

OCENJEVANJE PROGRAMSKE OPREME

Marjan Pivka
Univerza v Mariboru, Ekonomsko-poslovna fakulteta Maribor
Razlagova 14, 62000 Maribor

POVZETEK

Članek obravnava vprašanje ocenjevanja programske opreme in pokaže na razlike med ocenjevanjem, preskušanjem in testiranjem. Avtor poudarja, da je bistvo ocenjevanja v analiziranju semantične komponente programskega izdelka in analizira vprašanje merjenja količine kakovosti za posamezne karakteristike kakovosti po mednarodnem standardu ISO/IEC DIS 9126. Članek je delno povzet po referatu z enakim naslovom, ki je bil predstavljen na posvetovanju "Kakovost v programskem inženirstvu" septembra 1993 v Radencih.

ABSTRACT

In the paper, the differences between testing and assessing are analysed. Author emphasizes that assessment procedure is essentially a semantic analysis of software product. He researches measuring quality characteristics based on international standard ISO/IEC DIS 9126. The paper partly resumes the discourse author had at the Software Engineering Quality Conference, September 1993, in Radenci.



1. OPREDELITEV PROBLEMA

V vsakdanjem življenju nas proizvajalci na različne načine prepričujejo, kako kakovostni so njihovi izdelki. Proizvajalci uporabljajo vse mogoče metode, od bolj ali manj agresivnih reklam, do plačanih mnenj in ocen različnih strokovnjakov, institucij in podobno. Tem in podobnim ocenam kot uporabniki verjamemo, ali pa ne. Vse prevečkrat se zgodi, da šele po določenem času, ko smo izdelek kupili in ga začeli uporabljati, ugotovimo, da nam ali ne ustreza, ali ne izpolnjuje vseh deklariranih funkcij, ali pa preprosto ne deluje kot je deklarirano. Čeprav so programski izdelki na tržišču šele 10 do 15 let, pa so bolj ali manj, odvisno od razvitosti tržišča in aktualnosti programskih izdelkov, uveljavljeni vsi znani načini preverjanja kakovosti: mnenja neodvisnih strokovnjakov, objavljena v strokovnih revijah, različna potrdila ali certifikati, naročene ekspertize, mnenja itd. Razlog, zakaj se je v tako kratkem času razvila cela vrsta načinov preverjanja programskih izdelkov, je preprost: uporabnik ni sposoben preverjati vseh deklariranih funkcij, niti definirati vseh svojih potreb.

Postavlja se vprašanje, ali od uporabnika lahko pričakujemo, da bo ugotovil, ali programski izdelek res izvaja vse deklarirane funkcije na deklariran način. Vsakdanja praksa kaže, da ne. Razvoj informacijske tehnologije in s tem tesno povezane programske opreme pa kaže, da se to stanje le težko spreminja kljub večji osveščenosti uporabnikov.

Eden od načinov reševanja tega problema je, da ponudnik da kupcu potrdilo o ustreznosti izdelka. Potrdilo o ustreznosti ali certifikat je izjava, iz katere je razvidno, da programski izdelek ustreza določenemu standardu, predpisu, deklariranim funkcijam in podobno. Poznamo tri vrste potrdila o ustreznosti:

1. Potrdilo prve stranke (First party certificate). To potrdilo izda proizvajalec sam.
2. Potrdilo druge stranke (Second party certificate). Testiranje in izdajo potrdila je izvedel določen uporabnik.
3. Potrdilo tretje stranke (Third party certificate). To potrdilo izda neodvisna institucija na zahtevo proizvajalca.

Potrdilo o ustreznosti je končni rezultat procesa, ki ga imenujemo preskušanje in ocenjevanje. Cilj tega procesa je seveda ugotoviti, ali programski izdelek res ustreza zahtevam določenih standardov, predpisov in podobno. Certifikat je le formalna potrditev.

Neglede na certifikat pa ostaja formalna odgovornost uporabnika, da ugotovi, ali mu deklarirane funkcije izdelka ustrezajo ali ne. Certifikat sam torej ne daje nobene garancije, da izdelek izpolnjuje uporabnikova pričakovanja. Certifikat le potrjuje ustreznost izdelka določenim zahtevam, ki so običajno opredeljene v standardih in predpisih.

2. NIVO KAKOVOSTI IN KLASA KAKOVOSTI

Po definiciji mednarodnega standarda ISO 8402 Kakovost - slovar je klasa kakovosti definirana kot: "Kazalec kategorije ali nivoja kakovosti izdelka, procesa ali storitve, ki se nanaša na različne skupine zahtev za isto funkcionalno uporabo". Stopnja ali klasa ali razred kakovosti opredeljuje stopnjo funkcionalne dovršenosti nekega izdelka, procesa ali storitve. Namen opredeljevanja stopnje ali klase je pravzaprav poudariti odnos med stroški in funkcijami, ki jih neki izdelek, proces ali storitev, zadovoljujejo. Primer: Obračun proizvodnje ponuja na tržišču več proizvajalcev in vsak svoj izdelek deklarira za splošno uporabnega, cene pa so seveda različne. Kupec

ne ve, kateri izdelek naj izbere, zato je primoran analizirati vsakega ponujenega. Na koncu se, denimo, odloči za najcenejšega, ker ugotavlja, da izpolnjuje funkcionalne zahteve. Po nekaj mesečni uporabi ugotovi, da izdelek ne more slediti njegovemu razvoju in prisiljen je investirati v nov izdelek. Če bi bili programski izdelki označeni z razredom kakovosti, bi se uporabniki lažje odločali med različnimi proizvajalci tako iz vsebinskega, kot iz stroškovnega vidika.

Izdelek visokega razreda ali klase je lahko nekakovosten, če so posamezne funkcije proizvoda nekakovostno izdelane, imajo napake ipd. Dober primer za to je luksuzen hotel s slabo postrežbo. Ali programski izdelek, ki deklarirano izvaja celo vrsto funkcij, ni pa dokumentiran, nima zagotovljenega vzdrževanja, ima veliko število skritih napak, je neprijazen itd. Govorimo o nivoju kakovosti, ki opredeljuje, kako dobro je izdelek določene klase izdelan.

Nivo kakovosti opredeljuje stopnjo dovršenosti procesa, storitve ali izdelka. Ugotavljanje nivoja kakovosti je prav tako težavno. Tudi stopnja dovršenosti se s časom spreminja. To kar je bilo včeraj komaj mogoče (delo z več ekrani, grafična podpora ipd), je danes normalno.

Kakovosten izdelek, proces ali storitev zahtevajo visok nivo kakovosti za določen razred ali klaso kakovosti. To pa pomeni, da je tudi izdelek nizke klase lahko kakovosten in obratno: izdelek z deklarirano visoko klaso je lahko nekakovosten, če ni dovršen.

Klaso ali razred kakovosti lahko različno izražamo. Lahko s točkami (prva klasa), ali številkami (10 točk), ali pa kako drugače. V vsakdanjem življenju poznamo označevanje z zvezdicami za gostinske lokale in pet zvezdic pomeni pač lokal najvišje klase. V procesu izdelave programske opreme pa razvrščanja izdelkov, procesov ali storitev v različne klase ne poznamo. V naši državi zaenkrat nimamo izdelanih normativov, niti institucij, ki bi bile pooblaščenke za razvrščanje izdelkov v različne klase.

Tudi označevanje nivoja kakovosti je silno težavno, ker nimamo splošno sprejetih meril, ki bi izdelek, proces ali storitev, razvrstili na različne nivoje. Zato je označevanje nivoja kakovosti večinoma opisno. Primer: Help funkcija je lahko na dodatnem ekranu, dokumentacija za uporabnika in izvajalca je ločena, vgrajene so funkcije pomikanja ekrana v več oknih itd.

3. PRESKUŠANJE IN OCENJEVANJE PROGRAMSKE OPREME

Preskušanje je postopek, ko druga ali tretja stranka, to je laboratorij, priznan od proizvajalca in kupca, preskusi programski izdelek v skladu z določenim standardom ali predpisom. Če je izdelek preskus uspešno prestal, to pa pomeni, da je izdelek izdelan skladno z zahtevami standarda ali predpisa, izda preizkuševališče ustrezno poročilo. Tako poročilo se vselej nanaša na konkreten izdelek konkretnega proizvajalca. Tako poročilo ne razvršča enako imenovanih proizvodov različnih proizvajalcev v različne razrede ali klase kakovosti. Preskušanje je for-

malno gledano postopek, v katerem preskusni laboratorij preverja sintaktični in semantični vidik skladnosti med tem, kar je deklarirano za programski izdelek, in dejanskim stanjem.

Sintaktični vidik se nanaša na formalno skladnost med zahtevanim in pričakovanim, neglede na dejansko vsebino. Če standard zahteva da morajo vse deklarirane funkcije delovati tako kot je v dokumentaciji zapisano, potem je pa treba vse funkcije preskusiti in ugotoviti skladnost. Če predpisi ali zakoni nekaj zahtevajo, potem je treba preveriti, ali so rezultati programskega izdelka skladni s temi predpisi in zakoni. Rezultat preskušanja je odgovor DA (vsaj minimalna skladnost), ali NE. Rezultat preskušanja dveh enaki funkcionalni rabi namenjenih izdelkov ne pove, kateri je bolj kakovosten. Niti ne pove, kateri od njih izvaja več funkcij, je prijaznejši, bolj učinkovit itd.

Preskušanje ni testiranje programov ali izdelkov, s katerim odkrivamo napake v programih ter neskladnosti med definicijami in realizacijo.

Kadar govorimo o ocenjevanju programskega izdelka, pa ne mislimo le na njegov sintaktični, ampak tudi na njegov semantični vidik. To pomeni, da nas ne zanima le njegova sintaktična komponenta (skladnost s standardom, predpisom, zakonom,...), ampak tudi semantična, torej vsebinska komponenta iz vidika uporabnika in njegovih potreb. Zanimajo nas tiste lastnosti ali karakteristike izdelka, ki se nanašajo na njegovo vsebino in uporabnikovo okolje. Ocenjevanje programske opreme z razliko od preskušanja vsaj posredno vključuje tudi razvrščanje oziroma določevanje klase kakovosti in nivoja kakovosti izdelka. Nekateri avtorji (SCOPE) govorijo o razširjenem preskušanju. Zaradi vsebinske komponente procesa preskušanja pa je končni rezultat preskušanja ocena, ki je vsaj potencialno lahko osnova za rangiranje. Atributi ocenjevanja, torej lastnosti ocenjevanja, morajo biti tisti, ki jih zahtevata in pričakujeta stroka in uporabnik. Morajo biti enostavni za razumevanje, merljivi in rezultati meritev morajo biti ponovljivi. Pogoj merljivosti in ponovljivosti rezultatov merjenja lastnosti je zelo zahteven. Vendar pa le izpolnitev tega pogoja zagotavlja objektivnost ocenjevanja.

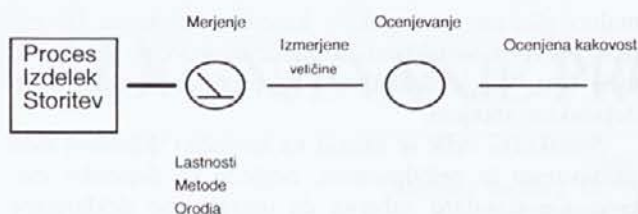
V zvezi z ocenjevanjem, ki je bistveno širše od preskušanja, pa nastopajo vsaj naslednji, vsebinsko zelo zahtevni in do danes še ne povsem rešeni problemi:

1. Katere so tiste karakteristike kakovosti programskih izdelkov, ki le-te najbolj ustrezno opredeljujejo?
2. Kakšen je pomen ali vpliv posameznih karakteristik kakovosti na kakovost celote?
3. Kako meriti (ugotavljati) količino posamezne karakteristike kakovosti?
4. Kako izraziti količino kakovosti za celoten izdelek?

3.1 Model postopka ocenjevanja

Model postopka ocenjevanja je preprost kibernetični model, ki ga prikazuje slika 1:

Problem, ki ga želimo poudariti v zvezi z ocenjevanjem, je, kako ugotoviti in izraziti količino posamezne last-



Slika 1: Kibernetički model ocenjevanja

nosti kakovosti. Na razpolago imamo široko paleto različnih metrik. Znani so na primer ciklomatično število, število vrstic kode, število operandov, operatorjev, stopnja vgnezenosti, vezljivost, sklopljenost itd. Vsaka od bolj ali manj znanih metrik pa se nanaša le na en vidik izdelka. Na primer na programsko kodo, kar pa je seveda daleč premalo za oceno celotnega izdelka po vseh mogočih karakteristikah. Še več! Kaže se potreba, da naj ocenjevanje upošteva tudi subjektivno, torej individualno mnenje ali oceno neke karakteristike. S tem pa je podatek iz objektivne metrike le eden od količinskih podatkov za ocenjevanje. Vprašanje je torej, s katerimi tehnikami ugotavljati količine kakovosti posameznih karakteristik in na tej osnovi izvajati ocenjevanje. Matematično gledano gre za proces preslikave ene metrične skale v drugo.

V programskem inženirstvu imamo na razpolago različne tehnike merjenja ali ugotavljanja količine posamezne lastnosti procesa ali izdelka. Vse seveda niso enako pomembne, niti niso primerne za vsako lastnost in za vsak primer. Nekatere se celo vsebinsko prekrivajo, niso eksaktne, celovite in še kaj. Pa vendar, teorija in praksa boljših ne pozna! V programskem inženirstvu so danes najbolj znane in v praksi uveljavljene naslednje tehnike:

- simulacija,
- testiranje,
- modeliranje,
- verifikiranje,
- programske metrike in
- preskušanje (kontrola).

Simulacija je tehnika, s katero prek nekega drugega sistema analiziramo dogajanje v obravnavanem sistemu.

Testiranje je postopek, s katerim odkrivamo napake v sistemu. Modeliranje se nanaša na razne modele zanesljivosti, stabilnosti, Markovi modeli ipd.

Verifikacija je proces formalne demonstracije sistema in njegovega pravičnega delovanja.

Programska metrika je proces pridobivanja in analiziranja podatkov o procesu, resursu ali izdelku na osnovi ustreznega modela.

Preskušanje (kontrola) je aktivnost preverjanja neke karakteristike na skladnost z doloženimi normami, predpisi itd.

3.2 Nivoji ocenjevanja

Izbira posamezne tehnike je odvisna ne le od obravnavane lastnosti, ampak tudi od kritičnosti obravnavanega

programskega izdelka ali njegovega dela (na primer programski modul). Dejstvo, da kritičnost programskega izdelka (ali njegovega dela) vpliva na izbiro tehnike ocenjevanja, vsiljuje odločitev, da je treba kritičnosti (rizičnosti) izdelka, ali z drugimi besedami pomenu izdelka za uporabnika, prirediti tehnike ocenjevanja. Manj kot je izdelek kritičen (na primer računalniške igrice), manj zahtevne so tehnike ocenjevanja. Bolj kot je programski izdelek rizičen (nevarnost za človeško življenje), bolj zahtevne so tehnike ocenjevanja (metrike, simulacije). Velja pa seveda splošno načelo, da zahtevnejši nivo ocenjevanja predpostavlja ustreznost vseh zahtev predhodnega nivoja. SCOPE projekt kot primer daje možnost ocenjevanja karakteristik kakovosti na štirih nivojih, označenih s črkami od A (najbolj zahteven nivo) do D (najmanj zahteven nivo). Za posamezen nivo ocenjevanja je treba izvesti:

Nivo D: Pregledati je treba uporabniško dokumentacijo, izvesti instalacijo ter preveriti, ali programi delujejo v skladu z dokumentacijo.

Nivo C: Zahteva oceno nivoja D, ter pregled specifikacije, zasnove izdelka in rezultate testiranja. Zahtevano je tudi funkcionalno testiranje skladno z specifikacijami.

Nivo B: Zahteva oceno nivoja C, pregledati je treba celotno proizvodno dokumentacijo, zahtevan je certifikat procesa po standardu ISO 9001 ali ekvivalentu. Izvesti je treba pregled specifikacije in zasnove ter ustrezna testiranja po načelu "bele škatle".

Nivo A: Zahteva oceno nivoja B, verifikacijo programske opreme (formalna, delno formalna, neformalna) in intenzivno testiranje po načelu "bele škatle".

Kdo pa je tisti, ki odloča, na katerem nivoju se bo izvedla ocena? V principu je to presoja in odločitev ocenjevalca in namena ocenjevanja. Ta odločitev pa se lahko spremeni na zahtevo uporabnika, kar pa se lahko nanaša le na zahtevo po višjem nivoju ocenjevanja, nikakor pa ne na zahtevo po nižjem nivoju.

Vidimo tudi, da se ocenjevanje na najnižjem nivoju, po vsebini prekriva s pojmom preskušanje, kakor smo ga v začetku opredelili.

3.3 Predlog tehnik ocenjevanja karakteristik kakovosti po ISO 9126

V Tabeli št.1 podajamo predlog tehnik ocenjevanja za posamezno karakteristiko kakovosti po ISO 9126 (ISO/IEC DIS 9126 Information Technology - Software product evaluation - Quality characteristics and guidelines for their use) za posamezen nivo ocenjevanja. Poudariti velja, da je to le predlog, oziroma nakazana povezava med tehnikami ocenjevanja, karakteristikami kakovosti in nivoji ocenjevanja (SCOPE). Pri tem ostaja popolnoma odprto vprašanje "kako" posamezno oceno izvesti na uvodu navedenih splošnih principih merjenja: enostavnost, razumljivost, objektivnost in ponovljivost rezultatov. To vprašanje je prepuščeno ocenjevalcu in njegovemu "know how". Vsekakor pa velja poudariti, da si mora ocenjevalec, oziroma firma, ki ima ambicije ukvarjati se z ocenjevanjem po principu druge in tretje stranke, izdelati

ali kupiti ustrezno delovno okolje, ki integrira manualne in avtomatizirane postopke v integralno delovno okolje ocenjevalca.

4 STROŠKI OCENJEVANJA

Ocenjevanje programske opreme seveda ni samo sebi namen, ampak je predhodna faza vrednotenja, torej obravnava pragmatičnega vidika programske opreme. V procesu vrednotenja dajemo oceni individualni, ali nek splošno priznani pomen. Na primer certifikat ustreznosti, ali pripustitev v obratovanje. Stroški ocenjevanja morajo biti v koleraciji z riziki in ne smejo presegati močne izgube, ki bi nastala pri vrednotenju brez ustrezne ocene.

Kolikšni so dopustni stroški ocenjevanja je seveda težko reči. Odvisni so pa od velikosti programskega izdelka in nivoja zahtevnosti ocenjevanja. Sposobnost in znanje ocenjevalca seveda nista nepomembni, predpostavljamo pa organizacijsko, kadrovsko in tehnično usposobljenost ocenjevalca. Naše dosedanje izkušnje in empirični rezultati iz SCOPE projekta in izjave ocenjevalcev iz TÜV KOLN kažejo okvirno oceno potrebnega časa za ocenjevanje izraženo v človek/mesecih, kot jo prikazujemo v tabeli 2. Ponovno poudarjamo, da morajo biti stroški ocenjevanja za vsaj nekritične programske proizvode (nivo D in deloma C), nižji od možnih rizikov in odgovarjati poslovnemu interesu proizvajalca.

Tabela 1: Predlog tehnik ocenjevanja za karakteristike kakovosti po ISO 9126:

ISO karakteristika	Tehnika ocenjevanja	Nivo
Funkcionalnost	Kontrolni pregledi, inspekcije	D
	Sledljivost, testiranje	C
	Testiranje ("bela, siva" škatla)	B
	Formalni dokaz	A
Zanesljivost	Pregled načina programiranja (jezik, pripomočki,...)	D
	Preskusi v realnem okolju, stohastična analiza sistema, dovoljena odstopanja	C
	Zanesljivostni modeli	B
	Formalni dokaz	A
Uporabnost	Kontrola uporabniškega vmesnika	D
	Kontrola na standarde (GUI), čitljivost	C
	Laboratorijski test, anketa uporabnika	B
	Uporabnikov model	A
Učinkovitost	Izvajanje in meritve	D
	Benchmark testi	C
	Kompleksnost algoritmov	B
	Izvedba evaluacije performans	A
Vzdrževalnost	Kontrolne liste	D
	Statična analiza	C
	Kontrola pravil programiranja	B
	Analiza sledljivosti procesa	A
Prenosljivost	Instalacija	D
	Kontrola pravil programiranja	C
	Analiza predpostavk za okolja in platforme	B
	Ocena zasnove programov	A

Tabela 2: Ocena časa ocenjevanja

Nivo	Ocena časa v človek/mesecih za izdelek velikosti:		
	Majhen	Srednji	Velik
A	7 - 13	8 - 16	9 - 19
B	5 - 9	6 - 11	7 - 12
C	3 - 6	4 - 8	5 - 9
D	1 - 2	2 - 3	3 - 4

Med majhne programske izdelke štejemo vse tiste z do 10.000 vrstic kode in s približno 50 stranmi dokumentacije.

Srednje veliki so tisti programski izdelki z do 50.000 vrsticami kode in 200 do 300 stranmi dokumentacije. To so tipične rešitve za osebni računalnik.

Veliki programski izdelki so pa tisti, ki imajo preko 100.000 vrstic kode in več kot 400 ali 500 strani dokumentacije.

5. ZAKLJUČEK

V referatu smo analizirali vprašanje ocenjevanja lastnosti kakovosti programske opreme, kot jih opredeljuje mednarodni standard DIS 9126. Posebno pomembna so naslednja spoznanja:

1. Posamezne lastnosti kakovosti programske opreme ne moremo ocenjevati z enakimi tehnikami in na enak način neglede na vrsto in pomen programske opreme. Programsko opremo je smiselno razvrstiti na nivoje zahtevnosti ali kritičnosti ali pomembnosti. Najmanj kritični so izdelki, ki ne povzročajo izgub ali škode, najbolj kritični so tisti, od katerih so odvisna človeška življenja.
2. Ocenjevalec potrebuje za učinkovito in objektivno oceno ustrezno delovno okolje.
3. Stroški ocenjevanja morajo biti v skladu z namenom ocenjevanja in primerljivi z možnimi izgubami zaradi nakupa neustreznega izdelka. To spoznanje ima seveda izjemo pri programskih izdelkih, od katerih so odvisna človeška življenja.

UPORABLJENA LITERATURA:

- BOOLINGER T.B., MCGOWAN C.: A Critical Look at Software Capability Evaluations. IEEE Software July 1991.
- ISO 9126 ISO/IEC DIS 9126 Information Technology - Software product evaluation - Quality characteristics and guidelines for their use.
- IEEE Master Plan for Software Engineering Standards. Ballot draft, 1993.
- PIVKA M., KAJZER F., GRABNAR D.: Bodo AOP zaradi PC izumrli? Revija Biro, št.2 1993
- PIVKA M., BORKO M.: Survey of software practices and software quality assesment in Slovenia. 3rd European Conference on Software quality. Madrid, 1992.
- SCOPE EGS Esprit Project 2151 SCOPE. Software Assessment and Certification Programme Europe.