisy zo zasadnutí CAB
- hlásenia manažmentu zmien (reporty)

Koncept manažmentu zmien

Poňatie manažmentu zmien v ITIL je procesne orientované, sú to ale viac manažérske procesy ako technické (na rozdiel napr. od manažmentu konfigurácii). Proces riešenia žiadosti o zmenu od jej prijatia manažmentom zmien po jej úspešné zavŕšenie je veľmi zložitý a prechádza veľkým množstvom fáz. Je však dôležité porozumieť princípu tohto procesu, pretože práve v ňom je podstata riešenia zmien a teda aj prístupu k údržbe podľa ITILu. Preto sa pokúsime jednotlivé fázy výstižne opísať. Na lepšie pochopenie nám dobre pomôžu priložené obrázky, ktoré zachytávajú časti diagramu opisovaného procesu. Na obrázku 3a vidíme prvú polovicu procesu. Obrázok 3b zachytáva druhú polovicu. Na obrázku 3c je zobrazený postup u tzv. štandardného problému. Na koniec, obrázok 3d demonštruje systém riešenia kriticky dôležitých žiadosti o zmenu.

Zaznamenávanie a filtrovanie zmien

Každá žiadosť o zmenu musí byť zaznamenaná a každý užívateľ musí mať právo podať žiadosť o zmenu. Avšak žiadosť by mala obsahovať tzv. schvaľovací podpis senior užívateľa alebo manažéra. V tejto fáze manažment zmien rýchlo posúdi opodstatnenosť žiadosti, aby vyfiltroval úplne neoprávnené a zlé žiadosti. Tie sa vrátia zadávateľovi so stručným zdôvodnením zamietnutia jeho žiadosti.

Posúdenie priority

Každá žiadosť musí byť posúdená a je jej priradená priorita v závislosti od závažnosti incidentu alebo problému. Posúdenia priority žiadosti sa má zúčastniť aj jej zadávateľ. Ak očividne nejde o vážny problém, CAB sa nemusí zúčastňovať tohto posúdenia. Podľa priority sa potom ďalej určí, ktoré zmeny sa začnú realizovať ako prvé a ktoré až neskôr. V prípade veľmi závažných problémov musí urýchlene zasadiť CAB alebo dokonca iba krízová komisia CAB/EC.

Kategorizácia zmeny

Musíme veľmi dôsledne posúdiť mieru rizika ohrozenia obchodnej činnosti. K tomu slúžia vopred zadané kategórie dôležitosti zmeny. Každá zmena musí byť zhodnotená a musíme určiť, do ktorej kategórie spadá. To určí ďalší postup v riešení žiadosti o zmenu. Chceme zdôrazniť, že toto posúdenie skúma dopad zmeny na organizáciu v závislosti od množstva zdrojov požadovaných na jej realizovanie. (Neskôr bude totiž žiadosť analyzovaná ešte raz, avšak s inými kritériami.) V zásade by sa mali rozdeliť žiadosti na žiadosti s nízkym dopadom, s veľkým dopadom a s kritickým dopadom. Systém posudzovania by mal byť nastavený tak, aby drvivá väčšina žiadostí patrila do jednej z prvých dvoch skupín. Priradenie kategórie danej zmene zároveň určí, kto bude zodpovedný za plánovanie a riadenie tej ktorej zmeny. U zmien s najväčším dopadom to musí byť CAB, ale u nepodstatných to môže byť iba istá poverená osoba. Môže však nastať ešte jeden prípad, ktorý spôsobí, že zmena nezapadne do žiadnej zo spomínaných kategórií. Posúdením totiž môžeme zistiť, že sa jedná o relatívne častú zmenu, pre ktorú už bolo prijaté riešenie, teda ide o tzv. štandardnú zmenu. To znamená, že nie je nutné, aby táto žiadosť ďalej prechádzala opisovaným systémom, nakoľko pre ňu už máme pripravený model, (predpripravené riešenie problému), ktorý sa na ňu môže hneď aplikovať a riešenie sa môže začať implementovať.

Zasadnutia CAB

Pravidelne, napríklad raz za 6 mesiacov, by malo prebehnúť zasadnutie CAB, ktoré sa bude venovať

rôznym témam, no v prvom rade kontrole priebehu dôležitých zmien.

Schválenie žiadosti o zmenu

CAB ešte raz veľmi komplexne posúdi žiadosť o zmenu berúc v úvahu prioritu žiadosti, ale aj vplyv žiadosti na všetky ostatné organizačné zložky, na poskytované služby a vydá rozhodnutie, či navrhuje schválenie žiadosti o zmenu. Vyjadrenie CAB má totiž iba odporúčaciu hodnotu. Definitívne rozhodnutie má byť v kompetencii riaditeľa IT alebo manažéra zmien alebo nimi poverených osôb. Pre navrhovanú zmenu je nutné získať finančné, technické a obchodné schválenie.

Časový plán

Jednotlivé zmeny môžu spolu súvisieť, jedna môže vyvolať vznik ďalších alebo jedna zmena môže predstavovať viacero menších zmien. Pri vytváraní časového plánu (change scheduling) treba brať ohľad aj na manažment vydání. Ak je to možné, zmeny by mali byť vydávané v balíkoch. A práve na to by sa mal brať ohľad pri vytváraní časového plánu.

Tvorba riešenia, testovanie a implementácia

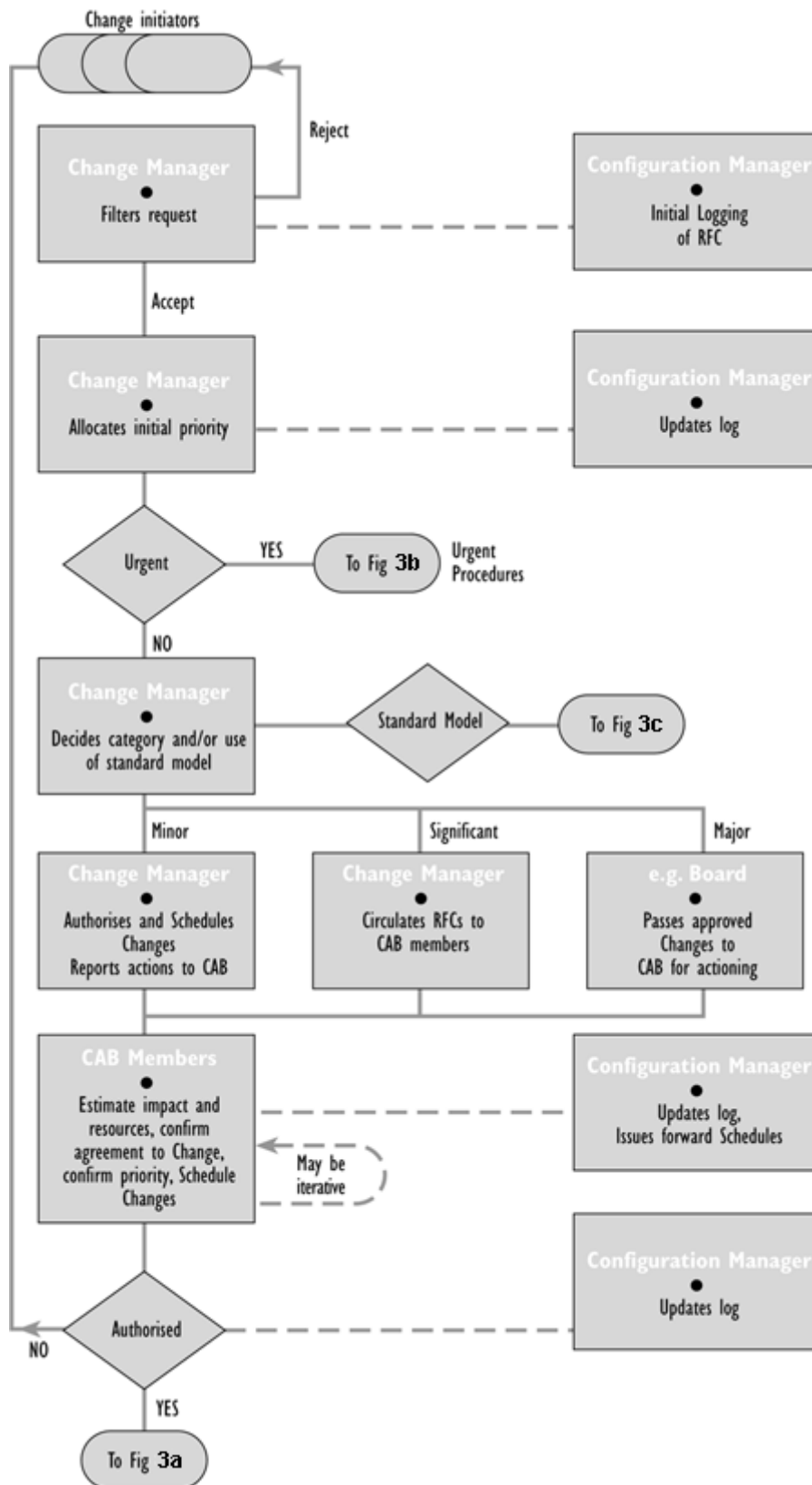
Schválené žiadosti o zmenu sú posunuté technickým skupinám na realizáciu a testovanie zmien. Testovanie má byť dôkladné a vyčerpávajúce, má zahŕňať aj testy na výkon, vonkajšiu bezpečnosť, dostupnosť a funkčnosť. Zodpovednosť za dodržanie časového plánu za tieto činnosti je opäť na manažmente zmien, aj keď samotnú realizáciu zmeny vykonáva iná zložka.

Urgentné zmeny

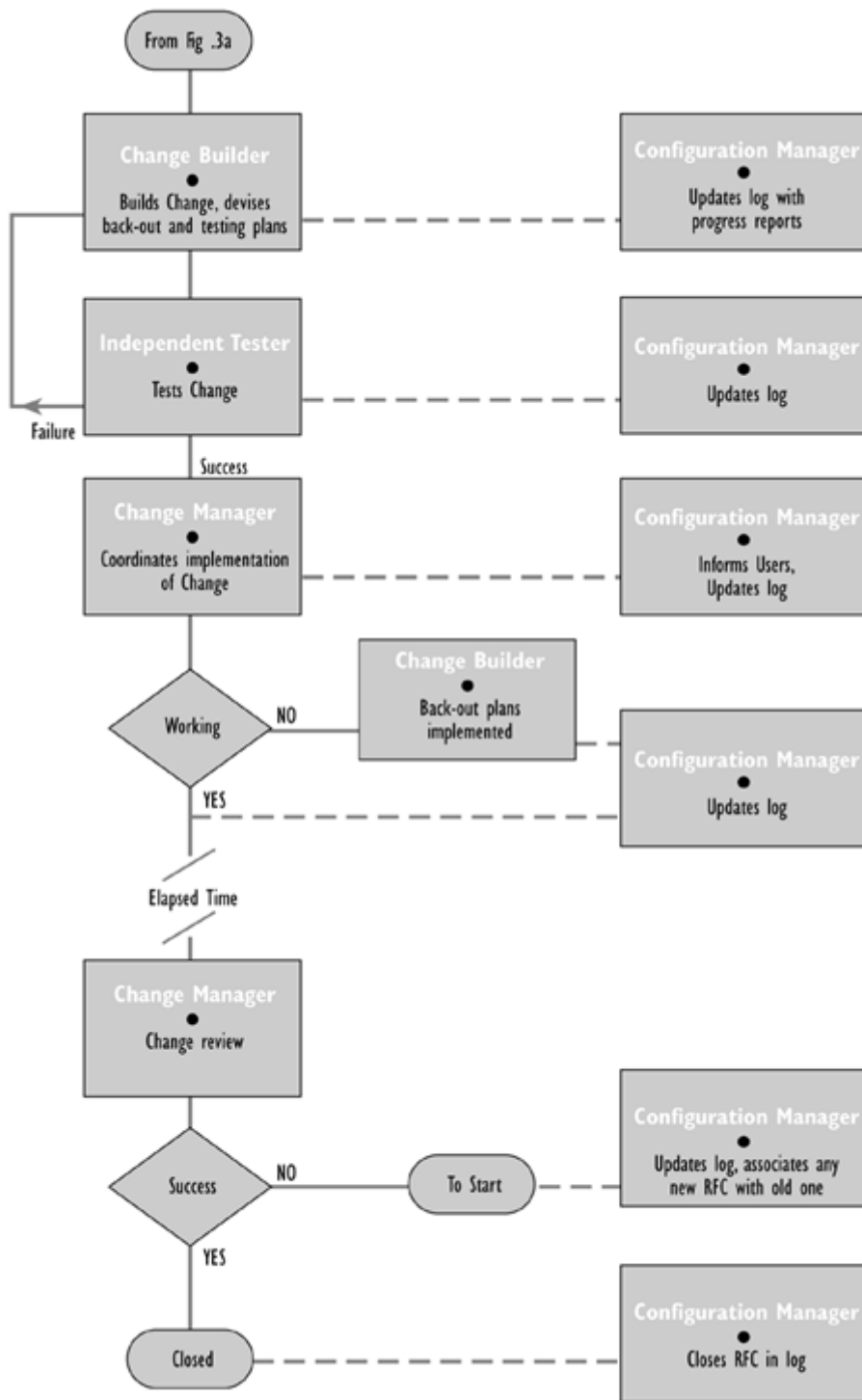
Skôr ako sa urgentná zmena začne riešiť, musí sa manažér zmien so zodpovedným technickým manažérom uistiť, že majú dostatok personálu a zdrojov, „aj keby ich mali vytiahnuť z postele“. Súčasne by sa mal začať vyvíjať aj „plán B“, respektíve zvážiť možnosti, ako vycúvať z neúspešného pokusu o riešenie urgentnej zmeny. S ohľadom na časové možnosti by sa tak zmena mala podrobiť dôslednému testovaniu, v maximálne možnom rozsahu. Ak to vyslovene nie je nevyhnutné, vôbec neotestované zmeny by sa implementovať nemali, ev. iba v skutočne výnimočných situáciách. Môže sa stať, že zrealizovanie urgentnej zmeny nevyrieši problém a teda pokus o riešenie zlyhá. V takom prípade je potrebné pokúšať sa opakovane o nové riešenie. Treba dohliadať aj na to, aby neúspešné zásahy boli promptne odstránené a dané do pôvodného stavu. Manažment zmien je zodpovedný za to, že sa stále hlavný dôraz kladie na obchodné potreby spoločnosti.

Zhodnotenie zmeny

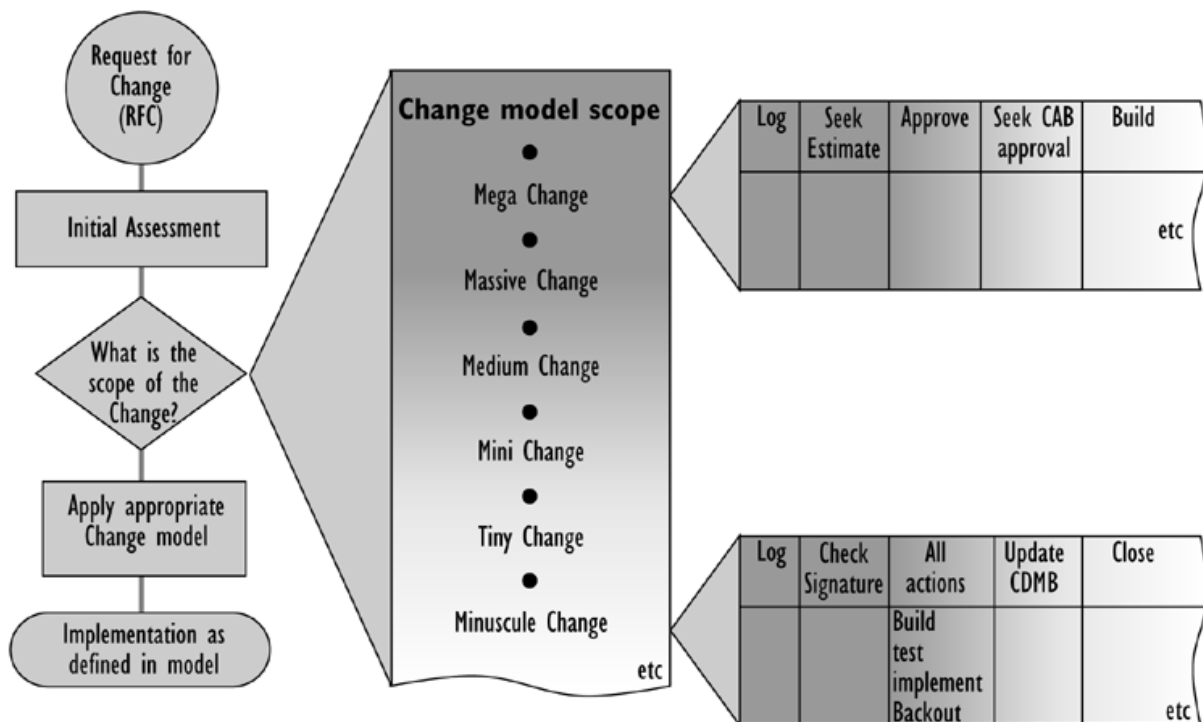
Po uplynutí vopred stanoveného času musia byť všetky implementované zmeny zhodnotené (change review). K tomu si manažment zmien môže opäť prizvať CAB alebo posúdenie môže prebiehať v rámci pravidelných CAB zasadnutí.



Obrázok 3a: schéma manažmentu zmien 1. časť



Obrázok 3b: schéma manažmentu zmien 2. časť

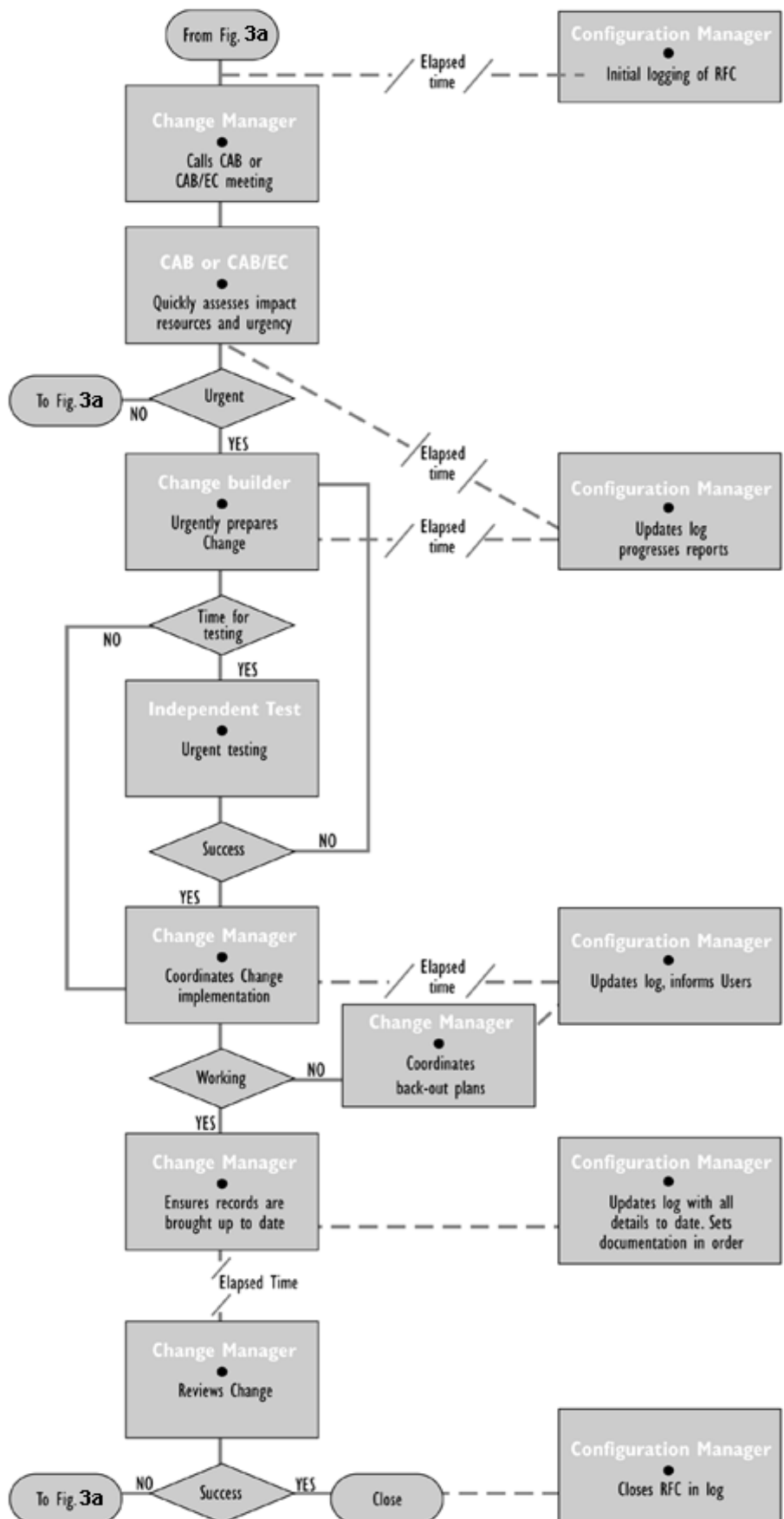


Obrázok 3c: schéma manažmentu zmien – štandardný problém

3.1.8 Manažment verzii

Verzia (vydanie) predstavuje súbor autorizovaných zmien zložený z nových alebo zmenených konfiguračných položiek, ktoré sú otestované a nasadené ako jeden celok do produkčného prostredia. Pojem *release* predstavuje tiež činnosť nasadzovania jednotlivých zmien. Proces manažmentu verzii zabezpečuje po organizačnej aj výkonnej stránke úspešné nasadenie zmien v podobe novej verzie do produkčného prostredia. Úlohy manažmentu verzii sú:

- plánovať a dohliadať na úspešné uvoľnenie softvéru i súvisiaceho hardvéru
- navrhovať a implementovať účinné procedúry na distribúciu a inštalovanie zmien do IT systému
- uistiť sa, že iba oprávnené a otestované zmeny sú inštalované
- komunikácia s používateľmi počas plánovania a uvoľnenia nových verzii.
- implementovanie nových vydaní v spolupráci s manažmentom zmien a manažmentom konfigurácií
- dohľad nad bezpečným uložením originálnych kópii konfiguračných prvkov a dohľad nad aktualizovaním CMDB



Obrázok 3d: schéma manažmentu zmien – urgentná zmena

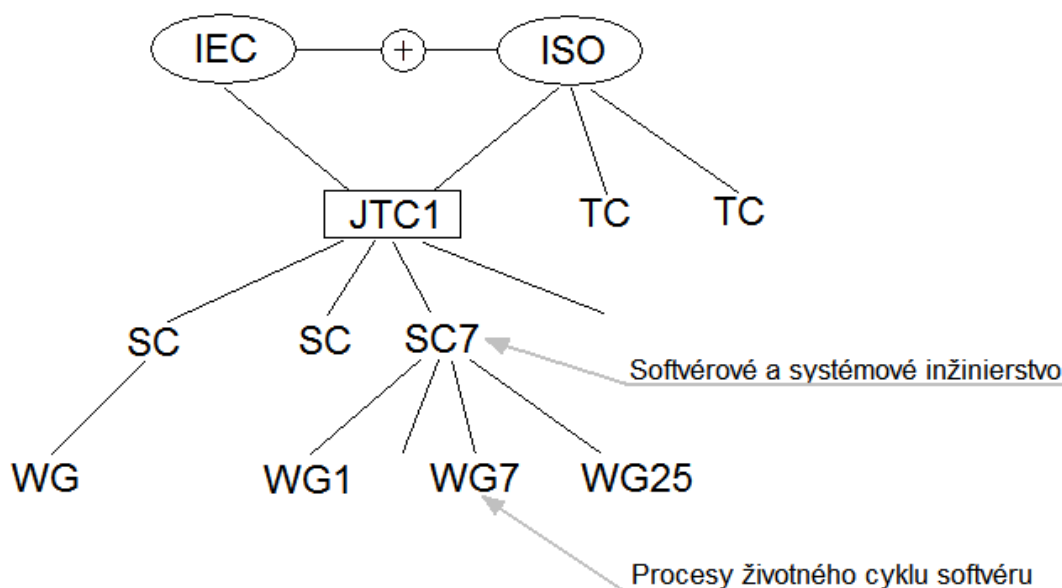
3.2 International Organization for Standardization

Počiatky noriem ako takých by sa dali datovať do roku 1926, keď bola založená organizácia ISA – International Federation of the National Standardizing Associations. Pôvodne bola určená pre strojársku oblasť, ale neskôr sa rozšírila aj do iných odvetví, existovala do roku 1942. Po skončení druhej svetovej vojny bola takmer celá Európa zdevastovaná a bolo potrebné zaviesť program čo najrýchlejšej obnovy priemyslu, a tým aj ostatných odvetví spoločnosti. Aj preto sa v roku 1926 v Londýne stretli zástupcovia 25 štátov, aby dali základ pre vznik organizácie, ktorá by sa starala o zjednotenie priemyselných štandardov týchto štátov, ktoré boli dovtedy rozdielne, a aby tak zabezpečil čo najrýchlejší rozvoj priemyslu a základ pre kooperáciu jednotlivých štátov, ktoré tieto štandardy prijali. Táto organizácia – ISO – oficiálne vznikla 23.2.1947 a odvtedy sa nepretržite rozvíja. Jej označenie v skutočnosti nie je skratkou názvu International Organization for Standardization, ako si väčšina ľudí myslí, ale bol odvodený od gréckeho slova ἴσος (isos), ktoré znamená *rovnaký* a odkazuje na cieľ organizácie - štandardizáciu. V súčasnej dobe je ISO najväčším svetovým tvorcom štandardov, jej členom je vyše 140 národných štandardizačných komisií, pričom každá národná komisia má svojho zástupcu v Ženevskom sídle organizácie.

3.2.1 Štruktúra ISO

Technické normy sú dokumenty, ktoré obsahujú technické parametre alebo iné stanovené kritériá, ktorých cieľom je zaručiť vhodnosť daných materiálov, výrobkov alebo služieb pre požadovaný účel. Technické normy zvyšujú efektívnosť výroby, spoľahlivosť a efektívnosť tovarov a služieb, ochraňujú záujmy spotrebiteľov akceptovaním úrovne kvality, zvyšujú kompatibilitu tovarov a služieb.

Svetovú a európsku normalizačnú činnosť zastrešujú organizácie ISO, IEC, CEN a CENELEC. ISO je celosvetová organizácia zabezpečujúca medzinárodnú normalizačnú činnosť, ktorá pokrýva svojimi aktivitami technickú normalizáciu vo všetkých oblastiach okrem elektrotechniky, elektroniky a telekomunikácií. Túto zvyšnú oblasť spravuje Medzinárodná elektrotechnická komisia IEC (International Electrotechnical Commission). Národné orgány, ktoré sú členmi ISO alebo IEC, sa spolupodieľajú na vývoji medzinárodných štandardov prostredníctvom technických komisií zriadených organizáciami zodpovedajúcimi za riešenie konkrétnych oblastí technickej aktivity. Technické komisie ISO a IEC spolupracujú na poli vzájomného záujmu. Na tejto práci sa v spojení s ISO a IEC spolupodieľajú aj ďalšie medzinárodné organizácie, tak vládne, ako i mimovládne. Na poli informačných technológií ISO a IEC zriadili spoločnú technickú komisiu (Joint Technical Commission) ISO/IEC JTC 1. Hlavnou úlohou spoločnej technickej komisie je zostavovať medzinárodné štandardy. Návrhy medzinárodných štandardov, prijaté touto komisiou, sú rozosielené národným orgánom na odsúhlasenie hlasovaním. Vydanie medzinárodného štandardu vyžaduje schválenie aspoň 75% národných orgánov zúčastnených na hlasovaní. Spojené technické komisie sa ďalej delia na množstvo subkomisií (SC). Jednou z nich je ISO/IEC JTC1 SC7, ktorá sa zaoberá softvérovým a systémovým inžinierstvom. Jej činnosť je rozdelená do pracovných skupín. Procesom životného cyklu softvéru sa venuje pracovná skupina (WG) s označením ISO/IEC JTC1 SC7 WG7.



Obrázok 4: štruktúra ISO

3.2.2 ISO normy zaoberajúce sa softvérom

Pôvodným cieľom ISO noriem bolo vytvorenie technických štandardov, ktoré mali uľahčiť spoluprácu a zvýšenie kompatibility v činnostiach spoločností, ktoré tieto štandardy prijali. Do veľkej miery sa však stávajú smerodajnými aj v ekonomických, či sociálnych oblastiach spoločností. Tým, že normy ISO stanovujú hranice, v ktorých by sa mali produkujúce spoločnosti nachádzať v rámci produkcie svojich výrobkov, chránia tak celý trh pred nevhodnými výrobkami, či už nepoužiteľnými alebo potenciálne nebezpečnými. ISO normy majú teda v princípe vplyv na všetky časti spoločnosti, pretože vplývajú na zákazníkov, vlády, výrobcov i na obchodníkov.

V oblasti informatiky, konkrétne v oblasti softvérovej údržby, má ISO pozíciu lídra na tomto poli. Dôsledkom toho je skutočnosť, že jeden z hlavných prístupov k softvérovej údržbe, ktorý je predmetom analýzy tejto práce, pochádza práve z dielne ISO.

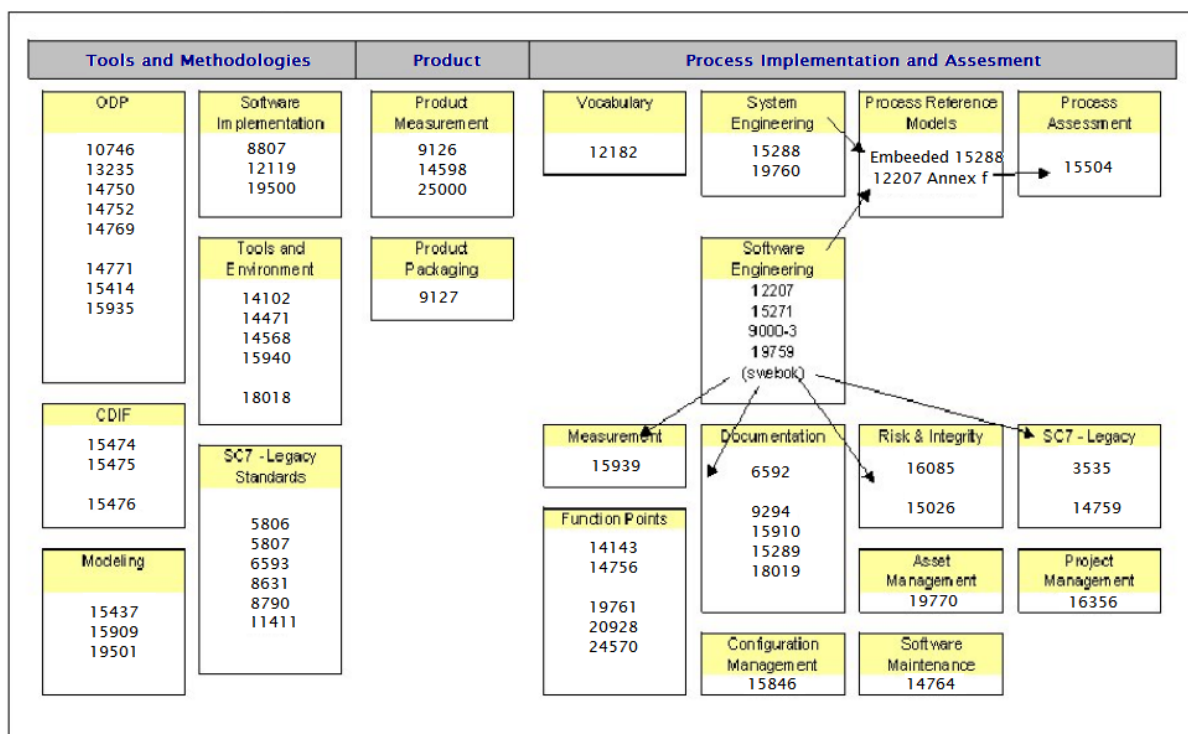
V súčasnej dobe normy ISO pokrývajú širokú škálu aktivít spojenú s vývojom softvéru. Základné normy ISO môžeme rozdeliť do troch skupín záujmov:

- Metodológie a prostriedky
- Produkty
- Implementácia a ohodnotenie procesov

Normy v skupine „Metodológie a prostriedky“ sú venované popisovaniu spôsobov správneho modelovania softvéru, správnej implementácie alebo napríklad vhodným prostriedkom a prostrediam na vývoj softvéru. Takisto zachováva aj štandardy, ktoré sa postupom času stali zastaranými.

Druhá časť „Produkty“ sa venuje meraniam vlastností softvérového produktu ako aj správne baleniu produktov.

Posledná skupina „Implementácia a ohodnotenie procesov“ je najširšia časť noriem venovaných vývoju softvéru. Tieto štandardy ponúkajú popis procesov a postupov pre zostrojovanie softvéru či systémov, pričom sa prihliada na všetky súčasti, ako správna dokumentácia produktov, manažovanie riskov, či integrity, správne manažovanie zdrojov, či projektu, ako aj údržba softvéru a iné súčasti. Jednotlivé väzby medzi normami sú zobrazené na obrázku 5.



Obrázok 5: prehľad ISO/IEC noriem

3.3 Medzinárodný štandard ISO/IEC 12207

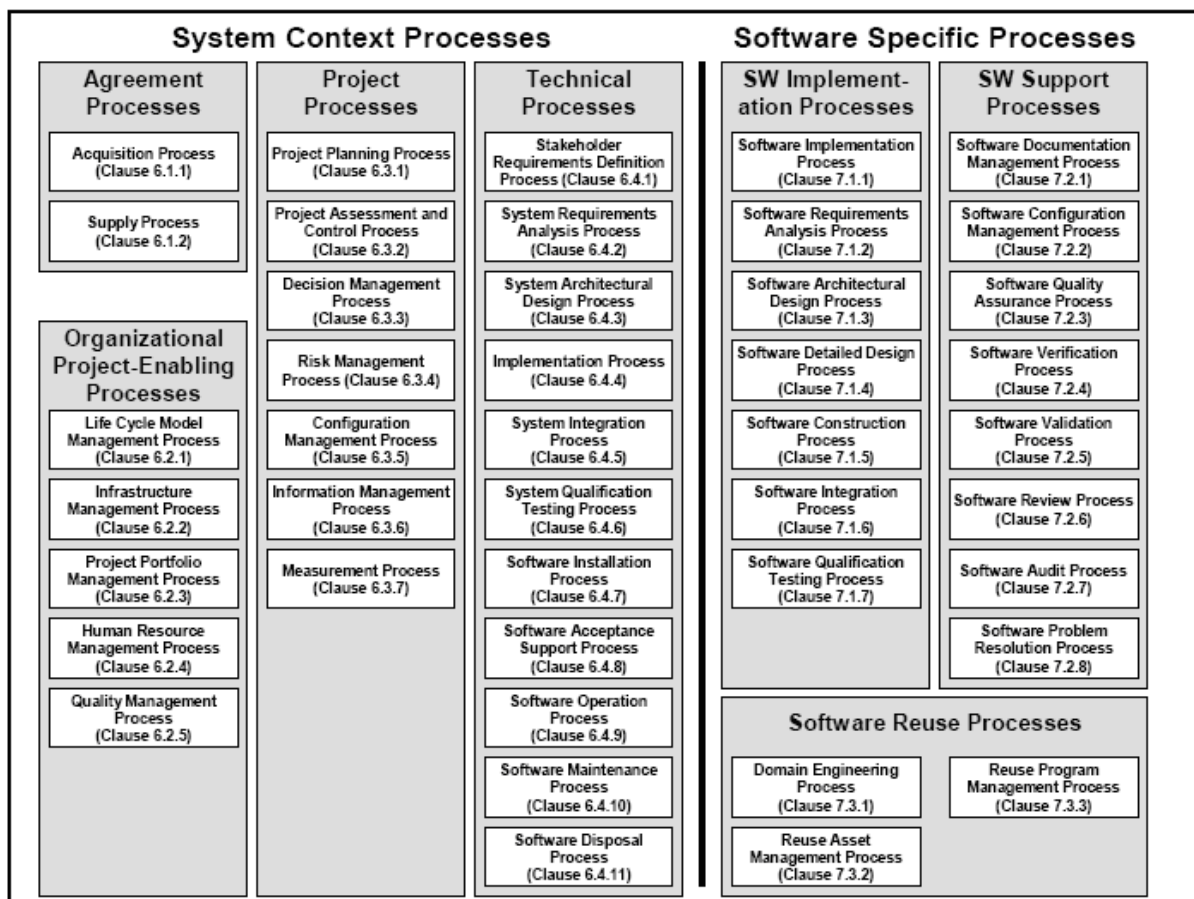
Norma ISO/IEC 12207 bola prvý krát vydaná v roku 1995 a predstavovala v tom čase prvý medzinárodne uznávaný štandard, ktorý poskytoval komplexný opis procesov, činností a úloh, ktoré sa vyskytujú v rámci celého životného cyklu softvéru, t.j. od momentu pretvorenia myšlienok do konkrétnej koncepcie vývoja softvéru až po skončenie používania softvéru. Uvedená norma prvý krát v histórii bola schopná definovať všeobecné zásady použiteľné pre softvérových inžinierov pri tvorbe a ďalšom riadení softvéru.

Podľa tejto normy sú činnosti a úlohy počas životného cyklu softvéru zoskupené do sústavy procesov, resp. do procesného modelu (konkrétny proces sa člení na súbor činností a každá činnosť sa člení na súbor úloh). Procesný model životného cyklu softvéru sa postupne vyvíjal a norma sa v ďalších vydaniach zameriava najmä na bližší opis činností a úloh v rámci konkrétnych procesov. Norma ISO/IEC 12207 sa dočkala v roku 2007 ďalšej revízie, ktorá čiastočne zmenila architektúru procesného modelu pre životný cyklus softvéru tým, že boli do modelu doplnené niektoré procesy a takisto niektoré pôvodné procesy boli premenované alebo rozšírené. Tiež bola definovaná nová úloha procesného modelu pre životný cyklus softvéru z toho dôvodu, že sa tento stal referenčným modelom pre hodnotenie spôsobilosti softvérových procesov, a takto úzko nadväzoval na vývoj iných súvisiacich noriem (najmä však noriem radu ISO/IEC 15504 zaoberajúcich sa hodnotením spôsobilosti procesov). ISO/IEC 12207 tvorí kompaktný dokument, ktorý má ambíciu definovať procesný model nevyhnutný na implementáciu dobrých praktík softvérového inžinierstva a stanoviť referenčný model alebo tzv. ideálnu scénu pre vykonávanie profesie softvérového inžinierstva, pomocou ktorých bude možné hodnotiť úroveň spôsobilosti procesov v organizácii zaoberajúcej sa vývojom, dodávkou a údržbou softvéru. Paralelne na vývoji tejto normy prebiehajú aj práce na tvorbe inej dôležitej normy ISO/IEC 15288, ktorá s ňou veľmi úzko súvisí. Táto norma dáva odpoveď na otázku ako správne a podľa akých dobrých praktík definovať, riadiť a neustále zlepšovať procesy životných cyklov

všeobecných systémov vytvorených ľuďmi – buď v rámci celej organizácie alebo v rámci konkrétneho projektu – ktorých súčasťou vo väčšine prípadov je aj softvér. Uvedená norma opisuje globálne uznávaný procesný model životného cyklu všeobecných systémov a je veľmi dôležitým dokumentom pre systémových inžinierov.

Pri bližšom pohľade na procesy životného cyklu softvéru budeme vychádzať z procesného modelu, resp. sústavy procesov uvedených vo verzii tejto normy ISO/IEC 12207(2007), ktorá predstavuje momentálne najaktuálnejší model. Procesy v rámci tejto sústavy sú rozdelené do 7 základných skupín:

- Zmluvné procesy (Agreement Processes) - 2 procesy
- Organizačné procesy (Organizational Project-Enabling Processes) - 5 procesov
- Projektové procesy (Project Processes) – 7 procesov
- Technické procesy (Technical Processes) – 11 procesov
- Procesy softvérovej implementácie (Software Implementation Processes) – 7 procesov
- Podporné softvérové procesy (Software Support Processes) – 8 procesov
- Procesy na znovupoužitie softvéru (Software Reuse Processes) - 3 procesy



Obrázok 6: Schéma skupín procesov štandardu ISO 12207

Tento referenčný model procesov sa nezaobera konkrétnymi spôsobmi ako implementovať tieto procesy, ani sa nesnaží vnútiť niektorý z modelov životného cyklu softvéru, metodiku alebo techniku. Tento model má byť prijatý organizáciou v súlade s jej produkčnými potrebami a aplikačnou doménou. Hlavný vplyv na konkrétnu podobu procesu, ktorý organizácia prijme do projektu, majú požiadavky užívateľov.

Výstupy procesov ukazujú, či sa podarilo dosiahnuť cieľ procesu. Tým napomáhajú v posudzovaní

vypelosti procesov, ktoré organizácia implementovala a poskytuje zdrojový materiál na plánovanie ďalšieho rozvoja procesov v organizácii.

Tento štandard popisuje procesy rozdelené do dvoch základných podskupín:

- Procesy, ktoré sa týkajú nezávislého softvérového produktu, služby alebo softvérového systému
- Procesy špecifické pre softvér, ktoré sú použité na implementovanie softvérového produktu alebo služby ako jednej z častí rozsiahleho systému

3.3.1 Procesy na dosiahnutie dohody

Tieto procesy definujú činnosti nevyhnutné k dosiahnutiu dohody medzi dvomi organizáciami. Patria sem tieto dva procesy:

- Proces akvizície (Acquisition Process)
- Proces podpory (Supply Process)

Procesy na dosiahnutie dohody popisované týmto štandardom sú vo všeobecnosti softvérovo-zamerané špecializácie zodpovedajúcich technických procesov podľa normy ISO/IEC 15288.

3.3.2 Organizačné procesy

Tieto procesy riadia schopnosť organizácie získavať a podporovať produkty alebo služby. Poskytujú zdroje a infraštruktúru nevyhnutnú na podporu projektov a na uistenie, že ciele organizácie a stanovené dohody budú úspešne splnené. Nesnažia sa byť komplexnou množinou obchodných procesov, ktoré umožňujú riadenie produkčnej činnosti organizácie. Skupinu tvoria nasledujúce procesy:

- Proces manažmentu pre modely životného cyklu (Life Cycle Model Management Process)
- Proces manažmentu infraštruktúry (Infrastructure Management Process)
- Proces manažmentu projektového portfólia (Project Portfolio Management Process)
- Proces manažmentu ľudských zdrojov (Human Resource Management Process)
- Proces manažmentu kvality (Quality Management Process)

Aj organizačné procesy popisované týmto štandardom sú vo všeobecnosti softvérovo-zamerané špecializácie zodpovedajúcich technických procesov podľa normy ISO 15288.

3.3.3 Projektové procesy

Projektové procesy sú v tomto medzinárodnom štandarde rozdelené do dvoch kategórií. Prvou sú procesy manažmentu projektu (Project Management Processes), ktoré slúžia na plánovanie, posudzovanie a riadenie priebehu vývoja projektu. Tieto procesy slúžia na zavedenie a rozvoj plánov projektu, na posúdenie aktuálneho plnenia projektu a na dohľad nad vykonávaním projektu až k jeho ukončeniu. Patria k nim tieto procesy:

- Proces plánovania projektu (Project Planning Process)
- Proces hodnotenia a riadenia projektu (Project Assessment and Control Process)

Druhou skupinou sú procesy podpory projektu (Project Support Processes), ktoré poskytujú zvláštne zamerané úlohy na dosiahnutie špeciálne zameraných cieľov manažmentu:

- Proces manažmentu rozhodovania (Decision Management Process)
- Proces Risk manažmentu (Risk Management Process)
- Proces manažmentu konfigurácii (Configuration Management Process)
- Proces manažmentu informácii (Information Management Process)
- Proces merania (Measurement Process)

3.3.4 Technické procesy

Technické procesy slúžia na definovanie požiadaviek na systém, na transformovanie požiadaviek do užitočného produktu, na umožnenie následného rozšírenia systému, ak to bude nutné, na používanie produktu, poskytovanie potrebných služieb, na udržanie sústavnej a správnej dodávky týchto služieb a na odstránenie produktu, ak bude vyradený z prevádzky..

Tieto procesy definujú činnosti, ktoré umožňujú maximalizovať úžitok a znižovať riziko technických rozhodnutí a krokov. Technické procesy majú za úlohu zaistiť vysokú mieru dostupnosti, nákladovej optimálnosti, spoľahlivosti, udržovateľnosti, použiteľnosti a ostatných požadovaných vlastností, ktoré plynú z dohody medzi zadávateľom a dodávateľom projektu. Taktiež umožňujú produktom prispôbiť sa legislatívnym požiadavkám napr. ohľadom vnútornej i vonkajšej bezpečnosti alebo environmentálnych faktorov. Tvorí ich nasledujúce procesy:

- Definovanie požiadaviek zainteresovaných strán (Stakeholder Requirements Definition)
- Analýzy požiadaviek systému (System Requirements Analysis)
- Architektonický návrh systému (System Architectural Design)
- Proces implementácie (Implementation Process)
- Proces integrácie systému (System Integration Process)
- Proces testovania spôsobilosti (System Qualification Testing Process)
- Proces inštalovania softvéru (Software Installation Process)
- Proces prevzatia podpory softvéru (Software Acceptance Support Process)
- Proces prevádzky softvéru (Software Operation Process)
- Proces softvérovej údržby (Software Maintenance Process)
- Proces vyradenia softvéru (Software Disposal Process)

Technické procesy popisované týmto štandardom sú vo všeobecnosti softvérovo-zamerané špecializácie zodpovedajúcich technických procesov podľa normy ISO/IEC 15288.

3.3.5 Procesy implementácie softvéru

Tieto procesy slúžia na vytváranie konkrétnych častí systému implementovaných v softvéri. Transformujú konkrétne vlastnosti, rozhrania a obmedzenia implementácie do činností vývoja, ktoré vedú k takej časti systému, ktorá spĺňa požiadavky odvodené z požiadaviek na systém.

Proces implementácie softvéru tvorí nasledujúcich 5 procesov nižšej úrovne:

- Proces analýzy požiadaviek softvéru (Software Requirements Analysis Process)
- Proces architektonického návrhu softvéru (Software Architectural Design Process)
- Proces detailného návrhu softvéru (Software Detailed Design Process)
- Proces konštrukcie softvéru (Software Construction Process)
- Proces integrácie softvéru (Software Integration Process)
- Proces kvalifikačného testovania softvéru (Software Qualification Testing Process)

3.3.6 Podporné softvérové procesy

Táto skupina poskytuje súbor špeciálne zameraných činností na vykonávanie konkrétnych softvérových procesov. Pomáhajú procesu softvérovej implementácie ako základnej činnosti, ktorej tieto procesy prispievajú k úspechu a kvalite celého softvérového projektu. V tejto skupine je 8 procesov:

- Proces dokumentácie (Software Documentation Management Process)
- Proces manažmentu konfigurácie (Software Configuration Management Process)
- Proces zabezpečovania kvality (Software Quality Assurance Process)
- Proces verifikácie (Software Verification Process)
- Proces validácie (Software Validation Process)
- Proces spoločného preskúmania (Software Review Process)
- Proces auditu (Software Audit Process)
- Proces riešenia problémov (Software Problem Resolution Process)

3.3.7 Procesy využívania skúseností a znalostí z predchádzajúcich aktivít

Táto skupina procesov pozostáva z troch procesov, ktoré podporujú schopnosť opätovne použiť softvérové prvky nad rámec hraníc projektu. Tieto procesy sú unikátne v tom zmysle, že sa nachádzajú mimo hraníc konkrétnych projektov. Sú to nasledujúce procesy:

- proces doménového inžinierstva (Domain Engineering Process)
- proces posúdenia znovupoužitia (Reuse Asset Management Process)
- proces programu znovupoužitia (Reuse Program Management Process)

3.4 Medzinárodný štandard ISO/IEC 14764

3.4.1 Rozsah

Tento medzinárodný štandard popisuje podrobnejšie manažment procesu údržby opísaný v ISO/IEC 12207, vrátane jej dodatkov. V úvode stanovuje definície viacerých druhov údržby. Poskytuje návody týkajúce sa plánovania, vykonávania, operatívneho riadenia, revízie, vyhodnotenia a ukončenia procesu údržby. Rozsah zahŕňa aj údržbu viacerých softvérových produktov s rovnakými zdrojmi pre údržbu.

Jeho použitie nie je obmedzené veľkosťou, zložitosťou, otázkou kritickosti, ani aplikáciou softvérového produktu. Používa model procesov na popis a zobrazenie každej fázy údržby softvéru. Stanovené kritéria platia pri plánovaní údržby softvéru, tak počas vývoja, ako aj pri plánovaní a realizácii údržby už existujúcich softvérových produktov. Podľa ISO 14764 by v ideálnom prípade plánovanie údržby malo začať už vo fáze návrhu vývoja softvéru. ISO 14764 poskytuje štruktúru, v rámci ktorej môžu byť vykonávané všeobecné i špecifické plány softvérovej údržby, môžu byť hodnotené a prispôbené tak čo do rozsahu, ako i do významu v daných softvérových produktoch.

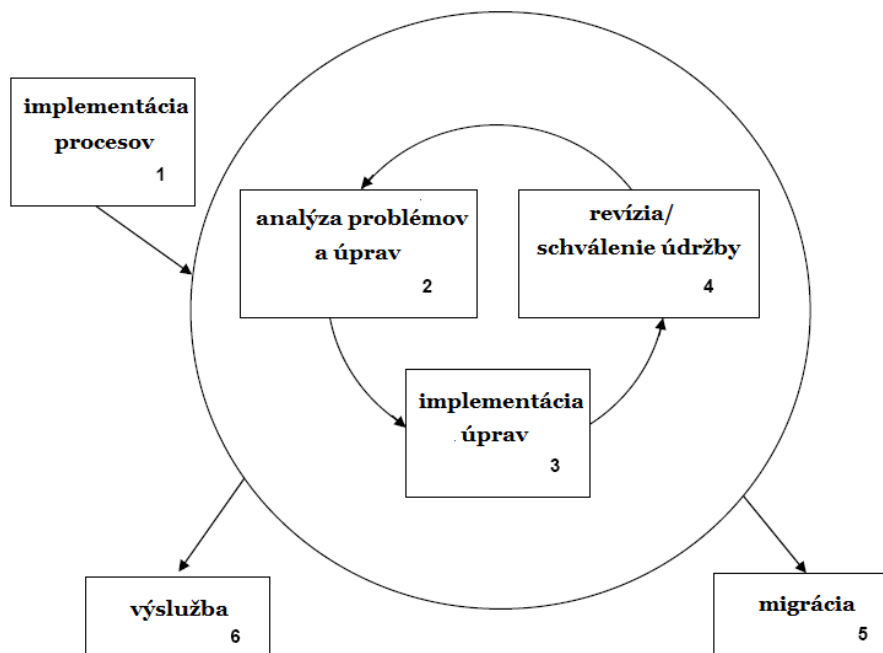
ISO 14764 poskytuje rámec pre procesy umožňujúce konzistentnú aplikáciu technológie (nástroje, techniky a metódy) pre údržbu softvéru. Určuje činnosti a úlohy údržby softvéru a poskytuje požiadavky na plánovanie údržby. Nevenuje sa prevádzke softvéru ani prevádzkovej funkčnosti, ako napr. zálohovaniu, obnovovaniu, správe systému, ktoré spravidla vykonávajú tí, ktorí softvér prevádzkujú. Tento medzinárodný štandard je vytvorený v prvom rade pre údržbárov softvéru

a v ďalšom rade pre tých, ktorí sú zodpovední za vývoj a zabezpečenie kvality. Môže byť použitý nadobúdateľmi, ale i užívateľmi, ktorí môžu poskytovať vstupy pre plán údržby. Rozpoznáva, akým spôsobom sa môže proces údržby uplatniť počas akvizície a prevádzky.

Tento medzinárodný štandard poskytuje pracovné postupy, ktoré sú príkladmi toho, čo vykonať, aby boli implementované údržbové aktivity a úlohy. V štandarde sa proces údržby rozdeľuje na nasledovné aktivity, ktoré ďalej detailnejšie opíšeme:

- implementácia procesov
- analýza problémov a úprav
- implementácia úprav
- revízia/schválenie údržby
- migrácia
- výslužba

Uvedené rozdelenie zachytáva jednotlivé procesy v chronologickom poradí, v ktorom sa bežne vykonávajú v priebehu životného cyklu softvéru. Ešte lepšiu predstavu o vzťahu týchto aktivít možno získať z nasledovného obrázku 7:



Obrázok 7: aktivity procesu údržby

Celý proces začína fázou implementovania procesov a končí odchodom softvéru do výslužby. Fáza migrácie sa nachádza medzi začiatkom a koncom životného cyklu, teda medzi implementáciou a odchodom do výslužby. Osobitosť fázy migrácie oproti ostatným je v tom, že nie každý softvér ňou počas svojej existencie prejde, hoci niektorý ňou môže prejsť aj viac krát.

Vo vnútri prebiehajú v cykle jednotlivé úpravy softvéru, čiže môžeme hovoriť o akomsi jednoduchom životnom cykle softvérovej úpravy, ktorý sa stále opakuje. Jednotlivé úpravy začínajú analýzou problému, realizujú sa vo fáze implementácie úpravy a zakončia sa otestovaním a schválením vykonanej úpravy.

Podrobný popis týchto šiestich základných údržbových aktivít je obsahom najdôležitejšej (piatej) kapitoly tohto štandardu. V ďalšej kapitole sa preto pokúsime podať zjednodušený pohľad na to, ako by podľa normy ISO 14764 mala prebiehať realizácia týchto fáz. V každej fáze popíšeme, čo tvorí jej

vstupy, aké úlohy je nutné vykonať k úspešnému dokončeniu a definujeme množinu výstupov. Keď to bude potrebné, popíšeme spôsob riadenia danej fázy. V predmetnej kapitole štandardu ISO 14764, ale aj v ostatných kapitolách, sa často odvoláva na ďalšie (podporné) procesy životného cyklu softvéru, tak ako ich špecifikuje štandard ISO 12207, ktoré napomáhajú úspešnému implementovaniu postupov údržby. Sú to predovšetkým procesy:

- dokumentácie
- manažmentu konfigurácií
- zabezpečenia kvality
- spoločných revízií
- riešenia problémov
- auditu
- verifikácie
- validácie

3.4.2 Implementácia procesov

Počas implementácie procesov údržbár zriaďuje plány a postupy, ktoré budú vykonané počas procesu údržby. Plán údržby by mal byť vyvinutý paralelne s plánom vývoja. Údržbár by tiež mal počas tejto aktivity zriadiť potrebné organizačné rozhrania. ISO 14764 kladie veľký dôraz na to, aby spoločnosti prevádzkujúce údržbu softvérov venovali dostatok úsilia na zavedenie plánov údržby. Hlavný, kľúčový dokument, ktorý má zachytávať všetky dôležité aspekty údržby, sa nazýva plán údržby. Do detailov rozpracované postupy pre konkrétne činnosti v rámci údržby majú byť vždy presne formálne uvedené v príslušnom pláne. Zoznam týchto plánov zároveň tvorí výstupy procesu implementácie.

Vstupy

Vstupy pre činnosti procesu implementácie zahŕňajú:

- relevantné základne
- systémovú dokumentáciu, ak existuje
- žiadosti o úpravu (MR) alebo hlásenia o probléme (PR)

Úlohy

Aby sa proces údržby mohol efektívne implementovať, údržbár by mal vyvinúť a zdokumentovať stratégiu výkonu údržby. K dosiahnutiu tohto cieľa musí preto údržbár vykonať nasledovné úlohy:

- vyvinúť plány údržby a postupy
- zaviesť MR/PR postupy
- implementovať manažment konfigurácií
- vyvinúť plán manažmentu konfigurácií (nemusí byť nevyhnutne pre MR/PR)

Výstupy

Výstupy tejto činnosti sú:

- plán údržby
- tréningový plán
- postupy údržby
- plán projektového manažmentu
- postupy riešenia problémov
- plán merania
- manuál údržby

- plány pre spätnú väzbu k užívateľom
- plán tranzície
- odhad udržovateľnosti
- plán manažmentu konfigurácie

Všetky výstupy by mali byť vložené pod manažment konfigurácie.

3.4.3 Analýza problémov a úprav

Touto činnosťou sa začína životný cyklus každej požiadavky na zmenu alebo hlásenia o chybe. Nezáleží na tom, či je požadovaná zmena preventívneho, adaptívneho alebo zlepšujúceho charakteru.

Vstupy

Vstupmi pre analýzu problémov a modifikácií sú tieto:

- MR/PR
- základňa (vychádzajúca z CMDB)
- softvérové úložisko (zdrojové kódy)
- systémová dokumentácia
 - informácie o stave konfigurácie
 - požiadavky ohľadom funkcionality
 - požiadavky na rozhranie
 - dáta projektového plánovania
 - výstupy z činnosti implementácie procesov

Úlohy

V tomto procese má údržbár vykonať kroky, ktoré vedú od prijatia MR/PR cez jej posúdenie až po schválenie povolenia pre jej vykonanie. Tieto činnosti vykonáva v nasledujúcom poradí:

- analyzuje MR/PR
- replikuje alebo overuje problém
- vyvíja alternatívy pre implementáciu úpravy
- dokumentuje MR/PR, výsledky a alternatívy vykonávania
- získava povolenie pre zvolenú alternatívu úpravy.

Predtým, než dôjde k začatiu práce na implementovaní zmeny, musí prebehnúť ešte (podobne ako to stanovuje ITIL) analýza o predbežnom odhade zdrojov. Na tomto posudzovaní sa majú zúčastniť zástupcovia rôznych oddelení. Výsledkom tejto analýzy potom môže byť rozhodnutie o začatí implementácie posudzovanej zmeny.

Výstupy

Výstupy tejto činnosti sú:

- analýza dopadu
 - správu o probléme alebo novú požiadavku
 - zhodnotenie problému alebo požiadavky
 - klasifikáciu požadovaného typu úpravy
 - východiskovú prioritu
 - dáta pre účely overovania (pre nápravné úpravy)
 - počiatočný odhad zdrojov potrebných na úpravu existujúceho systému.
- odporúčaná alternatíva
- schválená úprava

- aktualizovaná dokumentácia
 - aktualizovanú dokumentáciu testovania vrátane plánu testovania, postupy testovania a hlásenia z testovania
 - softvérovú dokumentáciu
 - aktualizované požiadavky

3.4.4 Implementácia úprav

Počas fázy implementácie úprav údržbár vyvíja a testuje úpravu softvérového produktu.

Vstupy

Vstupmi aktivity implementácie úprav sú:

- základňa
 - vymedzenie systémovej architektúry
 - záznam žiadosti o úpravu
 - zdrojový kód.
- schválené MR/PR
- schválená dokumentácia úprav
 - hlásenie analýzy dopadu
 - výstupy z aktivít analýzy problémov a úprav

Úlohy

V rámci implementovania úprav má údržbár vykonať analýzu, ktorej cieľom je stanoviť, ktorá dokumentácia, softvérové jednotky a ich verzie potrebujú úpravu. Výsledky dodatočných analýz by mali byť zdokumentované v softvérovej dokumentácii. Toto úsilie zahŕňa nasledujúce kroky:

- identifikovať elementy, ktoré treba upraviť na existujúcom systéme
- identifikovať elementy rozhrania ovplyvnené úpravou
- identifikovať dokumentáciu, ktorú treba aktualizovať
- aktualizovať softvérovú dokumentáciu

Počas toho, ako údržbári vykonávajú úpravu, nabáda tento štandard na dodržiavanie procesu vývoja tak, ako je popísaný v ISO 12207. Štandard na tomto mieste bližšie stanovuje, aké požiadavky majú byť doplnené do procesu vývoja.

Výstupy

Výstupy tejto činnosti by mali obsahovať:

- aktualizované plány a postupy testovania
- aktualizovanú dokumentáciu
 - aktualizované záznamy úprav
 - podrobné hlásenia analýz
 - aktualizované požiadavky
 - aktualizované plány testovania, postupy testovania a hlásenia z testovania
 - aktualizované školiace materiály
- upravený zdrojový kód
- hlásenie z testovania
- merania

3.4.5 Revízia/schválenie úpravy

Táto aktivita zabezpečuje, že úpravy na systéme sú správne a že boli vykonané v súlade so schválenými štandardmi a správnu metodológiou. Zároveň tu končí životný cyklus zmeny. Počas tohto procesu údržbár má získať schválenie o uspokojivom zavŕšení úpravy.

Vstupy

Vstupmi aktivít revízie/schválenia sú:

- aktuálna základňa (CMDB)
- upravený softvér
- výsledky testovania úprav

Úlohy

Údržbár má viesť revízie spolu s organizáciou, ktorá oprávňuje úpravu, aby sa stanovila integrita upravovaného systému. To znamená, že údržbár má:

- sledovať MR/PR od požiadaviek, k dizajnu a kódu
- overovať testovateľnosť kódu
- overovať súlad so štandardmi programovania
- overovať, či boli upravené iba nevyhnutné softvérové komponenty
- overovať, či nové softvérové komponenty boli správne integrované
- skontrolovať dokumentáciu, aby sa uistil, že bola aktualizovaná
- personál manažmentu konfigurácie vytvára softvérové položky na testovanie
- vykonať testovanie nezávislou testovacou organizáciou
- vykonať systémový test plne integrovaného systému
- vypracovať hlásenie o testovaní

Údržbár má tiež dostať schválenie za účelom uspokojivého zavŕšenia úpravy tak, ako je špecifikované v zmluve. Schválenie by malo byť obdržané aj v prípade, že sa údržba implementovala bez dohody. V súvislosti s týmto cieľom má údržbár:

- obdržať schválenie cez podporný proces zabezpečenia kvality životného cyklu
- overiť, že bol proces dodržiavaný
- upovedomiť operátorov
- vykonať inštaláciu a školenie v objekte operátora

Výstupy

Výstupy tejto aktivity sú:

- nová základňa s pridruženými schválenými úpravami
- zamietnuté úpravy
- hlásenie o prebratí
- hlásenie o audite a revízii
- hlásenie o teste spôsobilosti softvéru

3.4.6 Migrácia

Systém občas treba upraviť, aby počas svojho života mohol bežať v rôznych prostrediach. Aby mohol údržbár zmigrovať systém do nového prostredia, musí určiť činnosti potrebné na vykonanie migrácie a potom vyvinúť a zdokumentovať kroky potrebné k výkonu migrácie.

Vstupy

Vstupy pre migračné aktivity sú:

- staré prostredie
- nové prostredie
- stará základňa
- nová základňa

Úlohy

Údržbár v tejto fáze vyvíja plán migrácie, informuje užívateľov migrácie, poskytuje školenie, podáva oznámenie o ukončení, vyhodnocuje dopad na nové prostredie a archivuje dáta. Všetky výtvary z migračných aktivít sú riadené manažmentom konfigurácií.

Výstupy

Výstupy tejto aktivity sú:

- plán migrácie
- nástroje migrácie
- oznámenie o zámere
- zmigrovaný softvérový produkt
- oznámenie o ukončení
- merania
- archivované dáta

3.4.7 Výslužba softvéru

Ako náhle aplikácia dosiahla koniec svojho užitočného života, musí odísť do výslužby. Mala by byť vykonaná analýza, ktorá pomôže uskutočniť rozhodnutie odvolať aplikáciu do výslužby. Táto analýza je často založená ekonomicky a môže byť zahrnutá v pláne výslužby. Aplikácia môže byť nahradená novým softvérovým produktom, ale v niektorých prípadoch môže byť výsledkom analýzy rozhodnutie, že zastaraná aplikácia ostane naďalej v prevádzke. Analýza by mala určovať, či je nákladovo výhodné:

- udržať zastaranú technológiu
- zameniť za novú technológiu vyvinutím nového softvérového produktu
- vyvinúť nový softvérový produkt, aby sa dosiahla modularita
- vyvinúť nový softvérový produkt na podporu údržby
- vyvinúť nový softvérový produkt, aby sa dosiahla štandardizácia
- vyvinúť nový softvérový produkt na podporu nezávislosti od dodávateľa

Na to, aby softvérový produkt mohol odísť do výslužby, by mal údržbár určiť úlohy potrebné na dokončenie odchodu do výslužby a potom vyvinúť a zdokumentovať kroky potrebné na vykonanie odchodu. V rámci toho by mal zvážiť dostupnosť dát uložených softvérovým produktom, ktorý odchádza do výslužby. Všetky výtvary pochádzajúce z aktivít výslužby sú riadené manažmentom konfigurácií.

Vstupy

Vstupy aktivít výslužby sú:

- stará softvérová základňa, ktorá má odísť do výslužby
- nový softvérový produkt
- staré prostredie

Úlohy

Údržbár vykonáva aktivity odchodu do výslužby vyvíjaním plánu odchodu do výslužby, oznamovaním užívateľom odchod do výslužby, implementovaním paralelných úloh a školení, poskytovaním oznámenia o ukončení a archivovaní dát.

Výstupy

Výstupmi tejto činnosti sú:

- plán odchodu do výslužby
- oznámenie o zámere
- výsledky odchodu do výslužby
- vyškolení ľudia
- softvérový produkt vo výslužbe
- oznámenie o ukončení
- merania
- archivovaná základňa a dáta

Piata kapitola štandardu ISO 14764 detailne popisuje základné údržbové aktivity. Nasledujúca, predposledná kapitola, sa venuje úvahám o vykonávaní úprav a sporným bodom, ktoré treba zvážiť pri plánovaní údržby.

Posledná, ôsma kapitola, pojednáva o vývoji stratégie údržby softvéru. Táto stratégia sa má pripravovať pre ľudské a materiálne zdroje, potrebné na poskytovanie softvérovej údržby pre softvérové produkty. Údržbári by mali monitorovať vývojárske úsilie kvôli udržovateľnosti. Výsledky analýz udržovateľnosti by mali byť použité ako pomoc pri plánovaní údržby. Stratégia softvérovej údržby by mala pozostávať z nasledovných elementov:

- koncept údržby
- plán údržby
- analýza zdrojov

Kapitola 4

Výskum údržby v praxi

V tejto kapitole popíšeme ciele dotazníka, metodiku jeho zostavovania, návrh metriky udržovateľnosti i výsledky všetkých analýz.

4.1 Cieľ dotazníka

Pred začiatkom zostavovania dotazníka bolo potrebné položiť niekoľko zásadných otázok. Ich zodpovedaním sme získali hrubú počiatočnú predstavu o spôsobe jeho metodiky. Základnou otázkou, ktorá určovala podobu dotazníka, bolo stanovenie cieľa. Pritom sme museli brať do úvahy ďalšie dôležité okolnosti, predovšetkým možnosti oslovenia respondentov, konkrétne aká bude cieľová skupina respondentov, aké sú naše možnosti oslovenia v cieľovej skupine a koľkých respondentov potrebujeme osloviť a hlavne získať od nich potrebné informácie, aby sme z nich mohli vyvodzovať závery.

Pri stanovení cieľa dotazníka sme vychádzali zo základnej požiadavky - využiť teoretické vedomosti, ktoré sme získali štúdiom literatúry o softvérovom inžinierstve, softvérovej údržbe a o preštudovaných štandardoch a tieto vedomosti potom konfrontovať s reálnou praxou na rôzne fungujúcich IT oddeleniach alebo IT spoločnostiach. Takto sme stanovili 3 základné ciele.

4.1.1 Cieľ 1 – analýza riadenia procesu údržby

Prvým cieľom dotazníka je získať prehľad o riadení procesu údržby v hodnotených spoločnostiach. Pokúsime sa zistiť, ako sú procesy softvérovej údržby v týchto testovaných spoločnostiach implementované. V rámci tohto cieľa sme stanovili tri podciele.

- Prvým podcieľom v tejto časti bude identifikovať, aké procesy a postupy v rámci údržby softvéru sú v týchto spoločnostiach zavedené, napr. či sú implementované procesy plánovania údržby, analýzy vykonávaných zmien, činnosti ohľadne manažmentu konfigurácii a podobne.
- Druhým podcieľom bude zistiť nakoľko sú tieto procesy vyspelé, to znamená, či sú vykonávané systematicky, či sú jasne zadané, plánované, riadené, merané alebo

zlepšované a podobne.

- Nakoniec sa pokúsime zistiť, či prístup v týchto reálnych prostrediach zodpovedá prístupu nami analyzovaných noriem, a ak zodpovedá, posúdiť do akej miery. Predpokladáme totiž, že väčšina oslovených spoločností sa neriadí žiadnou medzinárodnou normou, avšak ich procesy môžu byť do istej miery týmto normám podobné. Pre istotu sme však otázku, či spoločnosť implementovala niektorú medzinárodnú normu do nášho dotazníka zahrnuli tiež.

4.1.2 Cieľ 2 – analýza vplyvu ďalších faktorov

Kým prvá časť dotazníka sa bude zaoberať celkovým prístupom hodnotených spoločností k softvérovej údržbe, druhá časť sa zameria už konkrétne na niektorú nimi udržiavanú aplikáciu. Otázkami už aj kvantitatívneho charakteru sa pokúsime získať čo najpodrobnejší pohľad na tieto aplikácie. Pomocou týchto údajov potom zhodnotíme udržiavateľnosť aplikácii, čiže množstvo úsilia potrebného na ich údržbu. Neskôr, po získaní dostatočného množstva vyplnených dotazníkov, budeme analyzovať, či tieto skutočnosti majú naozaj vplyv na udržiavateľnosť aplikácie.

4.1.3 Cieľ 3 - vytvorenie metriky udržiavateľnosti

K tom, aby sme mohli hodnotiť udržiavateľnosť aplikácií, skúmať vplyv rôznych skutočností na proces údržby, čiže, aby sme mohli zvládnuť nielen druhý, ale aj prvý stanovený cieľ, musíme nájsť spôsob, ktorým dokážeme rozlíšiť dobre udržiavateľný softvér od zle udržiavateľného - spôsob, ktorý každej skúmanej aplikácii pridelí *index udržiavateľnosti* - musíme nájsť vhodnú metriku udržiavateľnosti aplikácii.

4.2 Metodika tvorby dotazníka a otázky

Po zadefinovaní základných cieľov sme pristúpili k zostavovaniu súboru vhodných otázok. Množinu otázok sme rozdelili do dvoch okruhov, tak aby sledovali stanovené ciele dotazníka. Prvá skupina otázok bude mapovať prostredie v hodnotenej spoločnosti, otázky budú zamerané na spôsob akým je v týchto spoločnostiach údržba riadená a vykonávaná. Druhou skupinou otázok sa zameriame na v poradí druhý cieľ výskumu, čiže otázky v tejto časti budú smerovať na udržiavané aplikácie, programátorov, užívateľov, zdrojové kódy a podobne.

Množinu otázok dopĺňajú ešte rôzne ďalšie otázky, ktorými sme sa snažili získať čo najkomplexnejší pohľad na hodnotené spoločnosti.

4.2.1 Otázky ohľadom riadenia údržby

Prvou dôležitou skutočnosťou, na ktorú sme sa zamerali, boli žiadosti o zmenu a hlásenia o probléme (MR/PR).

Zaujímalo nás predovšetkým, akým spôsobom sú tieto žiadosti v oslovených spoločnostiach podávané, či v nich existujú stanovené postupy pre ich prijímanie, zaznamenávanie, vykonávanie a riadenie, či existuje v spoločnosti istá IT smernica, ktorá tieto postupy formálne stanovuje.

Veľký dôraz kladie štandard ISO 14764 na plán údržby. Preto sme sa ďalšími otázkami pokúšali zistiť, či v prvom rade existuje v oslovených spoločnostiach jasne zadefinovaný plán údržby, v ktorom sú opísané predovšetkým požiadavky na údržbu, definované zodpovednosti a odhady ľudských a finančných zdrojov na údržbu.

Zatiaľ čo v prvej časti nás zaujímalo iba to, či testované spoločnosti majú zavedené postupy pre spracovanie MR/PR, v ďalších otázkach sme sa už zamerali na podrobnosti ohľadom realizácie týchto procesov. Pýtali sme sa, či prijaté a zaznamenané MR/PR programátori najskôr analyzujú tým, že

ohlásený problém replikujú, overia jeho prítomnosť a navrhnú riešenia, ktoré nakoniec implementujú a zdokumentujú.

Kvalitu procesu prijatia a zaznamenania MR/PR sme testovali otázkou, v ktorej respondenti uvádzali, akým spôsobom prichádzajú nové MR/PR (emailom, telefonicky, písomne, ústne alebo iným spôsobom). Ak uviedli viacero spôsobov podávania, uvádzali aj percentuálny podiel pre každú možnosť.

Niekoľko ďalších otázok sa zaoberalo procesom implementácie a testovania zmien. Tak štandard ISO 14764, ako aj ITIL, kladie veľký dôraz na dôsledné a vyčerpávajúce testovanie zrealizovaných zmien pred tým, než budú schválené a uzavreté. Respondenti mali preto uviesť, či nasadené zmeny údržbári testujú a či pred schválením zmien vytvárajú aj písomný záznam o testovaní. Keďže norma ISO 14764 venuje osobitnú pozornosť úlohám, ktoré treba vykonať pri migrácii softvéru do nového prostredia, zisťovali sme, akým spôsobom oslovené spoločnosti postupovali a čo obsahoval vopred pripravený plán k migrácii.

Posledné otázky, ktoré smerovali k porovnaniu skutočných procesov údržby s požiadavkami štandardu ISO 14764, sa týkali fázy odchodu softvéru do výslužby. Opäť nás zaujímalo hlavne to, či aj v tejto fáze životného cyklu aplikácií mysleli ich údržbári na vypracovanie plánu a či bola táto činnosť patrične zdokumentovaná.

Ako sme už vyššie vysvetlili, ITIL zavádza pojem tzv. núdzovej opravy, ktorá predstavuje rýchle, ale nedokonalé riešenie vážnych problémov, aby sa čo najskôr obnovila dodávka postihnutej služby aplikácie, preto sme sa respondentov pýtali, či používajú, resp. ako často používajú tento štýl riešenia vážnych situácií. Z hľadiska vyhovenia štandardu ITIL bolo dôležité zistiť, či spoločnosti majú zavedené jasné pravidlá ohľadom kompetencií pre rozhodovanie pri riešení vážnych problémov. Špecifikom, ktoré odporúča implementovať ITIL je vytvorenie skupiny odborníkov a zástupcov všetkých oddelení spoločnosti, ktorá pomáha riešiť vzniknuté vážne situácie. Zisťovali sme preto, či spoločnosti takéto komisie zaviedli a využívajú. Analyzované normy nabádajú k tomu, že v praxi je veľmi dôležité zaviesť na vysokej úrovni fungujúci proces manažmentu konfigurácii. To sa týka okrem iného aj otázky nakladania s originálnymi zdrojovými kódmi či napr. dokumentácie, preto sme sa otázkami zamerali aj na túto problematiku.

Poslednou témou otázok zo životného cyklu zmien bola oblasť vydávania nových verzií aplikácií vo forme vydání (releasov). Zisťovali sme, či spoločnosti zaviedli procesy zodpovedné za plánovanie a kontrolu nad uvoľňovaním nových verzií.

Najdlhšou otázkou celého dotazníka bolo subjektívne posúdenie respondentov o vyspelosti ich procesov údržby softvéru. Na výber im bolo daných 5 úrovní kvality týchto procesov. Všetky úrovne boli stručne charakterizované niekoľkými vetami a postupne definovali procesy na úrovni: vykonávané, riadené, definované, kvantitatívne riadené a optimalizované procesy. Práve tieto kvalitatívne úrovne procesov rozoznáva známy model pre kvalitu procesov softvérového inžinierstva Capacity Maturity Model Integration (CMMI). My sme si však z celého modelu CMMI požičali iba definície škálovania vyspelosti procesov, ktoré CMMI rozoznáva, pretože sa nám toto rozdelenie zdalo vhodne použiteľné aj na konkrétnu oblasť udržiavateľnosti softvéru. Názor respondentov na celkovú úroveň kvality procesov údržby ich spoločnosti potom porovnáme s výsledkami na základe predchádzajúcich otázok a hlavne s celkovými výsledkami miery udržiavateľnosti ich IT produktov. Táto časť dotazníka pozostávala z tridsiatich otázok, respondenti do neho nevpisovali odpovede, v každej si iba vybrali z ponúknutých možností.

4.2.2. Otázky ku analýze aplikácií

Preštudovaná teória nám pomohla zostaviť množinu vlastností aplikácie, ktoré majú mať vplyv na jej udržateľnosť. Práve tieto ukazovatele boli preto predmetom otázok druhej časti dotazníka. (Vo výslednej podobe dotazníka, ktorý sa nachádza v prílohe sme umiestnili na úvod práve tieto otázky a až za nimi sme umiestnili otázky o procesoch údržby viažuce sa k prvému cieľu. Takéto poradie sme totiž pokladali za logickejšie.)

Na úvod sme chceli vedieť, kto bude tento dotazník vyplňovať. (V celom dotazníku sme sa však nepýtali na žiadne osobné údaje, ani ohľadom spoločnosti, ani ohľadom osoby respondenta, a taktiež pri oslovení spoločnosti sme zdôrazňovali, že dotazník je celkom anonymný.) Pre posúdenie výpovednej hodnoty dotazníka sme preto aspoň chceli vedieť, na akej pozícii pracuje osoba, ktorá tento dotazník bude vyplňovať. Veľmi dôležitou otázkou bolo určenie časovej doby, za ktorú bude aplikácia ďalej hodnotená. Nutná podmienka na to, aby sme mohli jednotlivé aplikácie porovnávať je totiž práve to, aby boli všetky hodnotené intervaly rovnako dlhé. Odporúčanú dĺžku hodnoteného obdobia sme stanovili na 12 mesiacov. V prípade, že niekto uvedie iný interval, pred analýzou ostatných dát najskôr tieto dáta prepočítame na spomínaných 12 mesiacov. Za dôležité pre posúdenie udržateľnosti sme pokladali aj typ a aspoň stručný popis aplikácie. Jednou z kľúčových vlastností aplikácií je jej veľkosť. Existuje množstvo spôsobov na určenie veľkosti aplikácií, avšak, aby sme mohli aplikácie porovnávať, vybrali sme iba najpoužívanejšie spôsoby: počet riadkov zdrojového kódu (KSLOC) a počet funkčných bodov (Feature Points – FP).

Jednou z ďalších hypotéz, ktoré sme chceli overiť, je otázka, či vek aplikácie vplyva na udržateľnosť. V literatúre sa často môžeme stretnúť s názorom, že s vekom aplikácie klesá počet ešte neobjavených chýb, no na druhej strane zdroje uvádzajú aj tvrdenia, že napredujúci vývoj IT (napr. hardvér alebo nové operačné systémy) oveľa viac zasiahne starší softvér, ktorý je potrebné adaptovať na nové podmienky. Podobne sme chceli preskúmať aj vplyv počtu verzií aplikácií, počtu operačných a databázových systémov, ktoré aplikácia podporuje. Preto sme sa respondentov pýtali aj na tieto údaje. Veľký vplyv na udržateľnosť aplikácie sme predpokladali aj v závislosti od tímu programátorov, ktorí túto údržbu vykonávajú. Venovali sme im teda viacero otázok, v ktorých sme sa zamerali na počet programátorov, fluktuáciu, dĺžku pracovnej doby a na ich úroveň znalosti použitých programovacích jazykov, technológií, samotnej aplikácie a na hlavnú obchodnú činnosť, ktorú táto aplikácia podporuje. Ďalší vplyv na udržateľnosť sme očakávali od vlastností zdrojového kódu a od kvality dokumentácie, ktoré vlastne aplikáciu tvoria. Pri zdrojovom kóde sme chceli vedieť aké programovacie jazyky (a v akých pomeroch) boli použité, matematickú a logickú zložitosť použitých algoritmov i to, či je zdrojový kód dobre štruktúrovaný, keďže sme predpokladali, že jednoduchá práca s kódom uľahčí programátorom údržbu. Pri softvérovej dokumentácii nás zaujímala teda okrem jej dostupnosti, rozdelenej do 16 celkov, aj kvalita každej tejto časti, aby sme získali čo najpresnejší obraz o tom, nakoľko pomáha programátorom riešiť problémy a zmeny. Z detailov vývoja sme si chceli overiť aj to, či svoj vplyv zohráva aj dostupnosť separovaného vývojového a testovacieho prostredia.

Ako sme už niekoľko krát spomenuli, za kľúčovú v tomto výskume považujeme všeobecne uznávanú definíciu, že udržateľnosť aplikácie je mierou úsilia programátorov potrebného na jej údržbu. Preto bolo nesmierne dôležité získať o každej aplikácii čo najpresnejšie informácie o množstve údržbového úsilia, ktoré si za hodnotené obdobie aplikácia vyžiadala. V dotazníku sme sa preto respondentom snažili v stručnosti vysvetliť, čo všetko považujeme za údržbu a na ktoré základné typy jednotlivé činnosti rozdeľujeme. Vychádzali sme opäť zo všeobecne používaného rozdelenia údržbových aktivít na opravy, vylepšenia, prispôbenia a preventívne zmeny. Okrem údržby nás však (hlavne kvôli filozofii ITIL) zaujímali aj činnosti programátorov pre podporu aplikácie a užívateľov. Vytvorili sme

preto 7 kategórii (4 pre údržbu a 3 pre podporu) a pre každú z nich sme chceli vedieť koľko úsilia jej programátori za hodnotených 12 mesiacov venovali. Ako jednotku úsilia sme v tomto prípade zvolili množstvo práce, ktoré vykoná programátor za 1 deň, z anglického označenia „man-days“ voľne preloženého ako „človeko-deň“. Predpokladali sme, že niektoré spoločnosti nebudú vedieť rozdeliť celkové úsilie do týchto kategórii, v tom prípade nás zaujímalo aspoň celkové vynaložené údržbové úsilie. Ukázalo sa však, že spoločnosti mali prehľad v tejto téme a až na ojedinelé prípady vedeli určiť požadované informácie.

Poslednou dôležitou skutočnosťou, ktorú sme chceli vyhodnotiť bol počet chýb, ktoré bolo nutné za sledovaných 12 mesiacov odstrániť. Zaujímali nás buď počty kritických, vážnych a drobných chýb alebo len celkový počet, ak organizácia nevedela podrobné záznamy o chybách.

4.2.3 Doplňujúce otázky

Celkové množstvo 40 otázok v tejto časti dopĺňali ešte otázky na ďalšie fakty, ktoré môžu mať vplyv na prevádzku a údržbu aplikácii, ako napríklad mieru dôležitosti hodnotenej aplikácie pre obchodné činnosti organizácie. Dá sa totiž predpokladať, že vyššie úsilie bude organizácia vyvíjať na údržbu tej aplikácie, ktorá zabezpečuje dostupnosť dôležitejšej služby. Respondenti uvádzali aj požadovanú dostupnosť služby, ktorú aplikácia poskytuje v hodinách do dňa a v dňoch do týždňa. Svoju úlohu počas prevádzky aplikácie zohráva aj počet koncových používateľov, ktorí aplikáciu používajú a úroveň ich skúseností.

Po skompletizovaní otázok sme pripravili úvod dotazníka, v ktorom sme sa mali za cieľ:

- uviesť respondentov do problematiky softvérovej údržby
- vysvetliť im základné pojmy, s ktorými sa budú v dotazníku stretávať
- opísať spôsoby vyplňovania dotazníka

Niektoré otázky, ktoré by mohli byť nesprávne pochopené alebo obsahujú nejaký výraz, ktorý treba vysvetliť, sme na záver doplnili stručnou nápodvedou.

Výsledný dotazník má spolu 72 otázok s 335 políčkami na vyplnenie alebo zaškrtnutie. Odhadovaný čas na jeho vyplnenie bol asi 25-30 minút.

4.3 Distribuovanie dotazníkov

Pri zostavovaní portfólia spoločností, ktoré sme požiadali o spoluprácu v tomto výskume sme si dali za cieľ osloviť čo najširšiu skupinu softvérových spoločností. Chceli sme získať prehľad o procesoch údržby tak v malých spoločnostiach ako aj vo veľkých nadnárodných organizáciách.

Z celkového počtu 30 oslovených spoločností bolo 25 vhodných pre dotazník (v piatich prípadoch spoločnosť nemala v danom čase vo svojich projektoch žiaden dokončený softvér, ktorý by už bol v prevádzke a na ktorom by prevádzali údržbu). Do stanoveného termínu sa nám naspäť vrátilo 21 vyplnených dotazníkov. Po zozbieraní údajov sme vybrali 15 tých, ktoré mali vyplnené všetky dôležité otázky. Vo zvyšných šiestich dotazníkoch buď chýbali niektoré kľúčové informácie alebo hodnotená aplikácia bola v prevádzke tak krátko (3-6 mesiacov), že uvedené hodnoty by príliš skresľovali výsledky výskumu. Ale aj z týchto dotazníkov sme využili niektoré odpovede, napr. v prehľadoch uvedených v nasledujúcej tabuľke.

	<i>minimum</i>	<i>priemer</i>	<i>medián</i>	<i>maximum</i>
dĺžka hodnoteného intervalu /mesiace	9	11,70	12	15
veľkosť aplikácie /KSLOC	3	125,19	50	530
rok spustenia do prevádzky	1999	2004,33	2004	2007
počet podporovaných verzií				
aplikácia	1	1,33	1	3
operačný systém	1	1,47	1	3
databázový systém	0	1,20	1	3
požadovaná dostupnosť aplikácie				
hodín/deň	4	18,80	24	24
deň/týždeň	5	6,53	7	7
počet užívateľov	2	1106,50	75	9000
prieskum spokojnosti užívateľ. / 1 alebo 0	0	0,44	0	1
počet programátorov	1	6,13	4	20
fluktuácia programátorov /%	0	15,93	10	66
skúsenosti programátorov /roky	2	3,84	3	6
aplikácia	1	2,77	3	4
technológie	2	3,20	3	6
programovacie jazyky	2	3,80	3	7
obchodná činnosť	1	2,53	2,5	6
dĺžka pracovnej doby /hodiny mesačne	160	166,57	160	200
všetko úsilie / údržba a podpora /CDr	36	620,40	500	2000
úsilie údržba /CDr = Človeko-deň za rok	20	432,07	320	1900
úsilie podpora /CDr	15	212,33	110	800
opravy /CDr	3	83,00	50	300
prispôsobenia /CDr	0	101,80	100	300
vylepšenia /CDr	6	222,27	80	1700
preventívna údržba /CDr	0	31,67	20	150
riešenie a analýza problémov /CDr	5	91,33	50	365
help desk /CDr	5	87,33	30	365
ďalšia drobná podpora /CDr	0	81,67	30	365
počet chýb				
extrémne	0	1,64	0	10
vážne	1	5,64	5	10
drobné	8	55,36	25	200
spolu	11	80,27	35	400
počet hovorov na Help Desk	0	116,92	40	900
priemerná doba opravy chýb /deň	0,3	4,72	2	30
existencia manaž. konfigurácii / 1 alebo 0	0	0,53	1	1
pokrytie dokumentáciou /%	0	59,33	70	100
použitie štandardu / 1 alebo 0	0	0,27	0	1
odhadované náklady na údržbu podľa respondenta /%	10	50,67	50	100
odhadovaná kvalita údržby podľa respondenta /1-5 (max)	1	2,14	2	5

Tabuľka 1: prehľad vybraných ukazovateľov výskumu

4.4 Analýza získaného portfólia

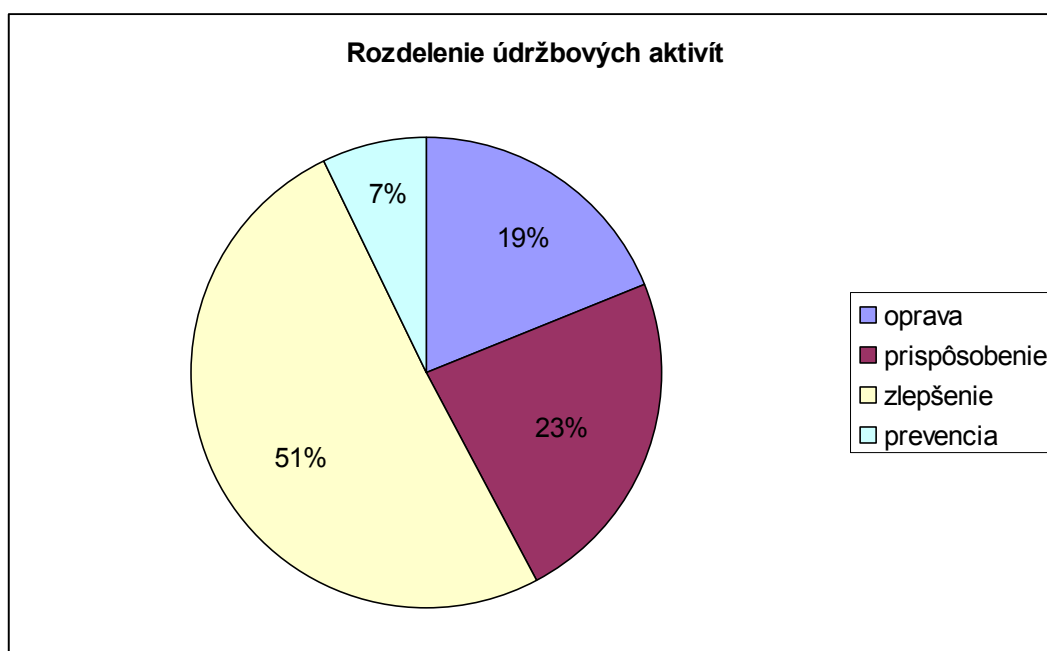
K získaniu prvých výsledkov prieskumu sme zo získaných dotazníkov zostavili tabuľku 1, ktorá zachytáva krajné a priemerné hodnoty odpovedí na niektoré otázky z dotazníka. Mnohé práve z týchto údajov (spolu s ďalšími) budeme v ďalších častiach práce podrobne analyzovať. Avšak niektoré z týchto údajov sme už ďalej bližšie nerozoberali, no považujeme ich za celkom zaujímavé a preto ich uvádzame aspoň v tomto prehľade.

Hodnotené aplikácie boli vyvíjané v 16 rôznych programovacích jazykoch, najpoužívanejšie boli C#, C++, JAVA, SQL a PHP. Priemerný užívateľ mal s používaním aplikácie zbežnú, približne ročnú skúsenosť.

Z vyplnených dotazníkov sa ukázalo, že viacero respondentov nevedelo vyjadriť veľkosť hodnotenej aplikácie v jednotkách FP, preto sme sa rozhodli v analýzach a meraniach používať iba veľkosť v jednotkách KSLOC. Uvedomujeme si, že táto miera nevystihuje presne napr. zložitosť aplikácie, no napriek tomu bude dostatočne charakterizovať rozsah aplikácie a od toho sa odvíjajúce úsilie potrebné na jej údržbu.

4.5 Dôkaz rozmanitosti portfólia

Prvým dôležitým a veľmi priaznivým výsledkom, ktorý sme dosiahli v tomto prieskume, bolo vypočítanie vzájomného pomeru 4 základných typov údržby a porovnanie výsledkov s údajmi, ktoré uvádza literatúra. Tie sme uviedli v úvode teoretickej časti tejto práce. Takmer úplná zhoda našich výsledkov a výsledkov z literatúry je dôkazom toho, že výskumná vzorka, ktorú sme získali je zostavená veľmi dobre. Hoci pozostáva iba z 21 subjektov, pravdepodobne dostatočne presne pokrýva široké spektrum IT spoločností a aplikácií. Môžeme teda očakávať, že i ostatné výsledky z prehľadovej tabuľky, ako aj ďalšie výsledky, ktoré ešte uvedieme, sú blízko reálnemu stavu priemernej IT spoločnosti.



Obrázok 8: rozdelenie údržbových aktivít z dotazníkov.

4.6 Meranie udržovateľnosti aplikácií

Dôležitým cieľom tohto výskumu je posúdenie vplyvu rôznych faktorov na udržateľnosť softvérovej aplikácie. Aby sme mohli tieto vplyvy analyzovať, musíme najskôr určiť spôsob, ktorým dokážeme ohodnotiť a číselne vyjadriť udržateľnosť ľubovoľného softvéru.

4.6.1 Analýza metrik udržovateľnosti

V literatúre sa uvádza niekoľko metrik, ktoré rôznym spôsobom hodnotia rozličné aspekty udržateľnosti aplikácií. [8] na strane 96 uvádza, že neexistuje spôsob, ktorým možno priamo merať udržateľnosť a preto musíme použiť nepriame metriky. Zároveň popisuje 2 metriky. Prvá hodnotí priemerný čas, ktorý uplynie od vzniku chyby po jej odstránenie. Druhá hodnotí pomer nákladov na údržbu ku celkovým nákladom na vývoj aplikácie. Avšak pomerne rýchlo sme tieto metriky zavrhlí, pretože ich prístup je síce logický, ale vôbec neberie do úvahy žiadne kľúčové vlastnosti aplikácie. Podobne sme skúmali aj ďalšie metriky v rôznych zdrojoch a hľadali sme vhodnú metriku ako základ pre ďalší výskum.

Stanovili sme podmienku, že daná metrika musí vychádzať z čo najrelevantnejších vlastností aplikácii, ktoré sme získali dotazníkom. Hľadaná metrika musí charakterizovať globálnu udržateľnosť v zmysle našich potrieb pre ďalší výskum a nemôže sledovať iba istý, veľmi špecifický aspekt udržateľnosti. Najbohatší prameň na metriky udržateľnosti sa nachádza v medzinárodnom štandarde ISO 9126. Obsahuje spolu 21 metrik, zameraných na 4 vlastnosti: analyzovateľnosť, meniteľnosť, stabilita a testovateľnosť softvéru. Tieto metriky sa ešte delia na interné a externé. Drvivá väčšina metrik sa ale tiež ukázala pre náš výskum ako nevhodná. Jedine oblasť meniteľnosti obsahovala 3 metriky, ktoré už čiastočne vychádzali z relevantných údajov. Tieto metriky hodnotili priemerný čas na odstránenie chyby, pomer veľkosti úsilia na zmenu ku veľkosti zmeny a pomer počtu neúspešných pokusov k celkovému počtu pokusov na odstránenie chyby. Niektoré zdroje uvádzajú aj pomerne zložité metriky, ktoré napríklad operujú s počtom naschvál vložených chýb do aplikácie. Detaily týchto výpočtov ako i ďalších metrik nájdeme napr. v [18].

4.6.2 Vytvorenie a overenie metriky udržovateľnosti

Posúdením rôznych existujúcich metrik udržateľnosti sme našli dve závislosti, ktoré sme sa rozhodli využiť a vhodne skombinovať vo výslednej vlastnej metrike.

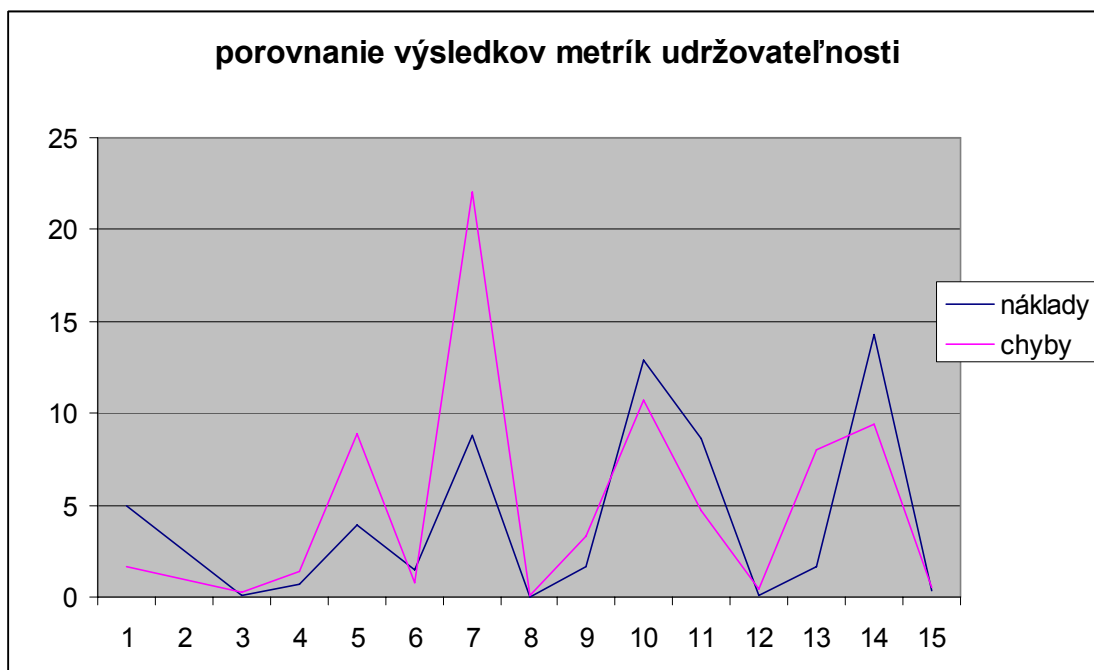
Prvá závislosť vyjadruje mieru úsilia, ktorú musia údržbári za hodnotených 12 mesiacov vynaložiť na údržbu 1000 riadkov zdrojového kódu.

Druhá závislosť sleduje počet chýb na 1000 riadkov zdrojového kódu aplikácie, ktoré údržbári za sledovaných 12 mesiacov odstránili.

Prvú závislosť sme ešte upravili tak, že z celkového úsilia sme odpočítali náklady na adaptívnu a zdokonaľujúcu údržbu, nakoľko tieto činnosti sa netýkajú priamo opráv. Množstvo vylepšení, ktoré si používatelia vyžadujú, môže byť v rôznych spoločnostiach výrazne odlišné a nemalo by mať vplyv na posudzované úsilie, ktoré potrebujeme vyhodnotiť. Ale na druhej strane podľa [1] i ďalších zdrojov, zdokonaľujúce úpravy sú tiež odstraňovaním chýb, ktoré našli a opravili programátori skôr ako stihli spôsobiť problém užívateľom.

Tieto dve metriky sme skúsili aplikovať na naše aplikácie. Niektoré spoločnosti neevidovali počet chýb podľa ich závažnosti a do dotazníka uviedli iba celkový počet, preto pôvodnú myšlienku - zohľadniť v druhej metrike aj závažnosť chýb (tým jednoduchším priradiť nižšiu váhu ako zložitým, či dokonca kritickým) - sme realizovať nemohli. Porovnanie výsledkov oboch metrik na skúmaných

aplikáciách v jednom grafe odhalilo zaujímavý výsledok: Oba grafy majú veľmi podobný priebeh, čiže aplikácie, ktoré obstáli dobre v jednej metrike boli úspešné aj v druhej a naopak. (Kvôli lepšej prehľadnosti počtu chýb je ich počet na grafe vynásobený dvoma.) Šťasti to možno vysvetliť tým, že v jednotlivých spoločnostiach pravdepodobne vyvíjajú približne rovnaké úsilie na opravenie jednej chyby (v prepočte ku veľkosti aplikácie), to však nezodpovedalo výsledkom, ktoré uviedli spoločnosti o priemernej dobe na odstránenie jednej chyby. Z dotazníkov vyplýva, že jednotlivé aplikácie sa líšia pomerom vážnych chýb a odstránenie zložitej chyby pravdepodobne stojí viac námahy ako odstránenie jednoduchšej, no tento fakt sme z už vysvetlených dôvodov nemohli do analýzy zahrnúť, preto sme brali do úvahy iba celkový počet chýb.



Obrázok 9: Graf kriviek pre zložky indexu udržovateľnosti

Vo výslednej metrike sme využili obe tieto metriky a sčítaním oboch hodnôt sme dostali výsledný index udržateľnosti aplikácie. Túto metriku budeme aj nazývať *index udržovateľnosti*. Čím vyššia je hodnota tohto indexu, tým horšie udržateľná je aplikácia. Nasledujúca tabuľka uvádza hodnoty oboch zložiek indexu udržovateľnosti: metriky nákladov (náklady) a metriky počtu chýb (chyby).

	min	avg	med	Max	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14	A15
náklady	0,04	4,15	1,67	14,29	5,00	2,50	0,08	0,73	3,89	1,50	8,82	0,04	1,67	12,93	8,67	0,12	1,67	14,29	0,36
chyby	0,04	4,88	1,67	22,00	1,67	1,00	0,30	1,43	8,89	0,80	22,00	0,04	3,33	10,69	4,67	0,46	8,00	9,43	0,48
kvalita	1	2,14	2,00	5	3	3	2	3	1	2	1	5	2	1	1	3	1	2	2
odhad	10	50,67	50,00	100	60	60	90	70	65	30	90	10	15	50	30	30	10	100	50

Tabuľka 2: hodnoty metrick udržovateľnosti a odhady

V tabuľke 2, ako i v ďalších tabuľkách budeme používať na sprehľadnenie výsledkov farebné zvýrazňovanie výsledkov aplikácií v rôznych hodnoteniach. Zelenou farbou zvýrazníme tie aplikácie, ktoré dosahujú výsledky lepšie ako je hodnota všetkých aplikácií (v niektorých prípadoch, hlavne ak je rozloženie výsledkov príliš nerovnomerné, použijeme ako určujúcu hodnotu medián). Červenými odtieňmi budeme zvýrazňovať aplikácie s podpriemernými výsledkami. V tabuľke 2 sme jasne zeleno zvýraznili tie aplikácie, ktorých index je nižší ako medián a červeno zvýraznené sú aplikácie, ktoré obstáli horšie ako medián. Aplikácie podobné mediánu ostali nezvýraznené. Pre porovnanie s

hodnotením údržby ich aplikácii podľa respondentov uvádzame ešte dva údaje z dotazníka. Tretí riadok (kvalita) predstavuje odpoveď na poslednú otázku dotazníka, v ktorej respondenti na stupnici od 1 do 5 hodnotili celkovú kvalitu udržovateľnosti ich prostredia a aplikácie. Štvrtý riadok (odhad) predstavuje odhad nákladov na údržbu v percentách z celkových nákladov na hodnotenú aplikáciu podľa predposlednej otázky dotazníka.

Zamerajme sa najskôr na tretí riadok. S výnimkou jedného prípadu je vo všetkých aplikáciách súlad medzi hodnotením aplikácie podľa našich metrík a hodnotením udržovateľnosti aplikácie od respondenta.

Všetky aplikácie, o ktorých ich zástupcovia tvrdili, že sú udržiavané v prostredí s najnižším stupňom kvality týchto procesov, dostali aj v našom hodnotení podpriemerné výsledky. Naopak, procesy údržby aplikácie A8, ktorá v našom hodnotení obstála najlepšie, dokonca s výrazným odstupom od ďalších aplikácii, ohodnotil jej zástupca tiež najvyšším možným stupňom kvality. Podobne dobre obstáli, až na spomínanú jednu výnimku, aj všetky ostatné aplikácie s nadpriemerným indexom udržovateľnosti. Myslíme si, že toto zistenie možno považovať za potvrdenie kvality oboch použitých metrík (a teda aj výslednej metriky).

Zaujímavé môže byť porovnanie indexu udržovateľnosti a odhadov nákladov na údržbu (štvrtý riadok), v ktorom aj vysoko hodnotené aplikácie spotrebovali veľké množstvo celkových nákladov. Porovnanie uvádzame ale naozaj iba ako zaujímavosť. Treba si totiž uvedomiť, že celkové náklady na údržbu zahŕňajú predovšetkým nemalé finančné náklady, napríklad na rôzny nový hardvér, softvér, mzdy a množstvo ďalších výdavkov. Oproti tomu, cieľom nášho výskumu je posúdenie kvality procesov údržby a skúmanie vplyvov rôznych detailov aplikácie na jej udržateľnosť. V nasledujúcej kapitole vysvetlíme, ako sme toto skúmanie realizovali a aké výsledky sme získali.

4.7 Metodika hodnotenia odpovedí

Najskôr uvedieme zoznam procesov, ktorých vplyv na udržateľnosť chceme skúmať.

Predovšetkým sa pokúsime zistiť, či zásady prístupu k údržbe, ktoré odporúčajú analyzované normy naozaj vedú k zvýšeniu udržovateľnosti aplikácii. Tieto zásady sme rozdelili do šiestich kategórií a na každú z nich sme sa respondentov pýtali v dotazníku niekoľkými otázkami, tak ako sme to už opísali vyššie v kapitole o metodike zostavovania dotazníka. Šesť hlavných tém štandardov, ktorých vplyv sme posudzovali tvoria procesy:

- plány údržby
- analýza MR/PR
- proces implementovania a testovania zmeny
- proces migrácie aplikácie
- manažment verzií
- manažment konfigurácii

V ďalšom kroku sme z odpovedí respondentov na otázky k jednotlivým témam získali číselné ohodnotenie, ktoré určilo, na koľko spĺňajú organizácie požiadavky štandardov k týmto témam údržby.

Otázky tejto časti dotazníka mali dve formy. Respondenti buď odpovedali na otázku odpoveďou „áno – nie – neviem“ alebo mali z ponúkanej množiny odpovedí zaškrtnúť všetky pravdivé odpovede v ich spoločnosti. Mieru vyhovenia spoločností požiadavkám štandardov sme hodnotili na stupnici od 1 do

10 nasledujúcim spôsobom. Hodnotenie 10 získala organizácia vtedy, ak v danej téme odpovedala na všetky otázky *áno* a zaškrtnula všetky možnosti pri druhom type otázok. Ak boli všetky odpovede záporné a nebola zaškrtnutá žiadna z ponúkaných odpovedí, výsledné hodnotenie bolo 0. Niektoré otázky poukazovali z hľadiska vyhovenia štandardom na významnejšie detaily, takže aj váha týchto otázok na výsledné hodnotenie bola vyššia. Snažili sme sa teda čo najpresnejšie ohodnotiť jednotlivé spoločnosti z hľadiska vyhovenia štandardom spôsobom, ktorý je sčasti formálny a sčasti intuitívny a vychádza z nášho vnímania dôležitosti požiadaviek štandardov.

Na záver prvej časti analýz sa zameriame na porovnanie výsledkov organizácií, ktoré sa riadia niektorým zo štandardov oproti spoločnostiam, ktoré žiadny štandard neimplementovali.

Okrem skúmania spoločností ohľadom vyhovenia medzinárodným normám sme analyzovali aj vplyv ďalších faktorov, či už ohľadom aplikácie alebo viažucich sa na jej programátorov. Aj tu sme posudzovali ich vplyv na kvalitu udržovateľnosti aplikácii, ktorú sme opäť reprezentovali navrhnutým indexom udržovateľnosti. Skúmali sme vplyvy nasledujúcich faktorov:

- dokumentácia
- zložitosť kódov
- dostupnosť testovacieho a vývojového prostredia
- vek aplikácie
- štruktúra kódu
- skúsenosti programátorov
- fluktuácia programátorov

Aj v týchto témach sme väčšinou jednotlivým organizáciám priradili hodnotenie od 1 do 10 podobným spôsobom ako v predošlých analýzach. Desať bodov za dokumentáciu získala organizácia, ktorá obsahovala dokumentáciu všetkých častí aplikácie a respondenti ju aj posúdili ako dostatočne užitočnú. Za zložitosť kódov dostali maximum bodov aplikácie s najjednoduchšími použitými algoritmami, za dostupnosť tie spoločnosti, v ktorých sú dostupné aj vývojové aj testovacie prostredia. V štruktúre kódov platilo, že čím lepšie organizovaná štruktúra zdrojových kódov, tým viac bodov. Skúsenosti programátorov respondenti v dotazníku hodnotili v štyroch oblastiach (programovací jazyk, použité technológie, znalosť aplikácie a znalosť obchodných procesov). Kľúčový dôraz sme kládli na skúsenosti s programovacím jazykom vyjadrený v rokoch používania. Tie sme potom podľa ostatných faktorov buď ešte čiastočne zvýšili alebo znížili. Maximálna známka v tomto hodnotení bola 6. Najviac bodov (10) za fluktuáciu programátorov získali organizácie s nulovou fluktuáciou, inak sa počítala známka ako 10 mínus fluktuácia zaokrúhlená na desiatky percent. Samotný vek aplikácie v rokoch priamo určoval výšku tohto hodnotenia.

4.8 Výsledky analýzy riadenia údržby

Všetky výsledky tejto časti sú ilustrované v tabuľke 3. Na rozdiel od predchádzajúcej tabuľky, aplikácie sú už zoradené vzostupne podľa indexu udržovateľnosti v poslednom riadku. Na farebné rozlíšenie nadpriemerných a podpriemerných výsledkov v každom riadku sme tiež použili podobný postup ako v tabuľke 2. Predposledný riadok vyjadruje sumu bodov vo všetkých šiestich kategóriách. Siedmy riadok ukazuje, či organizácia používa nejaký štandard životného cyklu softvéru alebo kvality softvéru. Áno znamená 10 bodov, nie znamená 0 bodov. Na výslednú hodnotu bodov v Sume ale štandard vplyv nemá, čiže iba samotná existencia štandardu v spoločnosti tejto aplikácii žiadne body do hodnotenia nepridá. Na nasledujúcich stranách podrobne vyhodnotíme vplyv týchto tém na udržovateľnosť.

	min	avg	med	max	A8	A3	A12	A15	A4	A6	A2	A9	A1	A13	A5	A11	A10	A14	A7
plány údržby	0	3,40	3	10	10	3	3	9	3	8	5	0	5	0	3	0	0	0	2
analýza MR/PR	0	5,20	6	10	10	2	9	8	10	7	8	0	6	2	0	0	4	6	6
implementovanie & testovanie	2	5,53	5	10	10	4	4	6	9	6	4	8	8	4	2	5	5	4	4
migrácia	1	5,90	7	10	10	7	0*	7	0*	1	7	4	7	0*	0*	0*	6	7	3
manažment verzí	2	6,07	6	10	10	6	6	10	5	6	7	8	9	4	2	4	8	2	4
manažment konfigurácií	0	5,33	10	10	10	10	10	10	10	10	10	0	10	0	0	0	0	0	0
štandard v organizácii	0	2,67	0	10	0	0	10	10	0	10	0	0	10	0	0	0	0	0	0
SUMA bodov	7	29,47	32	60	60	32	32	50	37	38	41	20	45	10	7	9	23	19	19
index udržovateľnosti	0,08	9,03	5,00	30,82	0,08	0,38	0,58	0,84	2,17	2,30	3,50	5,00	6,67	9,67	12,78	13,33	23,62	23,71	30,82
index2 pre migráciu	0	2,67	2,22	14,29	0,4	0,2	0,03	0,2	0,3	2,5	3	0,3	5	2,3	2,22	6,67	2,59	14,3	

Tabuľka 3: analýza procesov údržby v organizáciách I.

Skôr ako sa pustíme do analyzovania výsledkov po jednotlivých témach, všimnime si pomyslenú zvislú čiaru medzi A1 a A13, ktorá (ak nerátame biele časti) takmer presne rozdeľuje plochu tabuľky dve časti, napravo takmer výhradne oranžovú a naľavo s dominantnou prevahou zelenej. Veľmi pozoruhodný je ale fakt, že aj predposledný riadok - index udržovateľnosti, nenaruša toto rozdelenie ani v jednom stĺpci (aplikácii). Inými slovami, aplikácie, ktoré sme my označili na základe navrhnutého indexu za dobre udržovateľné, dosahujú veľmi často nadpriemerné hodnotenie aj v jednotlivých analyzovaných oblastiach. Pritom index udržovateľnosti je funkciou iba troch premenných (veľkosť, údržbové úsilie a počet chýb) a je preto celkom nezávislý od faktorov, ktoré sme analyzovali v tejto časti výskumu.

4.8.1 Plány údržby

Podľa priemerného hodnotenia jednotlivých tém má plán údržby najhoršie výsledky, to znamená, že v tomto smere je prax v hodnotených spoločnostiach najviac vzdialená od zásad, ktoré odporúčajú implementovať a dodržiavať štandardy. Až 33% spoločností nemá ani náznaky žiadneho plánovania v procesoch údržby. Ďalších 33% spoločností už síce nejaké plány údržby má, avšak ich kvalita je 2 alebo 3, čo na stupnici od 1 do 10 znamená veľmi slabé plány. Ak porovnáme známky plánov s indexmi udržovateľnosti, jasne vidíme, že spoločnosti bez plánov údržby sú výhradne v druhej polovici udržovateľnosti a naopak všetky spoločnosti s vyššou kvalitou plánov sú práve tie s najvyššími indexmi. Môžeme preto konštatovať, že zavedenie plánov údržby do spoločností pozitívne vplyva na udržovateľnosť aplikácií. Že kvalitné plány samé o sebe ešte nič neznamenajú však dokazuje spoločnosť, ktorá vykonáva údržbu aplikácie A1. Oproti väčšine ostatných spoločností je jej hodnotenie 5 síce nadpriemerné, avšak index udržovateľnosti radí ich aplikáciu až medzi priemerne udržovateľné. Dokonca o niečo vyšší index dosiahla aplikácia A9, ktorej organizácia nezaviedla vôbec žiadne plány na údržbu. Podobne veľmi dobrý index aplikácií A3, A12 a A4 dosiahli spoločnosti s pomerne slabými plánmi údržby. Aj to potvrdzuje, že samotné kvalitné plány ešte dobrú udržovateľnosť neprinesú. No napriek týmto, nie až tak výrazným výnimkám, vo všeobecnosti konštatujeme, že existencia plánov údržby a dodržiavanie zásad, ktoré s nimi súvisia, tak ako ich opisujú normy ISO 14764 a ITIL vedú k zvýšeniu udržovateľnosti aplikácií.

4.8.2 Analýza MR/PR

Priemerné hodnotenie prístupu oslovených spoločností k analýze MR/PR je v tomto prípade už 5,2, teda spoločnosti zhruba na úrovni 50% dodržiujú zásady, ktoré im v tomto smere odporúčajú štandardy ISO 14764 a ITIL. Všetkým spoločnostiam, ktoré venovali analýze MR/PR nadpriemerné úsilie ešte pred implementovaním zmien sa to vyplatilo, čo dokazuje vysoký, vždy nadpriemerný index. Naopak, spoločnosti, ktoré analýzu MRPR úplne vynechávajú alebo sa jej venujú len okrajovo, dosiahli

väčšinou iba podpriemernú, nanajväš priemernú udržovateľnosť. V tomto prípade však ešte viac ako v predošlom figuruje jediná výnimka, aplikácia A3. Táto spoločnosť napriek tomu, že analýze nevenuje takmer žiadne úsilie dosiahla veľmi dobré celkové výsledky udržovateľnosti. Podobne aj dve úplne posledné aplikácie v indexe udržovateľnosti z istých dôvodov dosahujú vcelku dobrý prístup k analýze. Ako sme však už viackrát uviedli, rôznych faktorov, ktoré majú vplyv na nami skúmanú údržbu je obrovské množstvo a niektoré z nich spôsobili aj prepád týchto dvoch aplikácií. Opäť však vyslovujeme záver, že dôsledné plánovanie analýzy MR/PR, ktoré zdôrazňujú skúmané normy ISO 14764 a ITIL má určite pozitívny vplyv na úroveň udržovateľnosti aplikácií.

4.8.3 Implementovanie a testovanie

Výsledky analýzy tejto fázy životného cyklu MR/PR sú sčasti podobné výsledkom predchádzajúcej analýzy. Pravdepodobne väčšina spoločností, ktorá na istej úrovni pristupuje k analýze MR/PR potom následne zhruba na rovnakej úrovni vedie aj proces implementovania a testovania zmien. Porovnaním výsledkov hodnotenia tohto procesu s indexom udržovateľnosti vidíme, že opäť všetci lídri s vysokým indexom aj v tomto prípade obstáli bez výnimky nadpriemerne. Opačných výnimiek nám však trochu pribudlo a až tri spoločnosti, ktoré venujú implementácii a testovaniu zmien iba mierne podpriemerné úsilie, dosiahli celkovo slušný index a z toho dve až veľmi vysoký index, jednou z nich je opäť spoločnosť s aplikáciu A3.

Stále je však preukázateľný vplyv kvalitného prístupu k implementovaniu a testovaniu zmien na výsledky udržovateľnosti aplikácie, preto aj v tomto prípade konštatujeme záver, že prístupy, ktoré v tomto smere zavádzajú normy ISO 14764 a ITIL vedú k efektívnemu odstraňovaniu problémov a k zvyšovaniu udržovateľnosti aplikácií.

V tejto kapitole nás ešte zaujímala príčina toho, prečo spoločnosť A3, ktorá obom týmto dôležitým procesom (ktoré -ako sme ukázali- majú obe výrazný vplyv na udržovateľnosť) venuje dosť málo úsilia, dosiahla inak veľmi dobrý index udržovateľnosti. Na tomto mieste trochu predbehneme výsledky analýz ďalších procesov, ktorých vplyv sme ešte v tejto práci skúmali a prezradíme, že v žiadnej z ďalších kategórií táto spoločnosť nedosiahla výrazne vysoké hodnotenie. Domnievame sa, že príčinu sme našli v programátoroch. Spoločnosť A3 síce ani v hodnotení skúseností programátorov nedosiahla nadpriemerné výsledky (pripomenieme, že toto hodnotenie vychádzalo z dĺžky skúseností s použitými programovacími jazykmi). Ako sme však neskôr zistili, títo programátori majú dobré teoretické vedomosti v informatickom odbore a tak predpokladáme, že práve to umožnilo programátorom efektívne realizovať žiadosti o zmeny a aj úspešne ich opravovať.

4.8.4 Migrácia

Priemerná známka kvality prístupu k procesu migrácie z hľadiska vyhovenia normám bola takmer 6 bodov. Treba však zdôrazniť, že až 5 aplikácií dosiaľ ešte nebolo potrebné upravovať z dôvodu zmien prostredí, v ktorých sú tieto aplikácie prevádzkované. Výsledky týchto aplikácií preto nebolo možné zhodnotiť a vo výsledkovej tabuľke majú uvedenú hodnotu 0*. Analyzovať budeme iba zvyšných 10 aplikácií. Súvis indexu udržovateľnosti s kvalitou procesu migrácie je viditeľný. Zo šiestich aplikácií, ktoré v tomto hodnotení obstáli nadpriemerne, má iba jedna podpriemerný index udržovateľnosti, naopak dve aplikácie s pomerne nízkym hodnotením prístupu k migrácii majú index nadpriemerný. V súvislosti s procesom migrácie však chceme podotknúť, že údržbové úsilie, ktoré vykonávajú programátori z dôvodu migrácie nepatrí z hľadiska rozdelenia údržby na 4 typy medzi opravy, ale k prispôbeniam aplikácie meniacemu sa vonkajšiemu prostrediu. Preto sme presnejšie výsledky získali, keď sme vypočítali nový index udržovateľnosti iba na základe množstva úsilia na prispôbenia a veľkosti aplikácie. Túto metriku sme nazvali *Index2* a jej hodnoty pre jednotlivé spoločnosti uvádzame v poslednom riadku tabuľky 3. Aj tieto výsledky potvrdili prítomnosť

skúmaného vplyvu, ako to vidíme z farebného zvýraznenia v tejto tabuľke. Záver tohto testovania je preto opäť pozitívny a konštatujeme, že riadenie procesu migrácie podľa návodov zo štandardov ISO 14764 a ITIL zvýši udržovateľnosť aplikácií.

4.8.5 Manažment verzii

Spomedzi všetkých posudzovaných tém v tejto časti vyhoveli testované spoločnosti normám najlepšie v tejto kategórii, ako to naznačuje priemerná hodnota známky- viac ako 6 bodov. Ak sa pozrieme na spoločnosti, ktorým buď veľmi dobrá známka v celkovom indexe nepomohla alebo naopak spoločnosti so slabým manažmentom verzii, avšak s vysokým indexom, nájdeme v oboch týchto skupinách iba po jednej výnimke. Zaujímavejšie sú až 3 spoločnosti s celkom priemernou kvalitou manažmentu verzii, ktoré ale dosiahli vysoké hodnoty indexu. Až 56 percent dobre udržovateľných aplikácií malo len priemerný, či v jednom prípade až mierne podpriemerný manažment verzii. Z opačného pohľadu ale 84% aplikácií s nadpriemerným manažmentom verzii dosiahlo aj nadpriemerný index udržovateľnosti, takže súvis medzi týmito dvoma skutočnosťami existuje a implementovaním kvalitného manažmentu verzii do vývoja a údržby aplikácií, tak ako to radí predovšetkým ITIL, ale aj ISO 14764 naozaj prispejeme k zvýšeniu udržovateľnosti daných aplikácií.

4.8.6 Manažment konfigurácií

Najjasnejšie výsledky porovnania skúmaných častí noriem a aplikácií priniesla práve téma manažmentu konfigurácií. Keďže niektoré aspekty manažmentu konfigurácií sme už hodnotili, v tejto analýze sme hodnotenie spoločností oproti ostatným zjednodušili a vychádzalo len z jednej otázky, v ktorej sme sa pýtali respondentov, či v ich spoločnostiach zaviedli proces manažmentu konfigurácií. Aj výsledné hodnotenie preto mohlo byť len 0 alebo 10. O to jasnejšie sú ale výsledky tejto analýzy, kde okrem jednej zanedbateľnej výmeny poradia uprostred zoradených spoločností, v indexe udržovateľnosti jednoznačne vedú práve tie spoločnosti, ktoré manažment konfigurácií implementovali. Spoločnosti bez manažmentu konfigurácií dosiahli každá maximálne priemernú udržovateľnosť. Vplyv implementovania manažmentu konfigurácií na výslednú udržovateľnosť je preto jasne preukázateľný. Záverom aj v tejto oblasti je preto odporúčanie implementovať manažment konfigurácií do organizácie tak, ako to popisujú štandardy ISO 14764 a ITIL.

	min	avg	med	max	A12	A15	A6	A1	A8	A3	A4	A2	A9	A13	A5	A11	A10	A14	A7
plány údržby	0	3,40	3	10	3	9	8	5	10	3	3	5	0	0	3	0	0	0	2
analýza MR/PR implementovanie & testovanie	0	5,20	6	10	9	8	7	6	10	2	10	8	0	2	0	0	4	6	6
migrácia	1	5,90	7	10	0*	7	1	7	10	7	0*	7	4	0*	0*	0*	6	7	3
manažment verzii	2	6,07	6	10	6	10	6	9	10	6	5	7	8	4	2	4	8	2	4
manažment konfigurácií	0	5,33	10	10	10	10	10	10	10	10	10	10	0	0	0	0	0	0	0
štandard v organizácii	0	2,67	0	10	ISO	ISO	CMMI	CMMI	0	0	0	0	0	0	0	0	0	0	0
SUMA bodov	7	29,47	32	60	32	50	38	45	60	32	37	41	20	10	7	9	23	19	19
index udržovateľnosti	0,08	9,03	5,00	30,82	0,58	0,84	2,30	6,67	0,08	0,38	2,17	3,50	5,00	9,67	12,78	13,33	23,62	23,71	30,82

Tabuľka 4: analýza procesov údržby v organizáciách II.

4.8.7 Organizácie so štandardom

Na záver prvej časti výskumu sa pokúsime porovnať výsledky spoločností, ktoré implementovali do svojho vývoja niektorý z medzinárodných štandardov (označme ich skupina A) s tými, ktoré sa žiadnym štandardom neriadia. Tieto ostatné môžeme ešte pre zaujímavosť a lepšie porovnanie rozdeliť podľa indexu udržovateľnosti na nadpriemerné (skupina B) a podpriemerné (skupina C). Predpokladáme, že rozdiel medzi najlepšimi a najhoršími (A verzus B+C) bude značný, ako tomu

nasvedčuje nová tabuľka s preusporiadaným poradím. Zaujímavejšie ale bude porovnanie úspešných spoločností so štandardom a bez neho (A verzus B). Kvôli lepšej prehľadnosti uvádzame upravenú tabuľku 4, v ktorej sú aplikácie preusporiadané v poradí A, B, C. Spoločnosť A12 implementovala štandardy ISO 12207 a ISO 27001 (ten sa však týka systému manažérstva informačnej bezpečnosti), spoločnosť A15 iba štandard ISO 12207 a spoločnosti A6 a A1 sa riadia štandardom CMMI. V súčasnosti sú obe na jeho druhej úrovni, no spoločnosť A6 sa v danom čase pokúšala dosiahnuť úroveň 3.

Najskôr sa zamerajme na výsledky skupiny A v porovnaní so všetkými ostatnými.

Len dve červené plochy v ľavej tretine tabuľky oproti obrovskému množstvu na druhej strane sú jasným dôkazom dominancie skupiny A. Vyjadrené v číslach je to nasledovné: priemerná známka v skupine A je 7,2 a v skupine B+C je to 4,5. Skupina A teda jasne vyniká vo všetkých hodnotených sférach. Aj porovnanie priemerného indexu udržateľnosti je jednoznačne v prospech skupiny A. Tento výsledok bol ale očakávateľný.

Dve zo štyroch spoločností v skupine A implementovali normu ISO 12207, ktorá, ako sme uviedli v teoretickej časti, bola základom pre jej rozšírenie v podobe normy ISO 14764 a práve táto norma bola jedna z kľúčových pri vytváraní rámca tejto časti dotazníka. Viac preto upúta pozornosť fakt, že spoločnosť A12 dosiahla iba 32 bodov za všetky posudzované kategórie, čo je len priemerná hodnota. Do istej miery je to jej 0* bodmi za migráciu, ale spoločnosť dosť slabo obstála aj v plánovaní údržby a v procese implementovania a testovania zmien. V tejto spoločnosti je teda ešte očividný priestor na zlepšovanie, v týchto dvoch, ale i v ďalších častiach. Ak by spoločnosť A12 implementovala aj štandard ISO 14764, nezabralo by to príliš veľa úsilia, pretože niektoré jeho časti má už teraz veľmi dobre zvládnuté a jej celková udržateľnosť by sa ešte zvýšila.

Výsledkom tohto porovnania je teda záver, že skupina A dosiahla oproti priemernej spoločnosti bez štandardu oveľa lepšie výsledky udržateľnosti.

Porovnanie skupín A a B už také veľké rozdiely neprináša. Výsledky hodnotenia sú v oboch skupinách veľmi podobné, dokonca hovoriace mierne v prospech skupiny B. Priemerný index aplikácie z B je o niečo málo vyšší ako priemer v skupine A, podobne aj priemerné známky skupiny B sú 7,4, čo je o 0,2 bodu viac ako v skupine A. Vysvetľujeme si to tým, že štandardy ISO 12207 ani CMMI nie sú zamerané priamo na softvérovú údržbu. Štandard ISO 12207 obsahuje stručnú kostru procesu softvérovej údržby, ale prípadných záujemcov o implementovanie tohto procesu odkazuje na štandard, ktorý je na to presne určený – ISO 14764.

Výsledkom tejto časti je preto konštatovanie, že v podmnožine spoločností s nadpriemernou udržateľnosťou nedosiahli spoločnosti, ktoré implementovali štandard ISO 12207 alebo CMMI lepšie výsledky ako ostatné spoločnosti.

Všetky spoločnosti skupiny A síce v porovnaní s (B+C) obstáli nadpriemerne dobre, ale zároveň vždy existovala aj spoločnosť (z B) bez týchto štandardov s rovnako dobrými výsledkami.

Celkové zhodnotenie výsledkov tejto časti výskumu popíšeme v nasledujúcej kapitole.

4.9 Vplyv ďalších faktorov na proces údržby

	min	avg	med	max	A8	A3	A12	A15	A4	A6	A2	A9	A1	A13	A5	A11	A10	A14	A7
dokumentácia	0	6,20	7	10	10	3	9	6	10	7	7	6	8	7	8	4	6	0	2
zložitosť kódu	0	4,67	5	10	0	5	5	5	5	5	5	5	10	5	5	5	5	5	0
vyvoj. a test. prostredie	0	8,17	10	10	10	10	7,5	0	10	10	10	7,5	10	10	7,5	0	10	10	10
štruktúra kódu	3	6,13	7	10	7	3	7	7	7	7	10	7	7	3	3	3	7	7	7
skúsenosti teamu	3	6,33	5	10	5	5	3	5	3	7	10	5	8	10	8	5	8	8	5
fluktuácia	3	8,47	9	10	10	7	10	9	6	9	9	9	3	10	9	8	10	8	10
vek aplikácie	1	3,67	4	9	5	4	2	6	4	2	9	4	4	1	1	3	4	5	1
SPOLU	28	43,63	43,5	60	47	37	43,5	38	45	47	60	44	50	46	41,5	28	50	43	35
index udržateľnosti	0,08	9,03	5,00	30,82	0,08	0,38	0,58	0,84	2,17	2,30	3,50	5,00	6,67	9,67	12,78	13,33	23,62	23,71	30,82

Tabuľka 5: analýza procesov údržby v organizáciách II.

4.9.1 Dokumentácia

Otázky v dotazníku ohľadom kvality a dostupnosti dokumentácie boli pomerne vyčerpávajúce. Analyzujeme, či v testovaných aplikáciách je nejaký súvis medzi touto dokumentáciou a indexom udržateľnosti.

Priemer tohto hodnotenia je približne 6 a úroveň dokumentácie v jednotlivých spoločnostiach bola približne rovnaká, väčšinou v rozmedzí od 6 do 8. Aplikácie s výrazne podpriemernou, či dokonca nulovou dokumentáciou, sú okrem jednej výnimky veľmi zle udržateľné. Avšak príliš veľa aplikácií s mierne nadpriemernou dokumentáciou je akurát uprostred poľa udržateľnosti, dokonca mierne smerom k horšej udržateľnosti. Výsledky preto nie sú zďaleka také jednoznačné ako pri posudzovaní prvej časti dotazníka. Každopádne istý súvis tu badateľný je, žiadna aplikácia s kvalitnou dokumentáciou neskončila výrazne zle udržateľná. Celkovo teda konštatujeme, že kvalitná dokumentácia má preukázateľný vplyv na udržateľnosť, čo si zrejme aj väčšina IT spoločností uvedomuje a dôkazom toho je pomerne vysoká kvalita ich dokumentácie.

4.9.2 Zložitosť kódu

Ak sme o kvalite dokumentácie tvrdili, že je medzi jednotlivými spoločnosťami pomerne vyrovnaná, v tomto prípade to platí ešte o rad viac. Na začiatok chceme podotknúť, že zložitosť kódov je vlastnosť, ktorú možno len ťažko ovplyvňovať a do veľkej miery je vopred daná typom aplikácie. Matematická a logická zložitosť algoritmov však určite je významným parametrom aplikácií. Domnievame sa, že čím zložitejšie sú tieto algoritmy, tým viac úsilia potrebujú programátori vynaložiť nato, aby v kódach našli chybu, implementovali opravu a úpravu otestovali. Zaujímalo nás, či tento vplyv preukážu aj odpovede respondentov. Otázka, ktorou sme sa pýtali na zložitosť algoritmov ich aplikácií obsahovala 3 možnosti – jednoduché, priemerne zložené a veľmi zložené algoritmy. Až 12 z 15 respondentov označilo ich algoritmy za priemerne zložené. Navyše obe aplikácie, ktoré obsahujú veľmi zložené algoritmy, sa umiestnili v celkovej udržateľnosti na prvom a poslednom mieste a jediná aplikácia s ľahkými algoritmami skončila uprostred poľa, čo nám znemožnilo vysloviť akékoľvek závery ohľadom vplyvu toho faktora na výslednú udržateľnosť. Za príčinu toho, že vplyv zložitosti kódu nemôžeme dokázať, považujeme nízky počet otázok s malým množstvom ponúkaných odpovedí a neočakávane častú rovnakú odpoveď veľkej väčšiny respondentov.

4.9.3 Dostupnosť vývojového a testovacieho prostredia

V literatúre [11] a v iných zdrojoch sa uvádza, že jedným z faktorov, ktoré majú vplyv na

udržovateľnosť aplikácii je kvalita testovania zmien pred ich nasadením do prevádzky. S tým úzko súvisí dostupnosť separovaného prostredia na vývoj a na testovanie. Skúmali sme preto, či aj naše aplikácie preukážu súvis tohto faktora na mieru udržovateľnosti. V otázkach 38 a 53 sme sa pýtali na dostupnosť testovacieho a vývojového prostredia a na to, či programátori zmenu pred jej nasadením testujú. V tejto téme dosiahli všetky aplikácie najvyššie priemerné skóre – viac ako 8, čiže drvivá väčšina spoločností odpovedala na všetky otázky kladne, takže dostupnosť vývojového a testovacieho prostredia bola skoro vo všetkých spoločnostiach samozrejmosťou. Jediným prekvapením je veľmi slušné štvrté miesto v celkovej udržovateľnosti aplikácie A15, ktorá nemá žiadne prostredie na testovanie. V závere preto konštatujeme, že na základe dotazníka nevieme zodpovedne posúdiť, či testovacie prostredie má vplyv na udržovateľnosť, avšak veľká väčšina spoločností toto prostredie má a zmeny pred nasadením do prevádzky najskôr testuje.

4.9.4 Štruktúra kódu

Ďalšou vlastnosťou zdrojových kódov, ktorú respondenti hodnotili bola kvalita štruktúry ich zdrojových kódov, o ktorej sme sa tiež domnievali, že môže mať vplyv na úsilie, ktoré musia programátori vynaložiť na opravy a realizovanie zmien. No aj v tomto prípade sa príliš veľká časť respondentov zhodla na rovnakej odpovedi a až 66% z nich označila ich zdrojové kódy za dobre štruktúrované. Iba v jednom prípade mala spoločnosť zdrojové kódy vytvorené využitím CASE nástrojov. Zvyšné 4 aplikácie mali túto štruktúru organizovanú iba slabo. Žiadna zo spoločností nedosiahla najhoršiu možnosť – chýbajúce časti zdrojových kódov. Navyše celková udržovateľnosť najúspešnejšej aplikácie v tejto kategórii je iba priemerná, zatiaľ čo výborný druhý najvyšší index udržovateľnosti dosiahla aplikácia so slabo štruktúrovanými kódmi. Podobne ako v predchádzajúcom prípade preto nemôžeme zodpovedne dokázať alebo vyvrátiť skúmaný vplyv.

4.9.5 Skúsenosti tímu

Tímom v tomto prípade myslíme programátorov, analytikov a testerov, ktorí sa podieľajú na údržbe. Pripomenieme, že ich hodnotenie vyjadrovalo, koľkoročné skúsenosti majú programátori v štyroch vybraných ukazovateľoch a hlavný vplyv mali skúsenosti s programovacím jazykom, v ktorom je ich aplikácia naprogramovaná. Predpokladali sme, že čím dlhšiu skúsenosť majú, tým menej úsilia zaberie každá oprava, či zmena v aplikácii. Podobný vplyv uvádza aj zdroj [11]. Ako však už prvý pohľad na príslušný riadok v tabuľke 4 ukazuje, nič nenasvedčuje tomu, že by skúsenosť tímu mala napomáhať rýchlejšim opravám. Dva najskúsenejšie tímy dosiahli len priemerný index udržovateľnosti. Päť mierne podpriemerných tímov sa pritom stará o aplikácie s najvyšším indexom. Ak by boli tieto tímy ešte o niečo slabšie, znamenalo by to skôr opačný vplyv na udržovateľnosť, ako sme predpokladali. Opäť preto nevidíme badateľný vplyv skúmaného faktora. Aj tu vidíme príčinu v tom, že výsledky spoločností boli pomerne dosť podobné.

Témy skúsenosti programátorov sme sa už dotkli skôr, pri posudzovaní vyhovievania normám ohľadom implementácie a testovania zmien. Načrtli sme iný prístup k hodnoteniu kvality programátorov, v ktorom by sa bral ohľad aj na ich teoretické znalosti z informatiky. Myslíme si, že v dotazníku takéhoto rozsahu nie je možné dostatočne komplexne posúdiť úroveň tak širokého pojmu, ako sú napríklad skúsenosti programátorov. Na druhej strane ale táto otázka v dotazníku priniesla aspoň zaujímavé údaje o priemerných skúsenostiach programátorov, ktoré sme uviedli už v tabuľke 1.

4.9.6 Fluktuácia programátorov

Pohyb, či stabilita prevádzkových programátorov je tiež jedným z faktorov, ktorý by mohol mať vplyv na efektivitu, s ktorou programátori zvládajú riešiť problémy objavené v aplikácii počas jej prevádzky. Programátor, ktorý na danom projekte pracuje už dlhší čas, pozná už aplikáciu a má väčšiu šancu prísť na príčinu problémov a rýchlo ju odstrániť. Podľa odpovedí v skúmaných spoločnostiach bola

maximálna fluktuácia 66%, avšak až u 10 z 15 spoločností sa pohybovala v rozmedzí od 0 do 15 percent. Ako vidíme aj v tabuľke spoločnosti s nulovou fluktuáciou dosiahli tak najlepšiu ako aj najhoršiu udržovateľnosť a nachádzali sa aj medzi priemernými. Ani spoločnosť A1 s najväčšou, až dvojtretinovou fluktuáciou nedosiahla zlý index. Skúsili sme sprísniť hodnotenie a zopár najhorších spoločností nepočítať, čím by sa zväčšili odchýlky medzi zostávajúcimi, no ani tak nebol vplyv fluktuácie preukázateľný.

4.9.7 Vek aplikácie

Poslednou skúmanou skutočnosťou bol vek aplikácie. Všetky spoločnosti hodnotili svoje aplikácie zhruba za rovnaké obdobie s koncovým dátumom v čase vyplnenia tohto dotazníka, takže spolu s dátumom uvedenia aplikácie do prevádzky vieme vypočítať vek aplikácie v čase hodnotenia a môžeme skúmať jeho vplyv na množstvo úsilia, ktoré si údržba za hodnotené obdobie vyžiadala. Bodová hodnota každej aplikácie je v tomto prípade priamo jej vek (v rokoch). Do tejto analýzy sme zahrnuli aj jednu z piatich aplikácií, ktoré sme na začiatku vyradili z ďalšieho spracovania, pretože ich dotazníky neobsahovali viacero dôležitých údajov. Zatiaľ čo priemerný vek týchto pätnástich aplikácií je asi tri a pol roka, spomínaná aplikácia vznikla už v roku 1980. Dotyčný respondent síce nevedel presne vyjadriť koľko úsilia si vyžiadali jednotlivé údržbové aktivity, ale uviedol k tejto téme zaujímavú poznámku, že opravy tejto aplikácie boli len minimálneho rozsahu, nakoľko za tak dlhé obdobie jej prevádzky už boli asi všetky chyby nájdené a opravené.

Situácia na týchto pätnástich aplikáciách síce jednoznačne nepotvrďuje teóriu, že množstvo opráv s pribúdajúcim časom klesá, ale mierny súvis badateľný je. Ak by naše skúmané portfólio obsahovalo aj podstatne staršie aplikácie, povedzme v rozmedzí od 1 až do 20 rokov, rozdiely by možno boli už markantnejšie. Ak v tomto prípade budeme posudzovať výnimočne aj spomínanú šesťnástu aplikáciu, o ktorej jej zástupca povedal, že si vyžaduje len minimum údržby, môžeme vysloviť záver, že množstvo úsilia na údržbu aplikácie s jej vekom klesá, avšak dobre viditeľné rozdiely možno registrovať až pri porovnávaní aplikácií s rozdielom vekov rádovo v desaťročiach.

4.9.8 Analýza spoločného vplyvu

Keďže v druhej časti výskumu sa nám nepodarilo preukázať vplyv viacerých faktorov na údržbu, domnievame sa, že ich vplyv je príliš slabý na to, aby sme ho v našom teste mohli preukázať. Avšak ak spojíme väčšie množstvo faktorov, ich spoločný vplyv už bude výraznejší a na výsledkoch skúmaných spoločností by sme ho mohli rozpoznať. Ako posledné sme vyhodnotili vplyv všetkých siedmych faktorov súčasne tak, že sme každej aplikácii spočítali body za predošlých sedem kategórií.

Zamerajme sa na rozloženie červených a zelených polí v predposlednom riadku tabuľky 4. Určite nemožno hovoriť o jednoznačnom vplyve, ale v rozložení farebných polí vidíme mierny posun zelených aplikácií vľavo. Domnievame sa, že tento vplyv by bol lepšie viditeľný, ak by sme poučení týmto dotazníkom vytvorili jeho novú vylepšenú verziu, v ktorej by sme predmetným témam venovali viac otázok. Tak by sme získali viac podrobností o každej zo skúmaných oblastí, podľa ktorých by sme vedeli vytvoriť presnejšie hodnotenie, ktoré by tieto aplikácie od seba viac rozlíšilo, a tým by bol možno skúmaný vplyv na udržovateľnosť markantnejší.

Celkové zhodnotenie výsledkov tejto časti výskumu popíšeme v nasledujúcej kapitole.

Kapitola 5

Zhodnotenie údržby

V tejto kapitole zhrnieme hlavné výsledky, ktoré sme získali vo výskume a podáme subjektívne posúdenie analyzovaných noriem a celkové posúdenie aktuálnej úrovne riadenia procesu údržby v spoločnostiach z výskumu.

5.1 Zhodnotenie výskumu

Prvým dôležitým úspechom výskumu bolo vytvorenie metriky, založenej na podstatných faktoroch údržovateľnosti aplikácií, o ktorej sme ukázali, že bude vhodná ako základ na posúdenie množstva úsilia, ktoré museli programátori hodnotených spoločností vynaložiť na údržbu ich aplikácií. Následne sme pristúpili k analýze procesu údržby v hodnotených spoločnostiach. Táto analýza pozostávala z dvoch častí

V prvej časti analýzy, rozdelenej do šiestich celkov sme skúmali, či proces údržby v softvérových spoločnostiach zahrňuje činnosti, ktoré sú definované v štandardoch ISO 14764 a ITIL. Na základe dotazníka sme každej spoločnosti priradili známky podľa toho, do akej miery spoločnosti spĺňali požiadavky štandardov pre každú zo šiestich častí. Následne sme skúmali, či u týchto spoločností existuje súvis medzi známkami a indexom udržovateľnosti. Pri všetkých šiestich procesoch sme ukázali, že vo veľkej väčšine prípadov dosiahnuté známky zodpovedali celkovej udržovateľnosti.

Na základe týchto výsledkov sme konštatovali, že riadenie údržby v softvérových spoločnostiach podľa pravidiel, ktoré sú obsiahnuté v štandardoch ISO 14764 a ITIL prináša spoločnostiam kvalitné návody, ktorých dodržiavanie pomôže spoločnosti efektívnejšie zvládnuť zmeny a tým zaručí spoločnostiam lepšiu udržovateľnosť ich aplikácií. Výsledky zároveň poukazujú na činnosti, ktoré sú väčšinou v prístupe spoločností k údržbe najviac zhodné s odporúčaniami noriem i na tie, ktoré sú zhodné najmenej.

V druhej časti analýzy sme skúmali vplyv siedmich faktorov na udržovateľnosť aplikácií. Podobne ako v prvej časti sme najskôr hodnoteným spoločnostiam priradili známku za každú kategóriu

a následne sme analyzovali, či u týchto spoločností existuje súvis medzi známkami a indexom udržovateľnosti. Podarilo sa nám ukázať, že kvalita softvérovej dokumentácie a vek aplikácie ma vplyv na udržovateľnosť. Nepodarilo sa nám preukázať vplyv ostatných piatich faktorov hlavne preto, že v týchto ukazovateľoch boli hodnotené spoločnosti veľmi podobné. Zároveň sme načrtli spôsob ako upraviť tento dotazník tak, aby bolo možné nepreukázané vplyvy preskúmať dôkladnejšie. Indície, že má zmysel sa o to pokúšať nám dáva záver poslednej analýzy, v ktorej sme ukázali mierny vplyv týchto faktorov, ak sme ich posudzovali všetky súčasne.

Výsledky tejto analýzy môžu v praxi pomôcť spoločnostiam, ktoré majú záujem zvyšovať kvalitu ich procesov, a to tak, že upriami pozornosť na činnosti, o ktorých sme dokázali, že majú preukázateľný vplyv na udržovateľnosť aplikácii, no mnohé spoločnosti im nevenujú dostatočnú pozornosť.

Ďalšie možnosti na rozvíjanie tejto práce vidíme napríklad v doplnení dotazníka o postup k vyhodnoteniu výsledkov alebo v rozšírení, prerobení a umiestnení tohto dotazníka na internet, tak ako to navrhujeme v závere tejto práce.

5.2 Posúdenie normatífov

5.2.1 Posúdenie štandardu ISO 14764

Tento štandard je neoddeliteľnou súčasťou množiny dokumentov ISO/IEC 12207 je preto nutné ho vnímať v kontexte tejto rodiny štandardov životného cyklu softvéru. V procese údržby zdôrazňuje predovšetkým udržovateľnosť softvérových produktov, potrebu modelov služieb údržby a potrebu plánu a stratégie údržby. Tento štandard však nešpecifikuje podrobnosti ako implementovať alebo vykonávať činnosti a úlohy zahrnuté v procese. Obsahuje ale množstvo príkladov zoznamov, ktoré môžu organizácie použiť ako základ pri vytváraní ich vlastného procesu údržby a podľa svojich potrieb patrične upraviť. Za hlavné prínosy tohto štandardu pokladáme:

- jasné definovanie činností a úloh pre primárny proces životného cyklu softvérovej údržby
- výstižné príklady postupov, ktoré je potrebné vykonať, aby mohli byť implementované základné údržbové aktivity a úlohy
- predloženie cenných úvah pre zavedenie procesu údržby, ktoré upozorňujú prípadných záujemcov na najdôležitejšie fakty, ktorými je nutné sa pri zavádzaní procesu údržby do organizácie zaoberať. Základná koncepcia tohto štandardu totiž tvorí iba hrubú kostru celého systému, ktorý je potrebný vytvoriť. Je dobrým základom, od ktorého sa môžu organizácie odvíjať a podľa vlastných potrieb dobudovať do takej podoby, aby výsledný proces presne vyhovoval podmienkam v danej organizácii. ISO 14764 im počas toho bude nápomocný hlavne cennými radami ohľadne:
 - včasnej angažovanosti údržbových požiadaviek vo vývoji
 - požiadavke monitorovania a hodnotenia kritérií udržovateľnosti vo všetkých fázach vývoja softvéru
 - stanovenia zmlúv o údržbe medzi dodávateľom a nadobúdateľom
 - merania softvérovej údržby
 - dokumentácii procesu
 - špecifických činností údržby v procese vývoja
 - vytvárania softvérovej dokumentácie
- návrh stratégie údržby, ktorý vhodne pojednáva o ľudských a materiálnych zdrojoch potrebných na poskytovanie softvérovej údržby pre jednotlivé produkty
- dôkladný koncept plánu údržby, v ktorom sa táto norma zameriava na všetky dôležité témy,

ktoré by mali byť zahrnuté v tomto pláne

Positívom toho štandardu je aj skutočnosť, že spoločná technická komisia ISO a IEC neustále pracuje na jeho ďalšom zlepšovaní, aby ešte presnejšie spĺňal požiadavky na neho kladené. Aktuálna verzia ISO 14764 bola vydaná v roku 2006. Podobne aj medzinárodný štandard ISO 12207, ktorý je vo veľmi úzkom vzťahu s ním, sa v roku 2008 dočkal nového rozšíreného vydania. Organizácie, ktoré budú chcieť prijať tento medzinárodný štandard, budú môcť onedlho siahnuť aj po jeho slovenskom ekvivalente, ktorý vydá Slovenský ústav technickej normalizácie ako Slovenskú technickú normu. Jej základom bude náš bilingválny preklad anglického originálu štandardu ISO 14764, ktorý vznikol ako vedľajší produkt tejto práce.

5.2.1 Posúdenie štandardu ITIL

Podobne ako štandard ISO 14764 nemožno vnímať osamote, ale v kontexte ostatných ISO štandardov životného cyklu softvéru, aj proces údržby podľa ITILu je časťou rozsiahleho rámca, ktorý popisuje spôsob riadenia IT služieb a infraštruktúry informačných a komunikačných technológií. To so sebou prináša isté výhody aj nevýhody. Nevýhodou v tomto prípade myslíme už spomenutý fakt, že ITIL treba vnímať ako veľký komplex množstva manažmentov, medzi jednotlivými procesmi je obrovské množstvo vzťahov a nie je možné vytrhnúť a implementovať z ITILu len tie procesy, ktoré priamo popisujú proces údržby. Z celkového pohľadu na ITIL je ale proces údržby veľmi kvalitne spracovaný, je logický a využiteľný tak pre menšie no v prvom rade pre veľké organizácie, ktorým prináša veľmi efektívny spôsob tak pre riadenie údržby ako aj pre podporu užívateľom, keďže vo filozofii ITIL sú tieto dva elementy vo veľmi úzkom vzťahu.

Podobne ako ISO 14764 používa aj ITIL pojem žiadosť o zmenu (hlásenie o probléme), líši sa však v prístupe k riešeniu týchto problémov v jednotlivých fázach ich životných cyklov.

Oproti ISO prístupu, zavádza ITIL pojem incidentu, a s tým súvisiaci manažment incidentov. V kritických situáciách, keď je postihnutá dostupnosť veľmi dôležitej služby, tento proces v kontexte s ostatnými procesmi riešiacimi zmeny a problémy v softvéri vytvára veľmi efektívny spôsob na promptne a maximálne účinné zásahy v snahe obnovenia dostupnosti postihnutej služby. Z výsledkov prieskumu totiž vyplýva, že približne tretina spoločností často rieši vzniknuté problémy formou núdzového riešenia a ďalšia tretina používa tento prístup vo vážnych situáciách.

Ďalším prínosom tohto prístupu je spracovanie témy vydávania nových verzií aplikácií, čo je vlastne logickým zavŕšením procesu implementovania zmeny alebo odstránenia problému. ITIL v tejto súvislosti zavádza manažment verzií, ktorým ponúka detailný návod ako tento problém riešiť, v úzkom vzťahu s manažmentmi zmien a konfigurácií.

5.3 Posúdenie aktuálnej úrovne riadenia údržby

Z výsledkov prieskumu vyplýva, že medzi hodnotenými spoločnosťami boli v úrovni riadenia údržby veľké rozdiely:

Zhruba v jednej tretine hodnotených spoločností bol proces údržby riadený na veľmi nízkej úrovni prípadne nebol riadený vôbec. Údržba v týchto spoločnostiach bola zvyčajne vykonávaná náhodne, dokonca až chaoticky. Spoločnosti neposkytovali stabilné prostredie na podporu tohto procesu. Ťažko bolo možné hovoriť o plánovaní, sledovaní a riadení činností údržby. Produkty týchto spoločností prekračovali rozpočet i termíny a spoločnosti nie sú schopné zopakovať prípadný úspech. Aj v celkovom hodnotení udržovateľnosti dosahovali aplikácie týchto spoločností najhoršie výsledky.

Približne 30% spoločností vykazovalo už základné znaky riadenia procesu údržby: Procesy v týchto spoločnostiach boli plánované, sledované a riadené pre individuálne projekty. Sledovali

dokumentovaný plán, mali odpovedajúce zdroje, boli určené zodpovednosti a právomoci počas vývoja, zodpovední ľudia boli školení a existovalo monitorovanie procesov s nadväznými opatreniami na odstránenie vznikajúcich nedostatkov. Stav prác na údržbe bol v každom čase známy pre vedenie firmy. Dodržiavanie týchto praktík malo v týchto spoločnostiach za následok menšie výkyvy v kvalite procesov, a tým aj v kvalite výsledného produktu. Ani v týchto spoločnostiach však nebol proces údržby v týchto spoločnostiach ešte dostatočne zadefinovaný a vo väčšine prípadov tieto spoločnosti nedosahovali uspokojivé výsledky v udržovateľnosti softvérových produktov.

Jasne zadefinovaný proces údržby malo asi iba 30% spoločností, čo sa takmer bez výnimky prejavilo aj na vysokej kvalite udržovateľnosti produktov týchto spoločností. Proces údržby týchto spoločností bol na rozdiel od predchádzajúcich dobre charakterizovaný, pochopený a bol opísaný v štandardoch, procedúrach a metódach. Spoločnosti mali vytvorené množiny štandardných procesov a postupne ich zlepšovali. Zatiaľ čo aj predchádzajúce spoločnosti mohli mať opisy procesov a procedúry odlišné pre rôzne projekty, tieto spoločnosti, na rozdiel od nich už všetky procesy odvádzali z množiny štandardných procesov spoločnosti a boli teda konzistentné. Procesy boli tiež definované presnejšie a obsiahlejšie, mali cieľ, vstup, vstupné podmienky, roly, kroky verifikácie, výstupy a podobne.

Vynikajúcu úroveň riadenia údržby dosiahla v prieskume len jediná spoločnosť. Proces údržby v tejto spoločnosti bol kvantitatívne riadený a optimalizovaný na základe štatistických alebo iných kvantitatívnych metód. Kvalita produktu a výkonnosť procesov bola sledovaná štatisticky a riadená počas celého životného cyklu. Za účelom riadenia boli postavené kvantitatívne ohodnotené ciele, bol určený spôsob merania výkonnosti procesov, kvality produktov a služieb. Proces údržby v tejto spoločnosti bol neustále zlepšovaný na základe znalosti podstatných príčin, spôsobujúcich odchýlky v procese. Technológie a spôsoby zlepšovania boli rozmiestnené systematicky v celej organizácii a úroveň zlepšenia bola merateľná a porovnávaná s kvantitatívnym cieľom.

Kapitola 6

Záver

V tejto práci sme čitateľa bližšie zoznámili s problematikou softvérovej údržby, vysvetlili sme jej význam údržby i motiváciu ku riadeniu údržby pomocou známych štandardov. Vybrali sme najdôležitejšie normy a opísali sme ich podstatu. Navrhli sme rozsiahly dotazník, ktorým sme oslovili široký okruh softvérových spoloční a získali sme veľké množstvo informácií potrebných k analýze. Našli sme spôsob, ktorým sme dokázali objektívne posúdiť a kvantifikovať mieru náročnosti údržby softvérových aplikácií. Následne sme skúmali a podarilo sa nám aj dokázať, že riadenie údržby podľa princípov analyzovaných noriem skutočne prinieslo spoločnostiam v prieskume dobré výsledky pri údržbe ich aplikácií. Naproti tomu spoločnosti, ktorých riadenie vôbec nezodpovedalo zásadám noriem, dosahujú v praxi vo väčšine prípadov o poznanie horšie výsledky a na údržbu musia vynaložiť oveľa viac úsilia. Podarilo sa nám teda dosiahnuť hlavný cieľ práce.

V druhej časti výskumu sa nám podarilo dokázať, že kvalita dokumentácie a vek aplikácie tiež vplyvajú množstvo úsilia vynakladaného na údržbu. Pri ďalších analyzovaných okolnostiach sa nám vplyv nepodarilo dokázať, ale ani vyvrátiť. Navyše, niektoré indície, ktoré v závere výskumu popisujeme, identifikujú pravdepodobné príčiny neúspechu a ukazujú možnosti, ako ešte viac skvalitniť dotazník i analýzu, a tým dokázať prítomnosť skúmaných vzťahov.

Výsledky tejto analýzy môžu v praxi pomôcť všetkým spoločnostiam, ktoré majú záujem zvyšovať kvalitu ich procesov, a to tak, že upriami pozornosť na činnosti, o ktorých sme dokázali, že majú preukázateľný vplyv na udržovateľnosť aplikácii, no mnohé spoločnosti im nevenujú dostatočnú pozornosť.

Ďalším prínosom tejto práce je vytvorenie základu pre budúcu Slovenskú technickú normu o údržbe softvéru, ktorú vydá Slovenský ústav technickej normalizácie podľa prekladu, ktorý sme ako vedľajší produkt vytvorili počas práce na tejto analýze.

Možnosti na ďalšie rozvíjanie tejto práce vidíme napríklad v doplnení dotazníka o postup k vyhodnoteniu výsledkov: Ľubovoľný manažér by tak pomocou tohto návodu vedel za svoju spoločnosť určiť známky jednotlivým častiam dotazníka, ktoré by potom mohol porovnať s výsledkami uvedenými v tejto práci. Iný spôsob ako pokračovať v práci je rozšíriť množstvo

analyzovaných faktov, u ktorých môžeme predpokladať vplyv na udržovateľnosť. K témam, u ktorých sa nám nepodarilo preukázať ich vplyv, sa môže rozšíriť množstvo otázok, ktoré presnejšie zmapujú daný stav v spoločnostiach. Vďaka detailnejším poznatkom o spoločnostiach sa možno podarí dokázať vplyv ďalších faktorov na udržovateľnosť. Dotazník by sme mohli prerobiť do webového prostredia a z doterajších i budúcich dotazníkov by mohla vzniknúť databáza. Jej rozširovanie by prinieslo ešte lepšie možnosti pre analýzu výsledkov spoločnosti, ktoré bude chcieť on-line zhodnotiť a porovnať svoj proces údržby a udržovateľnosť ich aplikácií s inými spoločnosťami.

Slovník

Incident

Akákoľvek udalosť, ktorá nie je časťou bežnej prevádzky služby a spôsobuje alebo môže spôsobiť prerušenie alebo obmedzenie služby

Work-around

postup, ako sa vyhnúť incidentu alebo problému buď dočasným fixovaním alebo ošetrením systému tak, že používateľ nie je závislý na tej konkrétnej časti služby, o ktorej vieme, že má problém. (Analogia s havarijnou nápravou /emergency change/ z ISO 14764)

Known-error

incident alebo problém, ktorého príčina je už známa, vyvinul sa dočasný Work-around alebo dočasná náhrada. Known-error bude pravdepodobne onedlho definitívne opravený.

Konfiguračná položka (Configuration item - CI)

je aktívum IT alebo sústava viacerých IT aktív, ktoré buď nejakým spôsobom závisia na ostatných IT procesoch alebo s nimi majú nejaký vzťah. Konfiguračné položky spravuje konfiguračná databáza CMDB.

Databáza manažmentu konfigurácií (CMDB)

CMDB (Configuration Management Database) je konfiguračná databáza, ktorá je kľúčovou časťou manažmentu konfigurácií obsahujúcou informácie o všetkých konfiguračných položkách tvoriacich ICT infraštruktúru a o vzťahoch medzi nimi vrátane odkazov na súvisiace záznamy z ostatných oblastí ITSM.

Change Advisory Board (CAB)

je skupina ľudí, ktorí dávajú odborné rady manažmentu konfigurácii ohľadom implementácie zmien. Táto komisia sa skladá zo zástupcov všetkých oblastí, tak z IT, ako aj z obchodného oddelenia. V prípade veľmi urgentného problému, keď nie je dostatok času na zvolanie celej rady, zvolá sa stretnutie len kľúčových členov tejto rady, ktorí tvoria tzv. havarijnú komisiu (emergency committee) CAB/EC.

Základňa (baseline)

Produkt alebo špecifikácia, ktorý bol formálne zhodnotený a schválený a ktorý následne na to slúži ako základ k ďalšiemu vývoju, pričom platí, že zmeny na neho možno aplikovať iba cez formálne procedúry na riadenie zmeny .

Tranzícia

Tranzícia alebo prechod softvéru je riadená a koordinovaná postupnosť činností, keď vývoj softvéru prechádza z organizácie, ktorá vykonávala počiatočný vývoj softvéru, na organizáciu vykonávajúcu softvérovú údržbu.

Žiadosť o úpravu (Modification Request)

Je všeobecným výrazom používaným na identifikáciu navrhovaných úprav softvérového produktu, ktorý sa práve nachádza v procese údržby. Môže mať formu napríklad screenshotu alebo formulára, slúži na zachytenie detailov žiadosti o úpravu niektorého z prvkov infraštruktúry, ktoré spadajú pod

manažment konfigurácií. Žiadosť o úpravu môže byť neskôr klasifikovaná ako oprava alebo vylepšenie a rozpoznaná ako opravná, preventívna, adaptívna alebo zdokonaľujúca údržba. V ďalšom texte budeme niekedy na označenie žiadosti o úpravu používať skratku MR.

Hlásenie o probléme (Problem Report)

Je výrazom používaným na rozpoznanie a popísanie problémov odhalených v softvérovom produkte. Hlásenia o probléme sú buď postúpené priamo, aby označovali chyby alebo sú vytvorené potom, ako je vykonaná analýza žiadostí o úpravu a objavia sa chyby. V ďalšom texte budeme niekedy na označenie Hlásenia o probléme používať skratku PR.

Literatúra

- [1] ISO/IEC JTC1/SC7/WG7. ISO/IEC 14764:2006(E): *Software Engineering — Software Life Cycle Processes — Maintenance*. 2006. ISBN 0-7381-4961-6 SS95534
- [2] ISO/IEC JTC1/SC7/WG7. ISO/IEC FDIS 12207: 2007(E): *Systems and Software Engineering — Software Life Cycle Processes*. 2007
- [3] Office of Government Commerce. *ITIL – The key to managing IT Services - Service Support*. Verzia 2.1. 2000. ISBN 0113308671
- [4] Office of Government Commerce. *ITIL – The key to managing IT Services - Service Delivery*. Verzia 2.0. 2003. ISBN 0113308930
- [5] Penny Grubb, Armstrong A Takang. *Software maintenance – Concepts and Practice, (2nd edition)*. World Scientific Publishing Co. Pte. Ltd. 2003. ISBN 981-238-425-1
- [6] Alain Abran, James W. Moore. *Svebok : Guide to Software Engineering Body of Knowledge*. 2004 Version. IEEE Computer Society. 2004. ISBN 0-7695-2330-7
- [7] Ian Sommerville. *Software Engineering* 6th edition, Chapter 27, 2000.
- [8] Roger S. Pressman. *Software engineering: a practitioner's approach – 5th. edition*. Kapitoly: 2, 4, 19, 30. McGraw-Hill. 2001. ISBN 0073655783
- [9] Fadi P. Deek. *Strategic Software Engineering - An Interdisciplinary Approach*. Auerbach Publications. 2005. ISBN 0-8493-3939-1
- [10] Len Bass, Paul Clements, Rick Kazman. *Software Architecture in Practice* (Second edition). Addison Wesley. 2003. ISBN 0321154959
- [11] Mária Bieliková. *Softvérové inžinierstvo - Princípy a manažment*. Vydavateľstvo STU, Bratislava. 2000. ISBN 80-227-1322-8
- [12] Ľubomír Majtás. *Otvorený softvér a udržiavateľnosť. Zborník esejí*. FIIT STU. 2005 <http://www2.fiit.stuba.sk/~bielik/courses/msi-slov/kniha/2005/group1/essay/7.pdf>
- [13] Pavol Gono. *Udržovateľnosť a Open Source Software. Zborník esejí*. FIIT STU. 2005 <http://www2.fiit.stuba.sk/~bielik/courses/msi-slov/kniha/2005/group2/essay/7.pdf>
- [14] Mária Bieliková, Pavol Návrat a kol. *Štúdie vybraných tém softvérového inžinierstva*. Vydavateľstvo STU, Bratislava. 2006.
- [15] Edmond VanDoren. *Maintainability index technique for measuring program maintainability*. *Software Technology Review*, SEI. 1997. www.sei.cmu.edu/str/descriptions/mitmpm_body.html.
- [16] Bianka Kováčová. *Zabezpečovanie a zvyšovanie kvality pomocou CMMI*. STU Bratislava. 2008. <http://www2.fiit.stuba.sk/~bielik/courses/msi-slov/kniha/2008/ssays/msipapersource32-kovacova.pdf>
- [17] Zuzana Halanová. *Manažment projektov: teória vs. Prax*. STU Bratislava. 2005. <http://www2.fiit.stuba.sk/~bielik/courses/msi-slov/kniha/2005/group3/essay/17.pdf>

- [18] OCG. *Information Technology Service Management*. oficiálna domovská web stránka. <http://www.itsm-officialsite.com/home/home.asp>
- [19] Omnicom. *Information Technology Service Management & Information Technology Infrastructure Library*. web stránka http://www.itsm.sk/sk/00_news/index.htm
- [20] Frank Niessink, Viktor Clerc. *IT Service CMM Questionnaire*.
<http://itservicecmmwebsite.googlepages.com/quest-0.3.pdf>
- [21] ISBSG. *Maintenance & Support Data Collection Questionnaire*
<http://www.isbsg.org/html/m&s%20data%20collection%20form.zip>
- [22] Szabolcs Molnár. *Meranie v softvérovom inžinierstve - Meranie pri testovaní, prevádzke a údržbe softvérových systémov*. STU Bratislava. 2001.
<http://www2.fiiit.stuba.sk/~bielik/courses/msi-slov/kniha/2001/meranie.pdf>
- [23] Peter Jehlár. *Softvérové inžinierstvo - etický kódex a procesy vývoja softvéru: Klasické modely*. STU Bratislava. 2000. <http://www2.fiiit.stuba.sk/~bielik/courses/msi-lov/kniha/2000/gwain.pdf>
- [24] Peter Bober. *Modely a štandardy pre zlepšovanie procesov v malých softvérových firmách*. FEI TUKE <http://lpi.fei.tuke.sk/kip/2000/1/bober/bober.html>

Príloha A DOTAZNÍK SOFTVÉROVEJ ÚDRŽBY

Každá firma aj každé IT oddelenie je jedinečné, vytvára jedinečné aplikácie, ktoré by sa dali popisovať a merať tisíc spôsobmi. Napriek tomu sa tento dotazník pokúsi porovnať kvalitu údržby rôznych aplikácií.

Každé IT oddelenie funguje svojím osobitným spôsobom, preto sa na úvod musíme dohodnúť na jednom spoločnom a tým pádom veľmi zjednodušenom modeli.

Hlavné zjednodušenie je v tom, že sa obmedzíme iba na **jednu aplikáciu**, ktorú vyvíjate. Náš zjednodušený model by mohol byť nasledovný: IT oddelenie má 10 členov (analytici, programátori, tester...). Všetci pracujú na trvalý pracovný úväzok. Vyvíjajú aplikáciu (vyberte si jednu), ktorú po dokončení buď predajú alebo ju vaša spoločnosť sama používa (napr. firemný informačný systém). V každom prípade po začatí prevádzky sa ich práca s touto aplikáciou nekončí. Objavia sa rôzne chyby, program nerobí to čo má, užívatelia požadujú doplniť ďalšiu funkcionálnosť, volajú na IT s rôznymi otázkami a žiadosťami... Práve tieto činnosti nazývame **údržbou a podporou aplikácie**. IT môže vyvíjať aj ďalšie nové aplikácie, ale základný predpoklad pre tento dotazník je ten, že rieši podporu tejto vybranej aplikácie!

Takmer všetky otázky sa preto budú týkať iba tejto jednej zvolenej aplikácie.

Prvá podmnožina otázok sa snaží získať základné údaje o skúmanej aplikácii, ako napr. veľkosť, typ, zložitosť, použité programovacie jazyky, dôležitosť pre obchodné procesy spoločnosti (hlavná podnikateľská činnosť organizácie, ktorá prináša zisky) a pod.

Druhá časť otázok je snaží zistiť, koľko času (úsilia) venujete údržbe a podpore aplikácie.

Údržbou sa v tomto prípade myslí úprava aplikácie, ktorá môže mať 4 dôvody:

1. oprava chýb
2. rozširovanie aplikácie o novú funkcionálnosť
3. zmenilo sa prostredie (HW, OS...), je potrebné túto aplikáciu prispôbiť, upraviť
4. Proaktívne zmeny: reorganizovanie kódov, zvyšovanie výkonu, dopĺňanie dokumentácie

Podporou užívateľom sa myslí predovšetkým:

1. skúmanie, či ich nahlásené problémy sú chybami aplikácie alebo len chybami užívateľa
2. radenie užívateľom s ich problémami (HELP DESK, telefón, mail, osobne), pomáhajúce užívateľom v ich práci rôznymi formami

Základom pre vaše hodnotenie je stanoviť istý počiatočný a koncový dátum, ktoré ohraničia obdobie, za ktoré budeme sledovať množstvo vykonanej práce na údržbách. Ideálne je obdobie trvajúce 1 rok, napr. rok 2007, alebo od apríla 2007 do dnes.

Nasleduje trochu zložitá časť s cieľom zistiť, koľko úsilia venovalo údržbe vaše IT oddelenie za stanovené obdobie. Použijeme známe vyjadrovanie náročnosti projektu v jednotkách z anglického man-days čiže „**človeko-deň**“. Je to množstvo práce, ktoré vykoná jeden programátor za jeden deň.

Potrebujeme poznať celkové množstvo práce **za všetkých** IT pracovníkov, ktoré venovali údržbe aplikácie **za obdobie 1 roka** (hodnotený interval).

Napríklad údržbe vybranej aplikácie sa venovalo 8 programátorov, z toho 4 všetok svoj pracovný čas a 4 iba polovicu (zvyšok času robili niečo iné). Sledované obdobie má 200 pracovných dní, takže spolu vykonali prácu $(4 + 4 \cdot (1/2)) \cdot 200 = 1200$ človeko-dní.

V ďalšej fáze sa pokúste sa určiť, koľko z tejto práce pripadalo na údržbu a koľko na podporu užívateľom. Napokon skúste ešte tieto hodnoty **rozdeliť podľa typov údržby**, ktorú ste vykonávali. Napr.:

údržba = 1000 ČH (oprava chýb 200, pridávanie novej funkcionality 500, prispôsobovanie 200, reorganizovanie kódov, dopisovanie dokumentácie 100)

podpora = 200 ČH (pomoc užívateľom 100, skúmanie ich sťažností a pod...100)

Prvú polovicu dotazníka uzatvárajú ešte otázky, ako napr. počet chýb, ktoré ste za sledované obdobie odstraňovali a otázky na dostupnosť a kvalitu dokumentácie.

Druhá polovica otázok je už rýchla, zaklikávacia a zisťuje, akým spôsobom pristupuje k údržbe vaše IT oddelenie.

<p>1. Aká je vaša súčasná pracovná pozícia v spoločnosti ?</p> <p> <input type="checkbox"/> Technická pozícia <input type="checkbox"/> Vývojárska pozícia <input type="checkbox"/> Analytická pozícia <input type="checkbox"/> Konzultantské pozície <input type="checkbox"/> Manažerské <input type="checkbox"/> Vedúce pozície <input type="checkbox"/> Iné pozície aj s IT znalosťami </p>	
<p>2. Uvedte prosím konkrétnejšie vašu pozíciu</p> <p> <input type="checkbox"/> IT manažér <input type="checkbox"/> Manažér kvality <input type="checkbox"/> Manažér údržby <input type="checkbox"/> Manažér podpory <input type="checkbox"/> Team leader <input type="checkbox"/> Vývojový pracovník <input type="checkbox"/> Manažér vývoja Iné: <input type="text"/> </p>	
<p>3. V akej oblasti pracujú koncoví užívatelia softvéru vašej spoločnosti ?</p> <p> <input type="checkbox"/> Elektrotechnický <input type="checkbox"/> Obchod <input type="checkbox"/> Informačné technológie <input type="checkbox"/> Poradenstvo <input type="checkbox"/> Reklama, propagácia a <input type="checkbox"/> Telekomunikácie <input type="checkbox"/> Cestovný ruch <input type="checkbox"/> Štátna správa <input type="checkbox"/> Finančníctvo, bankovníctvo, poisťovníctvo Iná: <input type="text"/> </p>	
<p>4. Interval hodnoteného obdobia</p> <p>Počiatkový dátum hodnoteného obdobia: <input type="text"/></p> <p>Koncový dátum hodnoteného obdobia: <input type="text"/></p>	<p>Odporúčame zvoliť dĺžku sledovaného obdobia minimálne 12 mesiacov. Pri zadaní kratšieho obdobia budú pred analýzou dáta primerane prepočítané.</p>

<p>5. Typ hodnotenej aplikácie</p> <p> <input type="checkbox"/> 3D modelovanie alebo animácie <input type="checkbox"/> katalóg alebo register predmetov <input type="checkbox"/> ovládač pre zariadenie alebo rozhranie <input type="checkbox"/> spracovanie obrazu, zvuku alebo videa <input type="checkbox"/> programy pre oddych a relax (hry) <input type="checkbox"/> logistika alebo podpora plánovania a riadenia <input type="checkbox"/> grafický alebo vydavateľský systém <input type="checkbox"/> účtovníctvo a spracovanie finančných transakcií <input type="checkbox"/> geografické informačné systémy a GPS <input type="checkbox"/> operačný systém alebo systémový nástroj <input type="checkbox"/> správa dokumentov <input type="checkbox"/> matematické <input type="checkbox"/> multimediálna utilita <input type="checkbox"/> antivírusový program </p> <p>Iný: <input type="text"/></p>	<p>Aplikácia, ktorú vyvíjate. Znalosť typu aplikácie môže pomôcť určiť náročnosť údržby a užívateľskej podpory pre túto aplikáciu.</p>
<p>6. Popis hodnotenej aplikácie</p> <p><input type="text"/></p>	<p>Stručne popíšte, čo vaša aplikácia robí.</p>
<p>7. Veľkosť hodnotenej aplikácie</p> <p>veľkosť aplikácie v KSLOC: <input type="text"/></p> <p>veľkosť aplikácie v FP: <input type="text"/></p>	<p>Veľkosť aplikácie je hlavný faktor k posúdeniu úsilia o údržbu. KSLOC sú tisíce riadkov zdrojového kódu, komentáre sa nezapočítavajú. FP (Feature Points) je množstvo funkčných bodov, ktoré aplikácia obsahuje.</p>
<p>8. Dátum spustenia aplikácie</p> <p>dátum spustenia prevádzky aplikácie: <input type="text"/></p>	<p>Dátum uvedenia aplikácie do prevádzky. Služí k určeniu veku aplikácie</p>
<p>9. Dôležitosť aplikácie pre obchodné procesy</p> <p>menej dôležité procesy <input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> nevie</p> <p>viac dôležité procesy <input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> nevie</p> <p>kritické procesy <input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> nevie</p>	<p>Niektoré obchodné procesy sú veľmi dôležité, iné menej. Zhodnoťte nakoľko dôležitá je pre tieto procesy vaša aplikácia.</p> <p>Áno znamená, že aplikácia je pre tieto procesy dôležitá.</p>
<p>10. Verzie produktu</p> <p>počet podporovaných verzií produktu: <input type="text"/></p> <p>počet podporovaných OS: <input type="text"/></p> <p>počet podporovaných DB systémov: <input type="text"/></p>	<p>Čím vyšší počet aktuálne podporovaných verzií, OS a DB systémov, tým vyššie možno očakávať úsilie potrebné na údržbu. DB (oracle, MSSQL,....)</p>

11. Dostupnosť aplikácie <input type="text"/> hodín/deň <input type="text"/> dní/týždeň	Koľko hodín do dňa s aplikáciou pracujú používatelia. Napr. v knižnici sa používa SW len 8 hodín do dňa a 5 dní do týždňa.																								
12. Počet koncových užívateľov <input type="text"/> užívateľov.	Celkový počet koncových užívateľov aplikácie. (nepočítajú sa internetoví používatelia)																								
13. Skúsenosť užívateľov <input type="checkbox"/> hodne skúsení (2 ročná prax) <input type="checkbox"/> zbežne skúsení <input type="checkbox"/> príležitostní užívatelia <input type="checkbox"/> úplne neskúsení (laická verejnosť)	Zvoľte úroveň skúseností priemerného užívateľa alebo väčšiny užívateľov. Skúsenosť užívateľov s aplikáciou pomôže určiť, nakoľko by si mohli byť svojpomocní pri istých problémoch.																								
14. Spokojnosť užívateľov organizácie : bol niekedy vykonaný prieskum spokojnosti užívateľov ? <input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> nevie	Prieskum spokojnosti zákazníkov je dotazník s otázkami ohľadom spokojnosti s aplikáciou.																								
15. Celkový počet zamestnancov, ktorí sa starajú o údržbu a podporu <input type="text"/> ľudí	Aktuálny počet všetkých zamestnancov (nie len zamestnaných na plný pracovný úväzok), ktorí sa starajú o údržbu a podporu aplikácie.																								
16. Fluktuácia oddelenia podpory a údržby v percentách <input type="text"/> percent	Počet neplánovaných odchodov zamestnancov z osobných dôvodov, ktorý môže mať nepriaznivý dopad na úroveň poskytovaných služieb.																								
17. Úroveň znalosti členov tímu pre údržbu a podporu <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th></th> <th colspan="3" style="text-align: center;">úroveň znalosti</th> </tr> <tr> <th></th> <th style="text-align: center;">< 1 rok</th> <th style="text-align: center;">1-5 rokov</th> <th style="text-align: center;">> 5 rokov</th> </tr> </thead> <tbody> <tr> <td>aplikácia</td> <td style="text-align: center;"><input type="text"/> %</td> <td style="text-align: center;"><input type="text"/> %</td> <td style="text-align: center;"><input type="text"/> %</td> </tr> <tr> <td>technológie</td> <td style="text-align: center;"><input type="text"/> %</td> <td style="text-align: center;"><input type="text"/> %</td> <td style="text-align: center;"><input type="text"/> %</td> </tr> <tr> <td>prog. jazyk</td> <td style="text-align: center;"><input type="text"/> %</td> <td style="text-align: center;"><input type="text"/> %</td> <td style="text-align: center;"><input type="text"/> %</td> </tr> <tr> <td>obch. činnosť</td> <td style="text-align: center;"><input type="text"/> %</td> <td style="text-align: center;"><input type="text"/> %</td> <td style="text-align: center;"><input type="text"/> %</td> </tr> </tbody> </table>		úroveň znalosti				< 1 rok	1-5 rokov	> 5 rokov	aplikácia	<input type="text"/> %	<input type="text"/> %	<input type="text"/> %	technológie	<input type="text"/> %	<input type="text"/> %	<input type="text"/> %	prog. jazyk	<input type="text"/> %	<input type="text"/> %	<input type="text"/> %	obch. činnosť	<input type="text"/> %	<input type="text"/> %	<input type="text"/> %	Percentuálne zhodnoťte úroveň skúsenosti IT personálu s aplikáciou, použitými technológiami a prog. jazykmi a oblasťou obchodu organizácie. Napr. 30% programátorov má skúsenosti s SQL viac ako 5 ročné....
	úroveň znalosti																								
	< 1 rok	1-5 rokov	> 5 rokov																						
aplikácia	<input type="text"/> %	<input type="text"/> %	<input type="text"/> %																						
technológie	<input type="text"/> %	<input type="text"/> %	<input type="text"/> %																						
prog. jazyk	<input type="text"/> %	<input type="text"/> %	<input type="text"/> %																						
obch. činnosť	<input type="text"/> %	<input type="text"/> %	<input type="text"/> %																						
18. Dĺžka pracovnej doby <input type="text"/> hodín/mesiac	Mesačná dĺžka pracovnej doby v hodinách u plného pracovného úväzku jedného zamestnanca oddelenia, ktorý sa stará o údržbu a podporu.																								

<p>19. Celkové úsilie vynaložené SPOLU na údržbu a podporu</p> <p><input type="text"/> človeko-dní/rok</p>	<p>Vysvetlené v úvode. Ročne množstvo úsilia všetkých zainteresovaných na údržbu a podporu. Nasledujúce dve otázky rozdeľujú túto hodnotu na 2 časti.</p>
<p>20. Celkové úsilie vynaložené na všetky údržbové aktivity</p> <p><input type="text"/> človeko-dní/rok</p>	<p>Úsilie na všetky druhy údržby. Ak je to možné, vyplňte aj otázky 22, 23, 24, 25, tak, aby ich súčet bol rovný práve tomuto číslu.</p>
<p>21. Celkové úsilie vynaložené na užívateľskú podporu</p> <p><input type="text"/> človeko-dní/rok</p>	<p>Úsilie na všetky druhy podpory užívateľom. Ak je to možné, vyplňte aj otázky 26,27,28, tak, aby ich súčet bol rovný práve tomuto číslu.</p>
<p>22. Celkové úsilie vydané na činnosti <i>opravnej údržby</i></p> <p><input type="text"/> človeko-dní/rok</p>	<p>Opravná údržba predstavuje činnosti podnikané kvôli oprave chýb v programe. Sú to modifikácie programu po jeho odovzdaní do prevádzky, ktoré opravujú vzniknuté problémy.</p>
<p>23. Celkové úsilie vydané na <i>prispôsobovanie aplikácie</i></p> <p><input type="text"/> človeko-dní/rok</p>	<p>Prispôsobenie sa chápe ako zmena, ktorá je potrebná, aby sa systém adaptoval zmenám prostredia, v ktorom sa daný softvérový systém nachádza.</p>
<p>24. Celkové úsilie vynaložené na <i>vylepšovanie aplikácie</i></p> <p><input type="text"/> človeko-dní/rok</p>	<p>Zmeny softvéru, ktoré pridávajú novú funkcionality, vylepšujú produkt, resp. jeho vlastnosti, ktoré neboli zahrnuté do špecifikácie pôvodného systému.</p>
<p>25. Celkové úsilie vynaložené na <i>preventívne zmeny</i></p> <p><input type="text"/> človeko-dní/rok</p>	<p>Preventívne zmeny sú podnikané s cieľom predísť poruchám (napr.: preorganizovanie kódov, optimalizácia kódov, aktualizovanie dokumentácie...)</p>
<p>26. Úsilie vynaložené na <i>vyšetrovanie problémov</i></p> <p><input type="text"/> človeko-dní/rok</p>	<p>Tieto činnosti predstavuje skúmanie, či hlásený problém je chyba aplikácie, chyba v dokumentácii, v školení alebo len chyba používateľa.</p>
<p>27. Úsilie vynaložené na <i>poskytovanie pomoci a rád užívateľom</i></p> <p><input type="text"/> človeko-dní/rok</p>	<p>Toto sú činnosti týkajúce sa podpory užívateľom poskytovaním pomoci, nápovedy a poradením.</p>

<p>28. Úsilie vynaložené na otázky a rýchle služby pre užívateľov</p> <p style="text-align: center;"><input type="text"/> človeko-dní/rok</p>	<p>Odpovedanie na ad hoc otázky a podobné služby užívateľom aplikácie</p>																																																
<p>29. Počet chýb za sledované obdobie</p> <p>extrémne chyby: <input type="text"/></p> <p>významné chyby: <input type="text"/></p> <p>drobné chyby: <input type="text"/> alebo</p> <p>celkový počet chýb: <input type="text"/></p>	<p>Celkový počet chýb na aplikáciách v jednotlivých kategóriách za hodnotené obdobie.</p>																																																
<p>30. Počet tel. volaní ohľadom aplikácie</p> <p style="text-align: center;"><input type="text"/> hovorov</p>	<p>Celkový počet hovorov na Help Desk za sledované obdobie.</p>																																																
<p>31. Priemerný počet dní na opravu</p> <p style="text-align: center;"><input type="text"/> dní</p>	<p>Priemerná doba, ktorá uplynie od nahlásenia chyby po jej opravu.</p>																																																
<p>32. Je v rámci organizácie zavedený manažment konfigurácií?</p> <p><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> nevie</p>	<p>manažment konfigurácií (Configuration management) je alebo nie je v organizácii zavedený.</p>																																																
<p>33. Programovacie jazyky</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%;">primárny</td> <td style="width: 15%;"><input type="text"/></td> <td style="width: 15%;"><input type="text"/></td> <td style="width: 15%;">(KSLOC)</td> <td style="width: 15%;"><input type="text"/></td> <td style="width: 10%;">(%)</td> </tr> <tr> <td>sekundárny</td> <td><input type="text"/></td> <td><input type="text"/></td> <td>(KSLOC)</td> <td><input type="text"/></td> <td>(%)</td> </tr> <tr> <td>terciárny</td> <td><input type="text"/></td> <td><input type="text"/></td> <td>(KSLOC)</td> <td><input type="text"/></td> <td>(%)</td> </tr> <tr> <td>d'alší</td> <td><input type="text"/></td> <td><input type="text"/></td> <td>(KSLOC)</td> <td><input type="text"/></td> <td>(%)</td> </tr> <tr> <td>d'alší</td> <td><input type="text"/></td> <td><input type="text"/></td> <td>(KSLOC)</td> <td><input type="text"/></td> <td>(%)</td> </tr> <tr> <td>d'alší</td> <td><input type="text"/></td> <td><input type="text"/></td> <td>(KSLOC)</td> <td><input type="text"/></td> <td>(%)</td> </tr> <tr> <td>d'alší</td> <td><input type="text"/></td> <td><input type="text"/></td> <td>(KSLOC)</td> <td><input type="text"/></td> <td>(%)</td> </tr> <tr> <td>d'alší</td> <td><input type="text"/></td> <td><input type="text"/></td> <td>(KSLOC)</td> <td><input type="text"/></td> <td>(%)</td> </tr> </table>	primárny	<input type="text"/>	<input type="text"/>	(KSLOC)	<input type="text"/>	(%)	sekundárny	<input type="text"/>	<input type="text"/>	(KSLOC)	<input type="text"/>	(%)	terciárny	<input type="text"/>	<input type="text"/>	(KSLOC)	<input type="text"/>	(%)	d'alší	<input type="text"/>	<input type="text"/>	(KSLOC)	<input type="text"/>	(%)	d'alší	<input type="text"/>	<input type="text"/>	(KSLOC)	<input type="text"/>	(%)	d'alší	<input type="text"/>	<input type="text"/>	(KSLOC)	<input type="text"/>	(%)	d'alší	<input type="text"/>	<input type="text"/>	(KSLOC)	<input type="text"/>	(%)	d'alší	<input type="text"/>	<input type="text"/>	(KSLOC)	<input type="text"/>	(%)	<p>Primárny programovací jazyk je ten, ktorý je použitý u najväčšieho počtu aplikácií portfólia. KSLOC je množstvo k zdrojového kódu bez komentárov v tisícoch. Namiesto KSLOC môžete uviesť podiel použitých jazykov v percentách.</p>
primárny	<input type="text"/>	<input type="text"/>	(KSLOC)	<input type="text"/>	(%)																																												
sekundárny	<input type="text"/>	<input type="text"/>	(KSLOC)	<input type="text"/>	(%)																																												
terciárny	<input type="text"/>	<input type="text"/>	(KSLOC)	<input type="text"/>	(%)																																												
d'alší	<input type="text"/>	<input type="text"/>	(KSLOC)	<input type="text"/>	(%)																																												
d'alší	<input type="text"/>	<input type="text"/>	(KSLOC)	<input type="text"/>	(%)																																												
d'alší	<input type="text"/>	<input type="text"/>	(KSLOC)	<input type="text"/>	(%)																																												
d'alší	<input type="text"/>	<input type="text"/>	(KSLOC)	<input type="text"/>	(%)																																												
d'alší	<input type="text"/>	<input type="text"/>	(KSLOC)	<input type="text"/>	(%)																																												
<p>34. Prevládajúca štruktúra kódu</p> <p><input type="checkbox"/> jednoducho a ľahko čitateľný a udržiavateľný kód za pomoci CASE prostriedkov a</p> <p><input type="checkbox"/> dobre organizovaný štruktúrovaný kód</p> <p><input type="checkbox"/> slabo organizovaný alebo z veľkej časti neštruktúrovaný kód</p> <p><input type="checkbox"/> chýbajúce bloky kódu</p>	<p>Podľa očakávaní jednoduchá práca s kódom znamená nižšie náklady na údržbu.</p>																																																

<p>35. Prevládajúca zložitosť algoritmov</p> <p><input type="checkbox"/> jednoduché logické a matematické algoritmy</p> <p><input type="checkbox"/> priemerne zložité logické a matematické algoritmy</p> <p><input type="checkbox"/> zložité logické a matematické algoritmy</p>	<p>Podľa očakávaní jednoduché algoritmy znamenajú nižšie náklady na údržbu.</p>																																																																												
<p>36. Dostupnosť a kvalita dokumentácie</p> <p style="text-align: center;">užitočnosť</p> <table border="0" style="width: 100%;"> <thead> <tr> <th style="width: 60%;"></th> <th style="width: 10%; text-align: center;">nízka</th> <th style="width: 10%; text-align: center;">priem.</th> <th style="width: 10%; text-align: center;">dobrá</th> </tr> </thead> <tbody> <tr><td><input type="checkbox"/> žiadna</td><td></td><td></td><td></td></tr> <tr><td><input type="checkbox"/> obchodný prípad (business)</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td><input type="checkbox"/></td></tr> <tr><td><input type="checkbox"/> dokument koncepcie systému</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td><input type="checkbox"/></td></tr> <tr><td><input type="checkbox"/> špecifikácia požiadavok</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td><input type="checkbox"/></td></tr> <tr><td><input type="checkbox"/> funkčná špecifikácia</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td><input type="checkbox"/></td></tr> <tr><td><input type="checkbox"/> technická špecifikácia</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td><input type="checkbox"/></td></tr> <tr><td> </td><td></td><td></td><td></td></tr> <tr><td><input type="checkbox"/> popis užívateľského rozhrania</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td><input type="checkbox"/></td></tr> <tr><td><input type="checkbox"/> užívateľská príručka (manuál)</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td><input type="checkbox"/></td></tr> <tr><td><input type="checkbox"/> entitno-relačný model</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td><input type="checkbox"/></td></tr> <tr><td><input type="checkbox"/> diagram dátových tokov</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td><input type="checkbox"/></td></tr> <tr><td><input type="checkbox"/> plány a skripty na testovanie</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td><input type="checkbox"/></td></tr> <tr><td><input type="checkbox"/> testovacie dáta</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td><input type="checkbox"/></td></tr> <tr><td><input type="checkbox"/> popis verzií (release)</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td><input type="checkbox"/></td></tr> <tr><td><input type="checkbox"/> zoznam incidentov / závad</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td><input type="checkbox"/></td></tr> <tr><td><input type="checkbox"/> manuál pre administrátorov</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td><input type="checkbox"/></td></tr> <tr><td><input type="checkbox"/> školiace manuály</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td><input type="checkbox"/></td></tr> <tr><td><input type="checkbox"/> žiadosti o zmenu / modifikácie</td><td><input type="checkbox"/></td><td><input type="checkbox"/></td><td><input type="checkbox"/></td></tr> </tbody> </table>		nízka	priem.	dobrá	<input type="checkbox"/> žiadna				<input type="checkbox"/> obchodný prípad (business)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/> dokument koncepcie systému	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/> špecifikácia požiadavok	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/> funkčná špecifikácia	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/> technická špecifikácia	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	 				<input type="checkbox"/> popis užívateľského rozhrania	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/> užívateľská príručka (manuál)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/> entitno-relačný model	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/> diagram dátových tokov	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/> plány a skripty na testovanie	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/> testovacie dáta	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/> popis verzií (release)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/> zoznam incidentov / závad	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/> manuál pre administrátorov	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/> školiace manuály	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/> žiadosti o zmenu / modifikácie	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<p>Toto je zoznam dokumentov, ktoré sú dostupné k prevádzke údržby aplikácie. Zhodnoťte kvalitu každej zložky dokumentácie posúdením toho, do akej miery pokrýva potrebné informácie. Kvalita dostupnej dokumentácie zvyšuje efektívnosť práce údržbárov.</p>
	nízka	priem.	dobrá																																																																										
<input type="checkbox"/> žiadna																																																																													
<input type="checkbox"/> obchodný prípad (business)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>																																																																										
<input type="checkbox"/> dokument koncepcie systému	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>																																																																										
<input type="checkbox"/> špecifikácia požiadavok	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>																																																																										
<input type="checkbox"/> funkčná špecifikácia	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>																																																																										
<input type="checkbox"/> technická špecifikácia	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>																																																																										
<input type="checkbox"/> popis užívateľského rozhrania	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>																																																																										
<input type="checkbox"/> užívateľská príručka (manuál)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>																																																																										
<input type="checkbox"/> entitno-relačný model	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>																																																																										
<input type="checkbox"/> diagram dátových tokov	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>																																																																										
<input type="checkbox"/> plány a skripty na testovanie	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>																																																																										
<input type="checkbox"/> testovacie dáta	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>																																																																										
<input type="checkbox"/> popis verzií (release)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>																																																																										
<input type="checkbox"/> zoznam incidentov / závad	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>																																																																										
<input type="checkbox"/> manuál pre administrátorov	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>																																																																										
<input type="checkbox"/> školiace manuály	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>																																																																										
<input type="checkbox"/> žiadosti o zmenu / modifikácie	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>																																																																										
<p>37. Celkové pokrytie aplikácie dokumentáciou</p> <p style="text-align: center;"><input style="width: 100px; height: 20px;" type="text"/> %</p>	<p>Zhodnoťte, aké je celkové pokrytie aplikácie dokumentáciou.</p>																																																																												
<p>38. Dostupnosť vývojového a testovacieho prostredia</p> <p>vývojové prostredie <input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> nevie</p> <p>testovacie prostredie <input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> nevie</p>	<p>Uvedte, či je dostupné separované prostredie pre vývoj a testovanie.</p>																																																																												

<p>39. Akým spôsobom sú podávané Žiadosti o Úpravu alebo Hlásenia o probléme (ďalej len ŽoU/HoP)?</p> <p>telefonicky: <input type="text"/> %</p> <p>emailom: <input type="text"/> %</p> <p>písomne (formulár): <input type="text"/> %</p> <p>ústne: <input type="text"/> %</p> <p>inak: <input type="text"/> <input type="text"/> %</p>	
<p>40. Existuje vnútro podniková smernica, ktorá postup riešenia problémov a úprav (ŽoU/HoP) upravuje?</p> <p><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> nevie</p>	
<p>41. Existuje v organizácii plán údržby, v ktorom sú opísané predovšetkým požiadavky na údržbu, definované zodpovednosti, odhadované náklady,...?</p> <p><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> nevie</p>	
<p>42. Ktoré z uvedených tém sú obsiahnuté v pláne údržby?</p> <p><input type="checkbox"/> úlohy a zodpovednosti</p> <p><input type="checkbox"/> dokumentácia</p> <p><input type="checkbox"/> popis procesu údržby (analýza, implementovanie, testovanie...)</p> <p><input type="checkbox"/> rozsah údržby a podpory (opravy, vylepšenia, Help Desk, ...)</p> <p><input type="checkbox"/> analýza finančných zdrojov (mzdy, školenia, licencie, technické publikácie, HW a SW pre testovanie)</p> <p><input type="checkbox"/> popis procesu údržby (analýza, implementovanie, testovanie...)</p> <p><input type="checkbox"/> analýza prostredia (vytvorenie separovaného prostredia pre vývoj, testovanie a prevádzku)</p>	
<p>43. Má vaša organizácia stanovené postupy pre prijímanie, zaznamenávanie, vykonávanie a riadenie žiadostí o zmenu?</p> <p><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> nevie</p>	

<p>44. Ktoré z uvedených plánov rozpoznávate vo vašej organizácii</p> <p> <input type="checkbox"/> plán údržby <input type="checkbox"/> plán merania <input type="checkbox"/> tréningový plán <input type="checkbox"/> manuál údržby <input type="checkbox"/> postupy údržby <input type="checkbox"/> odhad udržovateľnosti <input type="checkbox"/> plán projektového <input type="checkbox"/> postupy riešenia problémov <input type="checkbox"/> plány pre spätnú väzbu <input type="checkbox"/> plán manažmentu konfigurácie </p>	
<p>45. Prebieha u vás proces analýzy ŽoU/HoP ? (dopad na organizáciu, aplikáciu)</p> <p><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> neviem</p>	
<p>46. Prebieha u vás proces analýzy ŽoU/HoP ? (rozsah úpravy, kritickosť)</p> <p><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> neviem</p>	
<p>47. Prebieha u vás proces replikovania a overovania problému ?</p> <p><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> neviem</p>	
<p>48. Prebieha u vás proces hľadania alternatív pre implementáciu úpravy?</p> <p><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> neviem</p>	
<p>49. Prebieha u vás proces dokumentovania činností počas analýzy ŽoU/HoP?</p> <p><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> neviem</p>	
<p>50. Ktoré z nižšie uvedených tvoria výstup fázy analýzy ŽoU/HoP ?</p> <p> <input type="checkbox"/> analýza dopadu <input type="checkbox"/> odporúčaná alternatíva <input type="checkbox"/> schválená úprava <input type="checkbox"/> aktualizovaná </p>	
<p>51. Prebieha u vás analýza identifikujúca elementy, ktoré treba upraviť?</p> <p><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> neviem</p>	

<p>52. Prebieha u vás analýza identifikujúca časti dokumentácie, ktoré treba upraviť?</p> <p><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> neviem</p>	
<p>53. Prebieha u vás proces testovania aplikácie po implementovaní úpravy?</p> <p><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> neviem</p>	
<p>54. Ktoré z nižšie uvedených tvoria výstup fázy implementovania ŽoU/HoP ?</p> <p><input type="checkbox"/> aktualizované plány a postupy testovania</p> <p><input type="checkbox"/> aktualizovanú dokumentáciu</p> <p><input type="checkbox"/> upravený zdrojový kód</p> <p><input type="checkbox"/> hlásenie z testovania</p> <p><input type="checkbox"/> merania</p>	
<p>55. Prebieha u vás proces kontroly, že úpravy na systéme sú správne a že boli vykonané v súlade so schválenými štandardmi a správnu metodológiou?</p> <p><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> neviem</p>	
<p>56. Došlo počas prevádzky aplikácie k jej migrácii do nového prostredia?</p> <p><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> neviem</p>	<p>System bude potrebné upraviť, aby počas svojho života mohol bežať v rôznych prostrediach.</p>
<p>57. Ktoré z nižšie uvedených bolo výsledkom tohto procesu?</p> <p><input type="checkbox"/> plán migrácie <input type="checkbox"/> oznámenie o ukončení</p> <p><input type="checkbox"/> nástroje migrácie <input type="checkbox"/> merania,</p> <p><input type="checkbox"/> oznámenie <input type="checkbox"/> archivované dáta</p> <p><input type="checkbox"/> zmigrovaný softvérový produkt</p>	
<p>58. Nachádza sa vaša aplikácia vo fáze, keď sa zvažuje jej odchod do výslužby a nahradenie novou aplikáciou?</p> <p><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> neviem</p>	

<p>59. Ak áno, ktoré z nižšie uvedených činností v tejto fáze rozpoznávate?</p> <ul style="list-style-type: none"> <input type="checkbox"/> príprava plánu odchodu do výslužby <input type="checkbox"/> vydanie oznámenia o zámere <input type="checkbox"/> príprava výsledkov odchodu do výslužby <input type="checkbox"/> zaškolenie ľudí, ktorí odchod zrealizujú <input type="checkbox"/> vydanie oznámenia o ukončení <input type="checkbox"/> archivovanie dát 	
<p>60. Po zaznamenaní hlásenia o probléme, ktorý obmedzuje prevádzku aplikácie, pristupuje najskôr k núdzovému riešeniu, ktoré nehľadá príčinu, ale snaží sa akékoľvek riešenie vedúce k obnoveniu prevádzky?</p> <ul style="list-style-type: none"> <input type="checkbox"/> áno vždy <input type="checkbox"/> skôr nie <input type="checkbox"/> väčšinou áno <input type="checkbox"/> rozhodne nikdy <input type="checkbox"/> iba pri extrémnych problémoch 	
<p>61. Má vaša organizácia oficiálne schválenú radu, ktorá rozhoduje o spôsobe riešenia vážnych problémov?</p> <p style="text-align: center;"><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> neviem</p>	
<p>62. Má vaša organizácia oficiálne schválenú radu, ktorá je poradným alebo kontrolným orgánom v procese riešenia problémov?</p> <p style="text-align: center;"><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> neviem</p>	
<p>63. Existujú vo vašej organizácii procesy s cieľom plánovať a dohliadať na úspešné uvoľnenie softvéru (release) a súvisiaceho hardvéru?</p> <p style="text-align: center;"><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> neviem</p>	
<p>64. Existujú vo vašej organizácii procesy s cieľom navrhovať a implementovať účinné procedúry na distribúciu a inštalovanie zmien do IT systému?</p> <p style="text-align: center;"><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> neviem</p>	
<p>65. Existujú vo vašej organizácii procesy s cieľom uistiť sa, že iba oprávnené a otestované zmeny sú inštalované?</p> <p style="text-align: center;"><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> neviem</p>	

<p>66. Existujú vo vašej organizácii komunikácie s používateľmi počas plánovania a uvoľnenia nových vydaní?</p> <p><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> neviem</p>	
<p>67. Existujú vo vašej organizácii procesy na dohľad nad bezpečným uložením originálnych kópií konfiguračných prvkov (zdrojové kódy, verzie, dokumentácia...)?</p> <p><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> neviem</p>	
<p>68. Riadi sa proces údržby vo vašej organizácii podľa niektorého zo známych štandardov</p> <p><input type="checkbox"/> áno <input type="checkbox"/> nie <input type="checkbox"/> neviem</p>	
<p>69. Implementovali ste vo vašej organizácii niektorý zo štandardov alebo modelov?</p> <p><input type="checkbox"/> ISO 12207 <input type="checkbox"/> ISO 14764</p> <p><input type="checkbox"/> CMMI <input type="checkbox"/> ISO 15504 (SPICE)</p> <p><input type="checkbox"/> ITIL <input type="checkbox"/> COBIT</p> <p><input type="checkbox"/> SM3 (Software Maintenance Maturity Model)</p> <p><input type="checkbox"/> CM3 (Corrective Maintenance Maturity Model)</p> <p><input type="checkbox"/> Corrective Maintenance Maturity</p> <p>iný: <input type="text"/></p>	
<p>70. Pokúste sa odhadnúť koľko percent celkových finančných a ľudských prostriedkov pre danú aplikáciu spotrebuje údržba aplikácie (oprava chýb, vývoj vylepšení, dopĺňanie novej funkcionality, úpravy kvôli zmenenému prostrediu /OS, HW,.../, podpora užívateľom....)</p> <p><input type="text"/> % celkových prostriedkov</p>	

71. Ktorý z nasledujúcich stupňov kvality procesov údržby najlepšie charakterizuje stav praxe vo vašej organizácii?

Vykonávané procesy: Procesy sú zvyčajne vykonávané náhodne, dokonca až chaoticky. Spoločnosť neposkytuje stabilné prostredie na podporu procesov. Ťažko možno hovoriť o plánovaní, sledovaní a riadení procesov. Produkty často prekračujú rozpočet a termíny a spoločnosť nie je schopná zopakovať úspech.



Riadené procesy: Procesy tejto úrovne vyspelosti sú plánované, sledované a riadené pre individuálne projekty. Sledujú dokumentovaný plán, majú zodpovedajúce zdroje, sú určené zodpovednosťami a právomocami počas vývoja, zodpovedajúci ľudia sú školení a existuje monitorovanie procesov s nadväznými opatreniami na odstránenie vznikajúcich nedostatkov. Stav prác je v každom čase známy pre vedenie firmy. Dodržiavanie týchto praktík má za následok menšie výkyvy v kvalite procesov a tým aj kvalite výsledného produktu.



Definované procesy: riadený proces, ktorý je prispôbený zo sady štandardných procesov, používaných organizáciou. Definovaný proces jasne určuje vstupy, kritérium pre začatie, aktivity, úlohy, spôsob merania, kroky verifikácie, výstupy a kritérium na ukončenie. Štandardné procesy v organizácii sú postupne budované a časom overované.



Kvantitatívne riadené procesy: Na tejto úrovni sú definované procesy kvantitatívne riadené na základe štatistických alebo iných kvantitatívnych metód. Kvalita produktu, služby a výkonnosť procesov je sledovaná štatisticky a riadená počas celého životného cyklu. Za účelom riadenia sú postavené kvantitatívne ohodnotené ciele, je určený spôsob merania výkonnosti procesov, kvality produktov a služieb.



Optimalizované procesy: kvantitatívne riadený proces, ktorý je neustále zlepšovaný na základe znalosti podstatných príčin, spôsobujúcich odchýlky v procese. Zlepšovanie sa vykonáva po krokoch alebo skokom s ohľadom na celkové ciele firmy. Technológie a spôsoby zlepšovania sú rozmiestnené systematicky v celej organizácii a úroveň zlepšenia je merateľná a porovnávaná s kvantitatívnym cieľom.



72. Pripomienky, poznámky, námety.....



Ďakujem