

EVOLUTIVNO ODRŽAVANJE INFORMACIONIH SISTEMA INFORMATION SYSTEMS EVOLUTIVE MAINTENANCE

Vesna Oro Radovanović, Violeta Oro

REZIME: Prezentovan je problem kratkog životnog veka informacionog sistema preduzeća i mogućnost njegovog produžetka. Uvođenje informacionih sistema, s jedne strane iziskuje visoka finansijska sredstva, a takođe i velike kadrovske napore, a sa druge strane zastareva veoma brzo. Stoga je potreba za održavanjem postojećeg u životu, velika. U praksi je zato čest sličaj da se informacioni sistemi koriste znatno duže od njihovog predviđenog životnog veka. Pitanje je, da li je moguće obezbediti da informacioni sistem stariji od deset godina i dalje može da zadovolji potrebe preduzeća, kao kada je bio u punoj zrelosti. Ovaj članak se bavi tim problemom i razmatra mogućnosti realizacije informacionih sistema koji će kontinuiranim, evolutivnim promenama postići dugovečnost.

KLJUČNE REČI: informacioni sistem, životni vek informacionog sistema, održavanje informacionog sistema, evolutivno održavanje.

ABSTRACT: The problem of short lifetime of company information system and possibility of its extension is presented. Beside great financial means, implementation of an information system demands great effort of human resources and on the other hand, becomes obsolete too quickly. This is exactly where the strong need for keeping present information system alive comes from. That is also explanation why it is often the case in practice that information systems are exploited much longer than their predicted lifetime. The question is if it is possible that information system, older than ten years, is still able to fulfill the needs of the company, just as much as it was able when it was at its peak of maturity. The paper presents this problem and considers possibility of achieving longevity of information systems with continuous, evolving maintenance.

KEY WORDS: information system, information system lifetime, information system maintenance, evolutive maintenance.

1. UVOD

Uvođenje informacionih sistema predstavlja investiciju koja iziskuje velika materijalna sredstva, ali i puno kadrovske angažovanja. Kada bi menadžment naručioca razmotrio nekoliko osnovnih dimenzija ovakve investicije, kao što su visoka cena, dug rok izrade, mnogo sati razgovora, dogovora, ubeđivanja i testiranja i na kraju kratak rok upotrebe, tada ni jedan informacioni sistem ne bi bio uveden. Činjenica da se dužina razvoja, kao i cena informacionog sistema može porediti sa izgradnjom nekog velikog građevinskog objekta, a da se njegovo korišćenje može porediti sa rokom trajanja nekog tehničkog uređaja, poput već mašine, menadžment smatra neprihvatljivom. Menadžment prihvata činjenicu da će čitav proces iziskivati pored novčanih sredstava i velika dodatna angažovanja kadrova, ali ne može da prihvati činjenicu da se sav taj trud uloži da bi posle tri ili pet godina, sve to ponovo morao da radi zbog nekog novog informacionog sistema. Menadžment smatra da IS mora da traje znatno duže. Praksa to i potvrđuje.[6] Neki informacioni sistemi traju i po 10 ili 15 godina, ali je pitanje kako oni služe i kakva je njihova efikasnost. I već mašine se u nekim domaćinstvima koriste i po 20 godina, ali bruje, tresu se, prosipaju prašak, cure na sve strane, a od programa im radi samo iskuvavanje. To nije ono što se od IS-a želi. Ono što je potrebno je da on svoju operabilnost zadrži dugi niz godina, a da se njegovo prirodno brzo zastarevanje odloži što je moguće duže. Da li je to moguće?

Ukoliko bi se promenila perspektiva iz koje se IS posmatra kao jedna tehnička tvorevinu, koja ima svoj rok trajanja i isti sada posmatrao samo kao jedna pojava – jedno izdanje koje traje u nekom vremenskom periodu i ima sposobnost da se menja kroz sledeću generaciju - verziju tada bi se došlo do

sasvim nove perspektive u kojoj bi se mogao porediti sa biološkim sistemom. Poredeći IS sa biološkim sistemima, mogu se primeniti neki njihovi fenomeni, kao što je mogućnost selekcije kroz generacije, kao pokretače promena. Konstantne promene iz generacije u generaciju su nosioci dinamike sistema u cilju njegove primerenosti okruženju i vremenu, a čitav proces se naziva evolucija. Biološki sistemi stoga umesto da izumru, evoluiraju.

Da li ovakva filozofija može da se primeni i na informacione sisteme, tj. da li je moguće da umesto konstante borbe za održavanje sistema u životu i korišćenja zastarelog sistema, imamo uvek mlad i aktuelan sistem? Da li je moguće da umesto da imamo sistem jedne generacije, koja vremenom stari, zapravo promenimo paradigmu i učinimo da sistem bude dinamičan da se razvija i izrasta iz jedne u drugu generaciju? Da li je moguće primeniti teoriju evolucije i na tehnički sistem kakav je informacioni? Ukoliko je moguće, šta je potrebno da bi se to ostvarilo?

2. ŠTA JE TO EVOLUCIJA

“Evolucija (lat. *evolutio*: razvoj, razvitak), predstavlja razvoj iz nižeg u viši, iz jednostavnog u složeno, nekog prirodnog ili društ. procesa (npr. umnih sposobnosti, sredstava za proizvodnju); manifestacija neke pojave, događaja ili ideje sukcesivno, u jednom procesu (npr. slikarstva, pesništva, drame, tragedije itd.); u istorijskom razvoju društva, kvantitativna postepena promena kao suprotnost revoluciji; svaki razvoj koji je sličan rastu žive materije, nasuprot kreativnom shvatanju (npr. razvoj nekog izuma); kretanje koje se sastoji od niza povezanih i uzajamno uslovljenih radnji; proces obli-

kovanja nekoga finalnog proizvoda, isto tako i sam proizvod iz tog procesa, kao i svaka oblikovana etapa, jedinica iz jedne razvojne serije (npr. cveta iz pupoljka) itd.

U biologiji, evolucija predstavlja, skup naučnih teorija i hipoteza o nastanku života, o zakonima i putevima postupnog razvoja živog sveta; razvoj organizama od jednostavnih ka složenim; razvoj vrste, populacije, ili neke više sistematske grupe – filogenija. U najširem smislu, proces u kome nizom promena ili razvojnih stupnjeva živi organizam ili skup organizama stiče karakteristične morfološke i fiziološke karakteristike; teorija po kojoj različiti tipovi životinja i biljaka potiču od drugih preegzistentnih tipova. Ta teorija, prema kojoj se viši oblici života izvode iz nižih, temelji se na opsežnim posmatranjima, posebno na područjima paleontologije, embriologije, uporedne anatomije, genetike, molekularne biologije (molekularna evolucija), uporedne fiziologije itd.“[7]

Evolucija je pojava koja obuhvata neprekinuti niz generacija jedinki koje postepeno menjaju svoja svojstva, promenom svog genetskog materijala. Promene u genetskoj strukturi iz generacije u generaciju predstavljaju mikroevoluciju. Postoje dva osnovna uzroka mikroevolucije:

- Genetičke varijacije
- Prirodna selekcija

Genetičke varijacije obuhvataju individualne varijacije diskretnih i kvantitativnih karakteristika u okviru populacije i geografska variranja između populacija. Značaj genetskih varijacija u evolutivnom mehanizmu se ogleda u tome što individue najbolje prilagođene spoljašnjoj sredini ostavljaju najviše potomstva i ova prirodna selekcija rezultira u adaptaciju organizama odn. akumulaciju genetskih varijacija favorizovanih uslovima spoljašnje sredine.

Genetičke varijacije generišu mutacije i seksualne rekombinacije. Rekombinacije su korisne promene u genetičkom pulu i predstavljaju glavni izvor genetičkog variranja, dok većina mutacija nije korisna za organizme i najčešće se ne prenose na potomstvo.

Prirodna selekcija predstavlja odabir organizama koji su u interakciji sa drugim organizmima najbolje prilagođeni datim uslovima životne sredine. Ako se spoljašnja sredina promeni ili se populacija preseli, populacija će preživeti ako bar neka od individua može da podnese promenjene uslove. Veliki broj evolutivnih noviteta su u stvari modifikovane verzije starijih struktura. Evolutivni noviteti mogu da nastanu uz postepeno poboljšanje postojećih struktura. Ove strukture koje evoluiraju u jednom kontekstu ali se koadaptiraju za drugu funkciju (u drugom kontekstu) se zovu egzaptacije. Prirodna selekcija ne može da predviđa budućnost ali može da poboljša strukture u datom vremenu.

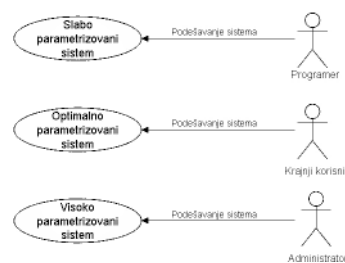
Stalne promene organizama dovode do bolje adaptiranosti i evolucije organizama (primer: stalne promene koje teku naizmenično kod gazele i lava, najbrža gazela preživljava a zatim lav koji najbrže trči da bi uhvatio plen takođe preživljava itd.)

Genetičke varijacije i prirodna selekcija su fenomeni evolucije. Međutim, to su mehanizmi koji su prisutni i u životnom ciklusu informacionog sistema. Prilikom razvoja i eksploatacije informacionog sistema, on prolazi kroz različite transformacije. Transformacije softvera su potaknute promenama, nekad iz okoline, a nekad iz same organizacije. Informacioni sistemi se vremenom uvećavaju, usložnjavaju, menjaju, dobijajući nove karakteristike, a sve u cilju prilagođavanja trenutnim uslovima. Neke karakteristike se vremenom pokažu kao favorizovane od strane korisnika, te preživaljevaju, dok druge bivaju napuštene. Na taj način informacioni sistem osigurava svoj opstanak. Ovaj proces se može stoga smatrati procesom evolucije.

3. RELACIJE IZMEĐU ODRŽAVANJA I EVOLUCIJE

Informacioni sistemi se nalaze u okruženju koje je podložno promenama. Svaka promena u organizaciji ili okruženju u kome se organizacija nalazi iziskuje potrebu da joj se IS prilagodi. Prilagođavanje sistema zavisi od veličine promene, ali i fleksibilnosti sistema.

Zavisno od veličine promene, kao i strukture informacionog sistema, aktivnosti, koje imaju za cilj prilagođavanje sistema, mogu izvršiti različiti akteri. Ukoliko je promena mala, a sistem tako postavljen da bude fleksibilan na promene, što se postiže parametrizacijom sistema, tada prilagođavanje može da izvrši običan korisnik, jednostavnom promenom parametara sistema. Međutim, ukoliko je promena velika, ili je sistem krut, slabo parametrizovan, te promene, zavisno od situacije, vrše administratori, programeri, projektanti. Stepem parametrizacije sistema određuje ko će intervenisati u određenoj situaciji, korisnik, administrator ili softveraš. Tako će u tri različite konkurentske firme na jednu promenu u poreskom sistemu na primer, reagovati tri različita aktera (slika 1). U jednoj će se možda morati angažovati programer, u drugoj će posao završiti dežurni administrator, dok će u trećoj firmi promena ostati nezapažena, jer će je korisnik sprovesti običnom promenom parametra kroz aplikaciju.



Slika 1. – Različiti učesnici u održavanju sistema u zavisnosti od stepena njegove parametrizacije[3]

Sve ove akcije prilagođavanja sistema predstavljaju njegovo održavanje. Međutim, ukoliko je intenzitet promene velik, tada i srednje, pa čak i visoko parametrizovani sistemi, moraju biti promenjeni kodiranjem, odnosno angažovanjem programera. Obzirom da se IS sastoji iz skupa aplikacija, promene se moraju sprovesti na aplikacijama na koje je promena imala

uticaja. Tada nastaje nova verzija ili verzije aplikacija informacionog sistema. Ako jednu verziju IS-a posmatramo kao jednu generaciju, tada prilagođavanje sistema nastalim promenama iz verzije u verziju možemo smatrati njegovom evolucijom. Obzirom da izrada novih verzija može da spada u domen održavanja, tada održavanje predstavlja nosioca evolutivnih promena. Međutim, nije svako održavanje evolutivno. Da bi postalo, potrebno je za početak da se ispune neki osnovni preduslovi.

4. OSNOVNI PREDUSLOVI ZA REALIZACIJU EVOLUTIVNOG ODRŽAVANJA

Prilikom nabavke informacionih sistema malo se razmišlja o njihovom održavanju. Međutim, održavanje je neophodno da bi informacioni sistem uopšte funkcionisao. Dovoljno je da se neki od parametara organizacije ili okolne sredine u kome organizacija funkcioniše promeni, pa da se pojavi potreba za prilagođavanjem informacionog sistema. To prilagođavanje informacionog sistema predstavlja zapravo njegovo održavanje, a time i održavanje njegove poslovne vrednosti.[2]

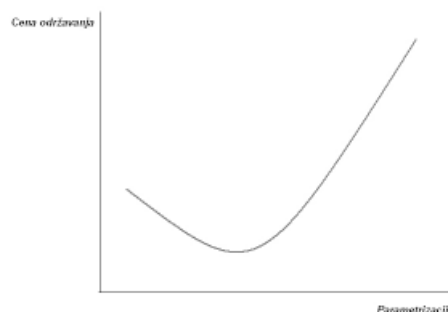
Obzirom da je evolucija stalni proces, dugoročni ugovori o održavanju predstavljaju dobar preduslov za evolutivno održavanje. Srednje i visoko parametrizovani sistemi su stoga dobri kandidati za evolutivno održavanje. Brand dobavljači, čiji su sistemi visokoparametrizovani, često izdaju nove verzije softvera, kao deo strateške bobre protiv konkurencije. Tako borba za opstanak na tržištu postaje pokretač evolucije brandiranih ERP(*Enterprise Resource Planning*, planiranje resursa preduzeća) sistema.

U slučaju softvera koji se prave po meri korisnika, koji su najčešće srednje parametrizovani, *outsourcing*(održavanje softvera od strane trećeg lica) i dugoročno ugovaranje održavanja može ograničiti mogućnosti inovativnog održavanja. *Outsourcing* kompanije ne žele da vrše inovacije na softveru i predmet njihovih ugovora su uglavnom korektivna održavanja. Tada je takođe teško povući liniju između održavanja i razvoja, te se često vodi borba oko toga šta spada u domen održavanja, a šta u domen razvoja. U tom slučaju, dobro je rešenje zapošljavanje određenog broja softveraša. Tada se ne postavlja pitanje predmeta njihovog rada, jer su zaposleni u preduzeću, te imaju obavezu da vrše i korektivno, ali i inovativno održavanje. Na taj način, zaposleni softverski stručnjaci se mogu uposliti da vrše konstatno razvoj softvera koji je u fazi eksploatacije.

Najčešća strategija menadžmenta preduzeća u pogledu održavanja informacionih sistema će biti da što više pojednostave i pojeftine održavanje sistema. Tada će najpovoljnija biti nabavka sistema koji će biti optimalno parametrizovan, u dobroj meri već skrojen prema specifičnosti sistema, odnosno, razvijan prema potrebama te firme. Održavanje će biti takvo da će ga uglavnom vršiti sami korisnici promenom vrednosti parametara sistema u toku samog rada.

Druga strategija prilikom nabavke informacionog sistema je okrenuta sigurnosti rešenja garantovana «brendom». ERP rešenja poznatih brendova podrazumevaju veliku opštost jer se

ne prave po meri jednog korisnika, već su namenjena širokom tržištu. Opštost ERP rešenja se postiže visokim stepenom parametrizacije. Na ovaj način, obično se gubi mogućnost podešavanja parametara od strane korisnika i uvode se ili administratori koji će taj posao obavljati u preduzeću ili će biti angažovano osoblje dobavljača ERP rešenja. Glavna vrednost ovakve solucije se nalazi zapravo u održavanju koji se najčešće ugovara na mesečnom nivou. To kod ovakvih aranžmana podrazumeva i s vremena na vreme dobijanje upgrade-a(nove verzije softvera sa novim ili unapređenim karakteristikama) isporučenog softvera ERP-a u kojim se nalaze ispravke grešaka iz prethodnih verzija, zatim prilagođavanja prema promenama nastalih u okruženju, na primer pravnom ili poreskom sistemu. Upravo zbog ovakvog aranžmana, ali i brenda, ova strategija može da bude mnogo skuplja od prethodne. (slika 2)



Slika 2. – Zavisnost cene održavanja od stepena parametrizacije[3]

Ipak, ovo rešenje može da osigura duži životni vek, jer brend proizvođač softvera, može da bude revnosan u pogledu unapređivanja sopstvenog rešenja, kako bi ostao u trci sa konkurencijom. S druge strane, visoka cena ovakvog aranžmana, može da odgovori menadžment od ovakvog način održavanja, odričući se novih verzija i prepuštajući podršku sopstvenom IT osoblju. Tada, naprotiv može da dođe do suprotnog efekta i do preranog zastarevanja informacionog sistema, pre svega, zato što osoblje koje se zapošljava da bi održavalo takav sistem, može da bude nedovoljno obučeno da to radi kao što bi to radio dobavljač ERP rešenja, a zatim što proizvođač ERP rešenja posle nekog vremena ne radi više ispravke starih verzija, već dorade sprovodi kroz nove verzije softvera.

Nedostatak prve strategije koja je utemeljena na sniženju troškova održavanja, se ispoljava u slučaju kada promena koju treba izvršiti prevazilazi fleksibilnost sistema, te je opet neophodno angažovati jednog ili više programera, da izmene uvedu u sam kod aplikacija informacionog sistema. Obzirom da se teži ka smanjenju troškova održavanja, za intervenciju se obično angažuje proizvođač softvera, samo po potrebi. Na ovaj način se održavanje svodi na prosto osposobljavanje softvera da radi u novim uslovima. Promene koje se sprovode su minimalne i neophodne. U kod se ne unose nikakve inovacije. Da bi softver bio inoviran mogao bi se ugovoriti neki dugoročniji aranžman sa proizvođačem sa obavezom isporučivanja novih verzija s vremena na vreme, koje bi pored obaveznih prilagođavanja softvera novim okolnostima,

sadržao i nove funkcionalne elemente, kao rezultat razvoja tehnologija i konstantnog istraživanja. Osim dugoročnih ugovora o održavanju sa proizvođačima softvera, postoji i jeftiniji, često i efikasniji način obezbeđenja dugoročne podrške sistemu, a to je zapošljavanje određenog broja specijalista za razvoj softvera-projektanata i programera.

Ipak, samo zapošljavanje programera ne garantuje i inovaciju sistema. Da inovacija ne bi zavisila samo od profesionalne odgovornosti programera, potrebno je doneti strategiju razvoja informacionog sistema i primenjivati je već od samog početka njegove eksploatacije. Ako bi svaka nova verzija sistema donosila poboljšanja u strukturi i procesima, a ne samo ispravke grešaka i dorade zbog zakonskih izmena, na primer, tada bi se informacioni sistem mogao uvesti u proces konstantnog razvoja, nalik nekom biološkom sistemu, koji se menja, raste i razvija kroz niz generacija.

5. KAKO EVOLUTIVNO ODRŽAVANJE SPROVESTI U PRAKSI

Evolutivno održavanje podrazumeva i korektivno i inovativno održavanje, ali nije svako održavanje evolutivno. Korektivno popravlja greške i prilagođava sistem izmenama nastalim u zakonskim regulativama, na primer. Inovativno donosi poboljšanja nastala kao rezultat novih znanja i tehnologija. Da bi održavanje poprimilo oblik evolutivnog, potrebno da ono bude sprovedeno planski, strateški i kontinuirano, da se odredi politika koja određuje ko, šta i kada radi. Evolutivno održavanje ne sme da ostane samo na odgovornosti samih učesnika u održavanju softvera, njihove raspoloživosti i motivisanosti. Neophodno je definisati strategiju ovakvog održavanja, određujući dinamiku realizacije novih verzija softvera, tačno svih akcija unapređenja i inoviranja softvera, aktera i rokova.

Šta to praktično znači? To znači da treba definisati sledeće:

Predmet promena

Potrebno je sagledati šta su kritične tačke u sistemu. Proceniti stanje baze podataka, koda, njegovu jasnost, opterećenost neaktivnim, suvišnim kodom, postojanje dokumentacije. Takođe treba realno proceniti adekvatnost i aktuelnost tehnologije na koju se sistem oslanja i otkriti kakvi su trendovi u toj oblasti, te na osnovu toga predvideti šta od toga bi moglo da se primeni u budućnosti. Takođe je neophodno postaviti prioritete i na osnovu svega toga treba odrediti ciljeve i zadatke u narednim godinama u pogledu revizije i inovacije koda, baze, telekomunikacione tehnologije itd...

Rokovi

Kada se odrede zadaci, neophodno je planirati i raspored aktivnosti, šta će se i kada obaviti. Potrebno je odrediti dinamiku izdavanja novih verzija aplikacija, kao i njihov sadržaj. Predmet promena kao i rokovi određuju i kadrove koji treba da se zaposle, ali se mogu formirati i prema raspoloživim kadrovima.

Sredstva

I kada je reč o inovacijama, ali i reviziji, u pozadini se nalazi tehnologija. Treba planirati koji će alati biti potrebni, da li su potrebni novi razvojni alati, da li je moguće obezbediti probne verzije za testiranje, da li su dostupni alati za softverske metrike, da li je moguće dobiti nove uređaje na probu, kao što su fiskalne kase, bar kod čitači, novi računari, itd. Takođe treba planirati finansijska sredstva za ono što je neophodno da se nabavi i od opreme, ali i od softvera.

Kadrovi

Kada se odredi šta treba da se menja, neophodno je odrediti minimalan broj kadrova, kao i njihov profesionalni profil. Ovo zavisi od veličine sistema, tehnologije koja se koristi za razvoj i broja ljudi koji je učestvovao u razvoju ovog sistema. Ovaj broj treba da obuhvati projektante, programere, analitičare i druge profile koji su neophodni za dati sistem. Treba imati na umu da održavanje iako na ovaj način praktično prerasta u razvoj, ipak predstavlja manje zahtevan posao u odnosu na inicijalni razvoj pre uvođenja u eksploataciju. Sistem koji se eksploatiše zahteva mnogo manje posla, te je potrebno angažovati verovatno tek trećinu ili još i manji broj specijalista nego u fazi razvoja. Ovaj broj ipak zavisi i od željene dinamike progressa sistema u eksploataciji. Takođe treba imati na umu da je prva godina eksploatacije vreme koje karakterišu brojni novi korisnički zahtevi, te će to biti period veoma intenzivnog razvoja sistema u eksploataciji. Kadrovske potrebe u ovoj fazi sa maksimalnim opterećenjem treba da budu mera i za naredni period.

Budžet

Kada se definiše sve što je potrebno u dužem vremenskom periodu, potrebno je odrediti i finansijska sredstva koja će to sve obezbediti. Usvojen budžet je taj koji verifikuje ili koriguje sve prethodne stavke plana.

U određivanju profila, potrebno je sagledati prirodu promena koje će biti predmet rada u narednim godinama. Ukoliko se planiraju promene strukture, neophodni su projektanti. Ukoliko je fokus na čestim organizacionim promenama u preduzeću, možda bi trebalo i nekog biznis analitičara uvrstiti u tim. Ukoliko je revizija koda od primarnog značaja, tada iskusni programeri treba da obavljaju dati posao. Ukoliko su inovacije ono što se najviše očekuje od tima, možda će biti angažovani i drugi specijalisti. Ne treba upasti u zamku i dati prevelik značaj jednom od ovih segmenata na štetu drugih. I revizija procedura i struktura baza podataka, kao i koda je od značaja za evolutivno održavanje. Takođe treba razmotriti i način angažovanja ovih profila. Možda će pojedini specijalisti biti neophodni samo kratko vreme, a kasnije samo opterećivati budžet IT službe, pa ih treba angažovati samo privremenim ugovorima. Takođe treba razmotriti i mogućnosti rotacija i premeštanja članova timova sa jednih na druge obaveze zavisno od faze kroz koju IS prolazi. Ali treba predvideti ko će od kadrova raditi na nepredviđenim situacijama, kako se proces razvoja ne bi ugrozio, zbog nekih nepredviđenih hitnih intervencija.

6. KAKO EVOLUTIVNO ODRŽAVANJE FUNKCIONIŠE

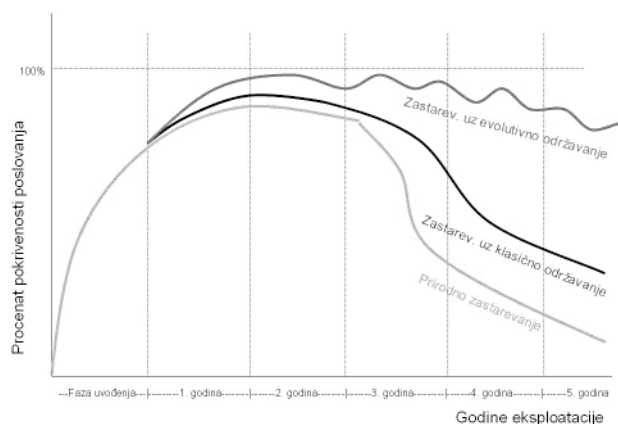
Informacioni sistemi se nalaze u stalnom procesu promena. [5] Kada je informacioni sistem uveden u firmu i kad korisnici počnu samostalno i svakodnevno da koriste aplikacije, tada počinju da upoznaju njegovu pravu snagu. Upoznajući mogućnosti informacionih tehnologija, korisnici se informaciono edukuju i počinju da dobijaju ideje kako da od njega dobiju više. Tada, ukoliko im kadar koji se bavi razvojem softvera stoji na raspolaganju, počinju sve intenzivnije da upućuju korisničke zahteve. Realizacijom korisničkih zahteva, softver počinje da raste.

Kada se održavanjem softvera bavi neka eksterna firma, postoji diskusija kakva dorada softvera spada u domen održavanja, a kakva u domen razvoja. U prvoj godini eksploatacije, teško je povući granicu između ova dva domena. Ukoliko je tim softverskih kadrova zaposlen u preduzeću na stalnom održavanju, tada se ovakvo pitanje i ne postavlja. Oni su tu da realizuju svaki zahtev. Zato informacioni sistem praktično nastavlja svoj razvoj u prvoj godini eksploatacije.

Međutim tek što su se korisnici navikli na softver i počeli da ga cene, u okruženju su se već desile promene i softver postaje sve manje upotrebljiv. Tada korektivno održavanje stupa na scenu. Softver se interventno menja, jer je rok za primenu novih propisa, na primer, iz poreskog sistema, obično surovo kratak, tako da je fokus softverskog tima na roku da se softver osposobi za rad, dok je kvalitet dorade nižeg prioriteta. Kada je posao obavljen, retko se ko vraća da na miru uradi kvalitetnije rešenje. To postaje kod lošeg kvaliteta. Evolutivno održavanje sada ili kasnije revidira ovakav kod i izbacuje verziju sa prečišćenim kodom.

U drugoj godini eksploatacije, kada su ideje korisnika iscrpljene, a slučajne greške u kodu napravljene u fazi razvoja ispravljene (za neke opcije softvera koje se ređe koriste, može proći dosta vremena dok se ne otkriju kodne greške), softver dostiže punu zrelost. Informacioni sistem u svom zenitu pokriva veliki deo poslovanja, a korisnici su zadovoljni. Međutim, vremenom na tržištu se pojavljuju nove mogućnosti, kao što je dostupnost preko interneta, mobilnih telefona, itd. Tada bi na ovaj način tim projekatana i programera angažovanih na konstantom, evolutivnom održavanju reagovao tako što bi pristupio izradi novih verzija softvera informacionog sistema, kako bi omogućio korišćenje novih tehnologija. Ove inovirane verzije mogle bi biti dosta različite u odnosu na prethodne. U terminologiji evolucije, izlazak nove verzije softvera bi predstavljao nastanak nove generacije. Nova generacija ovako može biti velikim delom drugačija od prethodne, te se informacioni sistem, takvim inovacijama može dosta udaljiti od početne verzije i transformisati u sasvim nov sistem. Međutim, obzirom da nije došlo do nove nabavke informacionog sistema, nove investicije, već se stari transformisao u novi, možemo smatrati da je došlo do podmlađivanja informacionog sistema i značajnog produženja njegovog životnog veka (slika 3)(slika 4).

Dugotrajnim korišćenjem aplikacija opcije softvera se diferenciraju po učestalosti korišćenja. Opcije koje se nikad



Slika 3. – Zavisnost životnog veka informacionog sistema od načina održavanja[4]

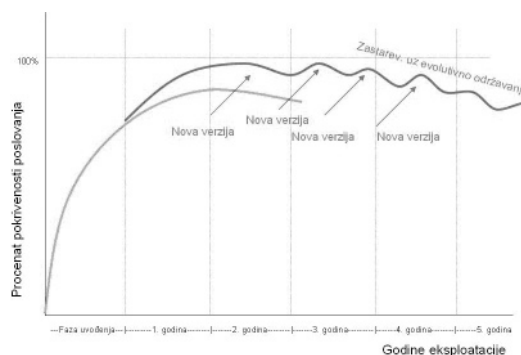
ne koriste tek posle nekog vremena postaju očigledno suvišne. Često korisnici, usled nekorišćenja, i zaborave čemu su koristile neke. To postaje višak koda.

Tek puna eksploatacija može i da pokaže menadžmentu da neki procesi nisu dobri i da treba da se menjaju. Tada programeri ne vole da se odriču koda, već napušteni kod ostavljaju neaktivnim (stave ga pod komentar), dodajući novi, u strahu da bi se korisnici ponovo mogli predomisliti i poželeti da se stvari vrate na staro. Kada prođe dovoljno vremena koje potvrdi koje rešenje je konačno, tada se već niko od programera ni ne seća da je ostalo koda koji stoji beskoristan i čeka da bude obrisan. Ovo se takođe mora otkloniti u nekoj kasnijoj verziji koja bi bila rezultat revizije koda.

Prilikom programiranja nekih hitnih stvari, programer dobije ideju ili bude zamoljen da kad bude imao više vremena uradi i neke druge stvari koje nisu neophodne za korisnika, ali mogu da budu od pomoći. Na primer neki sitan alat za preračunavanje. Kada prezauzetost programera prođe, programeri se retko sete šta su nameravali još da urade. To je nerealizovan kod. Evolutivno održavanje takođe mora da forsira da se ovakve stvari urade u nekoj od narednih verzija.

Klasično korektivno održavanje pored očigledne ispravke pogrešnog koda i prilagođavanja koda novim uslovima, može da ima za cilj i popravke/eliminacije:

- koda lošeg kvaliteta
- nekorišćenog koda
- neaktivnog (iskomentarisog) koda
- nerealizovanog koda



Slika 3. – Podmlađivanje informacionog sistema izradom novih verzija[4]

Da bi se bavilo ovakvim akcijama mora postojati visok stepen lične profesionalne odgovornosti softverskog kadra, kao i dovoljno vremena da im se posveti. Međutim, softverski kadar, angažovan na novim zadacima sa uvek kratkim rokovima, retko ima vremena da poboljšava unutrašnji kvalitet koda koji funkcioniše, sve dok to ne remeti njegov rad.

Evolutivno održavanje treba da se bavi ovim problemima. Pored klasičnih korekcija u čiju neophodnost ne može da se sumnja, da poboljšanje koda ne bi zavisilo od odgovornosti i kreativnosti onih koji ga obavljaju, potrebno je doneti strategiju koja obuhvata akcije koje treba sprovesti na softveru, njihovu dinamiku, sredstva, aktere. Aktivnosti koje treba da budu predmet evolutivnog održavanja su na primer:

- Revizija koda i eliminacija nekorišćenog, neaktivnog koda.
- Uprošćenje koda radi jednostavnosti održavanja, preglednosti i smanjenja broja linija koda.
- Revizija aplikacija i pronalaženje novih rešenja u cilju poboljšanja brzine izvršenja, jednostavnosti korišćenja.
- Realizacija pomoćnog obećanog koda

Međutim, glavna snaga evolutivnog održavanja leži u mogućnosti da objektivno sagledava valjanost postavke projekta u fazi razvoja i postupno i sukcesivno je menja u skladu sa novim otkrićima. Ovakve strukturne izmene zaista transformišu informacioni sistem u nov i eliminišu potrebu za novim investicijama u razvoj novog informacionog sistema. Strukturne promene su neophodne iz više razloga. Ne samo što se često dešavaju greške u projektu, zbog loše artikuliranih zahteva, već se često dešavaju i promene u samoj organizaciji preduzeća u njenoj strukturi i poslovnim procesima. Tada se aplikacije koje su skrojene po modelu jednog preduzeća koriste još samo domišljatošću samih korisnika, te ne daju adekvatne rezultate. Tada je potrebno postupno transformisati strukturu aplikacija, kao i procese i prevesti ga u nov sistem, koji će biti projekcija nove organizacije, a ne samo zastareo sistem koji očekuje jedne podatke, a dobija druge.

Pored toga, kadar koji treba da se bavi ovakvim održavanjem treba da prati savremene trendove u softverskoj industriji i kada prepozna da bi nešto moglo da unapredi rad sistema i olakša korisnicima njegovo korišćenje, da počne da uvodi nove funkcionalnosti. To može biti i prevođenje koda na nove programske jezike.

Osim toga, potrebno je pomenuti i da u kreaciji strategije evolutivnog održavanja treba definisati i pomoćne alate i tehnike, koje će pomagati učesnicima da otkriju sve one gore pomenute neželjene linije koda, kao što su na primer alati koji se bave softverskom metrikom.[1]

7. FINANSIJSKI I DRUGI EFEKTI EVOLUTIVNOG ODRŽAVANJA

Evolutivno održavanje dosta košta. Evolutivno održavanje angažuje više kadrova nego klasično održavanje. Mnogo je i više sati rada softverskih stručnjaka. Nove tehnologije su skupe. Pomoćni softver, softver za revizije koda, novi razvojni

alati, nove verzije sistema za upravljanje bazama podataka koštaju. Međutim, nabavka novih informacionih sistema košta više. Održavanje iziskuje konstantne troškove, a investicija predstavlja veliki udar na budžet, te je često potrebna i bankarska pomoć u vidu kredita sa određenom kamatnom stopom, koja dodatno poskupljuje nabavku softvera. Osim toga, nabavka softvera se vrši od nekog dobavljača koji u cenu softvera pored svih ovih troškova koje evolutivno održavanje vrši, uključuje i svoju zaradu. U slučaju kada održavanje rade kadrovi zaposleni u firmi, nikakvog dodatnog troška u vidu provizija, kamata i zarada posredovanja, osim troškova zarada i opreme, ne postoji.

Osim očiglednih direktnih finansijskih ušteda i drugačije dinamike ulaganja, ne treba zaboraviti velike uštede indirektnih troškova. Ti troškovi su sakriveni u satima intervjuisanja, dogovaranja, ubeđivanja i testiranja rešenja raznih od strane budućih korisnika, počev od menadžera do operatera. To vreme koje oni upotrebe na pripremu novog softvera, za njih nije produktivno i njihove redovne aktivnosti trpe, odlažu se i prenose u prekovremeni rad, koji predstavlja dodatni trošak za firmu. Ovakvi troškovi nastavljaju da postoje i pošto je novi sistem uveden, jer će rad biti sporiji zbog nedostatka rutine i potrebe da se radi paralelno na dva sistema u početnom periodu. Produkcija grešaka u korišćenju novog sistema će biti povećana, pa će takođe veliki broj sati biti utrošen na ispravku istih.

Osim svih ovih finansijskih efekata, kako direktnih, tako i indirektnih, ne sme se zaobići i vrednost podataka. Integracija novih sistema predstavlja bolan proces. Stari podaci se teško mogu iskoristiti u novoj bazi podataka, te obično bivaju napušteni, kada faza paralelnog rada dva sistema prođe. Podaci su blago firme i evolutivno održavanje to blago čuva i omogućava da se koristi.

8. ZAKLJUČAK

Informacioni sistem predstavlja veliku investiciju, koja odmah po uvođenju počinje da zastareva, udaljavajući se od poslovanja koje treba da pokriva. Kako bi povratio uložena sredstva, u interesu je organizacije da radi što duže.

Razlog zastarevanja je pre svega zbog promena u samoj organizaciji, zatim okruženju – tržištu, pravnom sistemu, tehnologiji koja se konstantno razvija. Da bi mogao da funkcioniše, sistem se mora prilagođavati ovim promenama. Akcije prilagođavanja predstavljaju servisne akcije, odnosno akcije održavanja. Međutim i samo održavanje hitnošću intervencija i postavljanje prioriteta brzine na štetu kvalitetu, doprinosi procesu starenja, opterećujući kod novim linijama, napuštenim kodom, deaktiviranim..

Učeći o drugim sistemima, kao što su biološki, možemo da promenimo perspektivu i promenimo proces starenja kod informacionih sistema. Činjenicu da biološki sistemi ne odumiru, jer evoluiraju, možemo primeniti na informacione sisteme. Konstatnim unapređivanjem koje bi se ogledalo u inovacijama, koje bi stalno pratila i revizija, odnosno čišćenje nepotrebnog, uspeli bismo da sistem održimo dugo mladim. Ovakvo održavanje bi se vršilo od samog uvođenja u eksploat-

aciju, a planiralo bi se i pre toga. Sistem bi se unapređivao postupno, kontinuirano i bez prestanka, kroz niz verzija. Na ovaj način bi sistem nastavio da se menja, razvija, raste i poput bioloških sistema, evoluiru u naprednije novije generacije sistema, aktuelne u datom trenutku. Jedna verzija bi bila jedna generacija, a promena u dužem vremenskom periodu bi ličila na evoluciju bioloških sistema. Obzirom na veliki broj promena realizovanih kroz generacije – verzije, to su ipak neki drugi sistemi drugačiji od polaznog. Ipak, kontinuitet u razvoju i evolutivna transformacija, kao i činjenica da nije došlo do nove investicije, čini da se ima utisak da je reč o istom sistemu, samo sistemu veoma produženog životnog veka.

Da ovakvo održavanje ne ostane samo na odgovornosti samih učesnika u održavanju softvera, njihove raspoloživosti i motivisanosti, potrebno je definisati strategiju ovakvog održavanja, određujući dinamiku realizacije novih verzija softvera, tačno svih akcija unapređenja i inoviranja softvera, aktera i rokova.

Iako ovakvo održavanje podrazumeva velike troškove održavanja usled većeg i stalnog angažmana određenog broja softverskih specijalista, kao i troškova svih sredstava koja bi se koristila, treba razmotriti uštedu kako finansijskih, tako i kadrovskih ulaganja u odnosu na ulaganja u novu investiciju u informacioni sistem svakih 3-5 godina, ali isto tako i ingritet podataka u dugom vremenskom periodu.

LITERATURA

- [1] Hackos, J.T., Using the Information Process Maturity Model for Strategic Planning, The Center for Information - Development Management, Denver, Colorado, USA, 2005, [Online]. Raspoloživo na: http://www.infomanagementcenter.com/download_center.shtml

- [2] Oro Radovanović, V., "Procenjivanje poslovne vrednosti informacionog sistema", *Zbornik radova 10. simpozijuma SymOrg*, Zlatibor, Srbija, 2006.
- [3] Oro Radovanović, V., Jut, A., Desančić, M., "Adaptibilnost informacionog sistema na organizacione promene preduzeća", *Festivalski katalog 13. festivala informatičkih dostignuća Budva 2006*, Budva, Crna Gora, 2006, 222-227
- [4] Oro Radovanović, V., Oro, V., "Produženje životnog veka informacionih sistema evolutivnim održavanjem", *Zbornik radova 11. simpozijuma SymOrg*, Beograd, Srbija, 2008.
- [5] Pereira Araujo, M., A., Kitchenham, B., "Evolutive Maintenance: Observing Object-Oriented Software Decay", Technical Report ES-XXX05, Keele University, UK, 2005.
- [6] Tamai, T., Torimitsu, Y., "Software Lifetime and its Evolution Process over Generations", *Software Maintenance 1992. Conference Proceed.*, Orlando, FL, USA, 1992, 63-69.
- [7] Wikipedia Slobodna enciklopedija, raspoloživo na: <http://hr.wikipedia.org/wiki/Evolucija>



Vesna Oro Radovanović, Štampa Sistem d.o.o, Beograd
Oblast interesovanja: poslovni informacioni sistemi, upravljanje projektima IS, poslovna inteligencija



mr Violeta Oro, Institut za zaštitu bilja i životnu sredinu, Beograd
Oblast interesovanja: ekologija, zoologija-nematologija, biotehnika

