

## ЦЕНА УПРАВЉАЊА КОД ИНКРЕМЕНТАЛНОГ ИНТЕГРАЦИОНОГ ТЕСТИРАЊА СОФТВЕРА

Зорица Михајловић, *Институт за нуклеарне науке "Винча", Београд*

**Садржај** – Овај рад разматра управљано инкрементално интеграционо тестирање софтвера који се развија на основама процедуралне парадигме. Под управљаним тестирањем подразумевамо планирано и праћено тестирање. Предност управљаног тестирања је његова видљивост јер постоји план тестирања у односу на који је у сваком тренутку могуће одредити напредак у тестирању. Циљ рада је да утврди цену управљаног инкременталног интеграционог тестирања и да је упореди са ценом неуправљаног тестирања. Резултати наше анализе показују да је цена управљаног инкременталног интеграционог тестирања већа од цене неуправљаног тестирања око 23 %.

### 1. УВОД

Планирање и праћење су два основне компоненте управљања у софтверском инжењерству [1], [2]. Оне су важне када се посматра појединачни софтверски пројекат (*software project*) али и када се посматрају поједини задаци у оквиру пројекта. Планирање обезбеђује основу за извођење задатка [2]. Праћење обезбеђује увид у стварни напредак задатка [2]. Инкрементално интеграционо тестирање софтвера један је од задатака софтверског инжењерства који такође тражи планирање и праћење.

Јединично и интеграционо тестирање су два најнижа нивоа генеричког задатка софтверског тестирања [3], [4], [1], [5]. За софтвер развијан на основама процедуралне парадигме најбољи начин овог тестирања је, према експертима [3], њихово извођење као једног интегрисаног нивоа инкременталног интеграционог тестирања. Током инкременталног интеграционог тестирања програм се гради креирањем инкрементата. Нова јединица и њени интерфејси налазе се у фокусу тестирања сваког инкремента. Под јединицом се подразумева софтверски елемент који се даље не дели на друге елементе.

Овај рад разматра управљачки аспект инкременталног интеграционог тестирања тако што третира управљано инкрементално интеграционо тестирање. Када кажемо управљано инкрементално интеграционо тестирање мислимо на планирано и праћено тестирање. Управљано инкрементално интеграционо тестирање заједно интегрише основне управљачке активности планирања и праћења као и коначно извештавање о тестирању са инжењерским активностима овог нивоа тестирања. Инжењерске активности су пројектовање тестирања, имплементација тестирања и извршавање тестирања. Супротно, неуправљано инкрементално интеграционо тестирање укључује једино инжењерске активности тестирања. Предност управљаног инкременталног интеграционог тестирања је то што чини тестирање видљивим. Другим речима, постоји план тестирања у односу на који се може у сваком тренутку оценити напредак у тестирању. Циљ рада је да одреди цену управљаног инкременталног интеграционог тестирања и да је упореди са ценом неуправљаног тестирања. При томе се цена управљаног

инкременталног интеграционог тестирања посматра као напор утрошен на овај ниво тестирања.

Одређивање цене управљаног инкременталног интеграционог тестирања има три корака. У првом кораку се потпуно и прецизно дефинише управљано инкрементално интеграционо тестирање (одељак 2.). У другом кораку се управљано инкрементално интеграционо тестирање примењује у развоју једног броја програма и прикупљају подаци о тестирању (одељак 3.). У трећем кораку се на основу ових података израчунава цена управљаног тестирања (одељак 4.). Када се управљачке активности раздвоје од инжењерских активности инкременталног интеграционог тестирања, могуће је на основу података за управљано тестирање израчунати и цену неуправљаног тестирања и упоредити је са ценом управљаног тестирања (одељак 4.).

### 2. УПРАВЉАНО ИНТЕГРАЦИОНО ТЕСТИРАЊЕ

У циљу дефинисања управљаног инкременталног интеграционог тестирања, развијена је посебна методологија названа *ITeM (Incremental Integration Testing Management)* [6]. Ова методологија потпуно и прецизно дефинише начин креирања плана инкременталног интеграционог тестирања и начин праћења извођења овог тестирања. Главна карактеристика методологије *ITeM* је њена превентивност према [7]. Превентивност значи да инкрементално интеграционо тестирање може да започне одмах након завршетка пројекта софтвера (*software design-a*) и да налажење проблема у софтверу почиње већ са планирањем и пројектовањем тестирања а не везује се једино за извршавање тестирања.

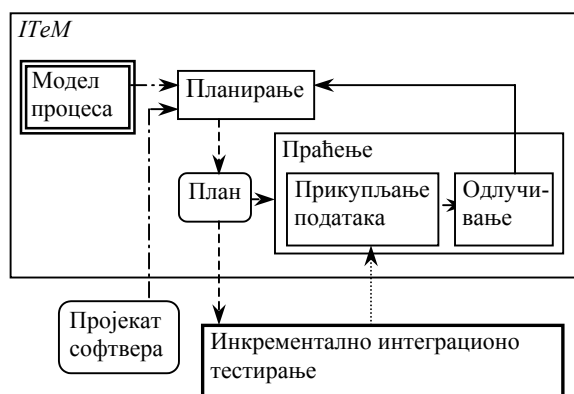
Креирање плана инкременталног интеграционог тестирања има осам корака када се користи методологија *ITeM* [6]: 1) одређивање карактеристика за тестирање, 2) идентификацију циљева тестирања, 3) идентификацију управљачких ограничења, 4) идентификацију расположивих ресурса, 5) креирање парцијалног плана, 6) процењивање, 7) распоређивање и 8) планирање за потребе праћења. Креирање парцијалног плана састоји се из поделе полазног задатка тестирања на мање активности дуж хијерархије активности до елементарних које се додељују на извршење једном инжењеру. Ова подела посла има две основе: процес и производ (Сл. 1.). Код прве основе користи се модел процеса инкременталног интеграционог тестирања [6], [8], [9], а код друге основе структурни дијаграм пројекта софтвера програма који се тестира. Код одређивања цена управљаног и неуправљаног инкременталног интеграционог тестирања од значаја је подела посла на основу модела процеса.

Праћење се одвија у паралели са извођењем појединачног задатка инкременталног интеграционог тестирања (Сл. 1.). Током извођења задатка прикупљају се подаци о тестирању у два дневника: дневнику активности и дневнику проблема. Проблем је било који догађај који се појави током тестирања а захтева даља истраживања. При записивању проблема, обавезан детаљ

је нотирање типа проблема према класификацији проблема дефинисаној у методологији *ITeM* [6]. Класификација води рачуна о:

- везама између проблема и докумената у развоју и тестирању софтвера и
- везама између проблема и фаза и активности у којима су проблеми нађени.

Тако се, на пример, прави разлика између проблема у пројекту софтвера нађеног током припреме за тестирање, планирања и пројектовања тестирања, и проблема у коду нађеног током извршавања тестирања. Дневници су инспирисани сличним документима предложеним у Личном софтверском процесу (*Personal Software Process*) [10] и прилагођени коришћењу код инкременталног интеграционог тестирања.



Сл. 1. Методологија *ITeM*

Извођење задатка инкременталног интеграционог тестирања може да одступи од плана мање или више. Неке девијације траже модификацију текућег плана, тј. креирање новог плана – репланирање. Проблеми нађени током тестирања одређују моменат репланирања. Репланирање је важан елемент праћења задатка инкременталног интеграционог тестирања. Модел процеса инкременталног интеграционог тестирања помаже у одлучивању о репланирању [6], [8], [9].

Типична секвенца активности инкременталног интеграционог тестирања може да има следећи облик:

ОПл К<sub>ПС</sub> ОПл П К<sub>ПС</sub> ОПл П К К<sub>ПС</sub> ОПл П К ДПл  
ИИЗвр Изв (1).

Изводи се на основу модела процеса инкременталног интеграционог тестирања [6], [8], [9]. У секвенци (1) ОПл означава опште, а ДПл детаљно планирање тестирања, П пројектовање тестирања, К кодирање, ИИЗвр имплементацију и извршавање тестирања, Изв извештавање о тестирању а К<sub>ПС</sub> корекције у пројекту софтвера. Све активности, осим последње две, припадају припреми за тестирање. Пролаз кроз пројекат софтвера је обавезан при пројектовању тестирања и тада могу да се открију евентуални проблеми у пројекту софтвера. Њих је потребно кориговати (К<sub>ПС</sub>) пре новог планирања тестирања и даље, пројектовања тестирања. То значи да пројектовање тестирања има способност да прочишћава пројекат софтвера. Исту ову способност имају и планирање тестирања и кодирање. То је и показано у секвенци (1).

Планирање тестирања у секвенци (1) има две форме: опште планирање ОПл и детаљно планирање ДПл.

Детаљним планирањем завршава се припрема за тестирање. Док је опште планирање тестирања усмерено на припрему за тестирање до детаља а остале фазе тестирања даје у скицама, то је детаљно планирање којим се завршава припрема за тестирање усмерено на имплементацију и извршавање тестирања и извештавање о тестирању до детаља.

Проблеми нађени у инкременталном интеграционом тестирању када се користи методологија *ITeM* деле се на две класе:

- проблеме фокусиране у развојним документима (пројекту софтвера и коду) и
- проблеме фокусиране у документима произведеним у тестирању (тестној спецификацији и драјверима, на пример).

Проблеми прве класе, посебно проблеми у пројекту софтвера, релевантни су за одређивање цена инкременталног интеграционог тестирања. Уз то, секвенца активности дата у (1) такође је релевантна за одређивање цена. У њој се разликују инжењерске активности тестирања, пројектовање тестирања, имплементација тестирања и извршавање тестирања, од управљачких активности: општег и детаљног планирања тестирања и извештавања о тестирању. Секвенца (1) одговара управљаном инкременталном интеграционом тестирању.

### 3. ПРИМЕНА УПРАВЉАНОГ ИНТЕГРАЦИОНОГ ТЕСТИРАЊА

У циљу израчунавања цена управљаног и неуправљаног инкременталног интеграционог тестирања, развијено је више програма у програмском језику *C*. Укупна величина ових програма прогнози око 3000 линија кода. Развој је укључио и инкрементално интеграционо тестирање на начин који одговара методологији *ITeM* чиме је осигурано извођење управљаног инкременталног интеграционог тестирања. Услови извођења инкременталног интеграционог тестирања били су исти за све развијене програме тако да је могуће сумирати прикупљене податке дуж свих програма.

При израчунавању цена коришћени су подаци из дневника активности и дневника проблема. Подаци релевантни за израчунавање цена извучени из дневника представљени су у Табели 1., Табели 2. и Табели 3. Табела 1. показује да припрема за тестирање троши 47 %, имплементација и извршавање тестирања 46 % а извештавање о тестирању 7 % укупног напора утрошеног на тестирање. Табела 3. показује да је већина проблема у пројекту софтвера посматраних програма који су нађени у припреми за тестирање, нађена после прве инстанце пројектовања тестирања.

### 4. АНАЛИЗА

Цена управљаног инкременталног интеграционог тестирања мерена преко стварног напора утрошеног на тестирање обухвата целокупни напор утрошен на овај ниво тестирања и износи 12733 минута (Табела 1.). На основу података из Табеле 2. и Табеле 3. индиректно се рачуна цена неуправљаног инкременталног интеграционог тестирања. Ради рачунања цене неуправљаног тестирања потребно је поделити активности инкременталног интеграционог тестирања на чисто инжењерске активности (активности тестирања и

развоја) и чисто управљачке активности. При томе је од помоћи секвенца (1). Пројектовање тестирања, имплементација тестирања, извршавање тестирања и кодирање су чисто инжењерске активности. Планирање тестирања и извештавање су управљачке активности, такође присутне у секвенци (1). Активности корекција пројекта софтвера по својој природи одговарају инжењерским активностима и због тога им се прикључују.

Табела 1. Дистрибуција стварног напора дуж фаза инкременталног интеграционог тестирања

Фаза	Стварни напор у минутима
Припрема за тестирање	6034
ИИЗвр	5858
Изв	841
Укупни напор	12733

Искључивањем прве, друге, треће, шесте, десете и тринаесте активности у секвенци (1) која одговара управљаном инкременталном интеграционом тестирању, добија се секвенца активности неуправљаног тестирања:

$$П \quad K_{ПС} \quad П \quad К \quad K_{ПС} \quad П \quad К \quad ИИЗвр \quad Изв \quad (2).$$

Она се користи у одређивању цене неуправљаног инкременталног интеграционог тестирања.

Цена неуправљаног инкременталног интеграционог тестирања има два дела: основну и додатну цену. Основна цена одговара стварно утрошеном напору на инжењерске активности према секвенци (2). Додатна цена компензује изостанак управљачких активности у неуправљаном тестирању и њихову способност да прочишћавају пројекат софтвера. Наиме, проблеми нађени и решени у току управљачких активности код управљаног инкременталног интеграционог тестирања, остају за налажење и решавање у току инжењерских активности код неуправљаног инкременталног интеграционог тестирања. Цена ових проблема одређује додатну цену неуправљаног тестирања. Реч је о проблемима у пројекту софтвера нађеним и решеним у току општег планирања тестирања пре појаве било које инстанце пројектовања тестирања (Табела 3.). Они одговарају трима првим активностима у секвенци (1): ОПл,  $K_{ПС}$  и ОПл. Наша претпоставка код одређивања цене ових проблема је да се ови проблеми код неуправљаног тестирања налазе и решавају у инжењерским активностима после прве појаве пројектовања тестирања у припреми за тестирање и под истим условима који за те проблеме важе код управљаног тестирања. Ова претпоставка одговара низу од седам активности у секвенци (2) идући од почетка:  $P \quad K_{ПС} \quad П \quad К \quad K_{ПС} \quad П \quad К$ . То је претпоставка која обезбеђује најмању могућу цену ових проблема код неуправљаног тестирања јер се они везују за активности које непосредно следе из оригиналних активности налажења и решавања ових проблема у управљаном тестирању. Како је познато да цена налажења и решавања проблема расте како развој софтвера напредује почевши од анализе, пројектовања, кодирања, коначног системског

тестирања до рада у пољу, и да је његова цена мања што се пре нађе и реши [1], то би свака друга претпоставка дала већу цену ових проблема.

Табела 2. Дистрибуција стварног напора дуж активности у припреми за тестирање

Активност	Стварни напор у минутима
ОПл и ДПл	1580
$K_{ПС}$ пре првог П	4
$K_{ПС}$ после првог П	399
П	2984
К	1067
Укупни напор	6034

Имајући у виду претходно, цена неуправљаног инкременталног интеграционог тестирања добија се по формули:

$$\text{Напор}_Н = \text{Напор}_П + \text{Напор}_К + \text{Напор}_{K_{ПС} \text{ после првог П}} + \text{Напор}_{ИИЗвр} + \text{Број}_и * (\text{Напор}_П + \text{Напор}_К + \text{Напор}_{K_{ПС} \text{ после првог П}}) / \text{Број}_П \quad (3),$$

где је  $\text{Напор}_Н$  стварни напор утрошен на неуправљано тестирање.  $\text{Напор}_Х$  је стварни напор утрошен на фазу/активност  $X$  тестирања (Табела 1. и Табела 2.),  $\text{Број}_и$  је број проблема у пројекту софтвера нађен и решен у припреми за тестирање пре првог П (Табела 3.) а  $\text{Број}_П$  је број проблема у пројекту софтвера нађен и решен у припреми за тестирање после првог П, све код управљаног тестирања. Прва четири члана у формули (3) представљају основну цену, а последњи члан представља додатну цену неуправљаног инкременталног интеграционог тестирања.

Применом формуле (3) на податке у табелама долази се до цене неуправљаног инкременталног интеграционог тестирања од 10404 минута. Поређењем две цене добија се да је цена управљаног тестирања 23 % већа од цене неуправљаног тестирања. Ако је додатна цена неуправљаног тестирања већа од оне дате нашем претпоставком, очекује се повећање цене управљаног тестирања мање од 23 %. Наравно да овај резултат важи под условима одређеним методологијом ИТеМ и сигурно за развијене примерке програма. За друге програме под сличним условима очекујемо сличан резултат.

## 5. ЗАКЉУЧАК

У циљу одређивања цене управљаног инкременталног интеграционог тестирања потребно је потпуно и прецизно дефинисати ово тестирање, применити га на низ програма, прикупити податке о тестирању и на крају израчунати цену као стварни напор утрошен на ово тестирање. Ако се раздвоје управљачке од инжењерских активности тестирања, такође је могуће израчунати и цену неуправљаног инкременталног интеграционог тестирања. То је управо урађено у овом раду.

Резултати показују да је цена управљаног тестирања 23 % већа од цене неуправљаног тестирања у најгорем случају. Ове резултате не бе требало схватити као коначан закључак већ као прелиминарно квантитативно поређење управљаног и неуправљаног инкременталног интеграционог тестирања. Они могу бити од помоћи код избора једног од ова два начина тестирања. Тада је потребно донети одлуку да ли је видљивост коју обезбеђује управљано тестирање (постојање плана и могућност одређивања напретка тестирања) довољно вредна да би прихватили већу цену од око 23 % овог начина тестирања у односу на неуправљано тестирање.

Табела 3. Проблеми нађени у припреми за тестирање код инкременталног интеграционог тестирања

Тип проблема	Број
Проблеми лоцирани у пројекту софтвера нађени пре првог П	2
Проблеми лоцирани у пројекту софтвера нађени после првог П	93
Укупан број	95

## ЗАХВАЛНИЦА

Овај рад је делом финансиран од стране Министарства науке и заштите животне средине Републике Србије под уговором број ПТР-2035А.

## ЛИТЕРАТУРА

- [1] R.S. Pressman, *Software Engineering: A Practitioner's Approach*, 4<sup>th</sup> edition, New York: McGraw-Hill, 1997.
- [2] M.C. Paulik et al. "Key Practices of the Capability Maturity Model", Version 1.1, *Technical Report, Software Engineering Institute*, CMU/SEI-93-TR-02, 1993.
- [3] G.J. Myers, *The Art of Software Testing*, New York: Wiley, 1979. D. Gelperin and B. Hetzel, "The Growth

of Software Testing", *Communications of the ACM*, Vol. 31, No. 6, June 1988, pp. 687-695.

- [4] B. Hetzel, *The Complete Guide to Software Testing*, second edition, New York: Wiley, 1988.
- [5] J.A. Whittaker, "What Is Software Testing? And Why Is It So Hard?", *IEEE Software*, Vol. 17, No. 1, January/February 2000, pp. 70-79.
- [6] З. Михајловић, Планирање и праћење јединичног и интеграционог тестирања софтвера засновано на знању, *Док. дисертација, Електротехнички факултет, Универзитет у Београду*, мај 2003.
- [7] D. Gelperin and B. Hetzel, "The Growth of Software Testing", *Communications of the ACM*, Vol. 31, No. 6, June 1988, pp. 687-695.
- [8] Z. Mihajlović, D. Velašević and N. Mastorakis, "A Preventive Process Model of the Incremental Integration Level of Software Testing", *WSEAS Trans. on Computers*, Vol. 2, Issue 1, 2003, pp. 30-35.
- [9] З. Михајловић и Д. Велашевић, "Модел процеса инкременталног интеграционог тестирања софтвера", у *Зборнику радова XLVIII конференције ЕТРАН-а, свеска III*, стр. 59-62, 2004.
- [10] W.S. Humphrey, *A Discipline for Software Engineering*, Reading, Massachusetts: Addison-Wesley, 1995.

**Abstract** – This paper considers the managed incremental integration testing for software developed through procedural paradigm, which means the planned and tracked testing. The advantage of the managed testing is its visibility because the plan of testing against which the progress of testing can be evaluated in any moment exists. The aim of the paper is to determine the price of the managed incremental integration testing and to compare it with the price of the unmanaged incremental integration testing. The results of our analysis show that the price of the managed incremental integration testing is greater than the price of the unmanaged testing of approximately 23 %.

## THE PRICE OF INCREMENTAL INTEGRATION SOFTWARE TESTING MANAGEMENT

Zorica Mihajlović