

UNIVERZA V LJUBLJANI
EKONOMSKA FAKULTETA

MAGISTRSKO DELO

**PREDNOSTI IZBRANE METODOLOGIJE PRI
RAZVOJU POSLOVNIH INFORMACIJSKIH
SISTEMOV NA PRIMERU ZAVAROVALNICE
SLOVENICA**

LJUBLJANA, MAJ 2002

MOJCA HITI ŠKETA

IZJAVA

Študentka Mojca Hiti Šketa izjavljam, da sem avtorica tega magistrskega dela, ki sem ga napisala pod mentorstvom docent dr. Jurij Jaklič in skladno s 1. odstavkom 21. člena Zakona o avtorskih in sorodnih pravicah dovolim objavo magistrskega dela na fakultetnih spletnih straneh.

V Ljubljani, dne 8.5.2002

Podpis:

Kazalo

1	Uvod	1
1.1	Opredelitev področja in opis problema	1
1.2	Tveganja vodenja projekta.....	2
1.3	Osnovne trditve in cilji	2
2	Projekt razvoja informacijskih sistemov	4
2.1	Opredelitev projekta	8
2.2	Nadzor projektnega vodenja	10
2.3	Upravljanje s tveganji	13
2.4	Kakovost projekta.....	17
2.5	Merilo uspešnosti projekta.....	17
3	Metodologija razvoja informacijskih sistemov.....	18
3.1	Uvod	18
3.2	Pomen metodologije pri neizkušenih projektne skupinah.....	20
3.3	ISO priporočila za kakovost razvoja IS	20
3.3.1	<i>Standard za vodenje in zagotavljanje kakovosti (ISO 9000-3)</i>	21
3.3.2	<i>Informacijska tehnologija: Življenjski cikel programske opreme (ISO 12207)</i>	22
3.3.3	<i>Informacijska tehnologija Zahteve za kakovost in testiranja (ISO 12119)</i> ..	22
3.3.4	<i>Vrednotenje programske opreme (ISO 9126)</i>	22
3.4	Pregled in ovrednotenje nekaterih metodologij	23
3.4.1	<i>COBIT</i>	23
3.4.2	<i>MSF</i>	24
3.4.3	<i>CHESTRA</i>	25
3.4.4	<i>Rational Unified Process</i>	26
3.5	Oracle CDM/PJM metodologija	27
3.5.1	<i>Procesi projektnega vodenja</i>	29
3.5.2	<i>Življenjski cikel projekta</i>	30
3.5.3	<i>Implementacija metodologije s CASE orodjem Oracle Designer</i>	32
4	Analiza uporabe CDM metodologije pri razvoju informacijskih sistemov v zavarovalnici Slovenica: je ena sama metodologija dovolj?	33
4.1	Definiranje obsega, ciljev in pristopa.....	34
4.2	Plan projekta	37
4.2.1	<i>Določitev nalog projekta, časovni potek projekta in definiranje izhodnih rezultatov projekta</i>	37
4.2.2	<i>Oblikovanje projektne skupine</i>	40
4.2.3	<i>Komunikacije na projektu (poročanje, kontrola, upravljanje s problemi, tveganji in spremembami)</i>	44
4.2.4	<i>Zagotavljanje kakovosti na projektu</i>	45

4.2.5	<i>Upravljanje s konfiguracijo</i>	46
4.3	Kontrolne točke projekta.....	48
4.4	Ključne naloge procesov v različnih fazah razvoja IS.....	49
4.4.1	<i>Definiranje poslovnih zahtev, prenova poslovnih procesov</i>	49
4.4.2	<i>Pregled obstoječega sistema</i>	53
4.4.3	<i>Tehnična arhitektura</i>	54
4.4.4	<i>Načrtovanje in implementacija baze podatkov</i>	56
4.4.5	<i>Načrtovanje in izgradnja aplikacije</i>	58
4.4.6	<i>Prenos in pretvorba podatkov</i>	60
4.4.7	<i>Testiranje</i>	62
4.4.8	<i>Dokumentacija in izobraževanje</i>	66
4.4.9	<i>Tranzicija</i>	68
4.4.10	<i>Produkcija</i>	71
4.5	Spremljanje projekta.....	72
4.6	Zaključek projekta.....	73
4.7	Pregled uporabe CDM metodologije pri upravljanju projekta na zavarovalnici Slovenica.....	74
5	Sklep	80
	Literatura	83
	Viri	84

Seznam slik

Slika 1:	Časovni potek projekta.....	5
Slika 2:	Organizacija projektnega vodenja v življenjskem ciklu projekta.....	30
Slika 3:	Procesi CDM metodologije skozi faze.....	32
Slika 4:	Shematičen prikaz izvedbenega načrta projekta na zavarovalnici Slovenica.....	40
Slika 5:	CDM/PJM projektne organizacije.....	41
Slika 6:	Organigram projektne skupine v Slovenici.....	43

1 Uvod

1.1 Opredelitev področja in opis problema

Uspešno rastoče podjetje zaradi povečanega obsega poslovanja, podedovane organizacijske strukture, organizacije dela ter poslovnih procesov, ki niso primerno zasnovani za nov življenjski cikel operacij, zahtevnost trga, obseg poslovanja ter zadovoljstvo strank, dostikrat na trgu izgublja konkurenčno prednost. Da ne pride do stagnacije ali propada podjetja, je potrebno zastaviti jasno in nedvoumno poslovno strategijo procesa preobrazbe, ki zajema tudi strategijo informacijskega razvoja. Takšni procesi preobrazbe so lahko izredno obsežni in zahtevni ter porabijo ogromno različnih sredstev, predvsem časa, ki pa je za uspeh načrtovane preobrazbe ključen, saj je zelo omejen. Po drugi strani se z enako hitrostjo razvija tudi razpoložljiva informacijska tehnologija, ki omogoča hitrejše odzive podjetij na tržne zahteve. Zato je upravljanje strategije informacijskega razvoja s stališča iskanja ravnovesja med zasnovo informacijskega sistema v smislu poslovnih procesov in razpoložljive tehnologije ob zasnovi projekta prenove informacijske tehnologije ter med njeno dejansko konkurenčnostjo in uporabnostjo v času njene produkcije izredno zahtevno.

Strategija informacijskega razvoja se mora voditi vzajemno s poslovno strategijo. Slabo definirana funkcionalnost informacijskega sistema oziroma zastarela funkcionalnost z vidika sodobnih poslovnih procesov vodi do prenove informacijskih sistemov, katere temeljni cilj je usmeriti organizacijo na pot poslovnega uspeha. V takšnih primerih se vodstvo oddelkov informacijskih tehnologij sooča z vrsto problemov, katerih rešitev naj bi prinesla večji poslovni uspeh organizacije, zadovoljstvo strank in višjo učinkovitost zaposlenih.

Celoten proces preobrazbe mora imeti jasno definirano vizijo in plan. Novi cilji morajo biti enolično definirani, razumljivi in dovolj dramatični, da dajo zadostno inspiracijo in motivacijo zaposlenim za dejansko akcijo, plan prenove pa mora opredeliti način in postopke za doseg ciljev.

Pri projektih izgradnje oziroma prenove informacijskega sistema (v nadaljevanju bo z izgradnjo mišljena tudi prenova, če ne bo eksplicitno drugače navedeno), ki so sestavni del procesa preobrazbe, se projektne skupine v številnih podjetjih srečujejo z različnimi problemi, ki so posledica neprimerne organizacije procesov projekta v dolgoročnem smislu. Vse preveč pozornosti se posveča učinkovitosti programske opreme, hitremu razvoju novih rešitev, premalo pa ostalim spremljajočim dejavnostim, kot so testiranje nove programske opreme, izobraževanje uporabnikov, izdelava ustrezne dokumentacije, prenos in pretvorba podatkov iz obstoječega v nov informacijski sistem, oblikovanje postopkov kasnejšega vzdrževanja aplikacije ter pravilno načrtovan in nadzorovan postopek zagona novega informacijskega sistema. Prav tako se v zagonu projekta ne oblikuje projektne skupine ustreznih strokovnjakov, ki bi bila kos posameznim nalogam.

Zaradi slabe metodologije za razvoj sistemov, ki nejasno opredeljuje aktivnosti in rezultate posameznih faz in ne vključuje ustreznega nadzora nad izvajanjem posameznih nalog, se lahko pojavijo različna tveganja. Tako projekt dostikrat ostane nezaključen ali pa je zgrajen informacijski sistem, ki ne podpira aktualnih poslovnih procesov v celoti.

1.2 Tveganja vodenja projekta

Tveganje vodenja projekta lahko opredelimo kot kakršnokoli prekoračitev obsega ali virov projekta (čas, denar, človeški viri) (McConnell, 1998, str. 93). Pri skrbno načrtovanih projektih je potrebno že pri pripravi projekta predvideti možna tveganja in določiti postopke, kako tveganja zmanjšati ali se jim popolnoma izogniti v smislu zviševanja kakovosti končnega izdelka. V kolikor se na projektu ne izvaja aktivnost upravljanja s tveganji, pomeni, da se ne upošteva rezultatov in izkušenj predhodnih uspešnih projektov.

S pravilnim in pravočasnim definiranjem potreb in študijo izvedljivosti se lahko že na začetku izognemo najpogostejšim tveganjem pri razvoju informacijskih sistemov, ki so v splošnem naslednji (Vallabhaneni, 1996, str. 755-758) :

- neprimerno vodenje življenjskega cikla projekta razvoja novega informacijskega sistema,
- nov informacijski sistem ne podpira uporabnikovih zahtev oziroma poslovnih procesov v celoti,
- neprimerni viri za razvoj in vzdrževanje novega informacijskega sistema,
- nepovezljivost različnih sistemov v poslovnem procesu, zaradi česar se izgublja poslovna prednost in podatki med sistemi,
- neprimeren in neučinkovit nadzor nad razvojem in/ali vzdrževanjem informacijskega sistema,
- prekoračitev stroškov razvoja in vzdrževanja novega informacijskega sistema,
- prekoračitev časovnih okvirov za razvoj novega informacijskega sistema in demotiviranje zaposlenih.

1.3 Osnovne trditve in cilji

Uporaba katerekoli priznane metodologije na projektu razvoja novega informacijskega sistema pomeni uporabiti najboljše izkušnje za doseg kakovosti na projektnem delu.

Metodologije razvoja novih informacijskih sistemov se v skladu z razvojem novih informacijskih tehnologij in pridobivanjem izkušenj na vodenju projektov nenehno spreminjajo. Prav tako so odvisne tudi od velikosti in zahtevnosti projektov ter zahtev naročnika. Na voljo je precej literature predvsem s področja vodenja informacijskih projektov, dosti težje pa je najti celovito metodologijo, ki pokriva tako vodenje kot vsebino projekta. Slednje so dostikrat razvite predvsem za uporabo na konkretnih orodjih za razvoj aplikativnih rešitev.

Podjetje Oracle je za potrebe projektov razvoja novih informacijskih sistemov z uporabo njihovih (Oracle) orodij razvilo svojo metodologijo, tako imenovano Custom Development Method Classic (CDM) v povezavi s PJM Advantage metodologijo (metodologija vodenja projektov informacijske tehnologije). CDM metodologija temelji na tradicionalnem, faznem pristopu k razvoju novega informacijskega sistema, njeni izpeljanki CDM Fast Track in CDM Lite pa sta skrajšani različici celotne metodologije, prilagojeni za manj obsežne in kratkotrajne projekte oziroma za izdelavo prototipov. Metodologija je usmerjena na razvoj informacijskih sistemov s pomočjo Oraclovih orodij (Oracle Designer, Oracle Developer), vendar je toliko splošna, da dopušča, v kolikor spremenimo tehniko izvedbe, uporabnost na katerikoli drugi tehnologiji in s tem uporabo na kateremkoli projektu razvoja novega informacijskega sistema.

Cilj magistrskega dela je na primeru uporabe Oracle CDM/PJM metodologije pri projektu razvoja novega informacijskega sistema za podporo produkcijskim poslovnim procesom v zavarovalnici Slovenica analizirati in prikazati, kako lahko pravilna izbira in uporaba preizkušene metodologije razvoja novih informacijskih sistemov dviguje kakovost projekta in produktivnost projektne skupine v smislu zmanjševanja znanih tveganj pri projektih informacijskih tehnologij.

Magistrsko delo se po eni strani opira na teoretična spoznanja o raznovrstnih pristopih k vodenju projektov, ki jih zasledimo pri različnih avtorjih, po drugi strani pa preverja ta spoznanja na samem projektu razvoja novega informacijskega sistema za podporo produkcijskim procesom v zavarovalnici Slovenica. Glede na to, da je projekt tako glede obsega (prenova poslovnih procesov dveh zavarovalnic v združitvenem procesu¹, zagotavljanje poslovanja na več kot 700 lokacijah po Sloveniji) kot glede virov (preko 100 članov projektne skupine na različnih lokacijah) izjemno obširen in zahteven, se bom osredotočila predvsem na analizo nalog, rezultatov in upravljanj s tveganji najbolj tveganih delov projekta. V okviru naloge sem uporabila različne metodološke pristope za proučevanje tematike:

- analitično-teoretično proučevanje različnih pristopov k vodenju projektov,
- proučitev priporočil CDM/PJM metodologije, ki so nastala na osnovi v praksi potrjenih izkušenj,
- analiza vpliva izbrane CDM/PJM metodologije na kakovost projekta izgradnje novega informacijskega sistema za podporo produkcijskim poslovnim procesom v zavarovalnici Slovenica s poudarkom na kritični oceni ustrezne definicije faz/procesov/nalog po metodologiji, podkrepļeni z izkustvenimi primeri iz konkretnega projektne delo.

¹ Danes sicer vemo, da do združitve trenutno ne bo prišlo, vendar v nalogi opisujem stanje tistega trenutka, ko so se aktivnosti izvajale.

CDM/PJM metodologija vsebuje priporočila za aktivnosti v življenjskem ciklu projekta in sicer od zaključka študije izvedljivosti do produkcije z novim informacijskim sistemom.

2 Projekt razvoja informacijskih sistemov

Projekt je množica enkratnih, časovno omejenih aktivnosti, ki z uporabo človeških, tehničnih in finančnih virov rezultirajo v vnaprej določen rezultat. Ne glede na to, ali je dejansko zaključen, kot je bilo definirano v planu, ali pa predčasno prekinjen, ima projekt vedno neko končno točko. Temeljne značilnosti projekta so:

- vsak projekt mora že na začetku imeti jasno opredeljen cilj,
- časovno mora biti opredeljen s svojim začetkom in koncem,
- porablja vnaprej opredeljene vire, katere je potrebno v času projekta upravljati, nadzirati, ocenjevati in korigirati.

Glede na vsebino projektov informacijskega razvoja ločimo (McLeod, B.I., str. 5-6):

- razvoj (lasten ali s pomočjo zunanjih partnerjev) novega informacijskega sistema,
- implementacija (z ali brez uporabniških modifikacij) novonabavljenega programskega paketa,
- uvedba sistemov za podporo odločanju oziroma skladišč podatkov,
- razvoj nove arhitekture informacijskega sistema,
- implementacijo nove tehnologije, kot je na primer uvedba novega operacijskega sistema ali mrežne programske opreme,
- prenova poslovnih procesov.

V magistrskem delu se bom omejila predvsem na projekte razvoja novega informacijskega sistema.

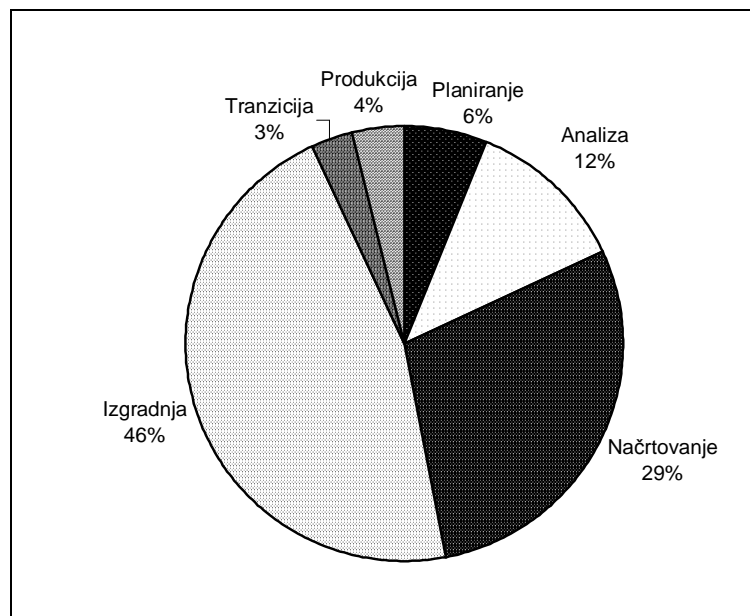
Opravila v času trajanja projekta razvoja novega informacijskega sistema običajno imenujemo življenjski cikel projekta. Opredelitev pristopov k definiranju življenjskega cikla informacijskega sistema se glede na različne parametre projekta (velikost, orodje, metodologija, razpoložljivost virov, znanje in izkušnje...) vodi na različne načine.

Tradicionalne metodologije navadno delijo razvoj novega informacijskega sistema na šest glavnih faz, in sicer na planiranje, analizo, načrtovanje, izgradnja, tranzicijo in produkcijo (Vallabhaneni, 1996, str. 750-753) (Slika 1). Faze razvoja projekta so različni avtorji različno poimenovali in tudi delili, vendar se tu omenjena delitev ujema z delitvijo, ki jo je Oracle uporabil tudi pri svoji CDM metodologiji (Oracle Method: CDM Classic Method Handbook, 2000, str. 1-1).

V fazi planiranja se zbirajo ekonomske, operativne in tehnične informacije o problemu, ki ga je potrebno rešiti. Definirati je potrebno področje in cilje predlaganega sistema, izdelati

grobo analizo podatkov in procesov, ki naj bi jih novi informacijski sistem zajemal, razumeti delovanje obstoječih sistemov in definirati potrebe uporabnikov, izdelati predhodne ocene stroškov in koristi ter določiti roke izvedbe. Identificirati in oceniti je potrebno alternativne sisteme ter analizirati stroške in koristi za te sisteme. Na koncu sledi izbor najboljše rešitve. Rezultat te faze predstavljajo opredeljeni cilji in študija izvedljivosti, ki zajema operativne, ekonomske in tehnične obrazložitve za vsak alternativni sistem. Izdelajo se priporočila za vodstvo družbe glede razvoja in predhodni izvedbeni načrt.

Slika 1: Časovni potek projekta



Vir: McConnel Steve, str.63-69, Oracle Method: CDM Classic Method Handbook str.1-20)

Sledi faza analize, ki zajema podrobno analizo podatkov in procesov, razumevanje delovanja obstoječih sistemov in potreb uporabnikov. Definirajo se zahteve sistema, poslovne zahteve, nadzor in varnost sistema.

Po CDM/PJM metodologiji so že v planu projekta predvidene aktivnosti zaključevanja posameznih faz, ki so namenjene oceni zaključene faze, analiziranju tveganj in kakovosti posamezne faze ter, glede na rezultate, morebitnim korekcijam izvedbenega načrta. Po zaključku faze analize so te aktivnosti še posebej pomembne, saj je za projektno skupino in vodstvo projekta tu zadnja priložnost, ko se še lahko dogovarja in spreminja izvedbeni načrt tako vsebinsko kot tehnično. Zavedati se je potrebno, da je dolžina projektov, vodenih na opisani, tradicionalni način, izredno dolga, tudi 3-5 let (Finkelstein, 1992, str.5)

in da zajemata fazi planiranja in analize po nekaterih podatkih petino celotnega razvojnega cikla projekta (18% (Oracle Method: CDM Classic Method Handbook, 2000, str. 1-20)). To pa je že časovno obdobje, ko lahko pride do velikih sprememb tudi v razpoložljivi informacijski tehnologiji (obstoječa tehnologija in njene zmogljivosti, cena, znanje in izkušnje projektne skupine, spremembe sestava projektne skupine in drugo). Po drugi strani pa lahko podrobna analiza poslovnih in sistemskih zahtev razkrije optimalnejšo tehnološko rešitev od v študiji izvedljivosti izbrane (na primer namesto novega transakcijskega informacijskega sistema, ki bi nadomestil enega ali več obstoječih sistemov, boljše izdelava skladišča podatkov, ki se polni iz obstoječih transakcijskih sistemov).

Fazi definiranja in analize lahko štejemo kot pripravo na nadaljnji življenjski cikel projekta, to je izvedbo novega informacijskega sistema, ki sestoji iz faze načrtovanja in izgradnje novega sistema.

Za uspešen in učinkovit sistem je potrebno v fazi načrtovanja informacijskega sistema izdelati splošno in podrobno zasnovo sistema, ki morata biti skladni s prej opredeljenimi zahtevami in cilji. Splošna zasnova zajema določitev namena in opis glavnih funkcij sistema in podsistemov. Opis mora zajemati tudi oceno teh funkcij, določitev njihovih vhodnih in izhodnih podatkov oziroma informacij ter zahteve glede nadzora in varnosti. Podrobna zasnova vsebuje specifikacije za potek programov, vhodne in izhodne datoteke, ekrane, testiranje, prenos in pretvorbo podatkov, dokumentacijo in uvajanje uporabnikov. Rezultat te faze so opisi sistemskih zahtev, logična in fizična zasnova podatkovnih baz, podatkovnih slovarjev, programske specifikacije ter druge značilnosti za vhodne, procesne in izhodne aktivnosti.

Faza izgradnje, ki je najobsežnejša faza glede obsega dela v celotnem življenjskem ciklu projekta, zajema programiranje in po poprejšnjih specifikacijah izvedbo testiranja programov, izdelavo dokumentacije, izdelava skript za pretvorbo in prenos podatkov iz obstoječega v nov sistem, pilotno pretvorbo in prenos podatkov iz starega v nov sistem in usposabljanje (vsaj ključnih) uporabnikov.

Temu sledita fazi zaključka projekta, to sta tranzicija in produkcija novega sistema.

V fazi tranzicije se izvede prevzemni test novega sistema ter pretvorba in prenos podatkov iz starega v nov sistem. Pripravi se produkcijsko okolje in na koncu tudi dejanski zagon novega sistema v produkciji.

Faza produkcije zajema izvajanje novega sistema v produkcijskem okolju, spremljanje njegovega delovanja, vodenje aktivnosti v zvezi s problemi uporabe novega informacijskega sistema, njegovo nadgrajevanje in vzdrževanje. Nekateri avtorji faze produkcije ne vključujejo v življenjski cikel projekta. Tudi po CDM metodologiji faza produkcije predstavlja samo prvi, vnaprej opredeljeni rok za izvedbo modifikacij in prilagoditev novega informacijskega sistema, ki običajno predstavlja prvih 90 dni

delovanja novega informacijskega sistema (Oracle: Project Management Process and Task Reference, 1996, str. 11-3). Po tem roku se izvedejo aktivnosti zaključka projekta.

Pri razvoju novega informacijskega sistema se uporabljata dva pristopa, in sicer:

- od zgoraj navzdol, uporaben v primerih, ko se izhaja iz potreb vodstva organizacije in opredeljene dolgoročne strategije

ter

- od spodaj navzgor, uporaben v primerih, ko se želi zadovoljiti določene potrebe ali cilje uporabnikov ali izboljšati in odpraviti pomanjkljivosti delujočega sistema.

V praksi se največkrat uporablja kombinacija obeh pristopov.

Način popolnega faznega razvoja informacijskega sistema je primeren predvsem za primere, ko so specifikacije popolnoma razumljive tako končnim uporabnikom kot razvijalcem novega informacijskega sistema, kar pa je v praksi skoraj nemogoče doseči. Zato so poleg tradicionalnih metod razvoja informacijskih sistemov glede na obsežnost projekta in nove razpoložljive tehnologije pojavljajo tudi druge metode, kot so na primer (Vallabhaneni, 1996, str. 750-754):

- **prototipiranje**, za katerega je značilno izdelava prototipa, s pomočjo katerega se uporabnikom prikažejo osnovne funkcije programske opreme ter vhodi in izhodi iz sistema. V prototipu niso zajete različne podrobnosti programske opreme, temveč se podfunkcije, kontrole ipd. na podlagi uporabniških zahtev postopno dopolnjujejo. Namen tega pristopa je, da uporabniki aktivno sodelujejo pri zasnovi programske opreme in da se pomanjkljivosti odpravijo že v začetnih fazah. Prototip se kasneje običajno zavrže.
- **evolucijski** način, kjer teče razvoj neprekinjenega verzioniranja oziroma nadgrajevanja obstoječega sistema, kjer je poudarek na pouporabi zahtev in dokumentaciji načrtovanja. Je podoben ekstenzivnemu prototipiranju, verifikaciji in regresijskem testiranju, kar vodi do končnega izdelka. Takšen način je primeren predvsem tam, kjer je potrebno kmalu pokazati končne, uporabne rezultate in se nov informacijski sistem postopno dograjuje. Pri takšnem razvoju je izredno pomemben proces nadzora sprememb konfiguracije (change management control).
- **spiralni način**, ki temelji na pogostih točkah, kjer se odloča o sprejemljivosti sistema v smislu ali in kako nadaljevati proces razvoja ali ga opustiti. Takšen način razvoja je dostikrat odvisen od tehnologije. Spiralni način je v bistvu kombinacija prototipiranja in evolucijskega načina razvoja.
- **objektni** pristop, ki rešuje nekatere klasične probleme razvoja, kot so ločeno obravnavanje statičnih in dinamičnih podatkov. Ta pristop temelji na objektih, ki vsebujejo podatkovne strukture in pripadajoče postopke na tipih objektov in sporočilih. Objekti se lahko združujejo v kompleksne objekte. Prednost takšnega

pristopa predstavlja večkratna uporaba istih objektov. Če se uporabljajo že obstoječi in zanesljivi objekti, to vsekakor pripomore k zanesljivosti rešitve.

Kljub vedno novim tehnikam in tehnologijam, ki se pojavljajo na trgu, pa tradicionalne metode razvoja novega informacijskega sistema ne izginjajo, ampak so še vedno zelo v uporabi na projektih velikih razsežnosti in z velikim številom uporabnikov (Warren, 1998, str. B4-4).

2.1 Opredelitev projekta

Plan projekta opredeljuje začetni pogled na projekt, to je začetno stanje, želeno končno stanje, začetni obseg projekta in plan projektne del. Ne glede na izbrano metodologijo razvoja novega informacijskega sistema je na začetku potrebno izdelati podroben plan projekta. Je nadgradnja začetnega plana, ki je rezultat postopkov pobude za projekt in študije izvedljivosti. Slednja postopka sta za kakovost izvedbe projekta izredno pomembna, čeprav ju nekatere metodologije (med drugim tudi Oracle PJM) ne obravnavajo v življenjskem ciklu projekta, saj definirajo začetek projekta s trenutkom, ko je projekt potrjen s strani sponzorja. Drugje je postopek izdelave podrobnega izvedbenega načrta projekta vključen v plan, izdelan v postopkih odločitve za projekt in študije izvedljivosti, katerih končni rezultat je dokument Zagonski elaborat (System Decision Paper) (Ruthberg, 1991, str. 5-2). Vendar menim, da je podrobni plan projekta zaradi svojega obsega nadgradnja odločitve za projekt. Glavno vlogo in odgovornost za izdelavo podrobnega plana projekta nosi vodja projekta, ki pripravi izhodišča za dokument, usklajuje vsebine dokumenta z naročnikom in ostalimi člani projektne skupine ter skrbi, da je dokument v celoti sprejet tako s strani naročnika kot s strani izvajalca ter da je odobren s strani sponzorja projekta in poslovnih direktorjev, ki pokrivajo področja, katera so vključena v vsebino projekta.

Podrobni plan projekta zajema (Oracle: Project Management Process and Task Reference, 1996, str. 1-8 do 1-14, Ruthberg, 1991, str. 5-7 do 5-19) :

- **cilje projekta**, ki so nadgradnja namenskih ciljev projekta iz planiranja projekta, in sicer:
 - **namenske cilje**, ki opredeljujejo želeni končni rezultat, in
 - **objektne cilje**, ki definirajo končne rezultate projekta, ki so potrebni za dosego namenskih ciljev;
- **obseg projekta**, ki ga definirajo:
 - **aplikacije** oziroma informacijski sistemi, ki jih bo novi informacijski sistem nadomestil oziroma za katera poslovna področja, ki danes še niso informatizirana, se bo razvijal nov informacijski sistem,
 - **lokacije** novega informacijskega sistema (na primer določitev mesta in števila strežnikov, distribucijo delovnih postaj,...),

- **prenova poslovnih procesov**, kjer je potrebno opredeliti ali se bo izvajala, za katera poslovna področja in v kakšnem obsegu,
- **sistemski vmesniki**, kjer se opredelijo povezave novega informacijskega sistema z drugimi,
- **arhitektura**, kjer je potrebno opredeliti osnovno zasnovo, saj vpliva na ceno projekta,
- **prenos in pretvorbo podatkov** iz obstoječih sistemov ter v kakšnem obsegu se bo izvajal,
- **testiranje**, in sicer kje in v kakšnem obsegu se bo izvajalo,
- **investicija**, in sicer velikost investicije, ki je odobrena za projekt;
- **usposabljanje projektne skupine**, in sicer koliko, katero in v kakšnem obsegu je predvideno usposabljanje za člane projektne skupine ter upoštevanje planiranega izobraževanja pri načrtovanju človeških virov v izvedbenem načrtu ter v načrtu investicij,
- **izobraževanje končnih uporabnikov novega sistema**, in sicer v kolikšnem obsegu in kje se bo izobraževanje izvajalo;
- **kontrolne točke**, kjer se na nekem vnaprej določenem nivoju pregleduje uspešnost vmesnih rezultatov projekta, zato je potrebno natančno opredeliti, kateri so ti vmesni rezultati, določiti datum kontrolnih točk ter izhodne rezultate projekta, ki se na posamezni kontrolni točki preverjajo ter opredeliti vloge in odgovornosti preverjanja;
- **izhodni rezultati projekta**, kjer je potrebno natančno definirati in opisati posamezne izhodne rezultate projekta ter določiti, kdo lahko oceni, ali je izhodni rezultat zaključen;
- **omejitve in predpostavke**, ki lahko vplivajo na projekt, kot so nerazpoložljivost članov projektne skupine, ki so zasedeni z delom na drugih projektih, ali datumi, do katerih morajo biti znane določene odločitve, ki lahko vplivajo na projekt;
- **tveganja**, ki jih je potrebno identificirati in predvideti način njihovega reševanja v smislu zmanjševanja vplivanja na potek projekta;
- **nadzor projekta**: na kakšen način in v skladu s katerimi že sprejetimi pravilniki oziroma poslovno politiko se bo izvajal;
- **relacije z drugimi projekti in sistemi**, ki lahko vplivajo na potek določenega projekta;
- **kritični faktorji uspeha**, kjer je potrebno opredeliti vse faktorje, ki zagotavljajo uspešen potek in zaključek projekta;
- **pristop**, kjer se določa način izvedbe projekta, to je:

- **projektne metode**, ki bodo uporabljene na projektu,
- **standardi**, ki se bodo na projektu uporabljali (možno jih je samo naštet, v kolikor še niso izdelani, in se jih pripravi v začetni fazi projekta),
- **strategijo** vodenja in nadzora projekta,
- **osnovno tehnično okolje** (podrobneje naj se opredeli po zaključku faze analize, ko je sprejeta odločitev o tehnologiji izvedbe),
- **grobi izvedbeni načrt**, kjer je na kratko opisano, katere so glavne faze projekta in kaj so ključni izhodni rezultati posameznih faz,
- **opredelitev zaključka projekta**, in sicer kdo ga lahko določi ter kaj definira uspešen zaključek .

Ko so vsi ti parametri znani, se lahko posvetimo natančnejšemu izvedbenemu načrtu projekta, to je postavitve nalog v medsebojno vsebinsko odvisnost znotraj določenih časovnih okvirov ob upoštevanju razpoložljivih virov. Pri tem je potrebno upoštevati, da je celoten čas, predviden za projekt, dostikrat omejen z definicijo časa iz poslovne strategije podjetja, ki ga določa najvišje vodstvo podjetja oziroma naročnik projekta. Vodja projekta mora to pri planiranju upoštevati in v primeru, da zahtevani obseg projekta in predlagana rešitev projekta nista izvedljiva znotraj predvidenega časovnega območja, to vodstvu tudi argumentirano predstaviti s predlogom drugačne rešitve (na primer drugačen pristop, zmanjšanje obsega, drugačna tehnična rešitev, ...), ki je glede na izhodiščno in želeno končno stanje uresničljiva z dodeljenimi fizičnimi, tehničnimi in finančnimi viri znotraj časovnega območja.

2.2 Nadzor projektnega vodenja

Postopki nadzora projektnega vodenja so podporni procesi življenjskega cikla projekta (ISO 12207, 1995, str.27). Z njihovo pomočjo zagotavljamo, da procesi življenjskega cikla projekta nemoteno tečejo in so dejansko operativni. Definiranje in izvedba kakovostnega nadzora projektnega vodenja pomeni zmanjšanje tveganj na projektu in dvig kakovosti projekta ter je ključna aktivnost za uspešen zaključek projekta.

Nadzor projektnega vodenja se v vsaki organizaciji in na vsakem projektu lahko vodi na svoj način, vendar pa morajo plani in postopki za vodenje in nadzor projekta zajemati nadzor nad naslednjimi projektnimi komponentami (Vallabhaneni, 1996, str 758):

- čas,
- obseg,
- kakovost in
- viri.

Čas je ena najpomembnejših komponent projekta, ki je tudi najtežje dosegljiva. Četudi so projekti zaključeni, se dostikrat zgodi, da niso zaključeni v začetnem planiranem roku, kar

zmanjšuje njihovo kakovost in konkurenčnost. Pri planiranju roka izvedbe projekta je zato potrebno upoštevati:

- kompleksnost projekta,
- soodvisnost in zaporednost posameznih projektnih nalog,
- usposobljenost projektne skupine,
- velikost projektne skupine (večja projektna skupina potrebuje več časa za koordinacijo in nadzor),
- usposobljenost in motiviranost končnih uporabnikov,
- podporo sponzorja in določeno prioriteto projekta,
- potreben čas za izvedbo nalog na aktivnostih nadzora projekta in zagotavljanja kakovosti.

Kljub temu je potrebno paziti, da je časovni okvir realen, saj prevelik odmik zaključka projekta lahko pomeni neaktualnost izvedbe v skladu s poslovno politiko in s tem povezane visoke finančne stroške.

Obseg projekta mora enolično določati meje projekta, in sicer glede:

- funkcionalnosti, ki bo v projektu implementirana,
- arhitekturne zasnove novega informacijskega sistema,
- potrebe po prenosu in pretvorbi podatkov,
- potrebe po novih sistemskih vmesnikih,
- uporabe avtomatskih testnih orodij,
- zahtev za dokumentacijo,
- zahtev po obsegu usposabljanja končnih uporabnikov,
- produkcije in vzdrževanje sistema.

Kakovost na projektu dosežemo s kvalitetnimi in natančnimi opisi poslovnih zahtev ter s skrbjo, da so vse poslovne zahteve tudi dejansko ustrezno implementirane. Višja kakovost opisov poslovnih zahtev predstavlja tudi višje stroške, nepopolno dokumentiranje poslovnih zahtev pa največkrat vodi v povečanje časa in virov projekta ter v najslabšem primeru tudi v nedokončanje projekta.

Viri se nanašajo tako na delovno silo kot na razpoložljivo informacijsko tehnologijo (razpoložljiva strojna in programska oprema). Pri človeških virih je potrebno tako pri planiranju kot nadzoru na projektu upoštevati njihovo število, razpoložljivost in usposobljenost. Predvsem pri velikih projektih skupinah je potrebno upoštevati povečanje virov (čas, denar) za nadzor nad človeškimi viri.

Vse omenjene komponente so med seboj soodvisne, zato kakršnakoli sprememba katere od njih vpliva tudi na preostale tri. Zato uspešne tehnike nadzora projektnega vodenja niso omejene samo na spremljanje posamezne komponente. Glede na to lahko govorimo o naslednjih skupinah postopkov za nadzor projektnega vodenja:

- nadzor obsega in vsebine projekta,
- nadzor časovnega poteka projekta,
- nadzor virov projekta (človeški, finančni, tehnični),
- nadzor kakovosti projekta,
- nadzor nad izhodnimi rezultati projekta (dokumentacija, programska koda, ...).

V planu projekta je že definirano, v skladu s katerimi veljavnimi pravilniki oziroma poslovno politiko se bo izvajal nadzor. V naslednjih poglavjih je natančneje opredeljeno, kaj zajema posamezen nadzor. V kolikor katera od komponent v poslovni politiki oziroma pravilnikih organizacije ni opredeljena, je to potrebno storiti v planu projekta.

Nadzor obsega in vsebine projekta zajema postopke:

- nadzor nad odprtimi vprašanji projekta (situacije, akcije, zadeve ali vprašanja, ki se pojavijo v času projekta in zahtevajo odgovor oziroma rešitev vodstva projekta),
- nadzor nad problemi projekta (razlika med pričakovano in dejansko kakovostjo izhodnega rezultata projekta),
- nadzor nad spremembami projekta, ki so lahko posledica problemov ali odprtih vprašanj ali pa se pojavijo kot posledica sprememb v poslovanju oziroma poslovnih odločitvah projekta.

Odgovornost nosi vodja projekta, v kolikor pa odgovori posledično vplivajo na obseg ali cilje projekta, je potrebno vprašanja posredovati na višji projektni nivo vse do sponzorja projekta.

Nadzor nad viri projekta zajema:

- spremljanje porabljenega časa in uspešnosti članov projektne skupine (časovno opredeljena poročila članov projektne skupine o delu na projektu, skladnost z izvedbenim planom in razpoložljivostjo virov),
- spremljanje porabe finančnih virov projekta (stroški delovne sile, tehnične opreme, nabave, zunanjih virov, nepredvideni stroški),
- spremljanje porabe tehničnih virov (predvsem na zahodu, medtem ko se pri nas običajno ne spremlja uporaba in obraba na primer posameznih računalnikov za namen projekta, razen ko gre za nabavo dodatne opreme, kar pa se spremlja pod finančnimi viri).

Nadzor časovnega poteka projekta zajema:

- redno spremljanje poteka aktivnosti v primerjavi z izvedbenim načrtom projekta,
- dokumentiranje vseh sprememb preko odprtih vprašanj, problemov in zahtev za spremembo,
- izvajanje rednih sestankov z vodji aktivnosti z namenom spremljanja in planiranja aktivnosti in obravnavanja odprtih vprašanj, problemov in zahtev za spremembo.

Nadzor nad konfiguracijo oziroma izhodnimi rezultati projekta mora celoten življenjski cikel projekta zajemati (ISO 12207):

- nadzor nad projektno dokumentacijo (določitev projektne knjižnice, oblike in postopka verzioniranja dokumentacije ter nadzora nad spremembami dokumentacije),
- nadzor nad konfiguracijo programskega paketa (identifikacija in definiranje osnovnih gradnikov sistema, nadzor nad modifikacijami in verzijami gradnikov, beleženje in poročanje o statusu gradnikov in zahtev za spremembe gradnikov, zagotavljanje popolnosti, konsistentnosti in pravilnosti gradnikov v celotnem sistemu, spremljanje sestave in verzioniranja sistemov (releases), nadzor nad hranjenjem, upravljanjem in izdelavo sistemov).

2.3 Upravljanje s tveganji

Tveganje predstavlja obstoj možnosti, da se v času projekta pojavi neželjeno stanje, ki lahko privede do zmanjšanja funkcionalnosti izdelka, povečanja virov projekta ali celo do nedokončanja projekta. Velikost in kompleksnost projekta in njegov vpliv na poslovanje organizacije že sama po sebi določata tveganost projekta. S prepoznavanjem in ustreznim upravljanjem s tveganji lahko precej povečamo verjetnost uspešnosti projekta.

Upravljanje s tveganji pomeni definiranje jasnih napotkov, kako tveganja beležiti in jih obravnavati. Proaktivno upravljanje s tveganji pomeni, da se že na začetku prepozna možna tveganja na projektu, da se tveganja dokumentirajo, da se opredeli njihova stopnja tveganosti ter da se izdela načrt, kako zmanjšati verjetnost pojava neželene situacije oziroma v primeru, če do tveganja vendarle pride, kako obravnavati njegove posledice. Pomembno je, da so s postopki upravljanja s tveganji seznanjeni vsi člani projektne skupine, saj običajno ravno izvajalci prvi opazijo, da se določeno tveganje lahko pojavi.

Uresničevanje upravljanja s tveganji na projektu sestoji iz štirih poglavitnih aktivnosti (McConnell, 1998, str. 95, Oracle: Project Management Process and Task Reference, 1996, str. 1-1 do 1-46):

- aktivnosti priprave projekta je potrebno voditi v smislu zmanjšanja verjetnosti, da projekt preide v neželjeno stanje,
- v fazi planiranja projekta je potrebno pisno definirati postopke za upravljanje s tveganji za primer, če projekt preide v neželjeno stanje,

- pri planiranju virov projekta je potrebno upoštevati tudi porabo virov na aktivnostih upravljanja s tveganji,
- v kolikor se v času projekta tveganje pojavi, je potrebno njegov vpliv skrbno spremljati in skrbeti, da je pod nadzorom oziroma v nasprotnem primeru njegov vpliv na projekt vključiti v izvedbeni načrt projekta.

Pomembno je, da se že v planu projekta implementirajo delovni postopki za obvladovanje tveganj na projektu (nadzor projektnega vodenja), kot so upravljanje z odprtimi vprašanji, postopek reševanja zahtev za spremembo ter postopki komunikacij in urejanja dokumentacije na projektu.

V življenjskem ciklu projekta je na vseh nivojih projektne organizacije potrebno upoštevati, da se ob nepredvideni situaciji uporabi v planu predviden delovni postopek in da se izbrani izhod iz neželene situacije ustrezno dokumentira.

Poseben problem pri upravljanju s tveganji predstavlja situacija, ko v fazi planiranja projekta tveganja prepoznamo, ne moremo pa jih zmanjšati ali odstraniti. V takšnih primerih je skoraj gotovo, da projekt ne bo uspešno zaključen. Takšnih projektov se je potrebno izogibati in na podlagi analize poiskati ustrežnejšo rešitev problema.

V nadaljevanju povzemam nekaj možnih, najpogostejših tveganj na projektih in predlog obvladovanja posameznih tveganj (Sturm,1999, str.148-155, McConnel, 1998, str.98-99):

- **Slabo definiran izvedbeni načrt projekta:** Plan projekta je potrebno definirati na podlagi vsebine projekta oziroma najti konsenz med obsegom projekta in strategijo podjetja. Najpogosteje se zgodi, da časovni okvir projekta določa sponzor projekta na podlagi strategije podjetja, zaradi česar obstaja velika verjetnost, da projekt ne bo zaključen v predvidenem časovnem roku.
- **Nenadzorovano naraščanje obsega projekta:** Meje projekta morajo biti natančno definirane, vsem članom projektne skupine mora biti jasno, kaj projekt zajema in kaj ne. V nasprotnem primeru lahko pride do nenadzorovanega naraščanja obsega projekta. Povečevanje obsega projekta je lahko tudi posledica ne vključenosti uporabnikov novega informacijskega sistema v pripravo projekta, določitev ciljev projekta ali pripravo projektne specifikacije. S prototipiranjem in ustreznim delovnim postopkom nadzora nad spremembami lahko zvišamo kakovost specifikacij novega sistema in povečamo informiranost in vključenost uporabnikov v pripravo projekta.
- **Nezadostnost virov:** Viri (človeški in finančni viri, čas, tehnična oprema, prostori,...) morajo biti nadzorovani. V nasprotnem primeru lahko pride do spornih situacij, kot so neenakomerna razporeditev dela med člani projektne skupine, preobremenitve posameznih članov projektne skupine, zviševanje projektne proračuna, prekoračenje časovnih rokov in drugo. Za nemoten potek projekta je potrebno glede na njegov plan že na začetku zagotoviti zadostne vire. Zaradi

premajhnega števila članov projektne skupine se lahko tudi zgodi, da ena oseba igra več projektnih vlog, ki so med seboj nezdržljive (na primer oseba je hkrati razvijalec nove programske opreme in član testne skupine). Zagotoviti je potrebno zadosti časa in poskrbeti, da, v kolikor ena oseba igra več vlog na projektu, te niso medsebojno nezdržljive. Prav tako lahko pride do nepredvidenega povečanja proračuna. V tem primeru se morda lahko odpovemo nekaterim, še neizdelanim in manj pomembnim funkcionalnostim.

- **Neizkušenost vodje projekta:** Neizkušen ali slab vodja projekta ne obvladuje izvajanja projekta, kar posledično vodi v kaotičen potek projekta in nekontrolirano porabo virov. Potrebno je pridobiti izkušenega managerja za vodenje projekta (morda tudi z najemom svetovalca za vodenje projektov) oziroma mu omogočiti dodatno izobraževanje v zvezi z vodenjem projektov.
- **Nezadostno poznavanje tehnologije:** Izkušnje projektne skupine pri uporabi izbranega orodja za delo na projektu so lahko slabe, zaradi česar tudi lahko pride do zamude na projektu. Zato je potrebno že v času projekta planirati dodatno izobraževanje članov projektne skupine in s tem v zvezi tudi dodatne stroške. Nepoznavanje izbrane tehnologije lahko privede tudi do nestabilnosti novega produkta. Zato je potrebno predvideti izvedbo projekta s tehnologijo, s katero imajo tako razvijalci kot uporabniki zadosti izkušenj. Upoštevati je tudi potrebno, da je za stabilnost projekta pomembno, da se ne uvedeta več kot eno ali največ dve novi orodji.
- **Nerealna pričakovanja uporabnikov:** Pričakovanja končnih uporabnikov so dostikrat nerealna, ko pričakujejo popolno funkcionalnost v kratkem časovnem obdobju. Uporabnikom je potrebno dodatno predstaviti projektno delo, razbiti projekt na manjše sklope, ki jih uporabnik postopoma prevzema in mu tako povečati zaupanje v nov informacijski sistem.
- **Nerazumevanje ciljev projekta:** Cilji projekta morajo biti sprejeti tako s strani članov projektne skupine kot uporabnikov. V kolikor člani projektne skupine ne razumejo skupnih ciljev projekta, se ne strinjajo z njimi ali celo ne vidijo svoje vloge na projektu, je lahko njihov prispevek k projektu nekvaliteten.
- **Nepoznavanje delovnih postopkov:** Delovni postopki (na primer komunikacije na projektu) naj bodo enaki, kot je standard v podjetju. V nasprotnem primeru (ali v primeru, da jih med projektom spreminjamo) se lahko zgodi, da jih uporabniki ne bodo uporabljali, da bodo imeli težave z uporabo ali jih ne bodo uporabljali v celoti. V takšnem primeru je potrebno članom projektne skupine nuditi izobraževanje in dodatno podporo k uporabi novih delovnih postopkov.
- **Časovna oddaljenost zaključka projekta:** V kolikor gre za enega večjih projektov, kjer je rezultate pričakovati šele po letu ali več dela, lahko zaradi dolžine

projekta pride do znižanja delovne vneme in demotivacije projektne skupine. Zato je potrebno projekt zastaviti po etapah, da so delni cilji in rezultati čim prej vidni.

- **Nedefiniranje kontrolnih točk:** Kontrolne točke projekta je potrebno definirati že v planu projekta, in sicer tako časovno kot vsebinsko ter z jasnimi nalogami in odgovornostmi. Pomanjkanje kontrolnih točk lahko pripelje v nepravočasno korigiranje projektne dela, posledica česar so lahko zamuda, prekoračitev proračuna ipd. Kasneje, ko je napaka odkrita, zahtevnejša in dražja je njena odprava.
- **Spremembe strategije podjetja oziroma ciljev projekta:** To v času projekta pomeni redefiniranje in preplaniranje projekta, posledica pa so lahko preseženi viri projekta in nenadzorovana rast obsega projekta. Zato je potrebno na začetku zagotoviti usklajenost ciljev projekta z dolgoročno strategijo projekta, pridobiti popolno podporo sponzorja projekta za delo na projektu in zagotoviti dober nadzor nad spremembami na projektu, ki jih mora odobriti ustrezno najvišje vodstvo organizacije (sponzor).
- **Nezadostna in prepozno vpeljana strategija testiranja:** Tehnologija testiranja mora biti zasnovana tako, da omogoča vzporedno izvajanje z razvojem novega sistema, zajemati mora dober nadzor nad potekom razvoja novega sistema ter zagotoviti zadosten čas za testiranje in odpravljanje napak. Izvajanje testiranja mora potekati z neodvisno testno skupino, v katero so vključeni predvsem bodoči uporabniki. V kolikor se testiranje izvaja šele, ko je sistem narejen, je cena odprave napak zelo visoka. V kolikor so v testiranje vključeni razvijalci, ne gre pričakovati realnih rezultatov testiranja, zaradi prekoračitve časa, potrebnega za razvoj sistema, pa lahko tudi zmanjka čas za testiranje sistema.
- **Nezadostna projektna dokumentacija:** Projektna dokumentacija mora biti sproti vodena, arhivirana in nadzorovana. V kolikor ni projektne dokumentacije, ni dokazov, kaj je že zaključeno, kakšne spremembe so bile narejene, katere napake so bile odkrite in odpravljene, kateri standardi se uporabljajo, kakšen je dogovor o delu na projektu ipd.
- **Nedefiniranje in neizvajanje postopkov nadzora nad spremembami:** Nadzor nad spremembami na projektu je potrebno pripraviti že v planu projekta in ga ves življenjski cikel projekta tudi uporabljati. V kolikor se na projektu ne izvaja ustrezen nadzor sprememb, ni nikjer dokazov, katere spremembe in s kakšnim namenom so bile na projektu izvedene.
- **Neuporaba standardnih knjižnic in pouporabljenih programskih komponent** večata čas in stroške projekta kakor tudi tveganje, da bo izdelek manj zanesljiv.
- **Slaba kakovost projekta:** Postavitev standardov in metodologij omogoča nadzor nad napredovanjem in oceno kakovosti projekta.

2.4 Kakovost projekta

Zagotavljanje kakovosti pri projektih razvoja novih informacijskih sistemov je planiran, sistematičen proces, ki zagotavlja, da so informacijski sistemi in procesi njihovega razvoja v skladu s sprejetimi standardi, načinom dela in postopki (Ruthberg, 1991, str.B-2). Na zunaj se izraža kakovost projekta v zanesljivosti novega informacijskega sistema in v zadovoljstvu uporabnikov pri njegovi uporabi.

Da bi lahko ocenjevali kakovost, je potrebno predhodno pripraviti merila za kakovost. Te obsegajo (Warren, 1998, str. B4-28):

- formulacijo sistemskih in programskih standardov,
- pregled dokumentacije načrtovanja informacijskih sistemov z namenom, da se ugotovi njihova skladnost s prej sprejetimi standardi in za zagotovitev, da bo imel novi sistem tudi nadzorne funkcije,
- pregled testiranja posameznih programskih komponent in testiranja celotnega sistema, nadzor nad paralelnim ali poskusnim izvajanjem novega informacijskega sistema z namenom zagotovitve skladnosti s standardi,
- potrditev, da se pri kodiranju in izdelavi sistema uporabljajo standardi,
- zagotovitev, da novi sistem vsebuje ustrezne kontrole, postopke varnosti in zaščite in revizijske sledi.

Nadzor kakovosti na projektu naj vrši notranji revizor, zunanji revizor, neodvisna skupina za zagotavljanje kakovosti ali v katerikoli kombinaciji naštetih.

2.5 Merilo uspešnosti projekta

Projekt je uspešno zaključen, če

- je zaključen v planiranem roku,
- ni presegel planiranega proračuna,
- ima zahtevano funkcionalnost,
- je izveden uspešen prevzem novega informacijskega sistema s strani končnih uporabnikov,
- izpolnjuje poslovne zahteve organizacije in
- zagotavlja ustrezne poslovne kontrole.

Vendar pa tako stroge kriterije dosega le malo število projektov. Glede na uspešnost projektov (po poročilu Standish Group(1998) je od skupno 1200 projektov samo 26% projektov uspešnih, to pomeni pravočasno zaključenih, brez prekoračitve proračuna in s funkcionalnostjo v skladu z naročnikovimi specifikacijami) večina projektov preseže vsaj katerega od zgoraj omenjenih faktorjev, pa se kljub temu vodijo kot zaključeni projekti. Tudi definiranje zaključka projekta v smislu, da so uspešno zaključene vse planirane

aktivnosti, ne vzdrži, saj dostikrat ostajajo vprašanja še odprta, niso odpravljene vse, največkrat lepotne, pomanjkljivosti in napake, in podobno. Zato je smiselno že na začetku projekta opredeliti faktorje, na podlagi katerih se bomo odločali, ali je projekt formalno zaključen in lahko preidemo na fazo delovanja novega informacijskega sistema.

Priporočljivo je, da se že v planu projekta opredeli, na podlagi česa se bo ocenjevala primernost zaključka projekta, in sicer vsaj njihov minimalni nabor, kot je navedeno v nadaljevanju:

- določitev nalog projekta, ki morajo za proglasitev zaključka projekta biti dejansko izvedene, zaključene, in za katere naročnik potrdi ustreznost,
- uspešno opravljen prevzemni test novega informacijskega sistema,
- izvedene vse kontrolne točke in potrjeni vsi tam sprejeti sklepi,
- doseženi vsi temeljni cilji, definirani v začetnem planu projekta,
- uspešno opravljene vse odkrite napake sistema, ki vodijo v zrušitev sistema ali v napačno funkcionalnost sistema (dopuščajo se kvečjemu lepotne napake in nekatere manjše težave, za katere pa mora biti dogovorjen način odprave v času produkcije).

3 Metodologija razvoja informacijskih sistemov

3.1 Uvod

Metodologija je množica najboljših napotkov, primerov in drugih informacij, zbranih z namenom, da se izkušnje enih predajo tudi drugim. Je formalizacija najboljših izkušenj, ki so v določenem trenutku dosegljive. V primeru metodologij za razvoj novih informacijskih sistemov je potrebno upoštevati, da se metodologije spreminjajo v skladu s hitrostjo razvoja in uporabe novih tehnologij, kar pomeni, da metodologija ne more obstajati neodvisno od tehnologije. Metodologija mora zato biti razvita in posodobljena v skladu z najboljšimi izkušnjami v določeni tehnološki veji (Clegg, 1994, str. 1). Kakovostno metodologijo razvoja informacijskega sistema sestavlja niz korakov, za katere so definirana orodja in navodila, kako jih na projektne delu uporabiti, druga pomembna zadeva pa je porazdelitev odgovornosti in vlog med glavnimi udeleženci procesa (Galliers, 1999, str. 69).

Metodologija strukturiranega razvoja novih informacijskih sistemov njenemu uporabniku omogoča fazen pristop k reševanju problema, z definiranimi izhodnimi rezultati posameznih faz, z določenimi nalogami, ki morajo biti izvedene z namenom izdelave izhodnih rezultatov in z dodatnimi tehnikami za merjenje uspeha procesa in kakovosti izhodnih rezultatov. Dobra metodologija omogoča projektne vodji določiti potrebne izhodne rezultate in oceniti tako čas kot potrebne vire projekta, potrebno tehnično in programsko opremo, vse to z namenom upravljanja celotnega procesa v smislu zniževanja tveganj za izvedbo procesa.

Eden glavnih namenov uporabe metodologije je zmanjšanje tveganja uspešnosti celotnega procesa, za katerega je metodologija uporabljena, v smislu predvidljivosti, ponovljivosti in izobraževanja njenih uporabnikov.

Dobra metodologija razvoja novega informacijskega sistema mora (Warren, 1998, str. B3-4):

- imeti strukturiran koncept,
- omogočati njenim uporabnikom enostavno uporabo,
- omogočati učinkovito vodenje projekta,

zato je pri izbiri ustrezne metodologije potrebno biti pozoren, da:

- deli življenjski cikel projekta v faze,
- ima natančno definirane postopke, zaporedje izvajanja postopkov in definirane izhodne rezultate postopkov,
- omogoča razvid statusa projekta v vsakem trenutku,
- vsebuje tehnike, ki izboljšujejo kakovost sistema in ima opredeljene postopke potrjevanja ustreznosti,
- vsebuje postopke planiranja in projektnega vodenja,
- ima opredeljene vloge in odgovornosti vseh sodelujočih na projektu,
- določa zahteve za dokumentacijo in njeno standardizacijo,
- vključuje postopke za zagotavljanje varnosti, nadzora in revizorske sledi v novem informacijskem sistemu in
- obstaja orodje, ki podpira metodologijo in zvišuje produktivnost programerja.

Metodologija naj ima ločeno opredeljene:

- izhodne rezultate posameznih aktivnosti (torej kaj mora biti narejeno),
- postopke, kako izdelati izhodne rezultate (kako mora biti narejeno),
- standarde izvajanja postopkov in izhodnih rezultatov.

Obstaja več vrst metodologij, ki jih različne skupine uporabljajo glede na svoje potrebe in zahteve:

- metodologije kot gotovi izdelki, ki so podprti z ustreznimi orodji in informacijsko tehnologijo,
- interne metodologije, razvite v posameznih organizacijah in
- kombinacija obeh.

Nobena metodologija, uporabljena pri izgradnji ali prenovi informacijskega sistema, sama po sebi ne zagotavlja uspeha, lahko pa s svojimi priporočili, opisi posameznih korakov, ciljev in tveganj zviša kakovost končnega rezultata ter zmanjša tveganja.

3.2 Pomen metodologije pri neizkušenih projektnih skupinah

Metodologijo sestavljajo aktivnosti, koraki ter napotki za izvedbo aktivnosti, definicije končnih izdelkov posameznih aktivnosti ter seznam možnih tveganj. Vse to omogoča projektni skupini kvalitetnejšo sliko o bodočih aktivnostih in njihovi soudeležbi na projektu.

Najpomembnejši del projekta se izvrši v prvi fazi, to je pri planiranju projekta. V kolikor projektna skupina ni izkušena, je na začetku zelo težko kvalitetno opredeliti ves projekt, določiti naloge projekta, planirati ustrezne vire projekta, opredeliti kakovost projekta ipd. V takšnih primerih je lahko ocena projekta posledica nerealnih pričakovanj, manjkajo aktivnosti na projektu, nedefinirani so postopki upravljanja projekta, nejasni so cilji projekta, ipd. Z uporabo metodologije na projektnem delu dosežemo:

- skrajšanje časov planiranja projekta,
- višjo stopnjo nadzora obsega in virov na projektu,
- kakovostnejše planiranje projekta,
- znižanje stroškov projekta,
- zagotavljanje rokov izvedbe,
- zmanjšanje možnosti izvajanja neplaniranih aktivnosti,
- zmanjšanje možnosti nenadzorovanega širjenja obsega projekta,
- zvišanje stopnje zaupanja v uspešen zaključek projekta,
- poenotenje pogleda na cilje projekta med člani projektne skupine,
- izboljšanje uporabe standardov in dogovorjenih postopkov na projektu,
- določitev vloge in odgovornosti članov projektne skupine na projektu,
- izboljšanje tehnologije testiranja na projektu.

Pri odločitvi za določeno metodologijo se je potrebno zavedati, da ni potrebno metodologije prevzeti že na začetku v celoti, prav tako tudi ni potrebno, da se jo striktno držimo. Pomembno je, da v določenem trenutku izberemo najboljšo pot, ki jo poznamo, pri čemer nam metodologija poveča množico ustreznih poti.

3.3 ISO priporočila za kakovost razvoja IS

Projekt razvoja informacijskega sistema je skupina aktivnosti, ki ima svoj začetek in konec, postopek pa je (seveda vsakič z drugo vsebino) ponovljiv. Z uporabo določene metodologije pridobimo smernice oziroma priporočila za izvedbo projekta. Vendar nam

dostikrat sama metodologija ne zadošča ali pa želimo določene aktivnosti še dodatno nadgraditi v smislu kakovosti procesov in izdelkov. Nasprotno tudi metodologije razvoja informacijskih sistemov slonijo na standardih in priporočilih. Zato sem v nadaljevanju povzela nekatere standarde, katerih uporabo v smislu zagotavljanja ustrezne kakovosti tako procesov razvoja kot tudi končnih izdelkov priporočam pri razvoju novih informacijskih sistemov.

3.3.1 Standard za vodenje in zagotavljanje kakovosti (ISO 9000-3)

Serijski standardi ISO 9000 se nanašajo na sisteme kakovosti. V njih so opisani modeli zagotavljanja kakovosti. Standard ISO 9000-3 (1997) je izpeljava standarda ISO 9001 za področje programske opreme. Navaja smernice pri razvoju, dobavi, instalaciji in vzdrževanju programske opreme. V tem standardu so navedene zahteve za sistem kakovosti, ki so namenjene zadovoljstvu uporabnikov in se nanašajo predvsem na dobavitelja programske opreme:

- odgovornost vodstva za zagotovitev kakovosti,
- način postavitve sistema kakovosti v pripravi projekta v smislu postopkov za zagotavljanje kakovosti,
- obvladovanje razvoja, ki zajema planiranje in dokumentiranje postopkov nadzora in verifikacije načrtovanja produkta v smislu, da bo produkt ustrezal specifikacijam, reševanje nejasnosti specifikacij, dokumentiranje rezultatov načrtovanja v smislu preverjanja ustreznosti z zahtevanimi vhodnimi specifikacijami projekta in možnost pregledovanja, verifikacije in validacije načrtov aplikacije ter spremljanje in dokumentiranje vseh sprememb pri načrtovanju,
- obvladovanje dokumentov in podatkov, to je upravljanje s projektno dokumentacijo in skrb, da so vse aktivnosti na projektu dokumentirane,
- priprava plana testiranja: določitev dokumentiranih postopkov za nadzor in testiranje, opis postopkov, ki morajo biti zajeti v planu kakovosti, pravila za prenos izdelka v produkcijo,
- metrika testiranja in kvalificiranje odkritih nepravilnosti, vzdrževanje statusov nepravilnosti, zagotovitev postopkov, ki onemogočajo inštalacijo produktov z nepravilnostmi v produkcijo,
- nadzor nad neustreznimi proizvodi (plan nadzora nad konfiguracijo gradnikov in celotnega produkta),
- odgovornosti in naloge priprave in izvedbe izobraževanja,
- vzdrževanje produkta.

3.3.2 Informacijska tehnologija: Življenjski cikel programske opreme (ISO 12207)

Standard ISO 12207 (1995) opredeljuje postopke življenjskega cikla nekega informacijskega sistema oziroma programskega paketa, ki ga deli na naslednje procese življenjskega cikla programskega produkta:

- primarne procese (raziskovanje možnosti za novo programsko opremo, dobavo oziroma razvoj nove programske opreme, prehod v produkcijo in vzdrževanje),
- podporne procese (dokumentacijski postopki, postopki upravljanja s konfiguracijo, postopki zagotavljanja kakovosti, postopki verifikacije in validacije, postopki revizije in postopki reševanja problemov v zvezi s produktom) ter
- organizacijske procese (upravljalški in infrastrukturni procesi, nadgrajevanje procesov življenjskega cikla projekta ter proces izobraževanja).

3.3.3 Informacijska tehnologija Zahteve za kakovost in testiranja (ISO 12119)

Standard ISO 12119 (1994) se nanaša na programske pakete. Naročnik oziroma kupec informacijskega sistema lahko uporabi standard pri preverjanju ustreznosti informacijskega sistema oziroma njegovega programskega dela s svojimi specifikacijami.

Standard ISO 12119 vsebuje dve poglavji in sicer:

- zahteve za programski paket (nanaša se na zahteve po kakovosti), ki zajemajo opis produkta, uporabniško dokumentacijo, uporabnost, zanesljivost in funkcionalnost programske opreme in podatkov ter
- navodila, kako se lotiti testiranja glede na zahteve za kakovost, ki zajemajo definicije predpogojev za testiranje, definiranje potrebnih testnih aktivnosti, način beleženja testnih aktivnosti, način poročanja o testiranju ter možnosti uporabe testnih postopkov in podatkov pri ponovnem testiranju.

3.3.4 Vrednotenje programske opreme (ISO 9126)

Glede na to, da sodoben razvoj omogoča pospešeno rast programskih aplikacij, je potreben tudi čedalje večji nadzor nad kakovostjo le-teh. Pri zagotavljanju kakovosti programske opreme se je pokazala potreba po posebni tehnologiji za definiranje in vrednotenje kakovosti tako končnega programskega izdelka kot tudi celotnega razvojnega procesa. S tem namenom je bil pripravljen standard ISO 9126 (1991), ki opredeljuje šest karakteristik, ki določajo kakovost programskega izdelka, ter v nadaljevanju opisuje priporočila za uporabo karakteristik za določitev kakovosti vrednotenja programske opreme. Te karakteristike so funkcionalnost, zanesljivost, uporabnost, učinkovitost, možnost vzdrževanja izdelka ter prenosljivost izdelka med različnimi okolji.

Pri razvoju novih informacijskih sistemov nam omenjeni standard lahko služi kot pomoč pri definiranju kakovostnih specifikacij kot tudi v zaključnih fazah (na primer pri testiranju).

3.4 Pregled in ovrednotenje nekaterih metodologij

Pri projektih informacijske tehnologije obstajajo metodologije:

- vodenja projekta,
- razvoja informacijskega sistema,
- kombinacija obeh.

Metodologije so lahko univerzalne, neodvisne od izbire tehnologije, ali pa so razvite z namenom uporabe na točno določenih orodjih.

Kot primer metodologije bom opisala osnovne lastnosti metodologije organizacije procesov informacijske tehnologije v smislu odgovora poslovnim potrebam, Control Objectives for Information and Related Technology (COBIT), Microsoft metodologijo Microsoft Solution Framework (MSF), CHESTRA metodologijo podjetja Siemens Business Services ter metodologijo Rational Unified Process (RUP). Za COBIT metodologijo sem se odločila predvsem zato, ker je strategija razvoja informacijskih procesov (katere aktivnost je lahko tudi projekt izgradnje informacijskega sistema) povezana s poslovno strategijo organizacije, COBIT metodologija pa obe področji povezuje. MSF metodologijo sem izbrala kot antipod Oracle metodologiji tako zaradi različne organizacije aktivnosti razvoja novega informacijskega sistema kot njune konkurenčnosti na trgu. CHESTRA metodologijo predstavljam kot metodologijo, ki je primerna predvsem za razvoj novih informacijskih sistemov, ki temeljijo na prenovi poslovnih ciljev in procesov. Zadnja, RUP metodologija je izbrana zaradi iterativnega pristopa k izvajanju posameznih faz, kar predstavlja kombinacijo CDM in MSF metodologije, njena uporaba ne temelji na določeni platformi ali izbiri izvedbenih orodij, poleg tega pa podjetje Rational ponuja tudi ustrezna integrirana orodja za upravljanje projekta, kar je bistvena pomanjkljivost preostalih, prej omenjenih metodologij.

3.4.1 COBIT

COBIT (IT Governance Institute, 2000) je metodologija, namenjena organiziranju procesov informacijske tehnologije, razvita v IT Governance Institute.

Ogrodje metodologije predstavljajo podatki in informacije, procesi informacijske tehnologije pa morajo z uporabo virov informacijske tehnologije (ljudje, aplikacijski sistemi, tehnologija, oprema in podatki) omogočiti, da so informacije v pravem trenutku na voljo poslovnemu procesu in da ustrezajo kriterijem kakovosti, zanesljivosti in zaščite. COBIT metodologijo sestavljajo trije nivoji upravljanja informacijske tehnologije, in sicer:

- najnižji nivo, ki ga sestavljajo posamezne aktivnosti in naloge,

- srednji nivo, kjer so sorodne aktivnosti in naloge združene v procese,
- najvišji nivo, kjer so procesi združeni v področja.

COBIT metodologija ločuje štiri področja:

- planiranje in organizacija,
- pridobivanje informacij in implementacija,
- predaja in podpora,
- spremljanje,

ki jih sestavlja 34 procesov. Za vsak proces metodologija predlaga korake izvedbe, opiše kritične faktorje uspeha posameznega procesa, vsebuje opis končnega rezultata, prilaga seznam in vloge človeških virov na procesu ter vprašalnik za oceno zrelosti posameznega procesa.

COBIT metodologija je uporabna za organiziranje IT oddelkov v smislu podpore poslovnim procesom. Ker lahko tudi projekt prenove informacijskega sistema opredelimo kot del strategije informacijskega razvoja (projekt je podmnožica oziroma del aktivnosti procesov informacijske tehnologije), lahko metodologijo učinkovito uporabimo tudi na projektih. V primeru, da imamo metodologijo že vpeljano, lahko njene procese na projektih enostavno pouporabimo, s čimer pridobimo čas pri planiranju projekta, saj so delovni procesi že utečeni in verificirani. Poleg tega metodologija omogoča v vsakem trenutku ocenitev kakovosti rezultata aktivnosti, kar pripomore h kakovostnejši izdelavi posameznega izdelka ter hitrejšemu odpravljanju napak in pomanjkljivosti. Poleg tega metodologija temelji tudi na tehničnih standardih (ISO, EDIFACT) in profesionalnih standardih za interni nadzor in revizijo.

3.4.2 MSF

MSF (Wilson, 1999) je projektni pristop oziroma metodološki okvir za izvajanje projektov, ki ga je razvilo podjetje Microsoft.

Po MSF metodologiji je prva aktivnost na projektu določitev članov projektne skupine in dodelitev vlog. MSF metodologija loči 6 enakovrednih vlog, in sicer produktna, programska, razvojna, uporabniška, testna in logistična, k čemur dodaja še vlogo vodje projekta.

Nato se začne razvoj, ki je implementiran skozi štiri faze:

- definiranje zahtev,
- načrtovanje,
- izdelava,
- stabilizacija.

Konec vsake faze označuje izdelek, ki je lahko dokument, program, načrt ali podobno. Glavna prednost MSF metodologije je delitev aplikativne rešitve na dele, ki jih ločimo z različnimi verzijami aplikacij.

V projektu je potrebno določiti točen dan predaje aplikacije v produkcijo. S tem se oblikuje jasen cilj, ki deluje motivacijsko, prav tako pa omogoča določitev prioritet in boljše načrtovanje projekta. Če je projekt preobsežen, ga je potrebno razbiti na manjše, obvladljive dele. V tem primeru uporabimo različne verzije znotraj projekta. Pri načrtovanju projekta je potrebno določiti interval (ne večji od enega tedna), v katerem moramo imeti delujočo aplikacijo, to je novo verzijo izdelka (seveda s funkcionalnostjo, ki jo je do takrat mogoče narediti). S tem zagotovimo sprotno testiranje in hitro odpravljanje arhitekturnih in programskih napak in zagotovimo boljšo kvaliteto izdelka.

Prednosti uporabe MSF metodologije na projektne delu so predvsem v:

- opredelitvi kratkoročnih ciljev,
- višji motivaciji projektne skupine, saj so rezultati vidni v zelo kratkem času, poleg tega pa je tedensko razviden tudi napredek dela na projektu,
- boljšem načrtovanju,
- hitrejšem zaznavanju napak in zato nižjih stroškov odprave napak.

Pomanjkljivosti metodologije so predvsem v tem, da so vse vloge, vključno z vodjem projekta, enakovredne (problem odločitev v kritičnih situacijah) ter da nima definiranih izdelkov za posamezne aktivnosti.

3.4.3 CHESTRA

CHESTRA (Siemens Business Services, 2002) je metodološki okvir za razvoj integriranih poslovnih rešitev, razvit v podjetju Siemens Business Services. CHESTRA metodološki okvir je primeren za pripravo, načrtovanje, izvedbo in prenos v produkcijo informacijskih sistemov, ki temeljijo na spremembah poslovnih procesov predvsem v velikih organizacijah. Nakazuje ustrezno pot za doseg zastavljenih ciljev pri izvajanju aktivnosti, kot so razvoj poslovne strategije in vizije organizacije, prenova poslovnih procesov, načrtovanje arhitekture informacijske tehnologije, vpeljava novega informacijskega sistema in drugo.

CHESTRA daje vpogled na organizacijo s šestih različnih vidikov, ki jih imenuje področja spremembe. Ta področja sprememb so proces, organizacija, lokacija, aplikacija, tehnologija in podatki. Za vsako področje predvideva izdelavo enega ali več modelov, ki se razgrajujejo in dopolnjujejo skozi posamezne faze razvoja informacijskega sistema, ki se znotraj CHESTRA metodološkega okvira delijo na:

- fazo vizije in strategije, kjer se na podlagi obstoječih poslovnih nalog in ciljev, najboljših izkušenj in pričakovanj naročnika določi poslovne cilje in njihova prioriteta ter izdela osnutek bodočega sistema,

- fazo arhitekture, katere namen je na podlagi rezultatov predhodne faze izdelava načrtov glavnih procesov, konceptualnega entitetno relacijskega modela, izdelava arhitekture aplikacije in tehnične infrastrukture novega sistema, izdelava načrta implementacij (posameznih verzij) novega sistema glede na želene poslovne rezultate v določenem časovnem obdobju ter analiza ekonomske upravičenosti sistema,
- fazo razvoja, kjer se izdelajo posamezne komponente ene verzije, ki zajema celotno poslovno rešitev (ta faza se lahko izvaja na različne načine: pospešen razvoj, iterativen razvoj,...)
- fazo integracije, kjer se združijo komponente, izdelane v fazi razvoja, izvedejo testi in izvrši pilotna implementacija ter
- fazo prehoda na nov sistem, kjer se dejansko postavi nov informacijski sistem v produkcijsko okolje vključno z izvršitvijo vseh testov, predajo sistema v produkcijo ter vzpostavitev ustreznega okolja za vzdrževanje novega informacijskega sistema.

Za vsako fazo so v metodološkem okviru opisani:

- proces in aktivnosti, ki se v posamezni fazi mora izvesti,
- izdelki, ki so rezultati posamezne faze,
- vloge in odgovornosti sodelujočih v posamezni fazi,
- orodja in tehnike, ki se uporabljajo za zagotovitev kakovosti izdelkov in produktivnosti ekipe.

Glavne prednosti Chestra metodološkega okvira so:

- orientiranost na poslovne procese,
- poudarek na odnos med razvijalcem in uporabnikom,
- podrobna členitev poslovnih specifikacij šele v fazi načrtovanja, kar omogoča vključevanje v zasnovo sistema tudi naknadne spremembe poslovnih procesov.

3.4.4 Rational Unified Process

Rational Unified Process (RUP) (Rational Unified Process Overview, 2002) je inženirski proces razvoja programske opreme, ki pokriva celoten življenjski cikel programske opreme, razvit v podjetju Rational Software. RUP zagotavlja izdelavo kakovostnega programskega paketa v skladu z naročnikovimi zahtevami ter dokončanje v okviru opredeljenega časa in proračuna. Združuje napotke za različne moderne tehnike in pristope, kot so objektna tehnologija, komponentni razvoj, modeliranje in UML, iterativni razvoj in podobno. Upošteva najboljše izkušnje razvoja programske opreme, in sicer iterativni razvoj programske opreme, upravljanje z zahtevami, komponentno arhitekturo sistema, vizualno modeliranje programske opreme, nenehno preverjanje kakovosti

programske opreme in nadzor nad spremembami programske opreme. Celoten proces je podprt s številnimi orodji, prav tako razvitimi v Rational Software.

Metodologija temelji na iterativnem ponavljanju tradicionalnega modela razvoja informacijskih sistemov, katerega izhod vsake iteracije je delujoč programski paket.

RUP definira proces razvoja informacijskega sistema z dvema dimenzijama, in sicer fazno (začetek, obdelava, izvedba in tranzicija) in disciplinsko (poslovno modeliranje, zahteve, analiza in načrtovanje, izvedba, testiranje, razvoj, upravljanje s konfiguracijo in spremembami, upravljanje s projektom in okolje). Prva dimenzija predstavlja dinamičen pogled na razvoj programske opreme v smislu posameznih faz, iteracij in kontrolnih točk, medtem ko druga dimenzija predstavlja statični pogled, to je opis komponent posameznih disciplin, aktivnosti znotraj disciplin, delovnih tokov, izdelkov in vlog. Vsaka faza se deli na eno ali več iteracij. Na koncu vsake posamezne iteracije mora biti izdelan delujoč programski paket.

Za vsako aktivnost znotraj disciplin in faz so opisane vloge, izdelki, ustrezna orodja ter napotki za uporabo orodij (tool mentors), na voljo pa so tudi predloge in navodila za izdelavo izdelkov.

Prednosti RUP procesa pred ostalimi metodologijami so predvsem v iterativnem pristopu k izvajanju posameznih faz ter v podpori ustreznih orodij, kar pa je zaradi visoke cene posameznih orodij lahko tudi pomanjkljivost.

3.5 Oracle CDM/PJM metodologija

CDM Classic (Custom Development Method) v povezavi s PJM Advantage metodologijo (metodologija vodenja projektov informacijske tehnologije) je strukturirana metodologija razvoja novih informacijskih sistemov, aplicirana na Oracle tehnologijo. Je zbirka izkušenj Oracle-ovih strokovnjakov pri svetovanju na področju razvoja novih programskih rešitev s pomočjo njihove tehnologije. Temelji na tradicionalni metodologiji popolnega faznega razvoja informacijskega sistema in nas vodi skozi celoten življenjski cikel razvoja novega informacijskega sistema od planiranja (naloge, ki jih metodologija opredeljuje, so definirane za življenjski cikel projekta po zaključku študije izvedljivosti) do produkcije (samo prvi del, kjer so zajete prilagoditve in postopki vzdrževanja in plani za modifikacije in nadgradnje).

CDM metodologija temelji na uporabi CASE orodij, s čimer zagotavlja, da so poslovne potrebe ustrezno dokumentirane in ostanejo razvidne skozi celoten življenjski cikel projekta. Ključni koncepti CDM metodologije so:

- pregled življenjskega cikla projekta,
- tipični inženirski pristop k razvojnim nivojem (poslovni, logični in fizični nivo),
- verifikacija sistema preko različnih testov (testiranje modulov, integracije modulov, sistema, integracije sistemov in prevzemno testiranje),

Metodologija je razvita predvsem za srednje in velike projekte. Njena največja učinkovitost je v povezavi z Oracle sistemom za upravljanje s podatki in CASE orodjem Oracle Designer, s spremembo tehnike izvedbe pa jo je možno uporabiti na katerikoli drugi tehnologiji.

Celoten sistem CDM/PJM metodologije sestavlja:

- Oracle Project Management metodologija (PJM) za spremljanje življenjskega cikla projekta,
- Oracle Custom Development Method (CDM) Classic priporočila za planiranje in izvedbo vsebine projekta,
- priporočila in standardi za izdelavo kakovostnih aplikacij z uporabo Oracleve tehnologije, ki zajemajo standarde in priporočila za (Oracle Method: CDM Standards and Guidelines Library, 2000, Volume 1- 4):
 - modeliranje poslovnih zahtev s pomočjo orodja Oracle Designer (procesno modeliranje, modeliranje poslovnih pravil, modeliranje entitetnih relacij in prototipiranje),
 - načrtovanje in izgradnjo večnivojskih internetnih aplikacij (podatkovni nivo, poslovni nivo in prezentacijski nivo),
 - gradnjo sistema s pomočjo Oracle programskih orodij (PL/SQL) .

Za vsako nalogo metodologije priporočila vsebujejo:

- kratek opis posamezne naloge,
- opis rezultata naloge, to je izhodnega dokumenta ali izdelka,
- seznam pogojev za pričetek naloge (izdelani izhodni rezultati drugih nalog oziroma zaključene naloge),
- opis pristopov k nalogi in priporočila za tehniko izvedbe naloge,
- vloge in odgovornosti članov projektne skupine na posamezni nalogi,
- smernice izdelave izhodnega izdelka s predlogo izhodnega izdelka (v kolikor gre za dokument) oziroma navodila za izdelavo izdelka s pomočjo Oracle tehnologije,
- predloge distribucije in uporabe izhodnega rezultata naloge,
- kriterije za oceno kakovosti izdelka.

Prednosti uporabe CDM metodologije na projektu izgradnje poslovnega informacijskega sistema so:

- s svojim konceptom razvoja informacijskih sistemov, temelječim na tradicionalnem, faznem pristopu z natančno določenimi aktivnostmi, nalogami in rezultati ter medsebojne odvisnosti med posameznimi nalogami omogoča hitrejši zagon projekta,

- ves razvoj (načrtovanje aplikacije, kreiranje testnih skript in scenarijev, pisanje dokumentacije, planiranje in izvajanje izobraževanja) temelji na poslovnih modelih, definiranih v prvi fazi projekta,
- procesno modeliranje omogoča natančnejši nadzor nad obsegom projekta, boljše razumevanje poslovnih procesov, možnost povezave s projekti prenove poslovnih procesov, boljše razumevanje s končnim uporabnikom,
- prihranek časa, kajti ni potrebno razvijanje standardov in priporočil,
- vprašalniki pri nalogah za sprotno ocenjevanje kakovosti rezultata naloge ter uporaba orodij za avtomatsko preverjanje uporabe dogovorjenih standardov zvišujejo kakovost končnega izdelka, to je informacijskega sistema.

3.5.1 Procesi projektnega vodenja

Oracle PJM metodologija omogoča standarden pristop k upravljanju projektov informacijske tehnologije po zaključeni študiji izvedljivosti skozi celoten življenjski cikel projekta. Organizacija projektnega vodenja po PJM metodologiji je procesno orientirana.

Procesi projektnega vodenja po PJM metodologiji se nanašajo na planiranje, nadzor in zaključek nalog na dveh nivojih projekta, in sicer projektne in fazne. Ti procesi so:

- nadzor in poročanje,
- upravljanje z delom,
- upravljanje z viri,
- upravljanje s kakovostjo,
- upravljanje s konfiguracijo.

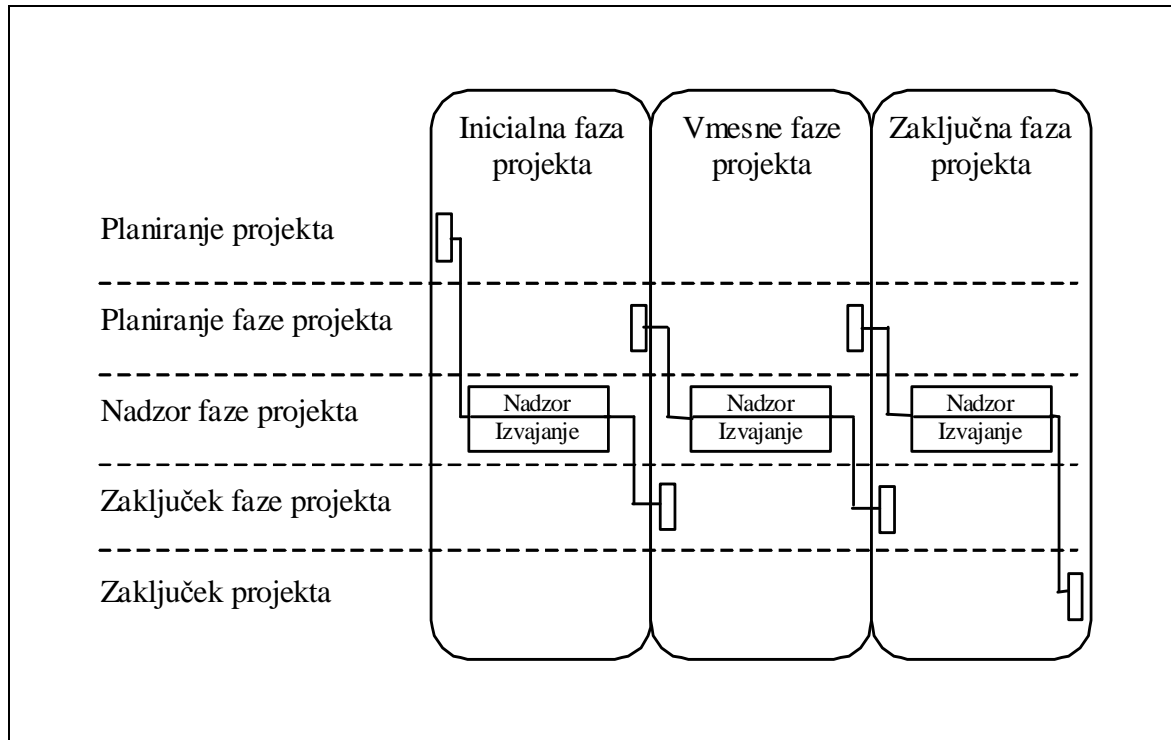
Vsebina procesov projektnega vodenja je podrobneje predstavljena v 4. poglavju.

Delitev nalog življenjskega cikla projekta v faze omogoča višjo stopnjo vodstvenega nadzora in zmanjšuje tveganja projektnega vodenja. Tako so v integraciji nalog projektnega vodenja z življenjskim ciklom projekta prepletene (Slika 2) :

- **naloge projektnega planiranja**, ki je enkratna aktivnost na začetku projekta in zajema naloge vseh petih procesov v smislu definiranja projekta in postopkov,
- **naloge planiranja posameznih faz projekta**, ki se izvede pred vsako posamezno fazo življenjskega cikla projekta in zajemajo naloge korekcij projektnega plana glede na rezultate predhodne faze tako v smislu vsebine kot nadzora projekta,
- **naloge nadzora izvajanja posameznih faz projekta** (na primer vrednotenje časovnega poteka projekta, redna tedenska poročila članov projektne skupine, redni kontrolni sestanki projekta,...), ki se vršijo ves čas izvajanja posamezne faze v smislu spremljanja napredovanja projekta, izhodnih rezultatov projekta in funkcij poročanja,

- **naloge zaključevanja posamezne faze projekta**, ki zajemajo naročnikov pregled in oceno izhodnih rezultatov posamezne faze,
- **naloge zaključka projekta**, ki je enkratna aktivnost, kjer se izdela zaključno poročilo o uspešnosti projekta in izvedejo naloge zaključevanja projekta.

Slika 2: Organizacija projektnega vodenja v življenjskem ciklu projekta



Vir: Oracle: Project Management Method Handbook, 1996, str 1-7

3.5.2 Življenjski cikel projekta

Oracle CDM metodologija se ukvarja z življenjskim ciklom projekta oziroma razvojem informacijskega sistema. Glede na obseg in vsebino projekta sicer ločimo tri metodologije, in sicer CDM Classic, Fast Track in Prototyping (v različici Oracle Method CDM 2.6 obstajata samo še Classic in Fast Track). V celoti bo predstavljena najboljšežnejša, uporabljena tudi na projektu izgradnje poslovnega informacijskega sistema v zavarovalnici Slovenica.

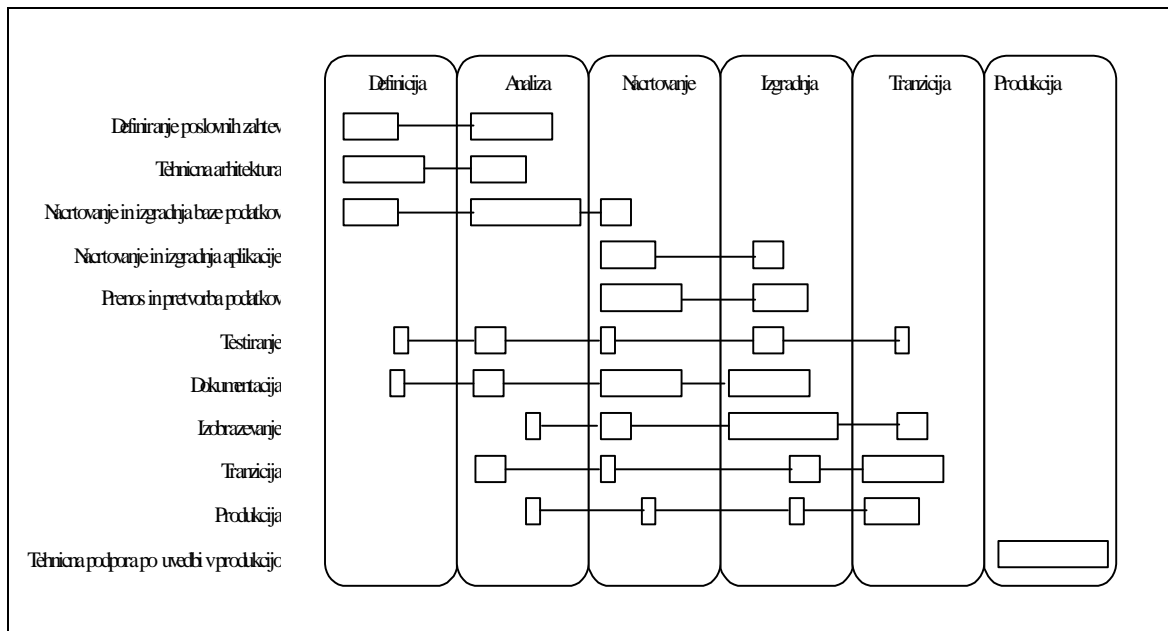
Oracle CDM metodologija temelji na tradicionalni, fazno orientirani metodologiji. Sestavlja jo šest faz, ki so:

- **definicija**, katere namen je v skladu s poslovnimi cilji grobo opredeliti poslovne in sistemske zahteve novega informacijskega sistema,
- **analiza**, katere namen je formuliranje podrobnih poslovnih in sistemskih zahtev ter izdelava poslovnih modelov procesov, funkcij in podatkov glede na opredeljene predhodne faze ter v skladu s tem definirati tehnično arhitekturo in strategijo prehoda na nov sistem,
- **načrtovanje**, kjer se poslovni modeli iz faze analize nadgradijo v sistemske modele in načrte novega informacijskega sistema z upoštevanjem zahtevane tehnične arhitekture in razpoložljive tehnologije,
- **implementacija**, ki zajema generiranje kode novega informacijskega sistema ter njeno testiranje, pripravo skript za prenos in pretvorbo podatkov, izdelavo dokumentacije in izobraževanje,
- **prehod na nov sistem**, kjer se izvede instalacija novega sistema, usposabljanje končnih uporabnikov in zažene nov sistem,
- **produkcija**, kjer poskrbimo za ustrezno podporo za vzdrževanje novega sistema, pregledamo in ovrednotimo trenutni sistem in planiramo nadaljnje funkcionalne nadgradnje sistema.

Že iz zgornjega opisa je razvidno, da se v posameznih fazah izvajajo različne naloge oziroma procesi. V skladu s tem je tudi po CDM metodologiji opredelitev aktivnosti v posameznih fazah opredeljena s procesi, ki jih sestavlja množica nalog, ki rezultirajo v točno določen objektni cilj. Iz slike 3 je razvidno, da se naloge določenega procesa pojavljajo samo v določenih fazah projekta. Takih skupin procesov je enajst, in sicer (tu so samo naštet, saj bodo podrobneje predstavljeni v 4. poglavju):

- definiranje poslovnih zahtev,
- pregled obstoječega sistema,
- opredelitev tehnične arhitekture sistema,
- načrtovanje in implementacija podatkovne zbirke,
- načrtovanja in izgradnja programske rešitve,
- prenos in pretvorba podatkov,
- izdelava dokumentacije,
- testiranje,
- usposabljanje,
- uvajanje novega sistema,
- tehnične podpore po uvedbi novega sistema v produkcijo.

Slika 3: Procesi CDM metodologije skozi faze



Vir: Oracle Method: CDM Classic Method Handbook, 2000, str.1-5

3.5.3 Implementacija metodologije s CASE orodjem Oracle Designer

CASE orodje Oracle Designer podpira celoten življenjski cikel projekta po CDM metodologiji. Omogoča tako izgradnjo poslovnih kot sistemskih in fizičnih modelov oziroma generiranje kode. Cilj modeliranja z orodjem Oracle Designer je na podlagi definicij poslovnih potreb čim hitreje izdelati fleksibilno, grafično odjemalec-strežnik aplikacijo z vso ustrezno tehnično dokumentacijo.

Orodje Oracle Designer lahko še nadgradimo z orodjem Headstart, ki ga sestavljata dve komponenti, in sicer:

- šablone za generiranje grafičnega uporabniškega vmesnika,
- orodje za preverjanje in uveljavljanje uporabe CDM standardov, s čimer se pospeši razvoj in zvišuje kakovost končnega izdelka.

4 Analiza uporabe CDM metodologije pri razvoju informacijskih sistemov v zavarovalnici

Slovenica: je ena sama metodologija dovolj?

V drugi polovici devetdesetih let je zavarovalnica Slovenica zelo povečala svoj obseg poslovanja. Da bi ostala uspešna in konkurenčna na trgu, se je morala temu prilagoditi. Sledila je prenova globalne strategije zavarovalnice in s tem povezana ugotovitev, da obstoječi informacijski sistem in njegov predviden razvoj ne sledita povečanju obsega poslovanja in ne ustrezata prenovljeni strategiji. Zato je prišlo do zasnove koncepta novega informacijskega sistema, ki naj bi :

- bil usklajen za vse subjekte poslovnega sistema,
- zagotavljal ustrezne in ažurne podatke in informacije,
- omogočal povezljivost z drugimi informacijskimi sistemi in poslovnimi partnerji.

Projekt prenove informacijskega sistema se naj bi izvršil v dveh stopnjah, in sicer najprej kot prenova produkcijskih poslovnih procesov, čemur bi sledila izgradnja novega informacijskega sistema za podporo prenovljenim produkcijskim poslovnim procesom.

Rezultat prve stopnje projekta, opis prenovljenih poslovnih procesov, naj bi služil kot specifikacija novega informacijskega sistema (v tem času je bila aktualna tudi izbira in nakup ustreznega programskega paketa za podporo produkcijskim poslovnim procesom, kar pa se je kasneje zaradi posebnosti poslovnih procesov izkazalo za neustrezno rešitev in je rezultiralo v izgradnjo popolnoma novega informacijskega sistema).

Istočasno s prvo stopnjo projekta je potekala priprava na drugo stopnjo, projekt izgradnje novega informacijskega sistema, ter vpeljava Oracle CDM/PJM metodologije v projektno delo. Razlogov, ki so vplivali na odločitev za naknadno vpeljavo Oracle CDM/PJM metodologije, je bilo več, ključni pa so bili predvsem:

- odločitev o izgradnji novega informacijskega sistema z zunanjim partnerjem (v nadaljevanju izvajalcem) je bila sprejeta šele ob zaključku prve stopnje prenove informacijskega sistema, to je ob zaključevanju prenove poslovnih procesov,
- tehnologija za izgradnjo novega informacijskega sistema je bila izbrana šele ob zaključevanju prve stopnje projekta, po izbiri ustreznega zunanjega izvajalca,
- pomanjkljivo izdelan obstoječi izvedbeni načrt za drugo stopnjo,
- pozitivne izkušnje izbranega zunanjega izvajalca pri uporabi CDM/PJM metodologije na projektih izgradnje informacijskih sistemov z orodjem Oracle Designer,
- želja zmanjšati znana tveganja na projektu.

Nadaljnja poglavja analizirajo uporabo CDM/PJM metodologije pri razvoju poslovnega informacijskega sistema v zavarovalnici Slovenica z oceno dviga kakovosti projekta in produktivnosti projektne skupine v smislu zmanjševanja znanih tveganj pri projektih informacijske tehnologije. Poudarjeni so predvsem problemi, katerih reševanje bi utegnilo biti zanimivo tudi za ostale projektne skupine na projektih izgradnje informacijskih sistemov. Predstavljene so naloge in procesi, kjer se je z uporabo CDM/PJM metodologije povečala kakovost izdelkov oziroma drugi pristopi, ki so bili izvedeni v smislu izboljšanja kakovosti ali zmanjšanja tveganja.

V poglavju 4.1 je opisan postopek definiranja obsega, ciljev in pristopov po PJM metodologiji. Ta naloga se na projektu ni izvajala v skladu s priporočili Oracle metodologije, saj je bil večji del naloge izveden pred njeno uvedbo na projekt. Kljub temu so bila nekatera metodološka priporočila upoštevana v fazi sklenitve pogodbe med naročnikom in izvajalcem, kjer je bilo tudi opredeljeno, da se metodologija uvede na projektno delo. Poleg tega se je opredelitev projekta dopolnjevala tudi z rezultati vzporedno izvajanega procesa izdelave plana projekta, kar je opisano v nadaljevanju. Ta del je potekal vzporedno z vpeljavo CDM/PJM metodologije, zato je del aktivnosti že narejenih v skladu z metodološkimi priporočili. Sledi analiza aktivnosti definiranja kontrolnih točk, katere ključni problem na projektu je bil opustitev tradicionalnih faz projekta v izvedbenem načrtu. Predstavljeni so posamezni procesi izgradnje informacijskih sistemov, opredelitev procesov po CDM metodologiji ter analiza uporabe metodoloških priporočil na projektu. V zadnjih dveh poglavjih je opisana organizacija aktivnosti spremljanja projekta in opredelitev zaključka projekta. Slednja je opisana zgolj po teoretični plati in načrtovani organizaciji dela, saj projekt še ni v sklepni fazi.

V času zaključevanja pisanja naloge je projekt razvoja novega informacijskega sistema v zavarovalnici Slovenica v fazi zaključevanja kodiranja in testiranja posameznih modulov aplikacije za prvo skupino zavarovanj ter pred uporabniškim (procesnim) in sistemskim testiranjem. Zato bodo tudi zaključna poglavja, ki opisujejo naloge in procese, ki se na projektu še niso izvajali, opisani zgolj po teoretični plati ter, kjer so se priprav ena nalogo že izvedle, z morebitnimi opisi zasnov, ki smo jih opredelili v času planiranja projekta.

4.1 Definiranje obsega, ciljev in pristopa

Definiranje obsega, ciljev in pristopa projekta je po CDM/PJM metodologiji prva aktivnost priprave projekta. V celoten cikel projekta, kot je definiran v 2. poglavju, se vključi po zaključeni študiji izvedljivosti. Je nadgradnja začetnega plana projekta in je temeljni dokument za izdelavo podrobnega plana projekta z vsemi aktivnostmi in opisi izhodnih izdelkov. Cilj dokumenta je (Oracle: Project Management Process and Task Reference, 1996, str. 1-8 do 1-14):

- da dobita naročnik in izvajalec usklajen in jasen pogled na namen in obseg projekta, in sicer na:
 - namenske in objektne cilje projekta ter kritične faktorje uspeha,

- obseg projekta, ki zajema določitev:
 - aplikacije novega IS oziroma obstoječe aplikacije, ki naj bi jih novi informacijski sistem zamenjal,
 - ali bo projekt zajemal prenovno poslovnih procesov,
 - ključnih izhodnih rezultatov projekta,
 - grobe arhitekturne zasnove sistema, lokacije in omrežje vključno z organizacijsko strukturo naročnika oziroma uporabnikov novega informacijskega sistema,
 - sistemskih vmesnikov med obstoječimi ali planiranimi informacijskimi sistemi (povezava z drugimi projekti) ter novim informacijskim sistemom,
 - vsebine in pogoje glede prilagoditev in nadgraditev novega informacijskega sistema po uvedbi v produkcijo,
 - prenosa in pretvorbe obstoječih podatkov,
 - potrebe po usposabljanju članov projektne skupine,
 - obsega izobraževanja končnih uporabnikov oziroma naročnika,
 - predvidenih testiranj novega informacijskega sistema,
 - obsega investicije;
- da naročnik in izvajalec prepoznata tveganje projekta, določita postopke zmanjševanja ali odprave posameznega tveganja, definirata termine in vsebine kontrolnih točk ter prepoznata omejitve in predpostavke, na katerih temelji projekt, ter kritične faktorje uspeha.
- da naročnik in izvajalec predvidita relacije z drugimi projekti in sistemi,
- da se določi pristop k projektному delu in projektne metode, ki bodo uporabljene,
- da se določi specifikacije in kriterije za prevzemni test novega informacijskega sistema ter postopek ugotavljanja zaključka projekta.

Tako definiran dokument je osnova za specifikacije za izdelavo aplikacij, za določitev obsega in predmeta systemske analize. Posredno je tudi osnova za pripravo testnega načrta, načrta za tranzicijo in načrta za izobraževanje uporabnikov.

Na projektu izgradnje novega informacijskega sistema v zavarovalnici Slovenica se ta aktivnost ni izvajala v skladu s PJM priporočili. Razlog za to je predvsem v tem, da se je metodologija uvedla na projekt šele pred izvedbenimi fazami. Kljub temu so se vsi omenjeni parametri določili, in sicer:

- prva opredelitev parametrov v zagonskem elaboratu projekta, pred pričetkom izvajanja aktivnosti prenove poslovnih procesov,

- v času prenove poslovnih procesov vse do sklenitve pogodbe z izvajalcem izvedbene faze projekta se je z različnimi dokumenti vsebina dopolnjevala (predvsem podrobna opredelitev kritičnih faktorjev uspeha in tveganj, obseg projekta zaradi združitvenega procesa dveh zavarovalnic ter interakcije in soodvisnosti z drugimi projekti)

in bila dokončno usklajena med sponzorjem, vodstvom projekta in izvajalcem v trenutku prehoda na izvedbeno fazo projekta.

Prednosti postopnega dopolnjevanja dokumenta Obsega, ciljev in pristopa vse do izvedbene faze projekta so predvsem :

- zaradi izvajanja prenove poslovnih procesov je bilo pred zaključkom te aktivnosti nemogoče predvideti najustreznejšo izvedbo (na primer nakup izdelanega produkta, modifikacija novega ali obstoječega produkta, nadgradnja obstoječega sistema, izgradnja novega sistema v lastni izvedbi ali z zunanjim izvajalcem,...). Tako se je koncept novega informacijskega sistema zasnoval na dejanskih potrebah, ki so jih pokazali prenovljeni poslovni procesi. S tem se je zmanjšalo tveganje, da nov informacijski sistem ne bo ustrezal prenovljenim poslovnim procesom glede na izbrano rešitev (to je izgradnja novega informacijskega sistema), ki je lahko zelo dolgotrajen proces (pred izvedbeno fazo je običajno opravljenega šele petino projekta). Še vedno pa je obstajalo tveganje, da bo razvoj tekel predolgo časa in s tem izgubil nekaj poslovne prednosti ali pa, da se bodo procesi do takrat že preveč spremenili.
- vzporedno s tem so tekale tudi aktivnosti priprave plana projekta, to je izvedbenega načrta projekta, oblikovanje projektne skupine ter organizacija podpornih procesov. Obe aktivnosti (Definiranje obsega, ciljev in pristopa ter Priprava plana projekta) sta se iterativno postopoma dopolnjevali. Zaradi tega so dobivali parametri dokumenta Obseg, cilji in pristop tudi pomensko konkretnejšo vsebino postopoma (ključne naloge in izhodne rezultate projekta, tveganja, kritične faktorje uspeha).

Vendar pa so se sčasoma pokazale tudi določene pomanjkljivosti neizvajanja aktivnosti po metodoloških napotkih, in sicer predvsem, ker projektna skupina ni bila seznanjena z obsegom, cilji in pristopom projekta v celoti, kot je predvideno v tej nalogi, saj so bile opredelitve projekta zapisane v različni dokumentaciji, ki pa ni bila dostopna in predstavljena vsem članom projektne skupine. Članom projektne skupine in končnim uporabnikom je potrebno Obseg, cilje in pristop na projektu predstaviti že v začetni fazi projekta, saj s tem med člani projektne skupine in uporabniki poenotimo poglede na obseg in cilje projekta, zvišamo njihovo motivacijo, jih vključimo v pripravo projekta, povečamo zaupanje v projekt in se izognemo nerealnim pričakovanjem končnih uporabnikov.

4.2 Plan projekta

Plan projekta temelji na dokumentu Definiranje obsega, ciljev in pristopa. Glede na vsebino plana projekta sta se, kot je omenjeno že v predhodnem poglavju, izdelka obeh procesov iterativno dopolnjevala vse do najoptimalnejše rešitve glede na znane parametre (časovna omejitev, razpoložljivost virov). Namen aktivnosti je, da se ustrezno definira celoten proces razvoja novega informacijskega sistema. PJM priporočila so v skladu s standardom ISO12207 (standard govori o tem, kako morajo biti postopki življenjskega cikla projekta urejeni, PJM metodologija pa vsebuje napotke za izvedbo urejenosti). Plan projekta sestavljajo:

- izvedbeni načrt projekta,
- načrt komunikacij na projektu,
- načrt zagotavljanja kakovosti na projektu,
- načrt upravljanja z dokumentacijo.

4.2.1 Določitev nalog projekta, časovni potek projekta in definiranje izhodnih rezultatov projekta

Cilj aktivnosti je opredelitev procesa upravljanja z delom, katerega rezultat mora biti izvedbeni načrt projekta z vsemi aktivnostmi, opis ciljev posameznih projektnih nalog in izdelkov projektnih nalog ter določitev standardov in postopkov spremljanja in izvedbe projekta.

PJM metodologijo za proces upravljanja z delom sestavljata dve skupini priporočil, in sicer:

- navodila za pripravo standardov (za spremljanje izvedbenega načrta projekta, replaniranje, poročanje o napredku na projektu in za upravljanje s finančnimi viri) in postopkov za nadzor nad projektnim delom (spremljanje porabljenega časa na projektu, poročanje o napredovanju izvedbe projekta, spremljanje projekta in replaniranje aktivnosti),
- navodila in pripomočke za izdelavo izvedbenega načrta projekta, kar zajema določitev vseh nalog na projektu, določitev časa, potrebnega za izvedbo posamezne naloge, pravilna razporeditev nalog glede na njihovo soodvisnost ter določitev časovnega zaporedja izvajanja nalog.

Pri pripravi standardov in postopkov za nadzor nad projektnim delom smo kot osnovo uporabili PJM priporočila, ki smo jih aplicirali in modificirali na izbrano orodje za planiranje in spremljanje projekta (Microsoft Project).

Za izdelavo izvedbenega načrta projekta, ki sta ga naročnik in izvajalec skupaj pripravila in usklajevala, se je prav tako uporabila PJM predloga izvedbenega načrta, ki že vsebuje vse aktivnosti CDM metodologije, z vsemi medsebojnimi soodvisnostmi med posameznimi

nalogami in ustrezno fazno razporeditvijo nalog vodenja (PJM) in vsebine (CDM) projekta.

Pri izdelavi izvedbenega načrta projekta se je upoštevalo naslednje:

- standardov in postopkov za upravljanje z delom (razen odločitev za Microsoft Project kot orodje za planiranje) v Slovenici ni bilo definiranih,
- definicija poslovnih potreb je bila rezultat aktivnosti prenove poslovnih procesov, tako da je bil poslovni model procesov že izdelan. Poslovni funkcijski in podatkovni model naj bi bila naloga izvajalca, zato se ju je predstavilo iz procesa definiranja poslovnih potreb v proces načrtovanje in izgradnja baze podatkov in načrtovanje in izgradnja aplikacije.
- Časovni okvir projekta je bil prilagojen strategiji organizacije, zato je bila pri izdelavi izvedbenega načrta tendenca po maksimalni izkoriščenosti časa in ljudi v fiksnem časovnem obdobju. Iz tega stališča je bil ukinjen tradicionalni fazni pristop, upoštevalo pa se je sosledje aktivnosti in čas, potreben za izvedbo aktivnosti.
- Glede na velik obseg projekta je bil s strani sponzorja projekta potrjen postopni način izgradnje informacijskega sistema za posamezna zavarovanja.

Prednosti uporabe CDM metodologije pri planiranju:

- ker v Slovenici do tedaj standardi za spremljanje dela niso obstajali, je bilo najenostavneje in najhitreje prevzeti kar PJM priporočila,
- z uporabo predloge za izdelavo izvedbenega načrta projekta, kjer so v pravilni soodvisnosti že vse naloge izvedbe projekta po CDM metodologiji, razporejene v skladu s tradicionalnim, faznim pristopom k izvedbi projekta, je bilo možno v kratkem času (dejansko en človek/dan) izdelati prvi predlog izvedbenega načrta projekta, kjer ni bila izpuščena nobena pomembna aktivnost,
- s pravočasnim definiranjem standardov in postopkov za nadzor nad izvedbenim načrtom projekta smo se izognili nekontroliranim spremembam izvedbenega načrta,
- ker CDM metodologija vključuje natančna navodila za izvedbo posamezne naloge vključno z opisom končnega izdelka, je naloga opisovanja izdelkov v planu projekta odpadla,
- z dobro definiranim izvedbenim načrtom se je zmanjšalo tveganje zaradi nepopolnega izvedbenega načrta projekta in s tem povezanim povečanjem časovnih virov,
- glede na postopen način izgradnje informacijskega sistema za posamezne skupine zavarovanj v treh zaporednih etapah smo časovno oddaljeni zaključek projekta razbili na manjše in bolj obvladljive sklope in s tem časovno približali zaključek

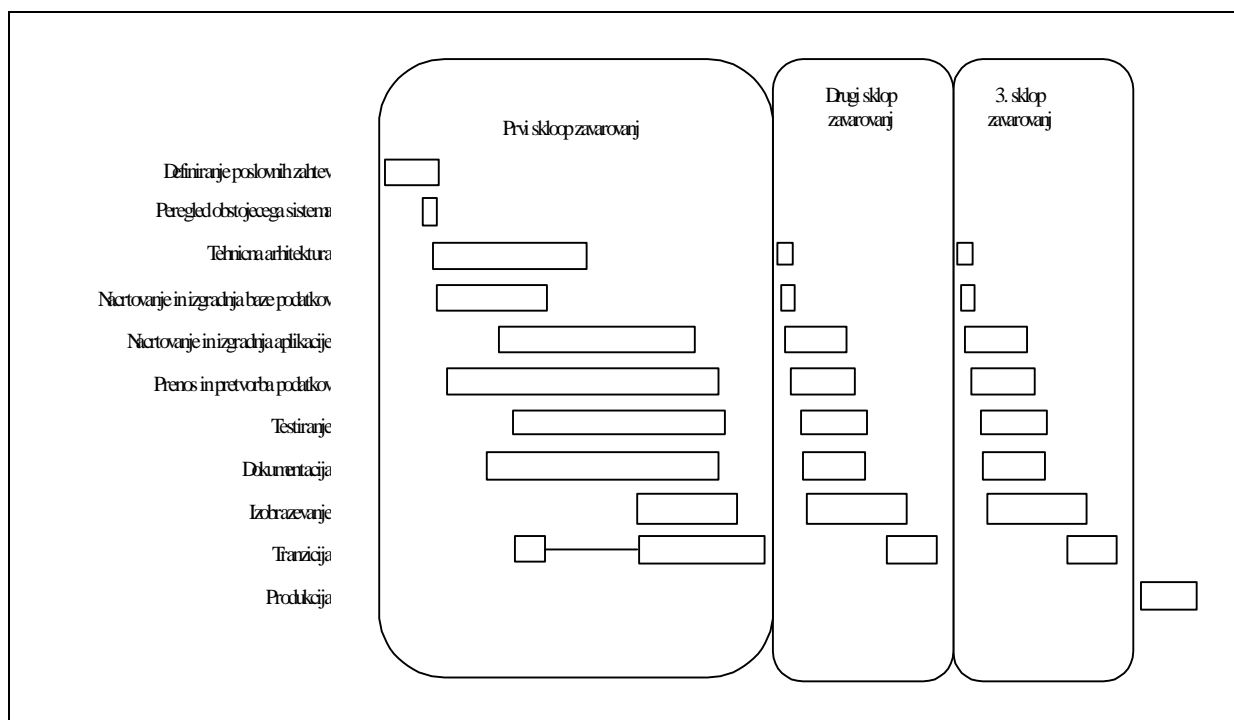
prve etape bodočim uporabnikom. Z etapno izgradnjo informacijskega sistema smo tudi skrajšali nekatere zelo obsežne naloge projekta, kot so na primer izobraževanje končnih uporabnikov, prenos in pretvorba podatkov, izdelava modulov in uporabniška dokumentacija, bistveno pa nismo pridobili pri izgradnji podatkovnega modela v prvi etapi, saj se je za vse sklope zavarovanj izdelal enotni podatkovni model (slika 4).

Ker je bil izvedbeni načrt dejansko izdelan ob zaključku faze analize, je bil izveden z vsemi znanimi dejstvi, ki so bila rezultat analize.

Zaradi prilagoditve časovnega okvira projekta strategiji organizacije se je plan modificiral v smislu zmanjševanja »rezervnega« časa med posameznimi nalogami in fazami, zaradi česar se je meja med tradicionalnimi fazami zabrisala (slika 4). Posledice tega so bile:

- slabo definirane kontrolne točke izvedbene faze projekta, saj ni bilo možno postaviti kontrolne točke na sredino aktivnosti. Rešitev bi lahko bila razdelitev aktivnosti na smiselno zaključene podaktivnosti (na primer naloga izdelava aplikacije pri nas opredeljena kot 8 mesecev trajajoča aktivnost), ki se v vnaprej določenih časovnih intervalih kontrolirajo (za prej omenjeno aktivnost izgradnje aplikacije delitev na izdelavo splošnih šifrantov, šifrantov sklepanja, šifrantov škod, sklepalnega dela aplikacije, ...). Na podlagi tega se razbije tudi aktivnosti na preostalih procesih (za zgornji primer na testiranju, prenosu in pretvorbi podatkov, izdelavi dokumentacije).
- izvedbeni načrt projekta ni vseboval nikakršnega rezervnega časa za popravljanje in dopolnjevanje izdelkov projekta, v kolikor je bilo tako določeno po opravljeni oceni kakovosti ali kot rezultat kontrolne točke, zaradi česar se je povečevala zamuda na projektu.

Slika 4: Shematičen prikaz izvedbenega načrta projekta na zavarovalnici Slovenica



Kasneje v izvedbeni fazi projekta, konkretno pri procesu testiranja, se je pojavila še ena pomanjkljivost CDM metodologije, in sicer neopredelitev časa, potrebne za odpravo napak, odkritih med testiranjem, posledica česar so bile preobremenjenost programerjev, zaostanki na procesu izgradnja aplikacije in nesprotno reševanje in odpravljanje napak. Tu je koristen nasvet MSF metodologije, ki ima posebej definiran čas za stabiliziranje novega informacijskega sistema, saj se lahko poleg napak, najdenih v postopku testiranja, pojavijo tudi težave s prenosom sistema iz razvojnega oziroma testnega v produkcijsko okolje.

Skrajšanje ali celo opustitev rezervnega časa pri posamezni aktivnosti zelo poveča tveganje, da projekt ne bo končan v postavljenih rokih. Rezervni čas je potrebno predvideti zaradi prej neznanih tehničnih (morebitne napake ali omejitve razvojnih orodij ali platforme) ali organizacijskih težav (nepredvidene odsotnosti izvajalcev aktivnosti, pomanjkljive ali napačne specifikacije,...).

4.2.2 Oblikovanje projektne skupine

Proces oblikovanja projektne skupine zajema identifikacijo vlog na projektu, določitev ustreznih strokovnjakov na posameznih nalogah ter dopolnitev izvedbenega načrta z ustreznimi izvajalci. Cilj je oblikovati ustrezno projektno skupino, ki v danih časovnih okvirih uspešno izvede zadane aktivnosti na projektu.

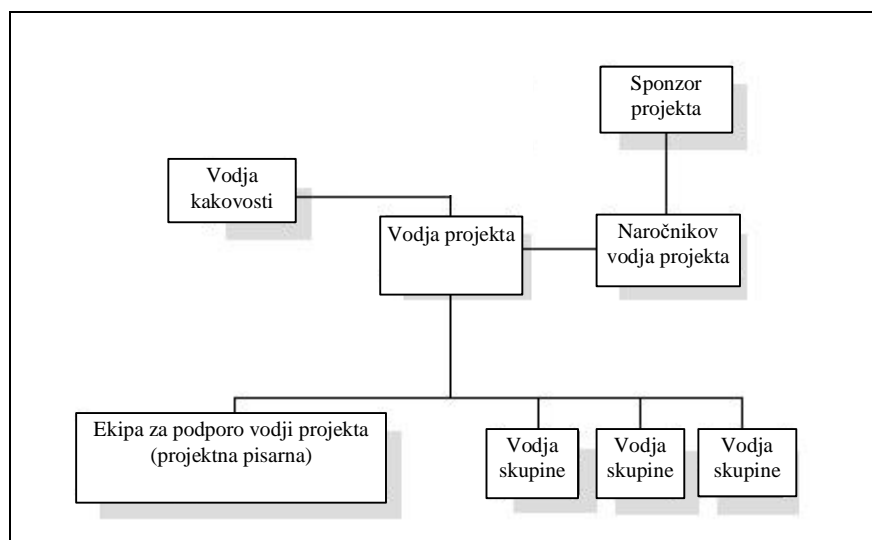
PJM metodologijo za proces upravljanja z viri za področje oblikovanja projektne skupine sestavljata dve skupini priporočil, in sicer:

- priporočila za sestavo projektne skupine,
- priporočila za implementacijo projektne skupine na izvedbeni načrt projekta.

CDM/PJM metodologija pri vsaki nalogi projekta za njeno izvedbo opredeljuje strokovno skupino ter vloge in odgovornosti članov strokovne skupine. Metodologija deli projektno skupino na dva dela, in sicer na vloge upravljanja projekta (PJM) in na vloge izvedbe projekta (CDM). Celotna organizacijska struktura temelji na dejstvu, da je naročnik izvzet iz izvedbe projekta ter da se vse aktivnosti projekta od definicije do produkcije izvedejo na strani izvajalca.

Tako definirana projektna organizacija je predstavljena na sliki 5.

Slika 5: CDM/PJM projektna organizacija



Vir: Oracle: Project Management Method Handbook, 1996, str.1-11

Za izvedbeni del projekta pa opredeljuje naslednje vloge (naštete so le najpomembnejše, podrobnosti v Oracle Method: CDM Classic Method Handbook, 2000, str. B-1 do C-10, Oracle Method: CDM Classic Process and Task Reference, 2000), ki so glede na potrebe, opredeljene za posamezne naloge, grupirane v ustrezne skupine:

- analitik, ki je odgovoren za vodenje aktivnosti v zvezi z definiranjem poslovnih zahtev naročnika in za pravilno interpretacijo podatkov pri prenosu in pretvorbi podatkov,
- načrtovalec kapacitet, ki je odgovoren za pripravo in vzdrževanje načrta kapacitet in za revidiranje načrtov novega IS z vidika performančnih zahtev,

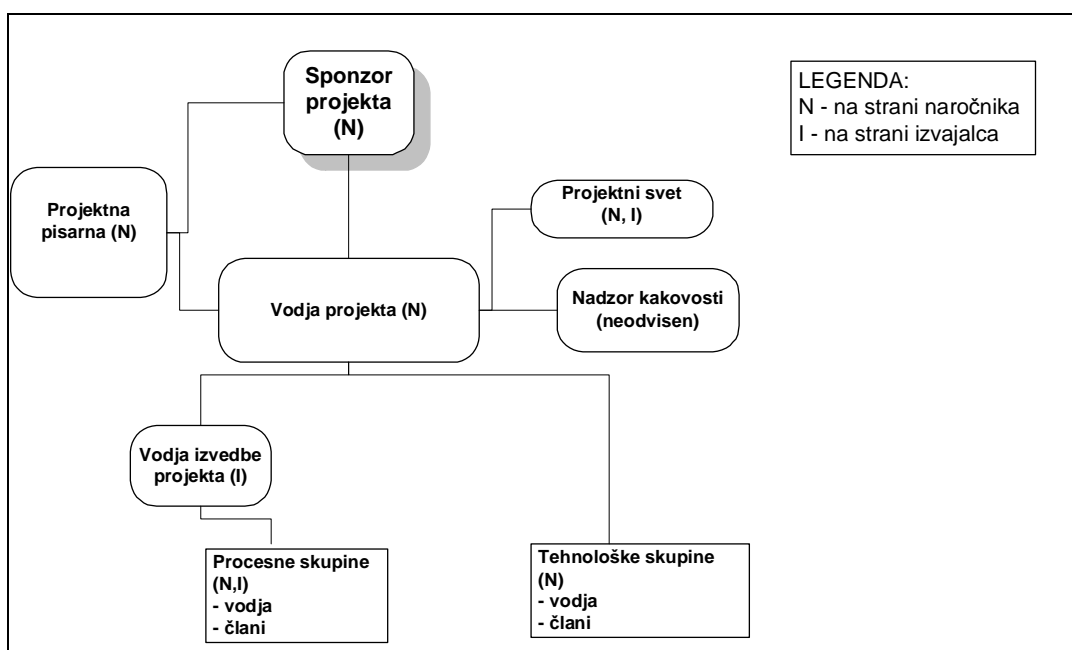
- administrator podatkov, ki je odgovoren za definicijo, integracijo in kvaliteto podatkov,
- administrator razvojne podatkovne baze, ki je odgovoren za inštalacijo, konfiguracijo, arhiviranje in obnavljanje razvojne podatkovne baze, kreiranje različnih podatkovnih baz (za potrebe razvoja, testiranja, izobraževanja,...) in vzdrževanje nadzora nad dostopom do baze,
- upravitelj sistema, ki je odgovoren za tehnično podporo sistema (standardi za IS, zagotavljanje tehničnih in poslovnih informacij o obstoječem sistemu, ...),
- načrtovalec aplikacije, ki je zadolžen za izdelavo načrtov aplikacije ter testiranja modulov,
- upravitelj mreže, ki svetuje, določi in implementira omrežno strojno in programsko opremo v skladu z zahtevami projektne skupine,
- administrator produkcijske baze podatkov, ki v fazi projektiranja skrbi za fizično implementacijo modela podatkov, definiranje particij in indeksov, v fazi vzdrževanja pa je zadolžen za dnevno delovanje baze podatkov (konsistentnost, dostopnost),
- programer, ki je odgovoren za izdelavo kode glede na uporabniške zahteve ter za diagnosticiranje in odpravljanje napak, najdenih pri testiranju programske opreme,
- programer konverzije podatkov je odgovoren za programiranje procesov konverzije podatkov,
- sistemski administrator je odgovoren za administracijo razvojnega sistema (konfiguracija strojne opreme, instalacije, konfiguracije in vzdrževanje operacijskega sistema in preostale programske opreme, zagotavlja dnevna arhiviranja, vzdržuje varnost sistema, nudi tehnično podporo razvojnemu sistemu),
- načrtovalec infrastrukture je odgovoren za načrtovanje tehnične infrastrukture in za varnostno strategijo,
- dokumentalist, odgovoren za pripravo tehnične, uporabniške in operativne dokumentacije,
- preizkuševalec, ki je odgovoren za razvoj in izvajanje testnih skript,
- inštruktor, ki je odgovoren za pripravo zahtev za izobraževanje, izdelavo načrta izobraževanja, pripravo materialov za izobraževanje in izvedbo izobraževanja,
- načrtovalec baze podatkov, odgovoren za načrtovanje in izdelavo logičnega in fizičnega modela podatkov,
- administrator aplikacije, ki je odgovoren za vzdrževanje in nadzor nad produkcijo, načrtovanje produkcijskih dnevnih procedur, definiranje dostopov uporabnikov in uporabniških skupin do podatkov in administracijo podatkov.

Pri pripravi organizacijske sheme projektne skupine je bilo v primeru projekta na Slovenici kar nekaj specifičnosti. CDM struktura je služila kot osnovno izhodišče dobre projektne organizacije, modifikacije pa so se izvedle glede na naslednje predpostavke:

- vodja projekta je bil na strani naročnika,
- definicija in analiza sta se izvedli na strani naročnika,
- organizacija izvedbenega načrta je bila orientirana procesno,
- pomanjkanje števila ustreznih strokovnjakov, zato je bilo več vlog dodeljeno eni osebi.

Na podlagi teh ugotovitev je bila oblikovana formalna organizacijska shema na projektu (slika 6).

Slika 6: Organigram projektne skupine v Slovenici



Vir: Slovenica, Adriatic: Organizacija projekta, vloge in naloge članov projektne skupine, str.6

Glavna razlika med obema organizacijskima strukturama je bila v izvzetju tehnoloških skupin, ki so izvajale aktivnosti definiranja poslovnih zahtev in prenove poslovnih procesov, iz izvedbe projekta ter v postavitvi procesnih skupin v sklopu izvedbenega dela projekta, ki so bile oblikovane zaradi procesne organiziranosti izvedbenega plana projekta.

S takšno projektno organizacijo in poimenskim seznamom strokovnjakov smo dopolnili izvedbeni načrt projekta. Pri kontroli obremenitev članov projektne skupine je bilo kar nekaj težav, saj se je zaradi podvajanja vlog in velike koncentracije različnih nalog v določenem časovnem obdobju pojavljala tudi do 500% obremenitev posameznih članov projektne ekipe. S številnimi iteracijami usklajevanja med organizacijsko strukturo in izvedbenim načrtom je sčasoma vendarle prišlo do realnega in uresničljivega izvedbenega načrta projekta.

Z uporabo metodologije CDM/PJM pri aktivnostih priprave izvedbenega načrta projekta in organizacijske strukture projekta smo dosegli:

- zagotovitev ustreznih strokovnjakov,
- zaradi pomanjkanja ustreznih strokovnjakov pravočasno planirano dodatno izobraževanje in izobraževanje ustrezno vključeno v izvedbeni načrt projekta,
- pravočasno planiranje resursov in usklajevanje njihovega dela na drugih projektih z delom na tem projektu (zasedenost, razpoložljivost in prerazporeditev),
- posamezniki so bili pravočasno, ob začetku projekta, seznanjeni s svojo vlogo, odgovornostjo in pričakovanji,
- optimizacijo izvedbenega načrta projekta,
- zmanjšanje tveganj glede neustrezne strukture projektne skupine ali da bi na kakšno vlogo pozabili.

4.2.3 Komunikacije na projektu (poročanje, kontrola, upravljanje s problemi, tveganji in spremembami)

Definiranje komunikacij na projektu je namenjeno opredelitvi načina komuniciranja vseh sodelujočih na projektu z namenom poenotiti načine in orodja za komuniciranje in podajo vsebin ter določiti potrebno frekventnost komunikacij.

Aktivnost komunikacije na projektu PJM metodologija opredeljuje z določitvijo:

- podpornih procesov za upravljanje z odprtimi vprašanji, problemi in zahtevami za spremembo,
- procesov za spremljanje izvajanja izvedbenega načrta projekta (postopki poročanja o napredovanju projekta, postopki za organizacijo in izvedbo različnih rednih sestankov ter postopki poročanja o delu na aktivnostih).

Na projektu smo za postopke uporabili naslednje predpostavke, priporočila in orodja:

- priporočila PJM metodologije,
- standardne postopke, ki so bili že v uporabi v zavarovalnici,
- razpršenost in informacijska nepovezljivost projektne ekipe,
- uporaba orodja Project Central ter elektronske pošte.

Na podlagi tega smo definirali postopke ter vloge in odgovornosti za:

- upravljanje z odprtimi vprašanji in zahtevami za spremembe (v skladu z definicijo v poglavju 2.2),
- delegiranje aktivnosti,
- poročanje o napredku dela na projektu in spremljanje napredovanja projekta (poročila o opravljenem delu in sledenje napredovanja projekta glede na poročila),
- tipe in vsebine sestankov na projektu,
- tipe in oblike poročanja o napredovanju projekta (poročanje sponzorju projekta).

Upravljanje s problemi v tej fazi ni bilo definirano, temveč je bilo prepuščeno testni ekipi, da definira postopek in orodje za upravljanje z napakami oziroma problemi v skladu s tehnologijo testiranja.

Z uporabo CDM metodologije smo poleg skrajšanja časov izdelave plana projekta dosegli tudi:

- da so postopki komunikacij bili definirani že v začetni fazi projekta, da so bili člani projektne skupine z njimi seznanjeni in so se jih v življenjskem ciklu projekta tudi posluževali,
- zmanjšali tveganje nedokumentiranja različnih odločitev na projektu,
- pravočasno izvedbo izobraževanje iz uporabe orodja Project Central za poročanje o napredovanju na projektih nalogah.

4.2.4 Zagotavljanje kakovosti na projektu

Standardi in postopki zagotavljanja kakovosti na projektu zagotavljajo konsistentnost med člani projektne skupine in določajo kriterije za oceno kakovosti. Opredelitev standardov na projektu lahko teče tudi v sklopu drugih projektih aktivnosti, če so le izdelani in sprejeti do trenutka, ko jih na drugih projektih aktivnostih potrebujemo.

CDM metodologija predlaga tri različne pristope za ocenjevanje kakovosti, in sicer:

- redne preglede v vnaprej definiranih časovnih intervalih,
- pregled vnaprej definiranih končnih rezultatov aktivnosti po planu projekta,
- metriko merjenja kakovosti.

Aktivnosti določitev standardov tečejo v procesih izgradnje informacijskega sistema in so opredeljeni znotraj CDM metodologije.

Posebne metrike merjenja kakovosti izdelkov na projektu izgradnje informacijskega sistema v zavarovalnici ni bilo opredeljene, prav tako so bile aktivnosti dolgotrajne in ni bilo opredelitev, ki bi dolgotrajne aktivnosti delila v manjše podsklope. Zato je bila

sprejeta odločitev, da se spremlja samo kakovost ključnih izdelkov v skladu s sprejetimi standardi in zahtevami, opisanimi v CDM opisih izdelkov.

Da bi kljub temu zagotovili zadosten nadzor nad izvajanjem projekta, smo v načrt izvajanja presojo kakovosti dodali v seznam izdelkov presoje predvsem izdelke, ki opredeljujejo različne standarde, načrte izgradnje (podatkovne baze in aplikacije) ter preostale plane (prenos in pretvorba podatkov, testiranja, tranzicije). S tem smo dosegli, da so se ti dokumenti pred izvajanjem aktivnosti pripravili, da so ustrezali zahtevani kakovosti in da so bili sprejeti s strani članov projektne skupine in končnih uporabnikov, s čimer smo povečali verjetnost za njihovo ustreznost in uporabo.

Poleg tega smo v času izvajanja projekta preverjali kakovosti končnega produkta tudi na druge, manj formalne načine, in sicer na rezultatih oziroma opažanjih drugih aktivnosti, na primer rezultatov vzporednega testiranja modulov ter na podlagi ocene naročnikovih izvajalcev v mešanih ekipah (predvsem v ključnih nalogah načrtovanja in izgradnje baze podatkov in aplikacije).

Glede na izbiro načina izvedbe projekta (nefazno izvajanje) so bili planirani zaključki ključnih aktivnosti, katerih rezultati naj bi bili podvrženi oceni kakovosti, šele tik pred zagonom novega informacijskega sistema v produkciji. Zato se je izkazal takšen način preverjanja kakovosti na projektu kot nezadosten, saj se je zelo pozno ocenjevala dejanska uporaba standardov in izvedba v skladu z načrti.

Z uporabo predlogov PJM metodologije na projektu smo vendarle dosegli, da so standardi, definirani za vse aktivnosti, za katere so potrebni, razumljeni in se jih tudi uporablja.

4.2.5 Upravljanje s konfiguracijo

Upravljanje s konfiguracijo je identifikacija, nadzor in spremljanje vseh izdelkov, ki nastanejo v procesu razvoja novega informacijskega sistema. Ti izdelki so lahko najrazličnejšega tipa in se od projekta do projekta razlikujejo. Postopki upravljanja s konfiguracijo morajo omogočiti:

- identificiranje in skrbno spremljanje posameznih gradnikov konfiguracije končnega produkta, ki se izdelajo znotraj posameznih nalog projekta,
- nadzor ustreznosti konfiguracije glede na poslovne in sistemske zahteve,
- izvajanje samo potrjenih sprememb na gradnikih in spremljanje vzroka sprememb,
- spremljanje vseh verzij posameznih izdelkov.

PJM priporočilo za definiranje plana projekta vsebuje aktivnost definiranja standardov in postopkov za upravljanje s konfiguracijo projekta, in sicer:

- definiranje tipov dokumentov, ki se na projektu generirajo in določitev postopkov za upravljanje z dokumenti (postopke za kreiranje, distribucijo, vzdrževanje in zaščito različnih tipov dokumentov),

- definiranje konfiguracije in upravljanje s konfiguracijo (sestave in lokacije konfiguracije, način verzioniranja elementov in celotnega sklopa konfiguracije) ter
- upravljanje z verzijami produkta (postopki za pripravo in izdajo posameznih verzij končnega produkta).

Za posamezne člane projektne skupine morajo postopki nadzora konfiguracije omogočati dostop do zelenega podatka na projektu oziroma vzpostaviti v kateremkoli trenutku razvoja določeno stanje projekta iz preteklosti (recovery falls from any point). Prav tako morajo postopki omogočati, da se na posameznih izdelkih vršijo samo potrjene spremembe in vzpostaviti tok dela med posameznimi člani projektne skupine (na primer pri potrjevanju ali zavračanju sprememb v fazi testiranja).

Od vsega, kar PJM metodologija ponuja, so se na projektu razvoja informacijskega sistema izvajali samo postopki za upravljanje z dokumenti.

Celotna organizacija upravljanja s konfiguracijo, kakršno zajema CDM priporočilo, je zelo obsežna in je v praksi neizvedljiva brez ustreznega orodja za podporo upravljanju s konfiguracijo. Takšnih orodij je na tržišču kar nekaj, vendar je njihova cena zelo visoka, saj zahtevajo kar nekaj različnih virov (finančne, človeške, čas) za ustrezno instalacijo, prilagoditve, uvajanje in podporo v času projekta oziroma produkcije. Pri izbiri orodja oziroma odločitvi za ali proti si je potrebno zato odgovoriti na nekaj ključnih vprašanj:

- kateri elementi bi morali biti pod nadzorom,
- ali orodje omogoča nadzor nad temi elementi konfiguracije,
- ali gre za enkratno uporabo orodja na projektu ali bo le-to uporabno tudi v času produkcije,
- koliko bo uporabnikov novega orodja in kakšna je njihova usposobljenost za njegovo uporabo,
- kakšne so alternativne variante nadzora nad konfiguracijo.

V zavarovalnici smo imeli pri izbiri ustreznega nadzora še dodatne težave zaradi sestave projektne skupine (tri lokacije, tri pravne osebe, mešana projektna skupina med naročnikom in izvajalcem).

Pri analizi odgovorov na zgornja vprašanja, smo najprej skušali razrešiti skupine elementov pod nadzorom. To so bile:

- modeli poslovnih procesov, izdelanih z ARIS orodjem,
- dokumenti (Word format) kot rezultat nekaterih aktivnosti projekta (opisi poslovnih procesov, izhodni rezultati nalog projekta, poročila, zapisniki, navodila,...),
- repozitorij orodja Oracle Designer, kjer se je nahajala vsa aplikacijska koda (baza, moduli, prožilci, knjižnice, ...).

Pri tem smo ugotovili, da orodja za verzioniranje tretjega sklopa (repozitorij) na tržišču ni. V času projekta je prišla na tržišče nova verzija Oracle Designer-ja, ki je omogočala verzioniranje repozitorija in njegovih posameznih elementov. Čeprav je bil razvoj še v začetni fazi in se je prestavil na višjo verzijo orodja, se samo verzioniranje zaradi neizkušenosti članov projektne skupine, zahtevnosti postopka verzioniranja ter obsežnosti projektne repozitorija ni uvedlo. Tako je bilo tu vzpostavljeno shranjevanje stanja repozitorija v določenih časovnih intervalih (konkretno 14 dnevni). Takšen način se je uporabil tudi pri testiranju (podrobneje v poglavju Testiranje), saj so bili rezultati testiranja vezani na določeno stanje repozitorija (na dan shranjevanja).

Na podoben način smo se odločili tudi pri ARIS modelih (redno shranjevanje določene stopnje razvoja ARIS modelov, pri čemer je bilo potrebno paziti, da so se tudi tekstovni opisi istočasno shranili, da se je ohranila vsebinska povezljivost med modelom in opisi). Posebno poglavje in zahtevnost priprav pa so bili word-ovi dokumenti. Ker je bilo uporabnikov veliko (150), je bilo upravljanje izjemno zahtevno. Tako smo se odločili za uvedbo zaščitenega datotečnega sistema, kjer je vsak član projektne ekipe imel možnost vpogleda v dokumente projekta, medtem ko so imeli pravico do ažuriranja in odlaganja dokumentov v datotečni sistem samo skrbniki za posamezna področja. Za vsako področje je bil določen natančno en skrbnik, ki je bil odgovoren za to, da se vse spremembe dogajajo nadzorovano in potrjeno ter da so ustrezno zabeležene.

Nadaljnji problem je bil tudi repozitorij oziroma njegovo shranjevanje. Glede na to, da je verzioniranje znotraj Oracle Designer-ja edina možna rešitev, bi jo morda vendarle bilo treba bolje preizkusiti in izrabiti na projektu. Problem se je dostikrat pokazal pri testiranju, ko so določeni moduli že bili uspešno testirani, pa se je kasneje izkazalo, da je modul ponovno nedelujoč. Ker ni bilo nikakršnega sledenja spremembam, je bilo izjemno težko določiti, kdaj in zakaj je prišlo do usodnih sprememb.

Konfiguracijo v začetni fazi nismo določili, kar se je kot velika pomanjkljivost pokazalo pri testiranju. Ker konfiguracija ni bila definirana, je bila postavitve testnega okolja izredno dolgotrajna in zahtevna, kar bo podrobneje opisano pri procesu testiranja (od razvojne skupine smo dobili repozitorij in nekaj dodatnih datotek, pri postavitvi testnega okolja pa smo skupaj z razvojno skupino izdelovali še konfiguracijsko shemo).

Poglavitne prednosti uporabe PJM metodologije na projektu v zavarovalnici so bile predvsem v pravočasnem in temeljitem definiranju vseh konfiguracij, ki se potrebuje pri projektne aktivnostih ter v postavitvi postopkov in standardov za upravljanje s konfiguracijo, s čimer se vzpostavi nadzor nad spremembami, potreben tako pri razvoju posameznih gradnikov aplikacije, verzioniranju aplikacije kot pri nadzoru projektna dokumentacije.

4.3 Kontrolne točke projekta

Kontrolne točke so namenjene pregledu napredovanja projekta in oceni opravljenega dela (Vallabhaneni, 1996, str. 762). Pomembno je, da so kontrolne točke postavljene tako, da so

merljive (Sturm, 1999, str.152), kajti v nasprotnem primeru ni možno izdelati oceno o stanju in napredku projekta.

CDM/PJM opredeljuje kontrolne točke kot strateške točke, na katerih se preverja napredovanje projekta. Določi se jih že v aktivnosti definiranja obsega, ciljev in pristopa tako na podlagi opredelitev v dokumentu, ki je rezultat omenjene aktivnosti, kot glede na izvedbeni načrt projekta.

Kontrolne točke naj bi bile postavljene na prelomnih trenutkih na projektu (na primer pri tradicionalnih faznih pristopih na prehodu iz ene v drugo fazo), lahko pa jih postavimo poljubno pogosto, v skrajnem primeru tudi na vsako posamezno aktivnost, če je takšna postavitev smotrna in upravičena. Glede na izvedbeni načrt, ki zaradi strukture ne omogoča jasne razmejitvene točke (kot je bilo v primeru izvedbenega načrta na opisovanem projektu), je potrebno kontrolne točke postaviti včasih tudi na sredino katerih ključnih aktivnosti in že v planu opredeliti, kaj od te aktivnosti mora ob kontrolni točki biti zaključeno (razbitje nalog po vsebini na manjše naloge).

Predlog, kako naj kontrolna točka izgleda, je stvar projektnega vodenja. Kontrolne točke so za izvajanje projekta ključne, saj se glede na vrednost kontrolne točke (ki naj bi bila binarna USPEŠNO/NEUSPEŠNO) najvišje vodstvo opredeljuje, ali in kako s projektom nadaljevati ali projekt morda celo opustiti. Zato menim, da je smotrno že v planu projekta opredeliti vsebino kontrolnih točk, ki naj zajema:

- opredelitev sodelujočih na posamezni kontrolni točki ter njihove vloge in odgovornosti,
- vsebina (kaj se bo ocenjevalo) in metrika (kaj bo merilo uspešnosti),
- natančno določen fiksen datum kontrolne točke v izvedbenem načrtu projekta,
- določitev vsebin zapisnika o izvedbi kontrolne točke, ki mora zajemati vsaj rezultat kontrolne točke, razloge za morebitna odstopanja od izvedbenega načrta ter sklepe in (morebitne) korektivne ukrepe.

Poleg tega je potrebno v skladu s postopki komuniciranja predvideti tudi, kaj storiti s projektom, če so ugotovitve kontrolne točke, da se stvari ne rešujejo in ne napredujejo v skladu s planom in dogovori. Če kontrolna točka ni uspešna, se to vedno rezultira v planu projekta in privede do povečanja virov projekta (časovni roki, povečana poraba proračunskih sredstev,...).

4.4 Ključne naloge procesov v različnih fazah razvoja IS

4.4.1 Definiranje poslovnih zahtev, prenova poslovnih procesov

Definiranje poslovnih zahtev za nov informacijski sistem je tisti del projekta, kjer naročnik oziroma končni uporabnik svoje potrebe prevede v specifikacije sistema. Proces je sestavljen iz treh povezanih aktivnostih (McConnel, 1998, str.114), in sicer:

- zbiranja informacij s pomočjo intervjujev potencialnih uporabnikov ter pregledovanjem konkurenčnih produktov, izgradnje prototipov in podobno,
- izdelava specifikacij poslovnih zahtev na ustreznem mediju (pisanje dokumentov ali uporaba specialnih orodij za popisovanje poslovnih zahtev),
- analiziranje in prečiščevanje specifikacij v iteracijah.

Cilj procesa je izdelati kakovostne, celovite in pravilno definirane poslovne zahteve, kajti v nasprotnem primeru lahko to postane glavni razlog neuspeha projektov izgradnje novega informacijskega sistema (Vallabhaneni, str.803). Celoten obseg definicij poslovnih zahtev zajema seznam in opise posameznih procesov in funkcij, opis sistemskih vmesnikov, zahteve za kakovost in performančne zahteve. Kakovostne definicije poslovnih zahtev morajo zadoščati naslednjim karakteristikam (Wiegers, 2001, str. 1-3):

- da je opisano vse, kar bo v novem sistemu implementirano,
- da so opisi medsebojno konsistentni,
- da so zahteve enoznačno indeksirane, kar omogoča nedvoumno naslavljanje posamezne specifikacije,
- da so zahteve zasnovane tako, da je možna sledljivost od specifikacij preko načrtov aplikacije do končnega izdelka, testnih primerov, vse v smislu možnosti preverjanj implementacije zahtev.

Cilj prenove poslovnih procesov je zaradi novih poslovnih potreb reorganizacija kritičnih in ključnih poslovnih procesov z namenom povečanja učinkovitosti, zmanjševanja stroškov in povečanja kakovosti storitev. Gre za radikalno spremembo poslovnih procesov, saj naj bi bilo spremenjeno vsaj 80% procesa (v kolikor gre za manjše spremembe na posameznih nalogah, govorimo o izboljšanju poslovnih procesov), sama redefinicija procesa pa lahko traja tudi nekaj let (Vallabhaneni, 1996, str.155). Po zaključku prenove poslovnih procesov je naloga sektorja informacijske tehnologije, da zagotovi ustrezno orodje (informacijski sistem) za podporo poslovnim procesom. Tveganje, da opisi prenovljenih procesov ne ustrezajo zgornjim karakteristikam kakovostne definicije poslovnih zahtev, je izredno veliko. V tem primeru je tradicionalen razvoj novega informacijskega sistema, kjer preteče kar nekaj mesecev, da se pokažejo prvi rezultati izgradnje novega informacijskega sistema, dokaj neprimeren. V kolikor je na voljo ustrezno orodje, ki omogoča na podlagi opisov poslovnih procesov hitro izgradnjo prototipov, s čimer se sproti preverja ustreznost specifikacij, je tveganje za uspešnost prenove poslovnih procesov dosti manjše. Da bo prenova poslovnih procesov uspešna, je treba zagotoviti, da (Butler, 1996, str. 23):

- je vsem sodelujočim jasno, katere procese je potrebno prenoviti,
- vsi razumejo poslovno prednost izboljšane procesa,
- vodstvo organizacije podpira prenavo poslovnih procesov,

- prenovu poslovnih procesov opravljajo analitiki z najširšim znanjem o poslovnih procesih.

Pri opisovanju poslovnih zahtev običajno sodeluje večje število ljudi iz različnih poslovnih področij. Zato je smiselno določiti enotno terminologijo znotraj vseh opisov in že pred začetkom procesa definiranja poslovnih zahtev pripraviti ustrezen terminološki slovar.

Po CDM metodologiji teče proces opisovanja poslovnih zahtev v prvih dveh fazah projekta. Metodologija vsebuje priporočila za planiranje in izvedbo naslednjih aktivnosti:

- izdelava poslovnih modelov podatkov, procesov in funkcij,
- izdelava sistemskih modelov podatkov, procesov in funkcij.

Priporočila vsebujejo poleg osnovnih napotkov za izvedbo aktivnosti še priporočila in standarde za izdelavo posameznega modela ter navodila in standarde za kakovostno uporabo CASE orodja Oracle Designer pri izdelavi posameznih modelov (podrobnosti so opredeljene v Oracle Method: CDM Standards and Guidelines Library, Volume 1 - Requirements Modeling Using Oracle Designer, 2000 in CDM Standards and Guidelines Library, Volume 4, 2000).

Po CDM metodologiji se proces definiranja poslovnih zahtev izvaja na podlagi tako imenovanega plana kakovosti, ki zajema rezultate vseh aktivnosti priprave projekta. Ključni izhodni rezultati procesa so:

- sistemski model procesov,
- sistemski model funkcij in
- sistemski model podatkov.

V zavarovalnici Slovenica se je proces definiranja poslovnih procesov pričel s prenovo poslovnih procesov. Rezultat prenove poslovnih procesov naj bi bil definicija poslovnih zahtev, na podlagi česar bi se zgradil nov informacijski sistem ali nabavil programski paket. V fazi prenove poslovnih procesov CDM metodologija še ni bila vpeljana, prav tako še ni bilo uradne odločitve, s katerimi sistemskimi orodji bo nov informacijski sistem implementiran. Glede na razpoložljivost, tehnične specifikacije in znanje uporabnikov je bila izbrana ARIS metodologija in ARIS orodje kot najustreznejši način za podporo prenovi poslovnih procesov. Opise prenovljenih poslovnih procesov so izdelovali tehnologi ob pomoči končnih uporabnikov in ob svetovanju analitikov informacijske tehnologije. V sklepni fazi izdelave opisov poslovnih funkcij so se tehnologom priključili tudi analitik bodočega izvajalca izgradnje novega informacijskega sistema. Celoten proces je trajal dobro leto dni, še eno leto pa se je na podlagi odprtih vprašanj dopolnjeval in nadgrajeval.

Rezultat aktivnosti prenove poslovnih procesov je bil poslovni model procesov in poslovni model neelementarnih funkcij, katerih opisi so bili izdelani v dokumentih (WORD format). Temu je bil dodan še tabelarni prikaz uporabe elementarnih podatkov na vhodnih in

izhodnih dokumentih posameznih funkcij in opisi poslovnih pravil. Poslovni podatkovni model, podroben funkcijski model in sistemski modeli niso bili izdelani.

Glede na kasnejši razvoj dogodkov na projektu in predvsem na odločitve za uporabo Oracle platforme za izdelavo novega informacijskega sistema, bi bilo na prvi pogled bolje, če bi se za CDM metodologijo in predvsem uporabo orodja Oracle Designer za modeliranje odločili že pred pričetkom aktivnosti prenove poslovnih procesov. S tem bi dobili možnost takojšnjega prototipiranja poslovnih procesov in preverjanja ustreznosti rezultatov prenove poslovnih procesov, kar bi bilo v skladu z zgoraj opisanimi postopki za zmanjševanje tveganj pri prenovi poslovnih procesov. Vendar pa CDM metodologija ne vsebuje priporočil za prenavo poslovnih procesov (ravno nasprotno: predlaga, da če prenova poslovnih procesov še ni izvedena, je pa planirana, da se izvede pred projektom izgradnje novega informacijskega sistema (Oracle Method: CDM Standards and Guidelines Library, Volume 1,2000, str2-3)). Z odločitvijo za ARIS metodologijo in ARIS orodje za prenavo poslovnih procesov smo pridobili naslednje:

- uporabo ARIS metodologije za prenavo poslovnih procesov in ustrezno informacijsko podporo enega najkakovostnejših orodij za prenavo poslovnih procesov in s tem zmanjšano tveganje uspešnosti prenove zaradi neizkušenosti ali neznanja projektne ekipe,
- glede na predhodno znanje in izkušnje z uporabo orodja smo izdelali kakovostnejše rezultate v smislu zmanjševanja tveganja zaradi nepoznavanja tehnologije,
- možnost povezljivosti modelov v ARIS-u z različnimi informacijskimi tehnologijami, s pomočjo katerih bi se nadaljevala implementacija poslovnih pravil in tako ohraniti sledljivost funkcij tudi v načrtih in izgradnji programske opreme,
- možnost indeksiranja posameznih funkcij in sledljivost funkcij v procesnih modelih,
- poslovni model, neodvisen od tehnologije (dandanes v času dveh let tehnologija toliko napreduje, da bi bil model ob zaključku že zastarel; ker je model izdelan z ARIS orodjem, ga je možno implementirati z najnovejšo tehnologijo.).

Neuporaba CDM metodologije (ali ustrezne druge kakovostne metodologije za fazo definicije in analize projekta, ARIS je vendarle samo metodologija prenove poslovnih procesov, kar pa ne zadošča za kakovostno pripravo na izvedbo projekta) v procesu definiranja poslovnih zahtev se je izražala tudi v nenatančni opredelitvi izhodnih rezultatov aktivnosti, ki morajo biti izvedene pred načrtovanjem novega informacijskega sistema, zaradi česar je bilo postavljeno slabše izhodišče za vse nadaljnje aktivnosti projekta. Poslovni modeli procesov in funkcij se niso nikdar v nadaljevanju projekta nadgradili s sistemskimi modeli (problemi bodo podrobneje opisani v naslednjih poglavjih, predvsem v 4.4.4 in 4.4.5), medtem ko poslovni model podatkov ni bil nikdar izdelan, temveč se je v procesu načrtovanja in izgradnje podatkovne baze izdelal samo sistemski model podatkov, kar bo podrobneje predstavljeno v poglavju 4.4.4. Kot velika pomanjkljivost na projektu se

je pokazala tudi neizdelava podatkovnega slovarja pred pričetkom procesa definiranja poslovnih zahtev. Poskusi izdelave slovarja so sicer naknadno bili (na njegovo potrebo je opozorila ravno CDM metodologija), vendar bi bilo prečiščevanje vseh opisov poslovnih funkcij in vseh modelov prezahtevno in neracionalno glede porabe virov (čas, ljudje), tako da slovar ni nikdar zares zaživel. Posledica terminoloških nekonsistentnosti pa je dostikrat v kasnejših aktivnostih projekta bilo vračanje k dopolnjevanju opisov.

V času priprave projekta je izvedbeni načrt zajemal tudi predlog aktivnosti izdelave prototipov za posamezne prenovljene poslovne procese, ki pa ni bil nikoli sprejet. Morda je v tem primeru vodstvu projekta zmanjkal tudi kakšen argument v prid prototipiranju, ki pa bi ga lahko našel ravno v ustreznih metodologijah (kot ustrezno rešitev ga med drugim navaja tudi Butler, 1996, str.19).

4.4.2 Pregled obstoječega sistema

Izvajanje procesa pregled obstoječega sistema je primerno samo v primeru, da gre za zamenjavo obstoječega informacijskega sistema z novim. Cilj aktivnosti je, da se v grobem pogleda, katere procese in funkcije opravlja obstoječi sistem, s katerimi drugimi sistemi je povezan (predvsem katere povezave se bodo ohranile tudi z novim sistemom) in pridobitev podrobnih informacij o podatkovnem modelu obstoječega sistema v primeru načrtovanja prenosa in pretvorbe podatkov iz obstoječega v nov informacijski sistem. Smiselno je, da se aktivnost izvrši v najzgodnejši fazi življenjskega cikla projekta, saj so ugotovitve tega procesa lahko osnova tako glede planiranja aktivnosti kot porabe virov (časa, ljudi) pri pripravi izvedbenega načrta projekta.

CDM metodologija ima aktivnost in priporočila razdeljeno v naslednje sklope nalog:

- opredelitev obstoječe organizacijske strukture glede na geografsko in fizično lokacijo, katere namen je pridobitev organizacijske sheme vseh, ki bodo v kakršnikoli zvezi z novim informacijskim sistemom,
- dokumentiranje procesnega modela obstoječega sistema z namenom pridobitve seznama aktivnosti, na katere mora sistem odgovoriti,
- zbiranje podrobne dokumentacije o obstoječem podatkovnem modelu (predvsem za potrebe prenosa in pretvorbe podatkov),
- dokumentiranje obstoječih sistemskih vmesnikov s poudarkom na tistih, ki bodo v interakciji tudi z novim informacijskim sistemom,
- dokumentiranje obstoječe tehnične arhitekture, omrežja in operativnih postopkov, v kolikor bodo uporabljeni tudi za nov informacijski sistem.

Namen aktivnosti je zbrati ustrezno dokumentacijo in nikakor ne njeno ponovno izdelovanje. Problem lahko nastopi le v primerih, ko ustrezna dokumentacija o obstoječem sistemu ne obstaja.

Predpostavke, na katerih je temeljilo izvajanje aktivnosti procesa pregleda obstoječega sistema v zavarovalnici Slovenica, so bile:

- podatkovni modeli obstoječih informacijskih sistemov so bili shranjeni v repozitoriju Oracle Designer-ja,
- predvidevala se je prenova organizacijske strukture,
- izdelan je bil opis obstoječih poslovnih procesov (eden od izdelkov prve stopnje projekta, to je prenove poslovnih procesov).

Dodatno je bila v skladu s CDM priporočili izvedena samo naloga pregled obstoječih sistemskih vmesnikov.

Z izvedbo te aktivnosti v skladu s CDM priporočili v prvi fazi projekta smo zmanjšali tveganje neustreznega načrtovanja in izgradnje baze podatkov in prenosa in pretvorbe podatkov, saj so bili vhodi za ta dva procesa pravočasno in kakovostno pripravljene.

4.4.3 Tehnična arhitektura

Na vsakem projektu izgradnje novega informacijskega sistema je potrebno ustrezno zasnovati tudi tehnično arhitekturo sistema. Potrebno je definirati strojno in programsko opremo okolja, v katerem bo tekel razvoj nove aplikacijske programske opreme (na primer sistemska programska oprema, razvojna orodja, strojna oprema in omrežje za potrebe razvoja,...), ter vse ostale specifikacije, ki bodo omogočale nemoteno uporabo izdelane programske opreme v produkciji (ustrezen sistem za upravljanje s podatki, postopki za varnost, nadzor, arhiviranje in obnovitev sistema, definiranje potreb za rezervno lokacijo informacijskega sistema, distribucijska arhitektura, in drugo).

CDM metodologija vse te aktivnosti združuje v sklopu procesa definiranja tehnične arhitekture, ki teče pretežno v prvih dveh fazah izgradnje informacijskega sistema, ki jo sestavljajo naslednji sklopi aktivnosti:

- določitev potreb ustrezne strojne in programske opreme in plan kapacitet za vsako posamezno okolje (razvojno, testno, izobraževalno in produkcijsko okolje) ter predvideno oceno stroškov,
- definiranje operativnih procesov posebej za čas razvoja in za produkcijo, in sicer postopke zagona in ugašanja operacijskega sistema in podatkovne baze, nadzor dostopov, zaščite, arhiviranja, postopke obnove ob zrušitvi, varnost in nadzor,
- določitev distribucijske arhitekture, to je fizične lokacije izvršilne kode in podatkov,
- določitev performančnih zahtev in kritičnih funkcij oziroma modulov s stališča performančnih zahtev.

Poleg teh aktivnosti po metodologiji spadajo v proces tehnične arhitekture tudi aktivnost definiranja sistemskih vmesnikov za povezovanje aplikacije z drugimi aplikacijami v

podjetju, ki je osnova tako za pripravo ustrezne tehnične arhitekture kot tudi za zasnovu podatkovnega modela, izdelavo aplikacijske kode, strategijo tranzicije ter prenosa in pretvorbe podatkov, ter aktivnost izdelave standardov uporabniškega vmesnika, kjer se opredeli uporabniški izgled aplikacije.

Na projektu izgradnje novega informacijskega sistema v zavarovalnici Slovenica je ves razvoj nove programske opreme, to je izgradnje podatkovne baze in izdelave aplikacijske kode, tekel pri zunanjem izvajalcu (outsourcing). Tako je tudi večina aktivnosti v zvezi s tehnično arhitekturo bila stvar organizacije dela izvajalca (predvsem v zvezi s postavitvijo ustreznega razvojnega okolja). Na podlagi razvojnih izkušenj se je potem skupaj s strokovnjaki na naročnikovi strani določila arhitektura testnega in produkcijskega okolja. Zaradi pravočasne priprave seznama tehnične opreme smo se tudi izognili nepredvidenim zastojem zaradi nenabavljene opreme, saj vemo, da je predvsem dobavni rok za strežnike lahko kar nekaj mesecev (potrebujemo pa jih že pri testiranju). Prav tako je glede na performančne zahteve novega informacijskega sistema, določene že ob podpisu pogodbe med naročnikom in izvajalcem, bila naloga izvajalca, da z ustrezno zasnovano podatkovne baze in načrta aplikacije doseže zahtevano performančnost novega informacijskega sistema.

Posebno poglavje na projektu je predstavljalo definiranje distribucijske arhitekture ter varnost in nadzor nad podatki, predvsem s stališča združevanja dveh zavarovalnic in s tem povezanega portfelja obeh zavarovalnic. Stopnjo zaščite podatkov mora določiti najvišje vodstvo obeh organizacij, problem pa se je reševal skupno z definiranjem načina tranzicije in prenosa in pretvorbe podatkov.

Ključni nalogi, ki sta se izvajali na strani naročnika, pa sta bili določitev standardov uporabniškega vmesnika in definiranje sistemskih vmesnikov.

Aktivnost določitev standardov uporabniškega vmesnika je bila ravno zaradi uporabe CDM metodologije časovno izredno nezahtevna. CDM metodologija (Oracle Method: CDM Standards and Guidelines Library, Volume 4 – CDM Standards, 2000) prinaša poleg ostalega tudi niz standardov, ki opredeljujejo izgled bodočega uporabniškega vmesnika (oblike menija, orodne vrstice, upravljanje in izgled posameznih oken, način navigacije po ekranu, upravljanje z različnimi vrstami sporočil). Ker so bodoči uporabniki imeli že nekaj izkušenj z uporabo Oracleovih aplikacij, katerih standardi uporabniškega vmesnika se nekoliko razlikujejo od standardov v Windows okolju, je bila pomembna novost zadnje verzije (6.0.0) Oracle standardov, ki smo jo uporabili tudi na projektu, možnost izbire enega, drugega ali kombinacijo obeh standardov. Prednost uporabe Oracle standardov je tudi v tem, da so ob vsakem standardu navodila, kako v Oracleovem orodju (Designer in Developer ter Headstart) globalno nastaviti implementacijo standarda. Na tak način je poenostavljen tudi nadaljnji nadzor uporabe dogovorjenih standardov pri razvoju, izvajalci pa so razbremenjeni skrbi za uporabo standardov, saj je njihova uveljavitev avtomatična oziroma orodje sproti opozarja na neustrezno rabo standardov. Tako je bila naloga naročnika:

- predstaviti prototipe različnih uporabniških vmesnikov bodočim končnim uporabnikom in v skupni kritični presoji najti najustreznejši izgled bodoče aplikacije,
- na podlagi izbire iz prejšnje alineje izbrati in dokumentirati ustrezne standarde in skupaj z izvajalcem poskrbeti, da se aplicirajo.

Druga aktivnost, to je definiranje sistemskih vmesnikov, pa je bila mnogo zahtevnejša. Sistemski vmesniki, ki so že obstajali v starem informacijskem sistemu in se naj bi v novem ohranili, so bili že zadovoljivo dokumentirani znotraj procesa pregled obstoječega sistema. Vzporedno s projektom razvoja novega informacijskega sistema za poslovne procese pa je tekkel tudi projekt prenove informacijskega sistema za finančno računovodske procese. Zaradi tega je bila izredno zahtevna opredelitev sistemskega vmesnika med obema sistemoma (ki je kasneje zaradi svoje obsežnosti postal podprojekt obeh omenjenih projektov). Zaradi vplivanja opisa sistemskih vmesnikov na druge, zgoraj naštetih aktivnosti, je izredno pomembno, da znamo kar najkvalitetneje opredeliti vse parametre bodočega sistemskega vmesnika, dostikrat pa niti ne vemo, če so vsi parametri opisani. Tu metodologija s svojimi koraki za postopno definiranje vmesnika usmerja iz grobega v podrobni opis sistemskega vmesnika in s tem olajša delo načrtovalca. Metodologija sicer zajema nalogo le enkrat, medtem ko je v našem primeru bilo potrebnih kar nekaj iteracij za razčiščevanje posameznih parametrov novega sistemskega vmesnika.

Poleg tega je obstajal še problem, ker je izdelava novega informacijskega in sistemskega vmesnika tekla pri dveh različnih izvajalcih. Glede na dogovor, da se uporabi ista tehnologija, sta oba izvajalca lahko uporabila iste standarde in dosegla enoten izgled aplikacij. Predvidevalo se je namreč, da bo določen krog končnih uporabnikov uporabljal obe aplikaciji, z enakim izgledom aplikacije pa bi se poenostavilo tudi njihovo delo.

4.4.4 Načrtovanje in implementacija baze podatkov

Proces načrtovanja in implementacija podatkovne baze je logično nadaljevanje razvoja poslovnega modela podatkov, izdelanega v fazi analize. Izdelki procesa so:

- logični podatkovni model, ki vsebuje tudi informacije o predvideni količini podatkov, prioriteto in način dostopa do podatkov (kar je pomembno za kasnejši fizični podatkovni model) ter omejitve glede performans, integritete, varnosti in zaupnosti (Vallabhaneni, 1996, str. 848-849),
- fizični podatkovni model, to je implementacija logičnega podatkovnega modela v ustrezen sistem za upravljanje s podatki (DBMS),
- indeksi,
- postopki za validacijo podatkov (da so v sistem vneseni pravilni podatki, da se pravilno procesirajo, da so procesirani podatki pravilno izpisani),

- postopki za nadzor procesov, ki zagotavlja, da so skupine zapisov ustrezno vnesene v sistem in ustrezno procesirane (na primer grupne funkcije za kontrolo pravilnosti),
- zasnova revizijskih sledi (sledenje transakcijam skozi sistem, sledenje izvornim dokumentom skozi zaporedne nivoje zbiranja, identifikacija sprememb na sistemskih parametrih in tabelah, ki vplivajo na obdelave in ponovne obdelave katerihkoli finančnih transakcij, spremljanje dostopov do sistema in njihovo uporabo, zapisovanje uspešnih in neuspešnih prijav, prepoznavanje uporabnikov, datum in čas prijave in odjave, oznako delovne postaje in shranjevanje podatkov in sprememb daljše časovno obdobje za potrebe kasnejših raziskav in kontrol).

CDM metodologija predlaga pet aktivnosti za načrtovanje in izgradnjo podatkovne baze, ki se načeloma izvajajo v fazah načrtovanja in implementacije, in sicer:

- izdelava logičnega podatkovnega modela na podlagi systemskega funkcijskega in podatkovnega modela ter standardov načrtovanja in implementacije, izdelanih v procesu načrtovanja in izgradnje modulov,
- izdelava načrta indeksov,
- izdelava avtorizacijske sheme za bazne objekte na podlagi strategije zagotavljanja varnosti in nadzora (opredeljeno v procesu tehnična arhitektura) ter načrta modulov (opredeljeno v procesu načrtovanje in izdelava aplikacije),
- izdelovanje fizičnega modela baze podatkov ter
- kreiranje datotek (DDL) za izdelavo produkcijske baze in njenih objektov.

Proces izdelave načrta in implementacije podatkovne baze na Slovenici je temeljil na dejstvu, da v procesu definiranja poslovnih zahtev poslovni podatkovni in funkcijski model nista bila izdelana. Zato smo v ta sklop nalog dodali tudi izdelavo systemskega podatkovnega modela (po CDM aktivnost procesa definiranja poslovnih zahtev), ki naj bi se izdelal na podlagi opisov poslovnih procesov in načrtov arhitekture novega sistema, medtem ko je bila izgradnja systemskega funkcionalnega modela (po CDM prav tako aktivnost procesa definiranja poslovnih zahtev), katerega del se implementira tudi skozi logični podatkovni model, predstavljena v aktivnost načrtovanja in izdelave aplikacije (ta aktivnost se kasneje ni izvedla, razlogi bodo opisani v naslednjem poglavju).

Načrtovanje in izdelavo baze podatkov je v celoti prevzel izvajalec.

Tako je naročnik, to je zavarovalnica, z uporabo CDM metodologije pridobil:

- nadzor nad potekom izvajanja aktivnosti načrtovanja in izgradnje baze podatkov,
- izvedba procesa v skladu z dogovorjenimi in sprejetimi standardi načrtovanja in preverjanje uporabe standardov (z orodjem Headstart Utilities izdelava poročil o ustrezni uporabi CDM standardov),

- glede na to, da je bilo za načrtovanje in izdelavo baze podatkov izbrano orodje Oracle Designer, je v njegovem repozitoriju bila zbrana vsa dokumentacija celotnega razvoja baze podatkov (ustrezen entitetno relacijski diagram podatkov, z vsemi komentarji in opisi, logični podatkovni model, načrt indeksov).

Žal CDM v sklopu načrtovanja in implementacije podatkovne baze ne vključuje tudi aktivnosti za implementacijo revizijske sledi v podatkovno bazo (del aktivnosti se izvrši v procesu načrtovanja in izgradnja aplikacije). V našem primeru smo med standarde načrtovanja dodali, da mora imeti vsak zapis v bazi 4 tako imenovane revizijske kolone (datum in uporabniško ime vnašalca in popravljalca) ter da se mora izključiti možnost brisanja zapisov.

Glede na to, da je celoten razvoj podatkovnega modela bil izveden na strani izvajalca, se je pri naročniku zastavljalo vprašanje ustreznosti zasnove glede na poslovne potrebe. V ta namen smo izvedli nekatere kontrolne aktivnosti, in sicer:

- predstavitev ključnih konceptov, upoštevanih pri zasnovi podatkovnega modela, za zagotovitev ustreznosti glede na ključne poslovne potrebe ključnim tehnologom, analitikom, administratorjem podatkovne baze in administratorjem podatkov,
- izvedba poskusnega vnosa in kontrola odzivnosti na različne transakcije (vnos, popravljanje, brisanje in iskanje) sistema na masovnih podatkih po eni ali več povezanih tabelah.

4.4.5 Načrtovanje in izgradnja aplikacije

Načrtovanje in izgradnja aplikacije je najobsežnejši in najzahtevnejši proces faze načrtovanja in izgradnje novega informacijskega sistema, ki skupaj obsegata kar 75% celotnega življenjskega cikla projekta. Glede na to, da so lahko procesi prenove informacijskih sistemov izredno dolgi (Finkelstein, 1992, omenja tudi 3-5 let), je torej takšna faza lahko dolga tudi 2-4 leta. Res, da je celotna faza izdelave aplikacije razbita na dva dela, to je načrtovanje in izgradnja, vendar še vedno predolgo obdobje, da bi vmes imeli samo eno kontrolno točko. Poleg tega v mnogih primerih zaradi časovne stiske obe aktivnosti tečeta vzporedno. Zato se v tem primeru nikakor ni smiselno držati tradicionalnega življenjskega cikla projekta, temveč razbiti faze na manjše, smiselne celote in potem za vsako posamezno celoto izvršiti cikel: načrtovanje, izgradnja, testiranje, ocena kakovosti in ustreznosti, v kar ima smiselno vključiti tudi konverzijo (o tem več v poglavju o prenosu in pretvorbi podatkov in testiranju).

Cilj procesa je, da se izdelata načrt celotne aplikacije, to je struktura programa, menijev, vse do oblik in vsebin posameznih ekranskih obrazcev in poročil, nato pa se dejansko izdelata (kodira) posamezen element aplikacije (meni, ekranski obrazec, poročilo). Priporočljivo je, da se fazi izvajata striktno zaporedno z vmesno kontrolno točko, saj je v primeru vzporednega načrtovanja in implementacije ugotavljanje in odpravljanje napak, storjenih v fazi načrtovanja in ugotovljenih šele v fazi implementacije, izredno zahtevno, zamudno in

drago opravilo. Možna je tudi varianta, ko se celoten načrt aplikacije razdeli na smiselne, zaključene celote. V tem primeru se celoten cikel (načrtovanje, preverjanje, implementacija, testiranje ter prenos in pretvorba podatkov) ponovi za vsak smiseln sklop posebej. V takšnem primeru lahko pričnemo z načrtovanjem drugega sklopa, ko je načrt prvega sklopa ustrezno potrjen in preide v implementacijo.

CDM poleg same izdelave načrtov in kode aplikacije zajema še pripravo standardov, ki pa jih lahko izdelamo že pred načrtovanjem in izgradnjo, kakor hitro je sprejeta odločitev o tehnologiji izvedbe, kar pa je lahko že v času priprav specifikacij novega sistema.

Aktivnosti načrtovanja in izgradnje aplikacije po CDM zajemajo:

- izdelavo standardov načrtovanja in izgradnje (pomembnih tudi za načrtovanje in izdelavo baze podatkov),
- izdelavo tehnične in funkcionalne dokumentacije,
- izdelavo strukture menijev,
- izdelavo načrta aplikacije,
- izdelavo pripomočkov za revizijo,
- pripravo razvojnega okolja,
- izdelavo kode aplikacije.

Na projektu razvoja informacijskega sistema v zavarovalnici Slovenica je celoten proces načrtovanja in izgradnje vodil izvajalec, skupine izvajalcev pa so sestavljali strokovnjaki naročnika in izvajalca.

S strani naročnika bi bilo smiselno, da zahteva postopen pristop k izdelavi aplikacije, kot je opisano zgoraj, saj omogoča boljši nadzor nad potekom aktivnosti.

V procesu definiranja poslovnih zahtev se je z ARIS orodjem izdelal funkcijski model prenovljenih poslovnih funkcij. CDM metodologija opredeljuje funkcijski model kot model tistih poslovnih pravil, ki niso implementirana s podatkovnim modelom (Oracle Method: CDM Standards and Guidelines Library, Volume 1, 2000, str. 5-4) . Funkcijsko drevo, izdelano z ARIS orodjem, ni imelo elementarnih funkcij. Nadaljnji problem je bil uporabnost funkcijskega modela. Orodje Oracle Designer omogoča izdelavo funkcijskega modela, vendar v času projekta še ni obstajala možnost za opisovanje poslovnih funkcij znotraj modela s pomočjo psevdokode, in orodja, ki bi omogočal pretvorbo psevdokode v prožilce. Tako je bila tudi zasnovana ideja CDM Rule Frame za nivo poslovne logike novega informacijskega sistema (Oracle Method: CDM Standards and Guidelines Library, Volume 2, 2000, str. 6-1 do 7-55), vendar produkt v času projekta še ni bil na tržišču. Torej bi navkljub izdelavi funkcijskega modela kodiranje poslovne logike še vedno ostalo ročno in ne bi obstajala neposredna povezava med funkcijskim modelom in kodo. Glede na to, da smo tudi procesne diagrame pustili v ARIS-u, se je izgubila sled med opisi poslovnih procesov in dejansko implementacijo. Del problema povezljivosti obeh sistemov

(predvsem za čas vzdrževanja sistema) bi lahko rešili z definiranjem natančno določenih zaporednih postopkov spreminjanja poslovnih procesov in aplikacije.

Prednosti uporabe CDM metodologije pri procesu načrtovanja in izgradnje aplikacije na projektu izgradnje poslovnega informacijskega sistema v Slovenici so bili:

- izdelava in uporaba standardov,
- uporaba prototipiranja z uporabo orodja Oracle Designer in s tem povezano hitrejše odzivanje bodočih uporabnikov na probleme načrtovanja in izdelave aplikacije,
- z uporabo orodja Headstart Utilities smo dosegli enostavnejšo in avtomatsko kontrolo implementacije Oracleovih standardov, pohitritev razvoja ter avtomatizirali izdelavo poročil o kakovosti izdelka.

Glede na obsežen sistem sta tako naročnik kot izvajalec imela precej težav pri planiranju potrebnega časa za posamezne aktivnosti načrtovanja in izgradnje aplikacije, predvsem z vidika, da ni bilo nikakršne ocene, kolikšen bo obseg aplikacije. Tudi v tem primeru bi bilo smiselno razbitje aplikacije na podsklope. Ko je prvi sklop zaključen (načrtovan in implementiran), se glede na porabljen čas in predviden obseg ostalih sklopov v primerjavi z že izdelanim naredi revizija izvedbenega načrta, ki da mnogo konkretnije rezultate.

4.4.6 Prenos in pretvorba podatkov

Pri implementaciji novega programskega paketa prav posebno mesto zavzema proces prenosa in pretvorbe podatkov iz obstoječega v nov informacijski sistem. To je izredno zahtevna in dolgotrajna aktivnost, ki mora biti opravljena pred produkcijsko instalacijo nove programske opreme.

Postopek prenosa in pretvorbe podatkov zajema naslednje stopnje (Vallabhaneni, 1996, str. 894):

- na podlagi analize določitev potreb za prenos in pretvorbo podatkov za obstoječe podatke,
- določitev strategije prenosa in pretvorbe glede načina prehoda na drug sistem, oblike trenutnega hranjenja podatkov in načina izvedbe (ročni ali avtomatski prenos),
- določitev pravil in odgovornosti za prenos in pretvorbo,
- izdelava časovnega razporeda izvajanja procesa,
- izdelava in testiranje programov prenosa in pretvorbe,
- izvajanje programov prenosa in pretvorbe in ročnega vnosa podatkov,
- ugotavljanje veljavnosti podatkov v novem sistemu, pravilnosti formata in točnosti.

Trajanje konverzije podatkov je odvisno od kompleksnosti novega sistema in od zasnove starega sistema. Če je bila stara programska oprema nabavljena pri različnih proizvajalcih

in ni bila povezana v celoto, če je instalirana na različni strojni opremi, uporablja različne baze podatkov in podobno, bodo postopki konverzije podatkov v nov sistem verjetno zapleteni in dolgotrajni.

CDM opredeljuje naslednje naloge:

- izdelava plana konverzije, ki je zajet v treh posameznih aktivnostih, in sicer izdelava seznama podatkov, za katere se zahteva prenos, ter opis posebnosti pri pripravi vsakega podatka (ali je potrebno predhodno čiščenje podatkov, validacija podatkov), opis strategije izvedbe prenosa ter izvedbeni načrt prenosa podatkov,
- načrtovanje, izgradnja in testiranje skript za prenos in pretvorbo podatkov,
- izvedba in verifikacija prenosa in pretvorbe podatkov.

Po CDM metodologiji se proces izvaja na podlagi:

- dokumentacije o obstoječem sistemu, predvsem na podrobnih opisih podatkovnih modelov obstoječih sistemov,
- arhitekturne zasnove novega sistema,
- logičnega podatkovnega modela novega sistema,
- strategije prehoda na nov sistem (za planiranje časovnega poteka in načina prenosa podatkov).

V nadaljevanju bom pojasnila tveganja, ki smo jih z uporabo CDM metodologije zmanjšali.

Pri izdelavi plana konverzije smo uporabili rezultate aktivnosti proučevanja obstoječega sistema, predvsem dokumentacijo o podatkovnem modelu obstoječih sistemov, katerih podatki bodo preneseni v nov sistem. Ker je bila aktivnost proučevanja obstoječega sistema že v izvedbenem načrtu in definirana med prvimi aktivnostmi projekta, nismo porabili dodatnega projektnega časa z raziskovanjem le-tega.

Kakovost podatkov v obstoječih sistemih vpliva na kakovost procesa pretvorbe in prenosa podatkov. Zaradi pravočasnega planiranja smo že zelo zgodaj vedeli, kateri podatki se bodo prenašali, zato je bilo zadosti časa za čiščenje podatkov. Glede na to, da so prihajali podatki iz več različnih sistemov, je bilo čiščenje še dodatno oteženo (na primer podatki o strankah v več različnih sistemih je potrebno predhodno prečistiti, da po združitvi ena stranka nastopa samo enkrat).

V nekaterih primerih se je izkazalo, da je potrebno prenesti iz starega v nov sistem tudi podatke, za katere ni obstajala preslikava. Ker je bilo potrebno stare podatke vendarle ohraniti, je bilo potrebno v skladu z obstoječimi podatki spremeniti tudi načrt baze podatkov. Ker smo te probleme zaznali že v procesu planiranja projekta, se je to pri načrtovanju baze ustrezno upoštevalo. V primeru, da bi to ugotovili šele ob zaključku izgradnje baze podatkov, bi bili stroški popravkov izredno visoki.

Proces prenosa in pretvorbe podatkov je smiselno izvajati vzporedno z načrtovanjem baze podatkov ter sproti izvajati testne prenose podatkov v smislu spremljanja porabljenega časa za izvedbo dejanskega masovnega prenosa podatkov. Lahko se namreč zgodi, da so kakšni prenosi izredno dolgotrajni. V kolikor gre za prenose dnevno občutljivih podatkov, je potrebno vsekakor to upoštevati pri načrtovanju aktivnosti prehoda na nov sistem ali, če ne gre drugače, tudi spremeniti načrt nove baze podatkov.

4.4.7 Testiranje

Proces testiranja se izvaja z namenom, da bo programska oprema delovala zanesljivo, zadovoljila funkcionalne in tehnične zahteve ter točne rezultate obdelav. Uspešno testiranje vključuje načrtovanje testiranja, izvajanje testiranja ter ocenjevanje rezultatov testiranja na podlagi predhodno določene metrike uspešnosti testiranja.

Izbiramo lahko med več metodami testiranja, med katere najpogosteje sodi (Vallabhaneni, 1996, str. 887):

- performančno testiranje glede na predefimirane parametre,
- testiranje funkcionalnosti izdelka (Black-Box),
- testiranje interne strukture in logike sistema (White-Box),

Testiranje programskega paketa je sestavljeno iz naslednjih vrst testiranja (Black, 1999, str. 2-10):

- testiranje posameznih enot,
- testiranje skupine enot oziroma podsistema,
- testiranje izdelka in integracijski testi,
- prevzemno testiranje.

Pri definiranju plana testiranja je potrebno odgovoriti na vprašanja, kot so:

- kaj vse bi lahko testirali oziroma določiti meje testiranega sistema,
- kaj je potrebno testirati, s čimer določimo področja novega sistema, ki lahko vplivajo na kakovost končnega izdelka, ter
- kaj se dejansko lahko testira glede na razpoložljive vire, kot so čas, denar in razpoložljivost testne ekipe.

CDM metodologija vsebuje priporočila za planiranje in izvedbo naslednjih aktivnosti:

- izdelavo plana testiranja, kjer se definira vrste, metode in obseg testiranja, način izvajanja testiranja (uporaba testnih orodij ali ročno testiranje), testna ekipa, testno okolje in delovni procesi testiranja,

- pripravo plana in izvedbo testiranja posameznih enot (ekranska maska s pripadajočimi procedurami ali poročilo) ter testiranje skupine enot oziroma podsistema v skladu s funkcionalnimi specifikacijami sistema,
- pripravo testnih skript in izvedbo testiranja sistema, kjer testna skupina testira skladnost procesov novega informacijskega sistema z opisi poslovnih procesov,
- testiranje integracije sistema, kjer se testira povezljivost novega informacijskega sistema z morebitnimi preostalimi informacijskimi sistemi,
- prevzemno testiranje sistema, kjer se izvede zaključni prevzemni test v skladu z definiranjem prevzemnega testa v planu projekta.

Po CDM metodologiji so med najpomembnejšimi izdelki, na podlagi katerih se izvaja proces testiranja, predvsem:

- funkcijski model ter načrt aplikacije, na podlagi česar se izdelajo plani testiranja posameznih enot in skupine enot,
- procesni model sistema, na podlagi česar se določijo plani sistemskih, integracijskih in prevzemnih testov,
- specifikacija zahtev za systemske vmesnike, na podlagi česar se določijo integracijski testi.

Spodaj je navedenih nekaj dejstev, ki so ključna za razumevanje poteka procesa testiranja na zavarovalnici Slovenica:

- obseg programskega paketa, namenjenega testiranju, naj bi obsegal približno 800 modulov in 9 vmesnikov do drugih informacijskih sistemov, med katerimi je najobsežnejši bil v nastajanju v vzporednem projektu,
- izvajalec je vodil aktivnosti načrtovanja in izgradnje, dejanski izvajalci posameznih aktivnosti načrtovanja in izgradnje pa so bili mešane ekipe tako naročnika kot izvajalcev.

S temi predpostavkami so se v nadaljevanju definirali vsi postopki testiranja.

Kot pri vseh ostalih procesih je bila tudi za proces testiranja na projektu v zavarovalnici Slovenica kot osnova CDM metodologija. V nadaljevanju navajam nekaj prednosti in slabosti pri uporabi metodologije in opisujem načine reševanja posameznih problemov.

Osnovna prednost uporabe metodologije je bila predvsem v njenem zgodnjem aktiviranju procesa testiranja, saj je celotna faza analize namenjena izdelavi plana testiranja, kar omogoča pravočasno pripravo vseh postopkov za izvajanje testiranja (obseg testiranja, sestava testne ekipe, postavitve testnega okolja, določitev postopkov testiranja, določitev postopkov upravljanja z napakami).

Vsi potrebni vhodni dokumenti za testiranje so že zajeti v izvedbenem načrtu projekta, kot ga priporoča CDM in v kolikor se projekt tudi izvaja v skladu z izvedbenim načrtom,

imamo pripravljeno vso dokumentacijo, ki jo pri testiranju potrebujemo (procesne in funkcionalne specifikacije sistema, tehnična in uporabniška dokumentacija sistema, standardi, dokumentacija o sistemskih vmesnikih, ...).

V kolikor za delo na projektu po CDM metodologiji uporabimo tudi orodja, ki jih metodologija pri posamezni aktivnosti priporoča, dobimo popolno sledljivost od opisov poslovnih procesov preko načrtov sistema do končnih izdelkov aplikacije in posledično sprotno izdelavo tehnične in uporabniške dokumentacije.

Z uporabo standardov načrtovanja in izdelave v skladu s CDM standardi in priporočili ter hkratno uporabo razvojnega orodja Oracle Designer, je s pomočjo orodja Oracle Headstart-Quality Check možno izdelati poročila o kakovosti izdelka. S tem se občutno skrajša čas testiranja kakovosti končnega izdelka.

Testna ekipa je bila mešana (člani naročnika in izvajalca). Projektna organizacijska struktura je bila postavljena v skladu s CDM metodologijo. Zato je bila testna skupina najprej odgovorna vodji izvedbe (torej izvajalcu) šele nato vodji projekta (naročniku). Pri tem je potrebno vedeti, da je CDM pisan za primer, da se celoten projekt izvaja znotraj programske hiše, od koder je tudi vodja projekta. V primerih projektne organizacije, ko je izvajalec dejansko zadolžen samo za izvedbo, naročnik pa za testiranje, je bolje sestaviti dve testni skupini, in sicer:

- testna skupina na strani izvajalca, ki je podrejena vodji izvedbe/izvajalca in katere namen je testiranje interne strukture in logike sistema ter funkcionalno testiranje,
- testna skupina na strani naročnika, ki testira skladnost izdelanega sistema s specifikacijami (funkcijskimi in procesnimi modeli) in zahtevami za kakovost.

CDM metodologija ne predvideva posebne aktivnosti za postavitve testnega okolja, testna ekipa pa je imela na začetku dober mesec dni nepredvidenega dela s postavitvijo testnega okolja. Zato smo v izvedbeni načrt testiranja testiranja dodali novo aktivnost, in sicer verificiranje postavitve testnega okolja (Build Verification Test, BVT). Namen naloge je bil preveriti, da vsak testni paket vsebuje (Hayes, 2001, str. 1-3):

- navodilo za postavitve testnega okolja,
- vse potrebne komponente,
- da so komponente ustreznih verzij, medsebojno povezljive, dosegljive in funkcionalne.

Takšen test je vzel le nekaj minut časa, prihranil pa je tedne brezplodnega dela pri pripravi testnega okolja.

Prva aktivnost testiranja je bila testiranje enot in skupine enot. Osnovno strukturno testiranje posameznih modulov naj bi izvršila razvojna skupina sama, testna skupina pa naj bi se posvetila testiranju funkcionalnosti in kakovosti modulov. Testiranje se je izvajalo v skladu s CDM priporočilom vzporedno z izgradnjo posameznih programskih enot. Zaradi

pogodbenega dogovora, da ves razvoj teče z orodjem Oracle Designer in da se celoten repozitorij preda naročniku, ki sam iz njega generira produkcijsko okolje, ni bilo možno prenašati posameznih programskih enot iz razvojnega v testno okolje, temveč vedno celoten repozitorij. Zato je bilo dogovorjeno, da se v rednih 14 dnevnih cikliih testno okolje nadgrajuje z novo verzijo iz razvojnega okolje. Aktivnost prenosa iz razvojnega v testno okolje je bilo potrebno dodati v izvedbeni načrt projekta, saj je aktivnost vsakič trajala 1 dan.

Zaradi obsežnosti aplikacije in planiranih zelo obsežnih sistemskih testov (tudi 100 in več ponovitev posameznih procesov na različnih kritičnih podatkih) ter pomanjkanja virov (časa in ljudi) se je raziskovalo možnost vpeljave avtomatskega testnega orodja. Takšno orodje bi moralo omogočati:

- izdelavo scenarijev – posnetkov procesov,
- samostojno izvajanje posnetih scenarijev poljubno mnogokrat,
- možnost izbiranja podatkov pri izvajanju skript iz vnaprej pripravljene zaloge podatkov (repozitorija),
- ponovljivost testov z istimi podatki,
- avtomatsko generiranje poročil o uspešnosti posameznega testiranja.

Testna ekipa je že v fazi analize raziskala tržišče s testnimi orodji in najustreznejšega tudi preizkusila. Rezultati preizkusa so pokazali, da bi za potrebe testiranja na samem projektu bil vložek v pripravo podatkov in scenarijev približno enak kot ročno izvajanje istega testiranja. V kolikor bi se pripravljene scenariji uporabljali tudi v času vzdrževanja in nadgrajevanja obstoječega programskega paketa (ali celo kot scenariji za performančne teste), bi bil vložek v nakup orodja sicer upravičen, saj bi omogočal ponovitev vseh že opravljenih testov ob vsaki modifikaciji sistema. Ker CDM metodologija prične s pripravo plana testiranja že v fazi analize, je v samem izvedbenem načrtu brez posledic na celoten plan projekta možno dodati aktivnost preverjanja testnega orodja in ji zagotoviti ustrezen čas. Žal se je v našem primeru izkazalo, da orodje ne podpira zadnje, izbrane verzije razvojnega orodja (Oracle Designer 6i), tako da se je celotno testiranje izvajalo ročno.

Čeprav je bil postopek upravljanja z napakami v skladu s CDM metodologijo postavljen v planu projekta in so ga vsi člani projektne ekipe uporabljali, je kmalu prišlo do zelo velikega obsega ugotovljenih napak, s tem pa se je pojavil problem sledenja napakam. Na projektu ni bil uveden noben sistem za upravljanje z napakami, ki bi temeljil na bazi podatkov, kar pa je skoraj obvezno na tako obsežnih projektih (veliko število komponent testiranja, veliko število razvijalcev in velika testna skupine), saj je drugače upravljanje med vsemi tremi komponentami izredno zahtevno.

Po CDM metodologiji je eden ključnih izdelkov projekta, na podlagi katerega se izvaja funkcionalno testiranje, rezultat aktivnosti Izdelava funkcijskega modela sistema. V fazi analize se je z ARIS orodjem grobi funkcijski model sicer izdelal, vendar se ni prenesel v

repozitorij orodja Oracle Designer (podrobnejši opis razlogov v poglavju 4.4.5). Tako ni bilo enolične povezljivosti med izdelanimi moduli in funkcijskim modelom. Delno smo veliko pomanjkljivost pri testiranju nadomestili s postopkom prenosa izdelanih modulov iz razvojnega v testno okolje z ustrezno predstavitevjo funkcionalnosti modulov s strani načrtovalca modulov in programerja, kar pa je bilo izredno zamudno (velika poraba časa, končni uporabniki pri tem niso sodelovali in niso mogli sami preverjati ustreznost izdelka z njihovimi funkcionalnimi zahtevami). V kolikor se funkcijski model izdelava v skladu s CDM priporočili z orodjem Oracle Designer (še posebno ob možnosti uporabe Oracle RuleFrame, ki v času odločitve načina izvedbe in pričetku aktivnosti izdelave načrta modulov še ni bil na trgu), ostane sled od funkcijskega modela preko načrta modulov do končnega izdelka enolično določena in s tem dokumentacija popolna. Zato je že v planu projekta smiselno razmisliti o možnosti povezovanja modelov med orodji za analizo in izvedbo.

Hkrati z izgradnjo modulov je v skladu s CDM priporočili nastajala tudi tehnična in uporabniška dokumentacija za posamezni modul (podrobneje opisano v poglavju 4.4.5 in 4.4.8), to pa je bilo v času testiranja tudi edina razpoložljiva dokumentacija s strani izvajalca za testiranje.

Ker je v izvedbenem planu bilo ukinjeno tradicionalno fazno izvajanje projekta, kot ga priporoča CDM/PJM metodologija, in ker so v skladu s tem bile zamaknjene tudi preostale aktivnosti (podrobneje v poglavju 4.2.1), so se aktivnosti načrtovanja aplikacije, izdelave aplikacije in testiranje modulov in integracije modulov izvajale sočasno. To pomeni, da so se testi modulov izvajali, ko načrt aplikacije še ni bil v celoti postavljen. Posledica tega je bil slabši pregled nad celotno strukturo aplikacije, kar je zmanjševalo kakovost testiranja, prednost pa je bila v zgodnejšem odkrivanju napak in hitrejšemu odpravljanju vpliva napak na preostali načrt in izvedbo.

Zaradi velikega števila odkritih napak se je že kmalu pojavil problem pomanjkanja časa za odpravo napak, saj je bila izvedbena skupina preobremenjena z aktivnostjo izgradnje novega sistema. Posledično je prišlo do zamude na projektu. Takšnemu tveganju se lahko izognemo, če že v fazi planiranja predvidimo dodaten čas (najbolje posebno aktivnost) za odpravljanje nepravilnosti. MSF metodologija problem rešuje s štirimi enako dolgimi fazami projekta (analiza, razvoj, testiranje, stabilizacija), kar omogoča, da že v izvedbenem načrtu projekta rezerviramo zadosti časa (faza stabilizacije) za odpravljanje pomanjkljivosti.

4.4.8 Dokumentacija in izobraževanje

Vsak informacijski sistem mora imeti naslednjo dokumentacijo (Black, 1999, str. 71):

- uporabniška navodila (kako programski paket uporabljati),
- opis načrta aplikacije (kako programska oprema deluje) in

- tehnično dokumentacijo (zakaj je programska oprema zgrajena ravno na tak način).

Poleg tega mora vsebovati tudi dokumentacijo o vzdrževanju sistema (Wilson, 1999, str. 625).

Eden glavnih problemov na projektu, predvsem med izvajalci in različnimi skupinami uporabnikov, je nerazumevanje zaradi neenotne terminologije. Zato naj bi poleg omenjene dokumentacije že v pripravah na projekt izdelali (ali pa uporabili že objavljen in sprejet) ustrezen terminološki slovar.

Odločilno je, da je dokumentacija izdelana hkrati z zaključkom razvoja, saj je potrebna tako za testiranje kot za izobraževanje. Lahko je v papirni obliki ali dosegljiva preko interneta na spletnih straneh (Wilson, 1999, str. 624).

Namen procesa izobraževanja je, da se bodočim uporabnikom predstavi pomen in vpliv novega informacijskega sistema na njihovo delovno mesto ter da se jih usposobi za uporabo osnovnih ukazov in postopkov, ki jih bodo pri delu uporabljali (Vallabhaneni, 1996, str. 828). Pri planiranju izobraževanja je potrebno upoštevati:

- kompleksnost poslovnih operacij,
- predhodno usposobljenost uporabnikov,
- kakovost dokumentacije,
- skupine uporabnikov glede na uporabo različnih modulov novega sistema.

Izvesti je potrebno dva tipa izobraževanja, in sicer:

- izobraževanje uporabnikov novega sistema in
- izobraževanje ekipe za vzdrževanje novega informacijskega sistema.

CDM predvideva dva ločena procesa za izdelavo dokumentacije ter za izvedbo izobraževanja.

Za izdelavo dokumentacije predlaga aktivnosti priprave podatkovnega slovarja, priprave standardov različnih tipov dokumentacije ter dejansko izdelavo vseh tipov dokumentacije.

Po CDM priporočilih naj bi se izdelala naslednja dokumentacija:

- uporabniški priročnik (user guide),
- priročnik za uporabo posameznih modulov (user reference),
- tehnična dokumentacija,
- priročnik za administratorje sistema,
- funkcijska dokumentacija.

V primeru izdelave aplikacije z Oracle orodji teče izdelava funkcijske in tehnične dokumentacije vzporedno z izdelavo posameznih modulov in je to naloga načrtovalca aplikacije.

Za izvedbo izobraževanja predlaga naslednje aktivnosti:

- določitev zahtev za izobraževanje ter izdelava načrta izobraževanja,
- priprava gradiva za izobraževanje ter ustreznega okolja (aplikacija, baza podatkov, podatki in primeri),
- izvedba izobraževanja za končne uporabnike in administratorje novega sistema.

Glede na izkušnje s predhodnih projektov se je v zavarovalnici v času priprave projekta odločilo, da bo najoptimalnejše, če vodenje procesov izdelave dokumentacije in izobraževanja prevzame ista oseba, saj le-ta skozi pripravo dokumentacije dodobra spozna nov sistem in je zato (poleg načrtovalcev in izdelovalcev aplikacije) najprimernejša za vodenje aktivnosti izobraževanja. Poleg tega se je zaradi množice načrtovane dokumentacije odločilo, da se na izobraževanju kot gradivo uporabi uporabniški priročnik (in ne posebna skripta s primeri za izobraževanje), s čimer se skrajšajo postopki priprav na izobraževanja, hkrati pa usposobi in navadi končne uporabnike uporabljati razpoložljivo dokumentacijo o novem informacijskem sistemu. Zaradi mešanih ekip in sodelovanja naročnika pri izdelavi novega informacijskega sistema se izobraževanje bodočih administratorjev ni načrtovalo. Po potrebi bi se izvedle le posamezne konzultacije.

Prednost uporabe CDM metodologije na procesu izdelave dokumentacije in izobraževanja so se pokazale predvsem v:

- pripravi standardov dokumentacije že v začetnih fazah projekta,
- sočasno z razvojem načrta aplikacije izdelava funkcionalne in tehnične dokumentacije znotraj Oracleovih orodij za razvoj aplikacije ter možnost izvoza te dokumentacije v tekstovno datoteko ali uporaba kot interaktivna pomoč,
- definiranje postopkov za pripravo ustreznega okolja za izobraževanje.

4.4.9 Tranzicija

Prva aktivnost procesa tranzicije, to je definiranje načina prehoda na nov sistem, se mora pričeti izvajati že v prvi fazi projekta, saj predstavlja izhodišče za načrtovanje aktivnosti celotnega projekta, predvsem pa na področju izdelave aplikacije, prenosa in pretvorbe podatkov, testiranja in izdelave načrta izobraževanja končnih uporabnikov.

Obstajajo različni tipi prehodov na nov sistem, med katerimi je izbira odvisna predvsem od časa, razpoložljivih finančnih virov, razpoložljivosti človeških virov in tveganosti poslovnega procesa, in sicer (Vallabhaneni, 1996, str. 894-896):

- **trenuten preklp** na nov sistem, primeren za male ali časovno občutljive sisteme,
- **sočasna operativnost** starega (za stare podatke) in novega (za nove podatke) informacijskega sistema, primeren v primeru, ko se po določenem času podatki v starem sistemu ne uporabljajo več,
- **verzioranje instalacije**, kar je primerno za obsežne projekte, ki jih je možno razdeliti na diskretne inkremente,
- **vzporedna operativnost obstoječega in novega informacijskega sistema s faznim vklapljanjem novega sistema** po segmentih, kjer organizacija vzporedno uporablja stari sistem in izvrši preklp posameznega segmenta na nov sistem šele tedaj, ko je ugotovljena ustrezna zanesljivost segmenta novega sistema, kar je primerno za lahko segmentirane sisteme,
- **vzporedno izvajanje s trenutnim preklpom**, ki je najprimernejši način za izredno kritične poslovne procese, saj je med vzporednim izvajanjem možnost testiranja, popravljanja in stabilizacije novega sistema, je pa izredno zahteven glede virov (čas, denar in človeški viri),
- **pilotna tranzicija**, kjer se vklaplja nov sistem postopoma po posameznih lokacijah in je primerna predvsem za geografsko izredno razpršene sisteme.

Na podlagi izbire ustrezne strategija prehoda na nov sistem se potem izvajajo tudi vse ostale aktivnosti v smislu priprave na prehod, kot so izdelava načrta inštalacij novega informacijskega sistema in priprava produkcijskega okolja, poleg tega pa tudi aktivnosti ostalih procesov, kot so prenos in pretvorba podatkov, izobraževanje in drugo, kot je opisano v prejšnjih poglavjih.

CDM metodologija za proces tranzicije vsebuje priporočila za planiranje in izvedbo naslednjih aktivnosti:

- definiranje strategije prehoda na nov sistem, kjer je potrebno na podlagi kritične analize možnih prehodov na nov sistem tudi v smislu izvedbe prenosa in pretvorbe podatkov, sistemskih vmesnikov in izobraževanja, izbrati ustreznega in ga čim natančneje dokumentirati,
- izdelava načrta inštalacij in priprava testnega, produkcijskega in vzdrževalnega okolja,
- izvedbo prehoda in ugašanje starega informacijskega sistema.

Po CDM metodologiji so najpomembnejši izdelki, na podlagi katerih se izvaja proces tranzicije:

- izdelki procesa prenosa in pretvorbe podatkov, predvsem načrt prenosa in pretvorbe podatkov za definiranje načina prehoda na nov sistem ter postopki prenosa in pretvorbe za dejanski prehod na nov sistem,
- zaključene aktivnosti izobraževanja, testiranja, uspešno opravljen prevzemni test,

- standardi načrtovanja in izdelave aplikacije ter aplikacijska koda,
- skripti za inštalacijo podatkovne baze,
- dokumentacija (uporabniška, tehnična, navodila za inštalacijo, operativnost in vzdrževanje novega sistema).

Spodaj so navedena dejstva, ki so ključna za razumevanje poteka procesa tranzicije na projektu izgradnje novega poslovnega transakcijskega informacijskega sistema v zavarovalnici Slovenica:

- sočasno s projektom izgradnje poslovnega informacijskega sistema poteka tudi projekt postavitve novega finančno-računovskega informacijskega sistema, zato je bila potrebna velika pazljivost pri usklajevanju terminov in definiranju sistemskih vmesnikov (v nadaljevanju je izgradnja systemskega vmesnika zaradi velikega obsega dela postala samostojen projekt),
- že v planu projekta je bilo predvideno segmentirano predajanje sistema v produkcijo po posameznih zavarovalnih vrstah,
- vodja procesa tranzicije je bil določen s strani izvajalca, njegova vloga pa je bila organizacijsko s čim manjšim tveganjem speljati prehod na nov sistem v času, ki ga je določil naročnik.

S temi predpostavkami so se v nadaljevanju izvajale vse aktivnosti tranzicije. V času izdelave te naloge sistem še ni bil pripravljen za produkcijo (to pišem v času pred uporabniškim testiranjem), zato bom navedla predvsem prednosti uporabe metodologije pri pripravi nalog tranzicije, predvsem definiranju načrta prehoda na nov sistem.

CDM metodologija priporoča čim zgodnejše definiranje začetne strategije prehoda na nov sistem. Na tak način že v prvi fazi projekta spoznaš problem prehoda. Dobra metodologija mora predlagati možne načine prehoda in korake pri posameznem načinu, tveganja ter primernost posameznega načina prehoda na različnih tipih informacijskih sistemov. S pomočjo metodologije projektna skupina določi ključne faktorje, na podlagi katerih lahko bolj kvalitetno oceni primernost posameznega prehoda na nov sistem. S pomočjo metodologije mora projektna skupina dobiti realno sliko prehoda z vsemi nalogami, pastmi, problemi in pričakovanim končnim rezultatom, na podlagi česar lahko lažje planira rok izvedbe prehoda. Glede načinov prehoda je CDM metodologija šibka, saj definira samo tri načine (fazen pristop, vzporedno delovanje obeh sistemov in zamenjava sistema). Zaradi zahtevne arhitekture novega informacijskega sistema zavarovalnice Slovenica (geografska razpršenost sistema po celi Sloveniji), kritičnosti poslovnega procesa, vzporednega izvajanja soodvisnega projekta, zagotavljanje konsistentnosti vnesenih podatkov ter segmentnega pristopa k izvedbi projekta po posameznih zavarovalnih vrstah) je bilo tudi način prehoda s strani Slovenice po predlogih CDM metodologije izredno težko definirati. V trenutku izdelave naloge še vedno ni popolnoma natančno definiran prehod, vsekakor pa je po zgornji opredelitvi različnih pristopov, kot jo opredeljuje Vallabhaneni

(1996), dosti bližje podobi prehoda na nov sistem v projektu zavarovalnice Slovenica, ki pa bo morala uporabiti kombinacijo opisanih načinov.

Ker CDM metodologija opredeljuje tudi vloge in odgovornosti posameznih članov projektne skupine pri opredeljevanju načina prehoda na nov sistem, je razvidno, da določitev načina prehoda nikakor ni samo naloga vodje projekta, temveč skupno delo tako članov projektne skupine (predvsem vodja prenosa in pretvorbe podatkov ter ostalega osebja sektorja informacijske tehnologije) kot tudi analitikov poslovnih procesov, vodje posameznih poslovnih oddelkov (v našem primeru oddelek zavarovanj, škod, trženja, aktuarstva, finančno-računovodskega) in uporabnikov, katerih naloga je določiti vsebino in kritičnost posameznih procesov in podatkov, ki se bodo prenašali in vnašali v nov sistem.

4.4.10 Produkcija

V tradicionalnem življenjskem ciklu projekta se proces produkcija izvaja po vključitvi novega informacijskega sistema v produkcijsko okolje. Nekatere metodologije zajemajo to obdobje ves čas uporabe novega informacijskega sistema do njegove izključitve, medtem ko druge (vključno s CDM-om) samo prvo, naprej opredeljeno uvodno obdobje. Slednji način je tudi bolj smiseln pri projektih, kjer je izvedba predana zunanjemu partnerju, saj običajno prvi, uvajalni del še spada pod pogodbene obveznosti za delo na projektu, medtem ko se za nadaljnje produkcijsko obdobje sklene (odvisno od potreb) vzdrževalna pogodba. Cilj aktivnosti je spremljanje in odzivanje na probleme, ki se pojavijo po prehodu na nov informacijski sistem, da se izvede nadgrajevanje sistema v smislu odpravljanja napak ter da se izdela načrt nadaljnjih nadgradenj novega sistema. V ta namen se izvajajo naslednje aktivnosti (Vallabhaneni, 1996, str. 822-831):

- pregled ustreznosti dokumentacije (uporabnost dokumentacije je razvidna šele po določenem času njene uporabe v produkciji),
- ocenjevanje odzivnosti sistema v realnih pogojih dela,
- ugotavljanje in odpravljanje napak in pomanjkljivosti, ugotovljenih v realnih pogojih dela,
- podpora uporabnikom pri uporabi novega informacijskega sistema,
- ugotavljanje potreb po dodatnih funkcionalnostih,
- preveri se ustreznost vzdrževalnih postopkov produkcijskega sistema in postopkov nadgrajevanja novega informacijskega sistema,
- izvede se neodvisna postimplementacijska revizija ustreznosti sistema.

CDM metodologija za to obdobje opredeljuje naslednje aktivnosti:

- revizija novega informacijskega sistema v smislu formalne ocenitve programske opreme z namenom, da se zagotovi pravilno delovanje sistema, s strani neodvisne osebe, običajno revizorja,

- določitev metrike (na podlagi načrta kapacitet in performančnih zahtev), izvedba merjenja in analiziranje problemov performančnih odzivnosti sistema ter posodobitve sistema zaradi ugotovljenih neustreznosti,
- upravljanje s problemi v času produkcije (gre za nadaljevanje aktivnosti ugotavljanje napak v času testiranja),
- načrtovanje, implementacija in testiranje skript za nadgrajevanje obstoječega sistema (nadgradnja instalacijskega plana v smislu nadgrajevanja obstoječega sistema),
- izdelava, testiranje in implementacija popravkov (apliciranje popravkov na podlagi problemov in zagotavljanja ustreznosti performans).

V času izdelave tega dela projekt še ni prišel v fazo zaključevanja, zato o izkušnjah na projektu ne morem govoriti. Kljub temu je prednost metodologije v tem, da se že v času planiranja zavedamo vseh zgoraj omenjenih aktivnostih, jih vnesemo v plan projekta in s tem zagotovimo:

- da so nam člani projektne skupine na voljo tudi v času produkcije in da ne odidejo takoj na druge projekte in delovne obveznosti,
- da že na začetku opredelimo potreben čas za vse te aktivnosti in ga, glede na tveganost projekta, ustrezno vključimo v življenjski cikel projekta.

4.5 Spremljanje projekta

Proces spremljanje projekta sestavljajo aktivnosti projektne vodnje v smislu (Vallabhaneni, 1996, str. 763-764):

- upravljanja z zahtevami za spremembo,
- spremljanje napredovanja projekta,
- modificiranje izvedbenega načrta projekta v smislu zagotavljanja pravočasnega zaključka projekta,
- komunikacije na projektu (sestanki, poročila o opravljenem delu).

V kolikor je bil plan projekta ustrezno pripravljen, so vsi standardi in postopki spremljanja projekta bili v fazi priprave projekta definirani.

PJM metodologija izvaja postopke spremljanja projekta preko naslednjih aktivnosti, ki se izvajajo vzporedno s posamezno tradicionalno fazo projekta:

- upravljanje z odprtimi vprašanji, zahtevami za spremembo, problemi,
- vodenje sestankov na projektu na različnih nivojih projektne organizacije,
- spremljanje in poročanje o napredovanju na projektu in porabi finančnih virov,
- spremljanje dela članov projektne skupine na nalogah projekta,

- spremljanje kakovosti na projektu,
- spremljanje upravljanja s konfiguracijo.

V skladu s PJM priporočili in vzpostavljenimi standardi in postopki na projektu so se zgornje aktivnosti izvajale tudi na projektu v zavarovalnici Slovenica. Bistvo PJM metodologije se je na našem projektu izkazalo v tem, da smo te aktivnosti definirali že na začetku in jih vnesli v projektni plan. Glede same izvedbe spremljanja projekta pa smo naleteli na dve težko rešljivi situaciji, ki ju predstavljam v nadaljevanju.

Spremljanje napredovanja projekta se je izvajalo elektronsko z orodjem Project Central. Naloga vsakega člana projektne skupine je bila izpolnitev obrazca v rednih dogovorjenih intervalih (tedensko) do določenega termina o delu na posameznih aktivnostih v preteklem obdobju. Vendar pa je ta informacija bila samo številčna (na primer porabljenih 15 ur na aktivnosti A od v celoti predvidenih 60 ur). Takšna informacija pa je bila praktično brez vrednosti, saj ni govorila o dejanski vsebini in kvaliteti opravljenega dela. Tako na primer vodja projekta ni imel nikakršnega vpogleda v zares opravljeno delo (na primer pri izdelavi aplikacije je bilo ugotovljeno, da je porabljenega že 40% časa na izdelavi modulov, nič pa o tem, ali in kateri moduli so že izdelani), s čimer je bil nadzor nad projektom precej okrnjen. Konkretna informacija bi bolje služila tudi ostalim članom projektne ekipe (na primer vodji testiranja za pripravo testnih skript, vodji prenosa in pretvorbe podatkov za pripravo testov prenesenih podatkov, za načrtovanje izdelave dokumentacije ipd.).

Nadzor kakovosti na projektu je bil definiran samo za oceno kakovosti končnih izdelkov. Zaradi zgostitve nalog na račun opustitve tradicionalnega faznega pristopa je bil zaključek velike večine ključnih aktivnosti skoncentriran šele ob koncu projekta. Poleg tega so se kritične aktivnosti, katerih ocena kakovosti prve je vplivala na izvedbo druge, izvajale vzporedno (na primer načrtovanje modulov se je izvajalo hkrati s kodiranjem modulov). Tako je bilo ob ugotovljenih nepravilnostih prve izredno zamudno in drago odpravljanje napak in popraviljanje rezultatov druge aktivnosti. Vsekakor menim, da je opustitev ključnih temeljev metodologije lahko zelo nevarno dejanje. Takšnemu stanju bi se delno lahko izognili, če bi izvajanje dolgotrajnejših aktivnosti razbili na manjše segmente in jih postopoma kontrolirali (na primer načrt modulov razbijemo na načrt šifrantov, načrt sklepalnih modulov, načrt škodnih modulov,... in šele, ko je ustrezna skupina modulov pregledana in ocenjena kot primerna, se prične z izvajanjem kodiranja modulov).

4.6 Zaključek projekta

Zaključek projekta je zadnja aktivnost projekta izgradnje novega informacijskega sistema. O zaključku projekta lahko govorimo, ko so zaključene vse projektne aktivnosti in izdelani vsi projektni izdelki. Namen aktivnosti je uradno zabeležiti zaključek pogodbenih obveznosti na projektnem delu med naročnikom in izvajalcem. Namen izvajalca je, da v tej aktivnosti pridobi podpis o ustreznosti vseh projektnih izdelkov ter jih preda naročniku oziroma ekipi za vzdrževanje novega informacijskega sistema v produkciji. Naročnik mora poskrbeti, da projektni izdelki zares ustrezajo poslovnim zahtevam, definiranim v planu

projekta in v prvih fazah projekta, da uporabniki novega sistema znajo sistem produktivno uporabljati pri svojem delu, da je sistem ustrezno testiran ter da ima ustrezno dokumentacijo. V kolikor je projekt dobro voden, je aktivnost zgolj formalna ureditev odnosov med pogodbenima partnerjema, saj se posamezni izdelki in njihova primernost in uporabnost ocenjujejo sproti. V nasprotnem primeru je to lahko precej konfliktna situacija. Da se temu izognemo, je pomembno, da se že v zgodnji fazah projekta opredeli kriterije za zaključek projekta ter da se vsi problemi in odprta vprašanja rešijo pred tem.

Celotna CDM metodologija je zasnovana tako, da omogoča sprotno preverjanje kakovosti vsakega posameznega izdelka ob zaključku naloge, poleg tega ima v okviru procesa testiranja predviden prevzemni test ter v okviru aktivnosti vodenja projekta (PJM) aktivnost spremljanja odprtih vprašanj in problemov, izvajanje kontrolnih točk ter postopek vodenja kakovosti. Na podlagi korektnega projektne vodenja se v nalogi zaključek projekta preverijo ustreznost novega informacijskega sistema glede na dokument Obseg, cilji in pristop, poročila o odprtih vprašanjih in problemih, poročilo o napredku projekta, poročilo o nadzoru kakovosti na projektu, poročilo o prevzemnem testu ter poročilo o prenosu sistema v produkcijsko okolje in postavitev okolja za vzdrževanje sistema.

Kljub vodenju projekta po CDM metodologiji v Slovenici so se že v fazi priprave projekta pojavile določene specifičnosti projektne organiziranosti predvsem glede projektne skupine in nefazne organiziranosti izvedbenega načrta projekta, zato smo že na začetku zastavili pogoje zaključevanja projekta nekoliko drugače. Glede na mešanost ekip med naročnikom in izvajalcem za izvedbo je bil poseben poudarek dan na uskladitev mnenj o ustreznosti projektne izdelkov predvsem na področju dokumentacije, izobraževanja, odprtih vprašanj in problemov, medtem ko je bila ustreznost aplikacije, prenosa in pretvorbe podatkov že stvar prevzemnega testa v fazi testiranja, ki se je izvedlo pred preklopom v produkcijo.

Pomembno je, da se vse te aktivnosti definirajo že v planu projekta, da je z njimi tudi izvajalec seznanjen, ki temu prilagodi tudi izvedbo ter da so jasno opredeljene vloge in odgovornosti članov projektne skupine.

4.7 Pregled uporabe CDM metodologije pri upravljanju projekta na zavarovalnici Slovenica

V poglavju predstavljam prednosti uporabe CDM/PJM metodologije na projektu izgradnje informacijskega sistema v zavarovalnici Slovenica kakor tudi probleme, na katere smo naleteli tako zaradi uporabe kot neuporabe metodologije ter načini ali predlogi, kako smo (ali bi) se težavam izognili.

Projekt je bil zasnovan in voden v zavarovalnici, torej na strani naročnika. Poleg prenove poslovnih procesov je bilo definiranje poslovnih potreb voden in izvajano na strani naročnika, medtem ko je izvedba, predvsem načrtovanje in izgradnja baze podatkov in aplikacije bilo predano zunanjemu izvajalcu. Procesna tehnična arhitektura in tranzicija sta

bila prav tako predana izvajalcu, medtem ko so bili za vodje procesov pregleda obstoječega sistema, prenosa in pretvorbe podatkov, testiranja, dokumentacije in izobraževanja določeni naročnikovi člani projektne skupine. Vse projektne skupine, razen tehnoloških, ki so opisovale poslovne procese, so sestavljali tako naročnikovi kot izvajalčevi člani projektne ekipe.

Zaradi odločitve, da bo razvoj novega informacijskega sistema potekal z orodjem Oracle Designer, se je zdelo smotrno, da se za vodenje in nadzor projekta uporabi ustrezna CDM/PJM metodologija, ki uporabo omenjenega orodja tudi predlaga. Metodologija je bila vpeljana v sklepnem delu izvajanja prenove poslovnih procesov. V tem času so bile v večji meri že definirane vsebine, katere PJM metodologija opredeljuje v dokumentu Definiranje obsega, ciljev in pristopa.

CDM metodologija na projektu izgradnje poslovnega informacijskega sistema s svojim konceptom razvoja informacijskih sistemov, ki temelji na tradicionalnem, faznem pristopu z natančno določenimi aktivnostmi, nalogami in rezultati ter medsebojne odvisnosti med posameznimi nalogami obljublja (podrobneje je opredeljeno v poglavju 3.5) hitrejši zagon projekta in skrajšanje časov planiranja projekta, kar se je na projektu razvoja informacijskega sistema za produkcijske poslovne procese dejansko izkazalo. Z uporabo CDM/PJM metodologije pri pripravi plana projekta so se zaradi napotkov in predlog, ki jih metodologija ponuja, glede na čas, ki je bil potreben za pripravo začetnega izvedbenega načrta projekta (pred prenovo poslovnih procesov), skrajšali časi izdelave izvedbenega načrta projekta, povečala se je popolnost izvedbenega načrta, zmanjšala možnost izvajanja neplaniranih aktivnosti ter povečala stopnja nadzora nad obsegom in viri projekta, medtem ko opisovanja izdelkov projekta sploh ni bilo potrebno izvesti, saj jih opisuje že CDM metodologija. Z uporabo predlogov organizacijske strukture projekta in opisom potrebnih strokovnjakov za izvedbo projekta smo že na začetku projekta zagotovili ustrezne strokovnjake za izvedbo oziroma pravočasno predvideli njihovo potrebno izobraževanje, pravočasno uskladili delo strokovnjakov na drugih projektih in nalogah v odvisnosti od izvedbenega načrta za projekt izgradnje informacijskega sistema, posamezniki so bili pravočasno oziroma že ob zagonu projekta seznanjeni s svojo vlogo, odgovornostjo in pričakovanji. Na podlagi kombinacije postavljene organizacijske strukture in izvedbenega načrta je bila možna optimizacija izvedbenega načrta projekta v smislu prezasedenosti ali neizkoriščenosti posameznih članov projektne skupine, zmanjšalo pa se je tudi tveganje postavitve neustrezne strukture projektne skupine ali da bi celo na kakšno vlogo pozabili. Z uporabo standardov in priporočil za komunikacije na projektu smo že na začetku projekta definirali ustrezne delovne postopke ter vso projektno skupino izobrazili za njihovo uporabo, s čimer smo zmanjšali tveganje, da bi zaradi kasnejših sprememb ali vpeljave nepoznanih postopkov člani projektne skupine le-teh ne uporabljali ustrezno ter zmanjšali tveganje nedokumentiranosti odločitev na projektu. Od postopkov zagotavljanja kakovosti na projektu smo vpeljali samo presojo ključnih, vnaprej definiranih izdelkov projekta predvsem v smislu, da se preveri pravočasna definiranost, ustreznost in razumljivost potrebnih standardov na projektu. S tem smo povečali nadzor nad

napredovanjem in oceno kakovosti projekta. S pomočjo uporabe napotkov za upravljanje s konfiguracijo na projektu smo že v planu projekta predvideli možne tipe konfiguracij ter določili postopke upravljanja s posameznimi tipi konfiguracije, s čimer smo vzpostavili ustrezen nadzor nad spremembami.

Nadalje CDM metodologija obljublja prihranek časa na račun razvijanja standardov in priporočil (v kolikor v organizaciji že ne obstajajo standardi razvoja, ki jih je potrebno pri izvedbi projekta upoštevati), saj le-ta metodologija tudi ponuja. Ker v Slovenici ustreznih standardov niso obstajali (obstoječi so bili neažurni in do tedaj tudi ne v celoti vpeljani), je bilo smiselno kritično sprejetje Oracle standardov. Z uporabo Oracle standardov in priporočil smo še pred izvedbeno fazo projekta pripravili ustrezen nabor standardov za načrtovanje in izgradnjo baze podatkov in aplikacije ter izdelali prototipe uporabniškega vmesnika, na podlagi katerega smo skupaj s končnimi uporabniki postavili zasnovo za bodoči izgled aplikacije. Ker je tekel razvoj informacijskega sistema za produkcijske poslovne procese in razvoj systemskega vmesnika (torej še ene aplikacije) pri dveh različnih izvajalcih, a z isto tehnologijo, smo z vpeljavo istih standardov načrtovanja dosegli enoten izgled obeh aplikacij, kar bo krogu uporabnikov, ki bodo uporabljali obe aplikaciji, omogočilo enostavnejšo uporabo. Z uporabo orodja Headstart Utilities smo dosegli pohitritev razvoja in povečanje implementacije standardov pri izdelavi informacijskega sistema ter avtomatizirali izdelavo poročil o kakovosti aplikacije. Zaradi uporabe orodja Oracle Headstart Quality Check testna ekipa ni porabljala dodatnega časa za izdelavo poročil o kakovosti končnega izdelka, s čimer se je skrajšal čas testiranja kakovosti končne aplikacije.

Ker po CDM metodologiji temelji ves razvoj (načrtovanje podatkovne baze in aplikacije, kreiranje testnih skript in scenarijev, pisanje dokumentacije, planiranje in izvajanje izobraževanja) na poslovnih modelih, definiranih v prvi fazi projekta, se vse aktivnosti projekta izvajajo urejeno in nadzorovano, prav tako pa se prihrani čas pri pripravi posameznih strategij. S tem je naročnik pridobil nadzor nad izvajanjem aktivnosti načrtovanja in izdelave baze podatkov in aplikacije, že v začetku izvedbene faze pa se je lahko začela aktivnost priprave strategije prenosa in pretvorbe podatkov v nov sistem, katere rezultati so pomembno vplivali tudi na načrt nove baze podatkov, s čimer smo znižali stroške naknadnega dograjevanja nove podatkovne baze za namen prenosa obstoječih podatkov (primer prenosa podatkov iz starega v nov sistem, za katerega ni obstajala enolična preslikava).

Uporaba CASE orodja Oracle Designer, ki omogoča hitro prototipiranje, omogoča tudi boljšo komunikacijo in razumevanje s končnim uporabnikom pri razvoju aplikacije.

Na področju spremljanja projekta in nadzora kakovosti na projektu vprašalniki pri nalogah za sprotno ocenjevanje kakovosti rezultata naloge ter uporaba orodij za avtomatsko preverjanje uporabe dogovorjenih standardov, ki jih CDM metodologija opredeljuje, zvišujejo kakovost končnega izdelka, to je informacijskega sistema. Nenazadnje nam je ravno opis posameznih izdelkov in kriterijev kakovosti posameznih izdelkov, kot jih CDM

metodologija definira, pomagal pri razčiščevanju nesoglasij med različnimi pogledi naročnika in izvajalca na končni izdelek posamezne aktivnosti.

Ker CDM metodologija opredeljuje tudi vloge in odgovornosti posameznih članov projektne skupine za posamezne naloge, je bilo že na začetku projekta jasno, da določitev načina prehoda nikakor ni samo naloga vodje projekta, temveč delo tako članov projektne skupine kakor predvsem analitikov poslovnih procesov ter vodij posameznih poslovnih področij in uporabnikov, katerih naloga je določiti vsebino in kritičnost posameznih procesov in podatkov, ki se bodo prenašali in vnašali v nov sistem.

Zaradi specifičnih okoliščin določene projektne aktivnosti niso bile izvedene v skladu s priporočili CDM metodologije, kar se je v veliki meri v izvedbeni fazi projekta pokazalo kot velika težava na projektu.

Prvi problemi so se nakazali že pri izdelavi izvedbenega načrta projekta in oblikovanju projektne ekipe. Ker je bil časovni okvir projekta prilagojen strategiji zavarovalnice, je zaradi tendence po maksimalni izkoriščenosti časa in ljudi v fiksnem časovnem obdobju bil ukinjen tradicionalni, fazni pristop k razvoju informacijskega sistema. Zaradi tega se je zmanjšal »rezervni« čas med posameznimi nalogami, skoraj nemogoče je bilo postaviti kontrolne točke na izvedbeno fazo projekta, oteženo je bilo izvajanje presoje kakovosti na projektu, saj so bile vse ključne aktivnosti izdelane šele tik pred prenosom sistema v produkcijo, projektna ekipa je bila polno zasedena z aktivnostmi na projektu, mnogi člani projektne ekipe so igrali več vlog hkrati na projektu in to na aktivnostih, ki so se odvijale vzporedno, zmanjkalo pa je tudi časa in človeških virov za nepredvidene aktivnosti, kot je bilo na primer odpravljanje napak v času testiranja. S tem se je povečalo tveganje, da projekt ne bo zaključen v predvidenem roku (ki se je kasneje tudi udejanilo, saj projekt kljub pretečenemu roku izvedbe še vedno teče).

Nadalje se je problem opustitve faznega pristopa pokazal tudi pri procesu načrtovanja in izgradnje aplikacije, saj je bila zabrisana meja med načrtovanjem in izgradnjo. Zaradi tega so se izvajale aktivnosti izgradnje brez predhodne potrditve ustreznosti načrtov aplikacije, zaradi česar so se napake in neustreznosti aplikacije odkrivala šele med testiranjem. To je izredno povečalo stroške in čas za odpravo napak, kar se je rezultiralo tudi v celotnem izvedbenem načrtu. Temu bi se kljub pomanjkanju časa za izvedbo projekta bilo možno izogniti, če bi se že pri pripravi izvedbenega načrta projekta uporabil iterativni pristop k razvoju aplikacije (podobno, kot ga opredeljujeta MSF in RUP metodologiji).

Zaradi neizvajanja naloge Definiranja obsega, ciljev in pristopa v skladu z metodološkimi priporočili je bil projekt definiran v dokumentih, ki niso bili dostopni in predstavljeni celotni projektni skupini. S tem ni bilo doseženo poenotenje pogleda na obseg in cilje projekta med člani projektne skupine in uporabniki, zmanjšano je bilo zaupanje v uspešnost projekta in ostala so tudi nerealna pričakovanja končnih uporabnikov.

Pri planiranju in nadzorovanju poteka projekta je bil tudi problem nadzora nad konfiguracijo aplikacijske kode, saj v fazi planiranja projekta ustrezno orodje za

verzioriranje repozitorija orodja Oracle Designer še ni bilo na tržišču. Zato so se predvsem v fazi testiranja pojavljali problemi v zvezi z referenciranjem odkritih napak. Tega problema nismo mogli rešiti ustrežneje, kot da se je odprava napake reševala na trenutnem razvojnem stanju repozitorija (in ne v dejanskem, kjer je bila napaka odkrita), posledica česar je bil predvsem nekonsistentni pogled na napako med skupinama za razvoj in testiranje. Podobno se je tudi problem nedefiniranja konfiguracije aplikacije po načelu 'bomo definirali, ko bomo potrebovali' prikazal šele pri postavitvi testnega okolja, saj je bilo ravno zaradi tega potrebno naknadno dodati aktivnost verificiranje postavitve testnega okolja v proces testiranja, kar je še dodatno upočasnilo projekt.

Ker po CDM metodologiji ves razvoj (načrtovanje aplikacije, kreiranje testnih skript in scenarijev, izdelava dokumentacije, planiranje in izvajanje izobraževanja) temelji na poslovnih modelih, definiranih v prvi fazi projekta, se je kmalu izkazalo za veliko pomanjkljivost, ker se tudi analiza ni izvedla v skladu s CDM metodološkimi priporočili, saj niso bili izdelani nekateri poslovni in sistemski modeli (dejansko so se izdelali model poslovnih procesov in grobi model poslovnih funkcij ter v nadaljevanju še sistemski podatkovni model). S tem so bila dana veliko slabša izhodišča za vse nadaljnje, zgoraj naštetih aktivnosti, zmanjšan pa je bil tudi nadzor uporabnikov nad izvedbo projekta, saj se je izgubila sled med poslovnim modelom procesov in dejansko izvedbo informacijskega sistema.

CDM metodologija izredno podrobno opredeljuje tudi procesa izdelave dokumentacije in izobraževanja. Glede na to, da so pri razvoju sodelovali tako naročnik kot izvajalec ter da so bili v razvoj in testiranje vključeni tudi tehnologi in končni uporabniki s strani naročnika, se je že v začetku planirala izdelava dokumentacije in izobraževanja v nekoliko zmanjšanem obsegu, kot ga predvideva metodologija. Obe aktivnosti sta bili združeni predvsem zaradi pomanjkanja članov projektne ekipe in podvajanja nalog (uporabniška dokumentacija in dokumentacija za izobraževanje) ter zaradi dobrega poznavanja novega informacijskega sistema dokumentalistov, za katere se je predpostavljalo, da bodo najustreznejši inštruktorji bodočim uporabnikom.

Kljub uporabi metodologije pa v skladu s trditvijo, zapisano v poglavju 3.1, niso bili rešeni vsi problemi na projektu. Določene probleme smo reševali na drug način z uporabo drugih tehnik in metodologij, nekateri problemi pa so ostali tudi nerešeni.

Na projektnem vodenju je bilo veliko nalog s področja komunikacij na projektu ter upravljanja z delom, viri in konfiguracijo (gre za delovne postopke), ki pa jih CDM/PJM metodologija opredeljuje samo s teoretičnim opisom poteka postopka, Oracle pa ne ponuja tehnične rešitve za informacijsko podporo aktivnostim. V kolikor želimo na obsežnem projektu upoštevati priporočila dokumentiranosti in sledljivosti teh aktivnosti, je potrebno uvesti informacijsko podporo delovnim procesom. Na področju planiranja projekta smo zato uporabili Microsoft Project ter Microsoft Project Central za področje spremljanja porabljenega časa, dela članov projektne skupine in napredka na projektu. Za področje spremljanja problemov v procesu testiranja smo sami razvili sistem za upravljanje s

problemi, saj na trgu nismo našli ustrezne enostavne in cenovno ugodne rešitve. Za upravljanje konfiguracije smo uvedli enostaven datotečni sistem za dokumente, medtem ko za verzioniranje aplikacije oziroma repozitorija orodja Oracle Designer nismo našli ustrezne rešitve, kar se je kot velika pomanjkljivost pokazalo predvsem pri testiranju.

Pri zagotavljanju kakovosti na projektu smo zaradi izvedbenega načrta in odločitve za izvajanje nadzora kakovosti samo na zaključenih izdelkih projekta, kar za ustrezen nadzor nad kakovostjo projekta ni zadoščalo, uvedli tudi manj formalne načine, kot sta spremljanje rezultatov aktivnosti, ki niso pod nadzorom kakovosti (na primer rezultatov testiranja modulov) ter ocene naročnikovih izvajalcev v mešanih ekipah, predvsem v ključnih nalogah načrtovanja in izgradnje baze podatkov in aplikacije.

Zaradi izdelave poslovnega modela procesov in funkcij z ARIS orodjem in odločitve, da se modele ne prenese v repozitorij razvojnega orodja Oracle Designer se je izgubila sled med poslovnimi modeli in implementacijo oziroma končnim izdelkom. Delno smo problem skušali rešiti s predpisom o dopolnitvi funkcionalne dokumentacije z referenčno številko procesa, česar pa se programerji niso striktno držali.

Velik problem je predstavljala tudi aktivnost definiranja sistemskih vmesnikov. Ker je vzporedno z razvojem poslovnega informacijskega sistema tekkel tudi projekt postavitve novega informacijskega sistema za finančno računovodske posle, je naloga definiranja sistemskega vmesnika med obema sistemoma prerasla okvir naloge, kot jo opredeljuje CDM metodologija. Zaradi tega je bil za ta vmesnik uveden nov projekt z drugimi izvajalci, s čimer se je razbremenil izvedbeni načrt (in tudi člani projektne skupine) projekta izgradnje poslovnega informacijskega sistema.

Procesa načrtovanja in izgradnje baze podatkov in aplikacije sta bila v celoti predana izvajalcu, poleg tega sta obe aktivnosti časovno zelo obsežni, zaradi česar je naročnik izgubil nadzor nad izvajanjem in kakovostjo izvedbe izdelkov omenjenih procesov. Glede nadzora ustreznosti podatkovnega modela in aplikacije smo problem reševali na projektu s predstavitvijo ključnih konceptov, upoštevanih pri zasnovi podatkovnega modela, ključnim tehnologom, analitikom, administratorjem baze podatkov in administratorjem podatkov, izveden pa je bil tudi poskusni vnos in kontrola odzivnosti novega sistema na različne transakcije pri masovnih podatkih, s čimer smo dobili okvirno sliko o performančnih odzivnosti novega sistema. Poleg tega smo celoten projekt razbili na sklope po posameznih zavarovanjih, kar je približalo časovno oddaljen zaključek projekta ter povečalo motivacijo projektne skupine.

Poseben problem je predstavljalo tudi planiranje potrebnega časa za razvoj aplikacije. Oba problema (tisti v prejšnjem odstavku in pravkar omenjeni) bi lahko rešili, če bi celoten sistem razbili na manjše, obvladljive podsisteme in bi se razvoja lotili iterativno za vsak posamezen sistem. S tem bi omogočili hitrejšo odkrivanje napak kakor tudi realnejšo oceno trajanja ključnih aktivnosti v izvedbenem načrtu projekta. V iteracijo bi bilo smiselno vključiti poleg načrtovanja, izgradnje in testiranja tudi prenos in pretvorbo podatkov, saj

vzporedno izvajanje pilotskih prenosov omogoča hitrejšo oceno zahtevnosti prenosa tudi s časovnega vidika, kar je potrebno poznati pri pripravi strategije prehoda na nov sistem.

Pri izvajanju testiranja se je pokazalo za veliko pomanjkljivost nekritična uporaba metodoloških priporočil, kot je bilo na primer neustrezna sestava testne ekipe in njeno mesto v projektni organizaciji, ki je prvotno bila odgovorna vodji izvedbe na strani izvajalca. V primeru mešanih razvojnih ekip je še posebej pomembno, da se formirata dve neodvisni ekipi (kar smo kasneje na projektu tudi storili), in sicer na strani izvajalca ekipa za testiranje modulov in integracije modulov ter na strani naročnika za testiranje skladnosti sistema s specifikacijami in zahtevami za kakovost. Poleg tega sta se pri testiranju pokazali še potreba po uvedbi dveh dodatnih aktivnosti, ki ju CDM metodologija ne predvideva, in sicer verificiranje postavitve testnega okolja in aktivnost odpravljanja napak.

In ne nazadnje, eden največjih problemov na projektu je vendarle bil postavitve strategije prehoda na nov sistem. Zaradi zahtevne arhitekture novega informacijskega sistema zavarovalnice Slovenica (geografska razpršenost sistema po celi Sloveniji), kritičnosti poslovnega procesa, vzporednega izvajanja soodvisnega projekta, zagotavljanja konsistentnosti vnesenih podatkov ter segmentnega pristopa k izvedbi projekta po posameznih zavarovalnih vrstah so se CDM priporočila izkazala za nezadostna, saj na kratko opredeljujejo zgolj tri najpogostejše tipe prehoda. Nalogo smo tako reševali s pomočjo drugih, podrobnejših virov in priporočil ter v precej razširjeni skupini, ki je vključevala ne samo informatike, temveč tudi analitike poslovnih procesov, sponzorja, vodje poslovnih področij in uporabnikov.

5 Sklep

Pri projektih izgradnje poslovnih informacijskih sistemov se projektne skupine v številnih organizacijah srečujejo z različnimi problemi, ki so posledica neprimerne organizacije procesov projekta v dolgoročnem smislu. Zaradi slabe metodologije za razvoj sistemov, ki nejasno opredeljujejo aktivnosti in rezultate faz in ne vključujejo ustreznega nadzora nad izvajanjem posameznih nalog, se lahko pojavijo različna tveganja, ki vplivajo na uspešen zaključek projekta. Ne glede na izbrani način pristopa k razvoju poslovnega informacijskega sistema je potrebno vedno najprej izdelati podroben plan projekta, ki zajema definiranje obsega in ciljev projekta ter postopke in standarde za nadzor nad projektnim vodenjem. Prepoznati je potrebno tveganja in določiti postopke upravljanja z njimi ter opredeliti zahteve za kakovost končnega izdelka. Hkrati s tem je potrebno izdelati podroben izvedbeni načrt projekta in opredeliti potrebne vire projekta.

Uporaba ustrezne metodologije na projektu razvoja poslovnega informacijskega sistema lahko pohitri izvajanje postopka priprave projekta in omogoča hitrejši zagon aktivnosti izvedbe projekta ter dviguje kakovost projekta in produktivnost projektne skupine v smislu zmanjševanja znanih tveganj. Dobra metodologija razvoja novega informacijskega sistema mora biti enostavna za uporabo ter omogočati efektivno vodenje projekta, zato mora imeti

definirane postopke, zaporedje postopkov in izdelke, deliti življenjski cikel projekta v faze, omogočati razvid statusa projekta v vsakem trenutku, vsebovati tehnike za zagotavljanje in izboljšanje kakovosti sistema, imeti opredeljene vloge in odgovornosti sodelujočih na projektu, vsebovati mora aktivnosti planiranja in vodenja projekta, določati zahteve za dokumentacijo in njeno standardizacijo, vključevati postopke za zagotavljanje varnosti, nadzora in revizorske sledi v novem informacijskem sistemu ter mora biti poleg tega tudi podprta z orodjem, ki vsem tem postopkom nudi informacijsko podporo z namenom zviševanja produktivnosti projektne skupine.

Na projektu razvoja poslovnega informacijskega sistema na zavarovalnici Slovenica je zaradi izbire razvojnega CASE orodja Oracle Designer kot najustreznejša bila izbrana CDM/PJM metodologija, ki ustreza kriterijem dobre metodologije. Celoten življenjski cikel projekta deli v tradicionalne faze projekta, znotraj katerih ima definirane procese in zaporedje posameznih nalog, kar omogoča razvid statusa projekta v vsakem trenutku, poleg tega pa vsebuje tudi opise izdelkov posameznih nalog, napotke za njihovo izdelavo, ima opredeljene vloge in odgovornosti članov projektne skupine na posameznih nalogah ter kriterije za oceno kakovosti posameznih izdelkov. Poleg tega vsebuje še priporočila in standarde za izdelavo kakovostnih aplikacij. Njena največja učinkovitost je v povezavi z Oracle sistemom za upravljanje s podatki in CASE orodjem Oracle Designer. CDM metodologija izgradnje informacijskega sistema je nadgrajena s PJM metodologijo, ki opredeljuje aktivnosti planiranja in vodenja projekta ter zagotavlja kakovost projekta in produktivnost projektne skupine v smislu zmanjševanja znanih tveganj.

Z uporabo priporočil CDM/PJM smo dejansko dosegli bolj organiziran potek projekta. Skrajšali smo čas planiranja projekta in pohitrili pričetek izvajanja aktivnosti izvedbe projekta. Skrajšal se je čas izdelave izvedbenega načrta projekta, povečala se je popolnost izvedbenega načrta, zmanjšala možnost izvajanja neplaniranih aktivnosti ter povečala stopnja nadzora nad obsegom in viri projekta, medtem ko opisovanja izdelkov projekta sploh ni bilo potrebno izvesti, saj jih opisuje že CDM metodologija. Z uporabo predlogov organizacijske strukture smo pravočasno formirali ustrezno in zadostno ekipo strokovnjakov za izvedbo projekta.

Nadalje smo na račun uporabe standardov in priporočil CDM metodologija prihranili čas, potreben za razvoj novih standardov. Z uporabo orodja Headstart Utilities smo dosegli pohitritev razvoja, povečali implementacijo standardov pri izdelavi informacijskega sistema, razbremenili programerje ter avtomatizirali izdelavo poročil o kakovosti aplikacije in s tem posledično tudi skrajšali čas testiranja kakovosti končne aplikacije.

Ker po CDM metodologiji temeljijo vse aktivnosti izvedbenega dela na poslovnih modelih, definiranih v prvi fazi projekta, se vse aktivnosti projekta izvajajo urejeno in nadzorovano, prav tako pa se prihrani čas pri pripravi posameznih strategij za izvajanje aktivnosti procesov.

S hitrim prototipiranjem, ki ga omogoča CASE orodje Oracle Designer, je bila pri razvoju aplikacije izboljšana komunikacija in razumevanje s končnim uporabnikom.

Uporabnost CDM metodologije je predvsem v fazi planiranja projekta izredno močna, medtem ko je njena moč v izvedbi orientirana predvsem na uporabo CASE orodja Oracle Designer ter na procese definiranja poslovnih zahtev ter načrtovanja in izgradnje baze podatkov in aplikacije, medtem ko je šibkejša predvsem v procesih testiranja in tranzicije. PJM metodologija ima standarde in postopke v grobem teoretično razdelane, vendar so potrebna dopolnila glede specifičnosti samega projekta. Vsi postopki so definirani za ročno izvajanje aktivnosti, vendar Oracle ne ponuja tudi ustrezne informacijske rešitve, kar je predvsem pri večjih in zahtevnejših projektih velika pomanjkljivost, saj gre lahko za veliko porabo časa in virov za vpeljavo ustrezne informacijske podpore delovnim postopkom oziroma za ročno izvajanje postopkov.

Kriteriji za sprotno preverjanje kakovosti izdelkov pri posameznih nalogah, kot jih opisuje CDM metodologija, ter uporaba orodja Headstart Utilities za implementacijo in preverjanje ustreznosti implementacije sprejetih standardov so avtomatizirali, skrajšali in poenostavili postopke spremljanja nadzora kakovosti izdelkov projekta, medtem ko so opisi posameznih izdelkov in kriteriji kakovosti, kot jih CDM metodologija definira, pripomogli k enotnemu pogledu naročnika in izvajalca na končni izdelek posamezne naloge.

Metodologije ni potrebno v začetku sprejeti v celoti, lahko jo uvajamo postopno ali prilagodimo svojim razmeram. Vendar je zelo nesmiselno opuščati glavne koncepte metodologije (kot v našem primeru fazni pristop), ker se lahko kasneje poruši ves koncept prednosti uporabe metodologije, predvsem v najbolj kritičnih nalogah. V primeru, da vendarle spreminjamo koncept, je potrebno najti ustrezno drugo rešitev (na primer z dodatno razdelitvijo projekta na manjše, bolj obvladljive dele in uvedbo iterativnosti izvedbenih procesov in aktivnosti za posamezne dele projekta, s prototipiranjem ali kaj drugega).

Kljub uporabi metodologije pa ne moremo pričakovati, da bodo rešeni vsi problemi na projektu. V primerih, da naletimo na nepredvidene probleme, je poleg postopkov upravljanja z odprtimi vprašanji in tveganji, ki jih moramo opredeliti v planu projekta, potrebno kar najširše poznavanje poslovne problematike in informacijske tehnologije.

Pri izgradnji informacijskega sistema je priporočljivo uporabiti metodologijo s smiselnimi prilagoditvami in spremembami glede na konkretni projekt, saj pravilna uporaba metodologije bistveno pripomore k uspešnemu izvajanju nalog projekta, kar je predstavljeno tudi v tej nalogi.

Literatura

1. Black Rex: Managing the Testing Process. B.k.: Microsoft Press, 1999. 381 str.
2. Butler Janet G.: Strategic Planning for Enterprise Information Systems. First Edition, B.k.: Computer Technology Research Corp., 1996. 182 str.
3. Clegg Dei, Barker Richard: CASE Method Fast Track, a RAD approach, Wokingam, England (etc.), Addison Wesley, 1994. 207 str.
4. Finkelstein Clive: Information Engineering: Strategic System Development, 1.del. Singapore: Addison-Wesley Publishing Company, 1992.
5. Galliers R.D., Leidner D.E. and Baker B.S.H.: Strategic Information Management. Second Edition. B.k.: Butterworth Heinemann, 1999. 590 str.
6. Hayes Linda G.: An Admission Ticket to Testing [URL: http://itmanagement.earthweb.com/appdev/print/0,,11963_713071,00.html], 25.4.2001
7. IT Governance Institute: COBIT 3rd Edition Executive Summary, July 2000, [URL: <http://www.isaca.org/exec1.htm>], 6.4.2002
8. McConnell Steve: Software Project Survival Guide. B.k.: Microsoft Press, 1998. 288 str.
9. McLeod Graham, Smith Derek: Managing Information Technology Projects. Cambridge: An International Thomson Publishing Company, B.l..386 str.
10. Oracle Method: CDM Classic Method Handbook. Release 2.6.0. B.k.: Oracle Corporation, February 2000. 248 str.
11. Oracle Method: CDM Classic Process and Task Reference. Release 2.6.0. B.k.: Oracle Corporation, February 2000. 845 str.
12. Oracle Method: CDM Standards and Guidelines Library, Volume 1 - Requirements Modeling Using Oracle Designer, Release 6.0.0. B.k.: Oracle Corporation, February 2000. 212 str.
13. Oracle Method: CDM Standards and Guidelines Library, Volume 2 - Design and Generation of Multi-Tier Web Applications, Release 2.0.0. B.k.: Oracle Corporation, February, 2000. 791 str.
14. Oracle Method: CDM Standards and Guidelines Library, Volume 3 – Building Sysems Using Oracle 8 Programming Tools, Release 6.0.0. B.k.: Oracle Corporation, February 2000. 102 str.

15. Oracle Method: CDM Standards and Guidelines Library, Volume 4 – CDM Standards, Version 6.0.0. B.k.: Oracle Corporation, March, 2000.
16. Oracle: Custom Development Deliverable Reference. Release 2.0.1. B.k.: Oracle Corporation, 1996. 388 str.
17. Oracle: Project Management Method Handbook. Release 2.0.1. B.k.: Oracle Corporation, 1996. 156 str.
18. Oracle: Project Management Process and Task Reference. Release 2.0.1. B.k.: Oracle Corporation, 1996. 236 str.
19. Ruthberg Zella G., Fisher Bonnie T., Lainhart John W. IV: System Development Auditor. B.k.:Elsevier Science Publisher Ltd, 1991. 279 str.
20. Siemens Business Services: Chestra, Siemens Business Services Methodology Framework [URL: http://www.sbs.lu/upload/doc/portfolio/lux/chesmet_en.pdf], 6.4.2002
21. Sturm Jake: Professional Visual Basic Project Management. Reprinted. B.k.: Wrox Press, December 1999. 551 str.
22. Vallabhaneni s. Rao: CISA Examination Textbooks, Volume 1: Theory. Second Edition. B.k.: SRV Professional Publications, 1996. 995 str.
23. Warren J. Donald Jr., Edelson Lynn W., Parker Xenia Ley: Handbook of IT Auditing. 1998 Edition. Warren, Gorham & Lamont, 1998. 1127 str.
24. Wiegers Karl E.: Writing Quality Requirements [URL: <http://www.processimpact.com/articles/qualreqs.html>], 18.5.2001
25. Wilson Scott F.: Analyzing Requirements and Defining Solution Architectures. B.k.:Microsoft Press, 1999. 724 str.

Viri

1. Mednarodni standard ISO 12119: Information technology – Software packages – Quality requirements and testing. First edition 1994-11-15, ISO/IEC 1994. 16 str.
2. Mednarodni standard ISO 12207: Information Technology – Software life cycle processes. First edition 1995-08-01, ISO/IEC 1995. 57 str.
3. Mednarodni standard ISO 9000-3: Quality management and quality assurance standards-Part3: Guidelines for application of ISO 9001: 1994 to the development, supply, installation and maintenance of computer software. Second edition 1997-12-15, ISO/IEC 1997. 32 str.

4. Mednarodni standard ISO 9126: Information technology – Software product evaluation – Quality characteristics and guidelines for their use. First edition 1991-12-15, ISO/IEC 1991. 12 str.
5. Rational Unified Process Overview: [URL: http://www.rational.com/demos/viewlets/rup/rup_platform/RUP_Platform_viewlet.html], 6.4.2002
6. Slovenica, Adriatic: Organizacija projekta, vloge in naloge članov projektne skupine
7. Slovenica: Zagonski elaborat projekta Izgradnja informacijskega sistema za podporo produkcijskih poslovnih procesov
8. The Standish Group International, 1998: CHAOS: A Recipe for Success, [URL: http://www.pm2go.com/sample_research/chaos1998.pdf], 5.3.2002