

UNIVERZA V LJUBLJANI
EKONOMSKA FAKULTETA

DIPLOMSKO DELO

**ZAMENJAVA INFORMACIJSKEGA SISTEMA V PODJETJU
ADRIA MOBIL, d.o.o.**

Ljubljana, maj 2003

GREGOR MALNARIČ

IZJAVA

Študent Gregor Malnarič izjavljam, da sem avtor tega diplomskega dela, ki sem ga napisal pod mentorstvom dr. Mojce Indihar Štemberger in dovolim objavo diplomskega dela na fakultetnih spletnih straneh.

V Ljubljani, dne 14. maj 2003

Podpis: _____

KAZALO

1.	UVOD	1
2.	ZAMENJAVA INFORMACIJSKEGA SISTEMA.....	2
2.1.	NAČINI ZAMENJAVE IS	3
2.2.	VZROKI ZA ZAMENJAVO	4
2.3.	METODOLOGIJA ŽIVLJENJSKEGA CIKLA SISTEMA.....	5
2.4.	OBJEKTNO ORIENTIRANE METODOLOGIJE.....	6
2.4.1.	Poenoteni jezik modeliranja – UML	6
2.4.2.	Objekt.....	7
2.4.3.	Razred.....	8
2.4.4.	UML diagrami.....	10
2.5.	PROTOTIPNI PRISTOP.....	11
3.	SHELLY/CASHMAN/ROSENBLATTOV ŽIVLJENJSKI CIKEL RAZVOJA INFORMACIJSKEGA SISTEMA	13
3.1.	NAČRTOVANJE PROJEKTA	13
3.2.	SISTEMSKA ANALIZA	15
3.2.1.	Strukturiranje potreb sistema	15
3.2.2.	Analiziranje potreb sistema	16
3.2.3.	Alternative in možne strategije	17
3.3.	NAČRTOVANJE SISTEMA.....	17
3.3.1.	Oblikovanje izhodov	18
3.3.2.	Oblikovanje vhodov	18
3.3.3.	Oblikovanje datoteke in baze podatkov	18
3.3.4.	Računalniška platforma.....	19
3.4.	IMPLEMENTACIJA SISTEMA	20
3.4.1.	Razvoj aplikacije.....	20
3.4.2.	Namestitev in vrednotenje.....	21
3.5.	IZVAJANJE IN VZDRŽEVANJE SISTEMA	21
4.	ZAMENJAVA IS V PODJETJU ADRIA MOBIL, D.O.O.	22
4.1.	PREDSTAVITEV ADRIE MOBIL, d.o.o.....	22
4.2.	INFORMACIJSKI SISTEM PRED ZAMENJAVO.....	23
4.2.1.	Analiza Infotehne	25
4.2.2.	Mnenje skrbnikov aplikacij.....	25
4.3.	PROJEKT PRENOVE IS	26
4.3.1.	Cilji projekta.....	27
4.3.2.	Opredelitev projektne organizacije	27
4.3.3.	Komunikacijska mreža	30
4.3.4.	Strojna oprema	31
4.3.5.	Izbira ponudnika IS	31
4.4.	UPRO – Upravljanje PROizvodnje.....	33
4.5.	PREVERBA REŠITEV Z UPORABNIKI	34
4.6.	PREVZEM PODATKOV IZ OBSTOJEČIH APLIKACIJSKIH BAZ PODATKOV	35
4.7.	NAMESTITEV PROTOTIPNE RAZVOJNE APLIKACIJE.....	35
4.8.	IZVEDBA POSEBNIH OZ. DODATNIH ZAHTEV V APLIKACIJAH	36
4.9.	USPOSABLJANJE UPORABNIKOV	39

4.10.	UVAJANJE IN NAMESTITEV PROGRAMSKEGA PAKETA	39
4.11.	ZAVAROVANJE PODATKOV	40
4.12.	PREDNOSTI IN SLABOSTI NOVEGA IS	41
5.	SKLEP	42
6.	LITERATURA	44
7.	VIRI	45

PRILOGE

KAZALO SLIK IN TABELE

Slika 1: Načini uvajanja novega sistema.....	4
Slika 2: Predvideno gibanje stroškov in koristi IS	5
Slika 3: Objekt.....	7
Slika 4: Razred	8
Slika 5: Diagram razredov.....	11
Slika 6: Življenjski cikel razvoja sistema.....	14
Slika 7: Mrežna arhitektura obstoječega sistema	24
Slika 8: Organizacijska struktura projektnega tima	28
Slika 9: Program UPRO	37
Slika 10: Obremenitev mreže pri uporabi SUBP Paradox	38
Slika 11: SUBP InterBase	38
Slika 12: Diagram razvrstitve računalniške platforme za Adrio Mobil	40

KAZALO PRILOG

Priloga 1: Poslovne in kapitalske povezave skupine Adria.....	I
Priloga 2: Poslovanje Adrie Mobil, d.o.o.....	II
Priloga 3: Moduli UPRO.....	V
Priloga 4: Dokument sprememb in dopolnitev	IX
Priloga 5: Slovarček slovenskih prevodov tujih izrazov	XI

1. UVOD

Informacijski sistem (v nadaljevanju IS) je del poslovnega sistema, ki ima skupaj z ostalimi sistemi in podsistemi cilj, da ustvarja pozitivne poslovne rezultate, in je eden od temeljnih sistemov za zagotavljanje nemotene proizvodnje in obvladovanje kakovosti. IS je niz postopkov, ki zbirajo, procesirajo, hranijo in dajejo informacije ter so podlaga za odločanje in vodenje.

Podjetje, ki se zaveda pomembnosti uporabe IS in moderne informacijske tehnologije, lahko močno poveča konkurenčno prednost organizacije, svojo učinkovitost in uspešnost. Splošna izhodišča za sodoben integrirani IS bi tako bila (Kovačič, 1997, str.11):

- izdelan mora biti s sodobnim orodjem, ki omogoča objektni pristop k razvoju in uporabi rešitve;
- imeti mora z odzivom na dogodke krmiljeno zasnovo;
- rešitev ni pogojena z določenim operacijskim sistemom;
- rešitev mora biti na voljo v izvorni ali vsaj parametrizirani kodi;
- rešitev mora uporabljati skupno in enovito bazo podatkov;
- omogočena mora biti tipizirana uporaba funkcij na vseh modulih;
- dobavitelj mora zagotavljati nemoteno in kvalitetno delo na programski rešitvi in
- vnaprej mora biti definirano vzdrževanje programske rešitve.

Diplomska naloga je sestavljena iz teoretičnega in praktičnega dela. V prvem delu bom predstavil vzroke, ki jih podjetja navajajo, ko se odločijo za zamenjavo IS. Nadalje bom predstavil tri pristope za razvoj sistemov. Pri izgradnji in načrtovanju sistemov lahko opredelimo tri pomembnejše. To so linearni pristop, objektni pristop in pristop prototipa (Kovačič, Vintar, 1994, str. 43-50).

Linearni pristop predstavlja niz zaporednih faz, za katere je značilno, da se nobena ne začne, dokler ni končana predhodna. V okviru tega pristopa bom v 3. poglavju opisal metodologijo življenjskega cikla razvoja sistema, ki sodi med starejše. Posebnost te metodologije je v ločevanju procesnega in podatkovnega vidika.

V 90-ih je linearni pristop razvoja IS izpodrinil objektni pristop. Razvile so se številne objektno orientirane metodologije. Pri teh metodologijah se kompleksnost problemov razbije na več manjših, ki so lažje rešljivi. Sam bom predstavil poenoteni jezik modeliranja – UML (ang. Unified Modeling Language), ki je med razvijalci najbolj razširjen.

S prototipnim pristopom pa skušamo ugotoviti možnosti izdelave novega IS, njegove lastnosti, stroške ipd. Po končanju prototipne faze se lahko prototipni sistem zamenja z novim, oz. se ga dokončno namesti in da v uporabo. Katero metodologijo bomo uporabili, prototipni pristop ne določa. Opisal ga bom v razdelku 2.5.

Metodologij življenjskega cikla razvoja in UML ne bom opisoval podrobno, ampak ju bom razdelil v dva sklopa. V 3. poglavju bom opisal življenjski cikel razvoja sistema. Pri tem bom dal več poudarka nalogam, ki jih je potrebno opraviti v posameznih fazah razvoja. Tej metodologiji bom posvetil več prostora, ker zelo nazorno prikaže sam potek razvoja, predvsem pa opozarja na vsebino posameznih faz in korakov. Ta del bom tudi najbolj približal vsebini praktičnega dela naloge. V razdelku 2.4. bom predstavil metodologijo UML. Tukaj se ne bom spuščal v podrobnosti, ampak bom opisal le osnovne pojme, zaradi katerih se tudi imenuje objektno orientirana metodologija. Izbral sem jo zato, ker je med najbolj uporabljenimi, hkrati pa je programska aplikacija za Adrio Mobil razvita po konceptih objektnega modeliranja.

V praktičnem delu bom v 4. poglavju predstavil proces zamenjave IS, ki so ga izvedli v podjetju Adria Mobil, d.o.o., iz Novega mesta, z osnovno dejavnostjo počitniških prikolic in avtomobov. Podjetje se je odločilo za nakup celovitega IS. To so sistemi, ki upravljajo vse razpoložljive vire, sredstva in aktivnosti v določeni organizaciji. Te sisteme tvorijo funkcionalni moduli, ki se jih da kupiti in implementirati ločeno po potrebi in neodvisno. V diplomski nalogi bom predstavil uvedbo takega sistema od faze odločanja, ali je delujoči sistem sploh potrebno zamenjati, pa vse do implementacije, vključno z ocenitvijo novega sistema. Navadno se pri uvajanju celovitih IS prilagaja potek organizacijskih procesov in ne obratno. V primeru Adrie pa se je sistem v celoti prilagodil potrebam podjetja, zato bom v nadaljevanju uporabljal termin razvoj IS.

Ugotovitve, ki izhajajo iz analize teoretičnega in praktičnega dela, bom zapisal v sklepu diplomske naloge. Namen naloge je najti povezave med metodologijama in dejanskim razvojem IS v Adrii Mobil.

Na koncu je naveden spisek literature in virov. Kot dodatek k diplomski nalogi je priloženih 5 prilog, ki razširjajo vsebino naloge.

2. ZAMENJAVA INFORMACIJSKEGA SISTEMA

Da lahko podjetja v konkurenčnem poslovnem okolju preživijo, morajo spreminjati svoje poslovne procese. To pa velja tudi za razvoj IS, ki se mora razvijati skupaj s podjetjem. Tako kot se spreminjajo kultura organizacije, struktura organizacije, poslovni procesi in sami ljudje v organizaciji, tako se mora razvijati tudi IS. Zato je pomembno, da je njegovo načrtovanje tesno povezano z razvojem organizacije.

Ko se podjetja odločajo za zamenjavo ali prenovo IS, je potrebno pretehtati in preučiti, kaj to za podjetje pomeni. Na eni strani imamo analitike in velika podjetja, ki zagovarjajo prenovo procesov. Ti naj bi se v večji meri prilagodili IS, ki je kot celovita rešitev ponujen na trgu. Bistvo celovitih rešitev je v tem, da poskušajo vzpostaviti večji pretok informacij in povezati različne službe v organizaciji, hkrati pa ponuditi avtomatizirane rešitve za čim večji obseg

poslovnih procesov. Na ta način naj bi se povečala učinkovitost in dodana vrednost. Na drugi strani so zagovorniki, ki pravijo, da ima vsaka organizacija svoje specifične poslovne procese in da ni IS, ki bi bil enako uspešen za več podjetij. V prid slednjim govorijo študije Boston Consulting Group, ki so pokazale, da je uspešna šele vsaka tretja implementacija celovitih programski rešitev. Najpogostejši vzroki za neuspeh so (Kovačič et al., 2002, str. 189-205):

- pomanjkljiva specifikacija zahtev;
- slabo definiran pogodbeni odnos med naročnikom in ponudnikom ter slaba medsebojna komunikacija;
- nezadosten strateški načrt naročnika;
- prikriti stroški uvajanja (s strani informacijske službe ali naročnika);
- neustrezno izobraževanje uporabnikov.

Trenutni trend je, da naj bi informacijske rešitve v povprečju pokrivalo največ 70 % potreb organizacije, za ostalih 30 % pa mora podjetje prilagoditi svoje procese. Pred tem je veljalo pravilo, da je nakup smiseln, če programska rešitev pokriva vsaj 80 % informacijskih potreb (Kovačič et al., 2002, str. 189-190).

Moderen IS bi lahko opredelili kot učinkovito uporabo strojne in programske opreme, podatkov, postopkov in ljudi, da dosežemo končne rezultate, ki koristijo organizaciji. Strojna oprema je ves fizičen inventar v organizaciji, ki se uporablja za vnašanje, shranjevanje, pretok in izpis podatkov. Programska oprema omogoča delovanje strojne opreme in procesiranje podatkov v informacije. Za to pa so potrebni postopki, ki jih izvajajo uporabniki v organizaciji (Shelly, Cashman, Rosenblatt, 1998, str. 1.4-1.6).

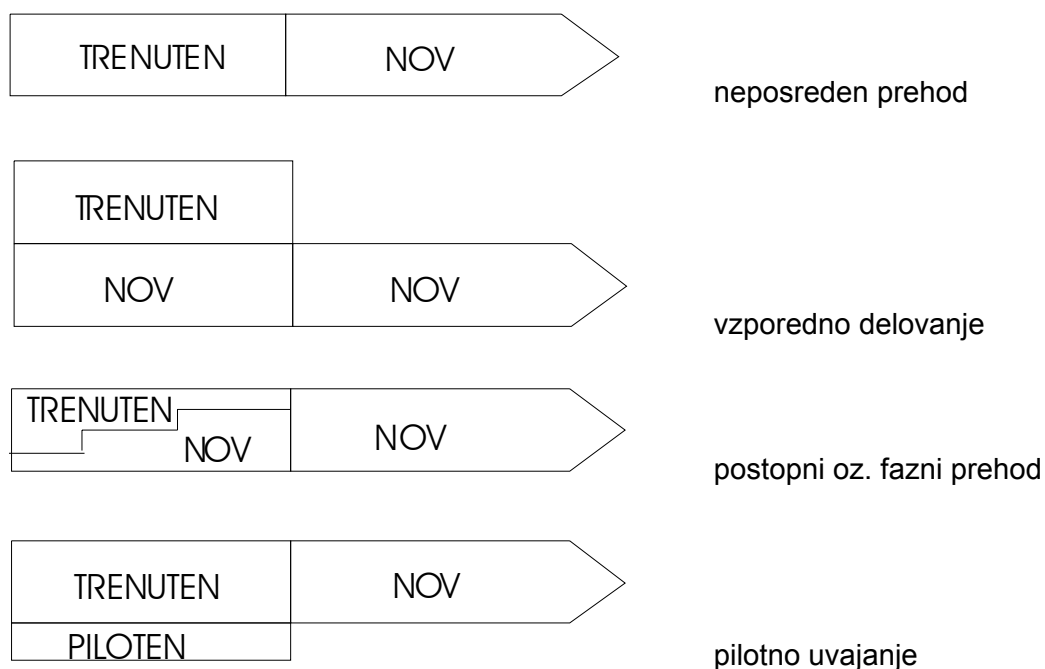
2.1. NAČINI ZAMENJAVE IS

Prehod na nov sistem je zelo pomemben in občutljiv, saj so ljudje že po naravi nezaupljivi do novih stvari. Zato je toliko bolj pomembno, kako jih na to pripraviti. Dobro je, da jim sistem predstavimo že med samim izobraževanjem. Glede izbire prehoda na nov sistem se v praksi uporabljajo 4 pristopi (Slika 1) (Shelly, Cashman, Rosenblatt, 1998, str. 11.10-11.13):

- **Neposreden prehod** je najlažji, najhitrejši, vendar povezan z največjim tveganjem. Smiselno ga je uporabiti, kadar smo prepričani, da je tak način prehoda možen in smiseln.
- **Vzporedno delovanje** je izredno naporno, saj zahteva dvojno delo, je pa najbolj varno. Če se izkaže, da ima nov sistem preveč pomanjkljivosti, lahko še vedno delamo na starem, dokler se pomanjkljivosti novega ne odpravijo. Vzporedno delo traja toliko časa, dokler se ne preizkusijo vse možne operacije.
- **Postopni prehod** pomeni, da postopoma vključujemo posamezne module novega sistema, ki zamenjujejo stare. Tak način je smiseln, ko sta si oba sistema zelo podobna in kadar delovanje modulov ni soodvisno. Traja pa ta način dlje časa.
- **Pilotno uvajanje**, ki so ga uporabili tudi v Adrii, se izvede na določeni lokaciji v podjetju, kjer se namesti nov sistem. Ko se izkaže, da je sistem zanesljiv, se namesti

celoten sistem. Ta način zmanjša tveganje in je cenejši kot pa pri vzporednem delovanju.

Slika 1: Načini uvajanja novega sistema



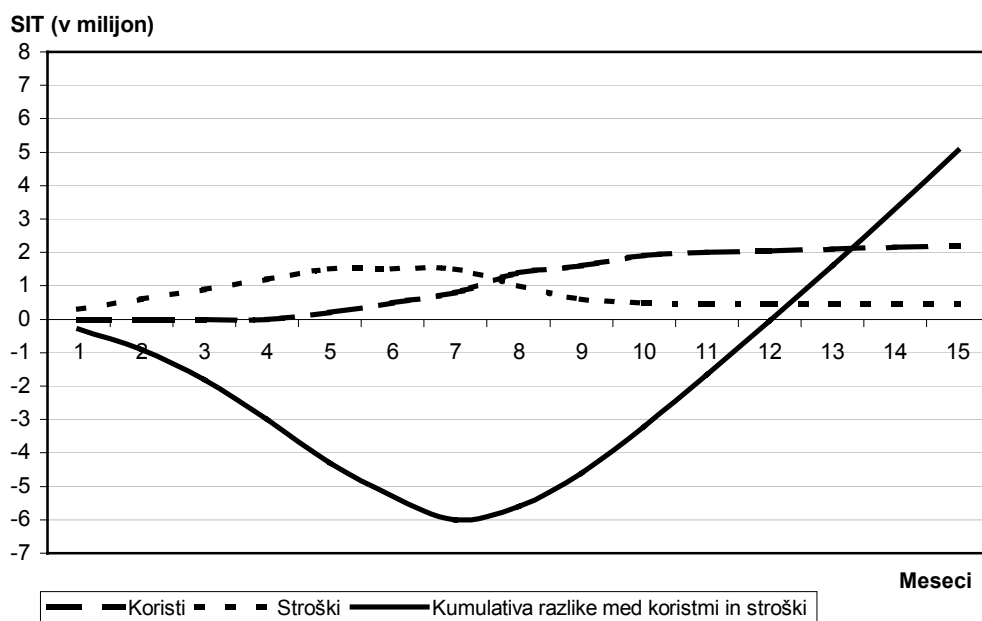
Vir: Gradišar, Resinovič, 2001, str. 428.

2.2. VZROKI ZA ZAMENJAVO

Najprej se moramo vprašati, kje so razlike med uspešnimi in manj uspešnimi podjetji. Te se med drugim pojavijo tudi zaradi učinkovitosti poslovnih in delovnih procesov v organizaciji in usklajenosti med njimi. IS, ki je še do pred kratkim zagotavljal nemoteno poslovanje in podjetju nudil konkurenčne prednosti, lahko postane kmalu neustrezen. Večletni nespremenjeni poslovni procesi, večje povpraševanje in ostalo silijo podjetja v prekomerno novo zaposlovanje, da zadostijo potrebam. Poslovno okolje pa se je z leti spremenilo, in to čez meje, do katerih so se zmožna podjetja prilagoditi (Hammer, Champy, 1995, str. 21). To je podobno, kot če računalnik nadgrajujemo z zmogljivejšim procesorjem, programske opreme pa ne spreminjamo. Računalnik ne bo deloval nič bolje ali drugače, le nekoliko hitrejši bo, pa še to ima svoje meje. Pri podjetju je enako in iz tega sledi, da okolje sili podjetja v prenavo. Organizacija je živ sistem, ki se razvija. Informacijske potrebe se spreminjajo z rastjo, zrelostjo in reagirajo tako na zunanje kot notranje sile. To še posebej velja za dinamično poslovno okolje, kjer so nenehne spremembe stalnica. Da je organizacija uspešna, potrebuje tak IS, ki bo kos nenehnim spreminjajočim se potrebam (Shelly, Cashman, Rosenblatt, 1998, str. 1.16). Seveda ga lahko nadgrajujemo. Vendar pa nadgradnja sčasoma postane predraga, sistem pa nezdružljiv z ostalo programsko in strojno opremo.

Zamenjava predstavlja za podjetje veliko investicijo. Vodstvu ni težko odšteti milijonov za nov stroj, strežnik, LCD¹ monitorje ali avtomobil. Pri vrednotenju IS pa so večinoma skeptični, saj izvajalec težko pokaže kaj oprijemljivega, medtem ko stroški (četudi ne upoštevamo stroškov strojne opreme) v začetni fazi hitro naraščajo. Koristi so večinoma posredne in se pokažejo šele kasneje (Slika 2). Vendar to ne bo tema diplomske naloge, je pa vzrok, da podjetja odlašajo z zamenjavo IS, s tem pa izgubljajo konkurenčno tekmo.

Slika 2: Predvideno gibanje stroškov in koristi IS



Vir: Gradišar, Resinovič, 2001, str. 420.

Za razvoj IS se uporabljajo različne metode² in metodologije³. Poleg metodologij življenjskega cikla razvoja sistema in UML ter metoda prototipa, ki ju bom opisal v nalogi, obstaja še veliko drugih (Kovačič, Vintar, 1994, str. 40). Med pogosteje uporabljene pa spadajo tradicionalni pristop, razvoj s strani končnih uporabnikov in uporaba programskih paketov.

2.3. METODOLOGIJA ŽIVLJENJSKEGA CIKLA SISTEMA

Življenjski cikel sistema je klasična metodologija. Uporablja se pri obsežnih projektih, ki imajo iterativno naravo procesa razvoja IS. Cikel poteka fazno in ima 4 korake: začetek, razvoj, uvajanje in izvajanje ter vzdrževanje. V praksi in literaturi se za ta pristop uporablja več sopomenk. Zasedimo lahko še tradicionalni življenjski cikel razvoja, tradicionalna sistemska analiza, model slapa, linearni pristop, življenjski cikel razvoja sistema (ang.

¹Ekran s tekočimi kristali – ploski ekran (ang. Liquid Crystal Display).

²Oblika načrtnega, premišljenega dejanja, ravnanja ali mišljenja za doseg kakega cilja; način, postopek (SSKJ, 1994, CD-rom).

³Skupek metod, ki se uporabljajo pri kakem raziskovanju, mišljenju (SSKJ, 1994, CD-rom).

Systems Development Life Cycle – SDLC). V nadaljevanju bom opisal Shelly/Cashman/Rosenblattov življenjski cikel razvoja sistema – SDLC. Ta metodologija ni vezana na noben programski paket za razvoj IS ali na programski jezik in uporablja zmes običajnega razvoja z možnostjo uporabe trenutnih tehnologij in orodij. Osnova metodologije je, da si faze, ki jih je 5, sledijo zaporedno. Za razvoj uporablja različne diagrame in postopke. Ti so podlaga naslednjim fazam, med njimi pa obstaja velika medsebojna odvisnost.

2.4. OBJEKTNO ORIENTIRANE METODOLOGIJE

Vse hitrejši razvoj tehnologij in potreb po informacijah so vzrok za vse večjo kompleksnost IS. V sredini 70-ih let so se zato začele razvijati prve objektno orientirane metodologije, katerih ideja je bila, da je potrebno zapletenost sistema z dekompozicijo zapletenih procesov razbiti na večje število enostavnejših. Te metodologije pa so se začele močno razvijati v začetku 90-ih let. Objektni pristop poskuša objekte realnega sveta preslikati v objekte objektnega modela. Ta pristop je glede na druge pristope naravnejši. Objektno modeliranje združuje modeliranje podatkov in modeliranje procesov, kar je prednost glede na ostale pristope (Veljović, 2002, str. 1-10).

Cilj objektnega pristopa je razvoj programske opreme po principu razvoja strojne, to je iz množice standardiziranih sestavnih delov. Posamezni objekti se lahko sestavljajo v kompleksne objekte po pravilih generalizacije, kar pelje v hierarhijo objektov. Prednost metodologije narašča s kompleksnostjo sistema, saj ta pristop omogoča (Kovačič, Vintar, 1994, str. 49):

- večkratno uporabo istih objektov, zato zagotavlja zanesljivost in kakovost rešitev;
- njihovo testiranje je neodvisno;
- omogoča lažje timsko delo in
- večjo možnost prilagoditve.

Glede na klasičen pristop, ki je opisan v naslednjem poglavju, omogoča objektno orientiran razvoj IS povečanje produktivnosti, povečanje kakovosti izvedenih projektov, poenostavitve vzdrževanja in nadgradnje. Objektno orientirane metodologije uporabljajo prvine, ki omogočajo boljše komuniciranje med razvijalci IS in uporabniki. Predvsem grafične poenostavitve omogočajo nazornejšo predstavitev problema tudi ostalim, ki niso razvijalci. S tem pa se zmanjša predvsem pogostost nespornostov, kar prihrani čas izvedbe.

2.4.1. Poenoteni jezik modeliranja – UML

Poenoteni jezik modeliranja je orodje za objektno modeliranje IS. Uporablja se za poslovno modeliranje, programsko modeliranje za vse faze razvoja in za vse vrste sistemov, tudi za generalno modeliranje, s čimer se definirajo statične strukture in dinamično obnašanje. To je jezik za specifikacijo, konstrukcijo, vizualizacijo in dokumentacijo programske opreme (Veljović, 2002, str. 2). UML ni vezan na noben določen programski jezik za razvoj objektno

orientiranih sistemov, ampak vsebuje splošni model načrtovanja. Zato lahko zaključimo, da je UML jezik za modeliranje oz. za opisovanje modelov. Njegova največja moč je v zmogljivosti modela, interaktivnem načinu dela in iterativnosti elementov skozi razvojni proces. Za prikaz teh modelov pa UML uporablja več diagramov, ki vsebujejo različne komponente ter medsebojne povezave, odnose in omejitve.

Razvoj IS z UML ima 4 faze. Te faze so (Veljović, 2002, str. 9):

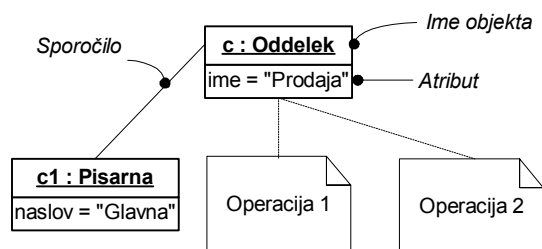
- **definiranje zahtev**, kjer se izdelata logični model funkcije in fizični model poslovnih procesov;
- v **objektno orientirani analizi** se izdelajo konceptualni model in diagrami sekvenc ter definirajo operacije;
- **objektno orientirano načrtovanje** je namenjeno izdelavi diagramov sodelovanja, razredov, stanj in definiranje paketov ter
- **implementacija**, ki vsebuje izdelavo aplikacije, definiranje tehnologij in mrežne arhitekture, testiranje, izvajanje ter vzdrževanje sistema.

Faze se za razliko od metodologije življenjskega cikla ne izvajajo zaporedno, ampak se prepletajo. Prav to pa je posebnost objektnih metodologij, saj se elementi, ki so bili razviti v prejšnjih fazah, lahko uporabijo večkrat. To pomeni, da se novega problema ni potrebno lotiti s ponovnim podrobnim izvajanjem začetnih faz. V diplomskem nalogi ne bom opisoval posameznih faz razvoja, ampak bom v nadaljevanju opisal le nekatere pomembnejše pojme in komponente, ki se uporabljajo pri UML jeziku.

2.4.2. Objekt

Osnova modeliranja z UML so objekti (Slika 3). Objekt je poljubna stvar, abstraktna ali realna, o kateri hranimo podatke in operacije nad njimi (Grad, Jaklič, 1996, str. 117-125). Objekt je lahko proces skladiščenja, račun, material itn. Objektni model je le približek realnega stanja, ki omogoča delitev problemskih stanj na manjše enote, da so razumljivejše in lažje rešljive.

Slika 3: Objekt



Vir: Booch, Rumbaugh, Jacobson, 1999, str. 196.

Za objekte so značilna stanja, obnašanje in enolična identiteta. Stanje je definirano z vrednostjo njegovih atributov. Obnašanje objekta opisujejo operacije, ki predstavljajo način za pregledovanje in spreminjanje vrednosti atributov. Enolična identiteta pa omogoča razločevanje med njimi. Skupina objektov je razred objektov. Interakcije med objekti ponazorimo v diagramih sodelovanja, diagramih zaporedij in v diagramih objektov. Grafično jih ponazorimo s pravokotnikom, ki ima dve polji, opombe pa označimo s pravokotnikom z zavihkom.

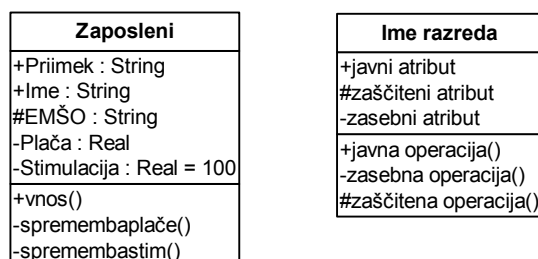
Objekti med seboj tudi komunicirajo. To jim omogočajo sporočila, ki zagotavljajo dinamične interakcije med objekti. Ob upoštevanju pravil in omejitev lahko objekte spreminjamo, kreiramo ali brišemo, lahko pa si tudi samo izmenjujejo podatke. Da se to izvede, so potrebne operacije, ki se uporabljajo za branje in za manipulacije nad podatki o objektu in izražajo obnašanje objekta. Sporočila oz. zahteve povzročijo, da se operacije nad objekti izvedejo. Vendar pa omogočajo manipulacijo nad objekti le preko njegovih operacij. Mehanizem skrivanja informacij pred zunanjim svetom z izključnim pristopom do podatkov preko operacij objekta se imenuje ograjevanje. Ta lastnost ščiti objekt pred izgubo podatkov in zunanjim vdorom. Operacije so uporabnikom sicer znane, ne vedo pa, kako se izvedejo.

Atributi so podatki, ki specifično določajo objekt. Z njimi dobi objekt svoje vrednosti in se uporabljajo za opisovanje primerkov razreda.

2.4.3. Razred

Razred (Slika 4) je najpomembnejši element v objektno orientiranih sistemih. Definicija predstavlja objekte, ki imajo enake attribute, operacije, povezave in pomen. Razred grafično ponazorimo s pravokotnikom, ki ima tri polja. Znaki »+«, »#« in »-« določajo tri nivoje zaščite atributov in operacij razredov. Ograjevanje se tako kot pri objektih, uporablja tudi pri razredih. Ta mehanizem preprečuje nepooblašcene dostope do podatkov in zagotavlja neodvisnost aplikacij.

Slika 4: Razred



Vir: Veljović, 2002, str. 128.

Med razredi obstajajo relacije oz. povezave, ki predstavljajo statično strukturo sistema in jo ponazorimo z diagramom razredov. UML uporablja štiri tipe povezav (Booch, Rumbaugh, Jacobson, 1999, str. 61-74, 135-153, 169-181; Veljović, 2002, str. 128-148, 166-169):

- asociacije,
- agregacija, kompozicija in odvisnost,
- generalizacija in
- realizacija.

Asociacija je strukturalni odnos, ki specificira povezavo objekta na eni strani z objektom na drugi strani. Prikaže nam abstraktno predstavitev odnosov realnega sveta. Primer asociacije med razredoma je povezava med njima.

Asociacije grafično ponazorimo s črto med razredi in z imenom asociacije (Slika 5, na str. 11). Za ime uporabljamo glagol, smer branja pa določimo s puščico. Za še podrobnejši opis damo na vsako stran črte tudi ime vloge, ki pove, kako razred vidi drugi razred preko asociacije. Ta opis je nujen, če uporabimo n-kratne asociacije, ki povezujejo več razredov ali pa, če je asociacija rekurzivna. Ta ne kaže na noben drug razred, ampak na samo sebe.

Asociacijam lahko določimo tudi kardinalnost oz. števnost. Kardinalnost določa, koliko objektov enega razreda je dovoljeno, da se povežejo z objekti drugega razreda. Poznamo 5 tipov kardinalnosti (Veljović, 2002, str. 132):

- **1** samo eden,
- ***** poljubno pozitivno celo število,
- **0..1** nobeden ali eden,
- **1..*** od enega do poljubnega pozitivnega celega števila,
- **0..*** od nobenega do poljubnega pozitivnega celega števila.

Agregacija je poseben tip asociacije, pri kateri se vzpostavi nesimetrična povezava tipa celota/del (ang. whole/part). Grafično jo označimo s črto z nepobarvanim romбом na strani agregata oz. celote. Del je pri agregaciji lahko skupen več agregatom.

Posebna oblika agregacije je **kompozicija** (Slika 5, na str. 11). Označena je enako kot agregacija, le da je romb pobarvan. Razlika pa je, da je kardinalnost kompozita oz. celote lahko le 0 ali 1. To pomeni, da podrejeni razred ne more obstajati brez nadrejenega razreda.

Odvisnost (Slika 5, na str. 11) predstavlja povezavo med dvema elementoma modela (razred, vmesnik ali paket), s katerim se definira funkcionalnost ali implementacija. V tej povezavi en element zahteva prisotnost drugega. Odvisnost povezuje elemente, pri katerih neodvisen element vpliva na obnašanje odvisnega. Odvisnost je šibkejša oblika povezave razredov, kjer odvisen razred nima podatkov o drugem razredu. Grafično jo ponazorimo kot prekinjeno črto, usmerjeno na odvisen element.

Paketi omogočajo deljenje modelov in grupiranje različnih elementov v procesu modeliranja. Vsebujejo razrede, objekte, povezave in komponente, odvisno od modela. Paket lahko vsebuje tudi podpakte, število nivojev pa ni omejeno. Grafično jih ponazorimo s pravokotnikom, ki ima na vrhu levo še en majhen pravokotnik. Pakete povežemo s tipom povezave odvisnosti. Uporabimo jih samo v fazi razvoja.

Generalizacija predstavlja hierarhično povezavo med razredi. Grafično jo označimo s črto in votlo puščico na strani nadrazreda (Slika 5, na str. 11). Z generalizacijo definiramo odnos med bolj splošnim elementom – nadrazred in bolj specifičnim elementom – podrazred. Podrazredi dedujejo vse attribute, operacije in asociacije od nadrazreda, poleg tega pa vsebujejo tudi svoje specifične attribute. Obratni postopek je specializacija. Dedovanje je lahko enostavno ali večkratno. Pri enostavnem ima vsak podrazred natančno en nadrazred. Pri večkratnem dedovanju pa ima podrazred dva ali več nadrazredov. Dedovanje pomeni identifikacijo in definiranje sistema hierarhij med razredi, tako da podrazred lahko podeduje attribute in operacije nadrazreda.

Realizacija je posebna vrsta povezave, ki jo lahko uvrstimo med agregacijo in generalizacijo. Grafično jo označimo kot prekinjeno črto z votlo puščico. Realizacija se uporablja za označevanje povezave med razredom in vmesnikom ali pa za komponente, ki oskrbujejo razred z operacijami. Vmesnik je zbir operacij, ki se koristijo za podroben opis razreda ali komponente. Vmesnik grafično prikažemo s krožcem, povezavo med vmesnikom in drugim elementom pa z navadno polno črto kot pri asociaciji (Slika 5, na str. 11).

2.4.4. UML diagrami

Z diagrami grafično in poenostavljeno predstavimo sistem, objekte, dogodke in stanja v različnih pogledih. UML uporablja 9 vrst diagramov. Vendar pa UML omogoča razvitje tudi novih. Prav tako pa lahko tudi z manjšim številom diagramov pridemo do cilja. Pomembno je, da izberemo ustrezne diagrame za prikaz sistema, odvisno od posamičnega obravnavanega problema. To pa je naloga 1. faze – definiranje zahtev.

Diagrami so razvrščeni v dve večji skupini. Delijo se glede na statičen ali dinamičen prikaz sistema in so naslednji (Booch, Rumbaugh, Jacobson, 1999, str. 91-104; Veljović, 2002, str. 68-159):

- **strukturalni diagrami** – statičen pogled:
 - diagram razredov,
 - diagram objektov,
 - diagram komponent,
 - diagram razvrstitve;

- **diagrami obnašanja** – dinamičen pogled:
 - diagram primerov uporabe,
 - diagram zaporedja,

od grobega k podrobnemu. Prototip predstavlja orodje modeliranja, kjer se preizkušajo ideje, odločitve ter tehnične in logične rešitve, in sicer brez vpliva na delujoči sistem. Spremembe in dopolnitve se izvajajo hitro in enostavno, s tem pa se izboljšuje kakovost prototipa, ki je v končni fazi osnova za bodoči IS. Nastal je kot posledica slabosti linearnega pristopa (Kovačič, Vintar, 1994, str. 47).

Uporaba prototipov izboljšuje faze raziskave, analize in načrtovanja sistema. Še posebej je uporabna v primerih, ko (Avison, Fitzgerald, 1996, str. 77):

- področje rešitve ni dovolj dobro definirano;
- nova tehnologija, ki je potrebna za izdelavo rešitve, še ni dovolj dobro poznana v organizaciji;
- komunikacija med uporabniki in analitiki ni zadostna in je prepočasna;
- bi bil strošek zavrnitve rešitve uporabnikov oz. naročnikov visok ter je zagotovitev pokritja uporabniških potreb bistvenega pomena in
- je potrebna ocena vpliva bodočega IS.

Neko splošno prepričanje velja, da mora prototip vsebovati naslednje lastnosti, oz. mora biti (Kovačič, Vintar, 1994, str. 238) :

- dovolj majhen za izvedbo,
- dovolj vpliven in prepričljiv,
- dovolj poceni in dolgoročen in
- dovolj nujen za izvedbo.

Namen prototipa je praktična predstava videza in funkcionalnosti bodoče rešitve. Rezultati, pridobljeni pri obravnavi prototipa se lahko uporabijo v (Avison, Fitzgerald, 1996, str. 77):

- fazi analize za razumevanje obstoječega sistema in predlog funkcionalnih zahtev alternativnega sistema;
- fazi izgradnje prototipa za ocenjevanje s strani uporabnikov;
- več ponavljajočih se fazah spreminjanja in ocenjevanja prototipa in
- fazah načrtovanja in razvoja novega sistema, kjer je prototip uporabljen kot del specifikacije.

Običajno prototip ni namenjen nadaljnji uporabi in se po dokončanju novega sistema zavrže. Alternativni pristop je uporaba prototipa kot osnove bodočega operativnega sistema. V okviru prototipnega pristopa se prototipu v več iteracijah dodajajo in spreminjajo funkcionalnosti, dokler ne ustreza potrebam uporabnika. V tej fazi se prototip preda v operativno rabo. Pri izdelavi takega prototipa je treba posebej paziti na dobro načrtovanje. Prototip mora biti sposoben obvladovati količine podatkov, ki so predvidene v nadaljnji operativni rabi. Potrebno je tudi ustrezno dokumentiranje sistema zaradi prihodnjih sprememb in tudi za pravilno uporabo uporabnikov in njihovo izobraževanje. Če bi ta korak izpustili, bi lahko prihajalo do problemov, kot so: nezanimanje uporabnikov za uporabo prototipa, načrtno diskriminiranje bodočega IS ipd. To bi vsekakor dalo napačne analize o uporabnosti in funkcionalnosti prototipa. Kadar je prototip namenjen kasnejši operativni rabi, je izjemnega

pomena kontrola nad procesom razvoja, predvsem pa se je potrebno izogibati iskanju bližnjic, ki neizbežno povzročijo težave pri nadaljnji uporabi tako nastale rešitve.

3. SHELLY/CASHMAN/ROSENBLATTOV ŽIVLJENJSKI CIKEL RAZVOJA INFORMACIJSKEGA SISTEMA

Metoda življenjskega cikla razvoja sistema je najbolj primerna za izgradnjo velikih IS, ki bodo v uporabi dlje časa. Življenjski cikel razvoja sistema (Slika 6) ima pet faz, ki si sledijo zaporedno (Shelly, Cashman, Rosenblatt, 1998, str. 1.16). Seveda je nerealistično pričakovati, da se v razvoju ne bi nikoli vrnili v predhodno fazo. V praksi se v posamezni fazi odkrivajo napake, ki so bile storjene v prejšnjih, zato je vračanje nekaj normalnega. V naslednjih petih razdelkih bom opisal koncept življenjskega cikla razvoja sistema.

3.1. NAČRTOVANJE PROJEKTA

Namen prve faze je, da se točno identificira celotno področje problema. V tej fazi izvedemo predhodno raziskavo, ki je ključnega pomena, saj bo rezultat tega dela vplival na celotni proces razvoja. Analitik mora biti v celoti seznanjen, kaj se od novega sistema pričakuje in zahteva. Običajne potrebe bi lahko bile (Shelly, Cashman, Rosenblatt, 1998, str. 2.7):

- izboljšanje odnosa s strankami;
- več informacij (pomembnost izkopavanja podatkov);
- točnost podatkov (kontrola pri vnosih, zaščita in varnost podatkov);
- boljše delovanje sistema (odzivni časi pri poizvedbah, fleksibilnost, možnost nadgradnje) in
- zmanjšanje stroškov (manj vzdrževanja – gledanje na dolgi rok).

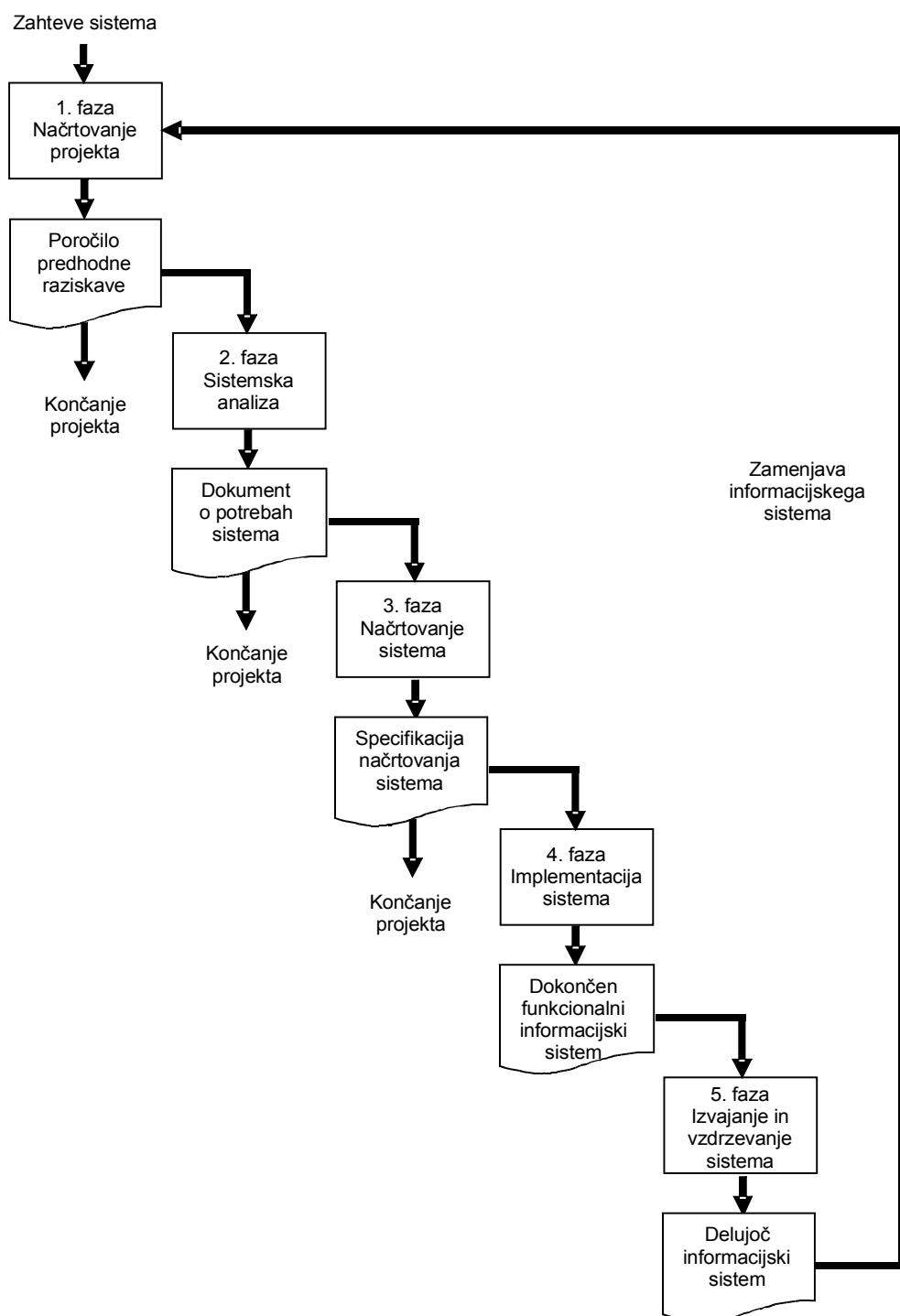
To so samo grobe potrebe, ki jih je potrebno dodatno razčleniti. V ta del vključimo udeležence, ki bodo sistem uporabljali. To so informacijska služba ter potrebe uporabnikov in vodstva. Misliti pa je potrebno tudi na zunanje udeležence, kot so: konkurenti, stranke, cena strojne in programske opreme, tehnologija, zakoni itn.

Ko so te potrebe zbrane, jih je potrebno oceniti. To storimo na treh področjih (Shelly, Cashman, Rosenblatt, 1998, str. 2.13):

- **analiza operativnih izvedljivosti.** Tukaj se vprašamo, ali so zaposleni dovzetni za spremembe, kako bo nov sistem vplival na zaposlenost, koliko bo potrebno dodatnega usposabljanja zaposlenih, predviden čas postavitve sistema;
- **analiza tehničnih izvedljivosti.** Ali ima organizacija že zadostno stopnjo tehnologije za nov sistem, in če ne, kakšne so možnosti za dobavo le-te. Misliti je potrebno vnaprej in računati na večje kapacitete in možnost nadgradnje, na zadostno zanesljivost operacijskega sistema ipd;

- **analiza ekonomskih izvedljivosti.** Pri tej analizi pa se seveda osredotočimo na zmožnost organizacije, da zagotovi zadostna sredstva za uspešno izvedbo projekta v vseh fazah razvoja.

Slika 6: Življenjski cikel razvoja sistema



Vir: Shelly, Cashman, Rosenblatt, 1998, str. 1.17.

Namen predhodne raziskave je zbrati dovolj informacij, na podlagi katerih se lahko odločimo ali je smiselno nadaljevati z nadaljnjim razvojem IS. V tej fazi ni bistvo definirati vseh

problemov in njihovih možnih rešitev, ampak razumeti samo naravo problema, definirati področje zelenega sistema, ugotoviti koristi, potreben čas in stroške ter o tem seznaniti vodstvo (Shelly, Cashman, Rosenblatt, 1998, str. 2.17). Za to moramo imeti vso podporo vodstva pri raziskovanju, dostop do potrebnih informacij, razumeti organizacijsko strukturo in analizirati informacije, ki smo jih pridobili. Načrtovanje projekta zaključimo s poročilom predhodne raziskave, na podlagi katerega se vodstvo organizacije odloči o nadaljevanju ali zaustavitvi projekta.

3.2. SISTEMSKA ANALIZA

V drugi fazi se sistemski analitik točno seznanja z delovanjem trenutnega sistema, ne glede na to, ali je informatiziran ali ne, ugotovi pomanjkljivosti in odkriva ter išče nove rešitve. Na podlagi tega razvije logični model predlaganega sistema in o tem seznanja vodstvo podjetja (Shelly, Cashman, Rosenblatt, 1998, str. 3.3). Odkrivanje pomanjkljivosti starega IS je težavna naloga. Podatki in dokumentacija o zgradbi trenutnega sistema so velikokrat nepopolni in iskanje logike delovanja le-tega bi bilo izredno težavno. Zato se je potrebno posvetiti nadaljnjim korakom v fazi analize sistema, ki so (Shelly, Cashman, Rosenblatt): strukturiranje potreb, njihova analiza ter vrednotenje alternativ in možnih strategij. Strukturirane tehnike za razvoj IS so se razvile v 70-tih letih, njihov glavni cilj pa je bil (Gradišar, Resinovič, 2001, str. 183):

- povečanje produktivnosti pri razvoju,
- zmanjšanje števila napak,
- odkrivanje napak v čim zgodnejših fazah razvoja,
- boljša komunikacija razvijalcev z uporabniki,
- krajši čas razvoja in
- večja kvaliteta razvitega sistema.

Analitik mora dobiti jasno sliko o dinamiki IS in o informacijskih potrebah podjetja. Najprej ugotovi, kakšno je trenutno stanje v podjetju in šele nato lahko išče rešitve. V nadaljevanju bom v razdelku 3.2.2. navedel nekaj orodij za strukturirano analizo, ki so del CASE⁴ orodij.

3.2.1. Strukturiranje potreb sistema

Ta korak je ključnega pomena, saj bo rezultat vplival na karakteristike novega IS. Te potrebe razvrstimo v pet kategorij (Shelly, Cashman, Rosenblatt, 1998, str. 3.6):

- **izhodi** so informacije, ki jih sistem generira na osnovi vhodov. S poznavanjem zelenih kategorij in zahtev izhodnih informacij se lahko razvije kvaliteten sistem, ki bo zadostil potrebam podjetja;

⁴ Computer Aided Software Engineering – računalniško podprta gradnja informacijskega sistema je metodološko podprt problemsko usmerjen pristop, zbirka metod in znanj, ki uporablja računalniško podprte sisteme za bolj učinkovit, natančen in celovit proces analize, načrtovanja, razvoja in vzdrževanja IS. (Kovačič, Vintar, 1994, str. 277-279).

- **vhodi** so vse kategorije podatkov, ki vstopajo v sistem in bodo osnova za kvalitetne izhode;
- pri **procesih** je potrebno paziti na njihovo kompleksnost izvajanja. Izvajati se morajo na način, da bo sistem obremenjen minimalno in da ne bodo vplivali na stabilnost in hitrost delovanja sistema;
- **časovna komponenta** je odvisna od izvajanja procesov zahtevanih nalog;
- **kontrola** je pomembno vršiti pri vnosu, spreminjanju, brisanju, iskanju in pregledovanju podatkov. Dostop naj bi imeli le pooblaščen. Posebej pri vnosu podatkov vstavimo razne kontrole, ki preprečujejo napake, podvajanja, količinsko неконтролиран vnos in vnos nepotrebnih kategorij.

Eden od najbolj pogostih in učinkovitih načinov zbiranja informacij o potrebah IS so intervjuji. V prvi fazi (načrtovanje projekta) smo sodelovali pretežno z vodstvenimi delavci, v tej fazi pa zbiramo informacije od zaposlenih na vseh nivojih organizacije, ki bodo sistem uporabljali. Z intervjuji odkrijemo tudi neformalne povezave, ki so prav tako pomembne kot formalne. Na ta način lahko zaključimo, da so nekateri poslovni procesi nepotrebni in da delo poteka hitreje preko neformalnih komunikacij. Vendar pa intervjuji niso dovolj. Pregledati je potrebno tudi dokumente, kot so: izpisi plač, fakture, skladiščne listine ipd. Dokumenti programskih paketov, ki jih podjetje uporablja ali jih želi uporabljati, nam povedo, v kolikšni meri bo nov IS z njimi združljiv. Naslednji način je zgolj opazovanje zaposlenih pri delu. Pri tem pa moramo biti pozorni na *hawthornov efekt*⁵. Če hočemo dobiti mnenje večine, se poslužujemo vprašalnikov.

3.2.2. Analiziranje potreb sistema

Ko zberemo vse zahteve, jih je potrebno skrbno analizirati in izdelati podroben načrt za rešitev problemov. Celotni IS sestavljajo procesi, poslovna logika in podatkovni model. Ena od možnosti je, da uporabimo različna orodja strukturiranega analiziranja.

Prvo orodje strukturiranega analiziranja so diagrami toka podatkov - DTP (ang. data flow diagrams – DFD). Naslednje grafično orodje za analizo sistema so sistemski diagrami. Tretji način pa so diagrami prehajanja stanj (ang. state-transition diagrams – STD) (Shelly, Cashman, Rosenblatt, 1998, str. 5.27-5.28).

Podatkovni slovar se uporablja za zbiranje, dokumentiranje in urejanje specifičnih lastnosti o sistemu ter vsebuje vse informacije o bazi podatkov obravnavanega sistema. Če izdelamo dober podatkovni slovar, je vsaka nadgradnja ali popravek IS veliko lažji in hitrejši (Gradišar, Resinovič, 2001, str. 189).

⁵ Raziskavo je izvedel Elton Mayo v tovarni Hawthorn Western Electric leta 1920. V raziskavi je hotel ugotoviti, kako različne spremembe pri delu vplivajo na učinkovitost delavca. Med drugim je tudi ugotovil, da so delavci bolj produktivni, če jih osamimo ali jih pri delu opazujemo (Encyclopedia Britannica, 1999, CD-rom 1).

Ker diagrami sami ne povedo, kakšna je logika preoblikovanja procesov, moramo procese natančno določiti. Opis procesov opisuje logična pravila in procese z uporabo strukturirane angleščine⁶ ali psevdokode. Končni produkt strukturne analize je logični model, ki pokaže, kaj mora sistem delati, manj pa kako to fizično doseči (Shelly, Cashman, Rosenblatt, 1998, str. 4.32-4.33).

3.2.3. Alternative in možne strategije

V zadnjem koraku se mora vodstvo na podlagi prejšnjih faz in korakov odločiti, ali se projekt zaustavi ali se z njim nadaljuje. Če se odloči za slednje, je potrebno identificirati programsko opremo, ki bo zadostila potrebam IS. Pri tem imajo podjetja več alternativ. Lahko se odločijo za popolno razvitje sistema, nakup različnih programskih paketov ali za nakup celovitega IS z možnostjo prilagoditve. Podjetja se ponavadi odločajo za kombinacijo med programskimi paketi, ki so na trgu, in za razvoj programskih aplikacij, ki jih izdelava zunanji sodelavec ali informacijska služba v podjetju. Podjetje se odloči za nakup standardnih programskih paketov za urejanje podatkov in besedil, za risanje in oblikovanje, za skupinsko delo ipd., saj je to ceneje in preizkušeno, implementacija pa je hitra. Vse ostale potrebe, ki jih nakup različnih programskih paketov ne more zadostiti, pa je potrebno razviti. In o zadnjem v diplomskem delu tudi pišem. O izbiri in strukturi strojne opreme se odločamo v sistemski analizi, načrtovanju sistema in v fazi implementacije. Strojna oprema mora podpirati delovanje programske opreme, vendar to danes ni več vprašanje, saj so procesorske moči in podatkovni mediji že tako veliki, da zadostijo skoraj vsej programski opremi, ki je na voljo.

Zaključek analize sistema je izdelava dokumenta o potrebah sistema, ki vsebuje potrebe za nov IS. V njem so opisane alternative o izboru programske in strojne opreme ter daje natančne predloge in priporočila vodstvu podjetja. Ta dokument je osnova za merjenje ustreznosti, točnosti in zapletenosti končnega IS, preden nadaljujemo z načrtovanjem sistema (Shelly, Cashman, Rosenblatt, 1998, str. 5.16).

3.3. NAČRTOVANJE SISTEMA

Cilj je izdelati IS, ki bo učinkovit, zanesljiv in bo imel možnost nadgrajevanja. Da lahko dosežemo te kriterije, opravimo 4 korake (oblikovanje izhodov, oblikovanje vhodov, oblikovanje datoteke in baze podatkov ter računalniška platforma). Z njimi izdelamo fizični model IS, ki mora zadoščati kriterijem v dokumentu o potrebah sistema. Osnova za izdelavo fizičnega modela, ki pove, kako potrebe doseči, je logični model iz prejšnje faze, zato nam mora biti popolnoma jasen njegov koncept (Shelly, Cashman, Rosenblatt, 1998, str. 6.3-6.17).

⁶ Structured English – skupek angleških besed, ki točno in natančno opisujejo logične procese tako, da so razumljivi uporabniku, ni pa to programiranje.

3.3.1. Oblikovanje izhodov

Najbolje je, da začnemo z oblikovanjem izhodov, saj ti nakazujejo, kaj mora IS »producirati«, da bo zadostil poslovnim zahtevam. Uporabniki bodo ocenjevali uspešnost sistema po tem, kako dobri bodo izhodi, ki jih potrebujejo za uspešno delo. Tiskana poročila, ki so namenjena za notranjo ali zunanjo uporabo, morajo vsebovati toliko informacij in biti v obliki, ki jo potrebuje uporabnik oz. prejemnik poročila. Ker pa uporabnik dela predvsem preko zaslona kot vidnega medija, mora biti izgled na ekranu, delo preko tipkovnice in miške uporabniku prijazno in funkcionalno. Izhod je tudi tip podatka. Ta mora biti v obliki, ki bo delovala tudi v ostalih programih izven IS. Pri izhodih pa je bistveno, da ne prihaja do napak. Zato moramo zagotoviti kontrolo izhodov, ki bo zagotavljala, da bodo informacije točne, dokončne, zanesljive in da bodo do njih imele dostop le pooblašcene osebe (Shelly, Cashman, Rosenblatt, 1998, str. 6.14-6.52).

3.3.2. Oblikovanje vhodov

Kvaliteta izhodov je odvisna predvsem od kvalitete vhodov. Ta koncept je poznan tudi kot smeti noter, smeti ven (ang. garbage in, garbage out – GIGO), kar pomeni, da netočni vhodi vodijo do netočnih izhodov. Da pridemo do čim kvalitetnejših vhodov, je potrebno razviti uporabniku prijazen vmesnik za vnos podatkov, ki bo zagotavljal kvaliteto, točnost in pravočasnost. Napake se pojavijo pri vnosu, zato je pomembno, da izberemo tak vmesnik, ki bo zagotavljal najmanjše število možnih napak (Shelly, Cashman, Rosenblatt, 1998, str. 7.1-7.15). Vnos je potrebno čim bolj avtomatizirati s pomočjo terminalov (kreditne in delovne kartice), skeniranja, črtne kode. Ostale možnosti so še prepoznavanje magnetnega črnila, naprave za digitalizacijo in prepoznavanje zvoka ter video signalov, senzorji, ekrani na dotik ipd. Še vedno pa je najpogostejši vnos podatkov preko tipkovnice. Da bi bilo napak čim manj, poskušamo zmanjšati število vnosov tako, da jih sistem generira sam (datum, šifranti) oz. jih dopolni ali izračuna iz predhodnih, enkrat že vnesenih podatkov (ime, cena, popusti, zaloge). Dodatno pa namestimo kontrole, ki skrbijo za manjše število napak (kontrola vnosa za števila, črke in posebne znake, omejeno št. mest pri vnosu).

3.3.3. Oblikovanje datoteke in baze podatkov

Logični model iz analize potreb sistema služi za izdelavo fizičnega načrta baze podatkov. Baza podatkov je za IS in za samo podjetje izrednega pomena, saj hrani osebne podatke (zaposleni, stiki, podatki o plačilih itn.), podatke za izvajanje poslovnih operacij (zaloge v skladiščih, odsotnost itn.), zagotavlja operativne podatke (zgoščene oblike podatkov) in zagotavlja podatke iz okolja organizacije (konkurenca, gospodarska gibanja ter napovedi itn.) (Shelly, Cashman, Rosenblatt, 1998, str. 7.16-7.21). Od tega, kako je oblikovana in kako se uporablja, pa je odvisno delovanje organizacije in s tem njena konkurenčnost. Zato lahko

bazo podatkov pojmuje kot enega osnovnih življenjskih virov organizacije, ki mora biti oblikovana tako, da:

- omogoča hiter dostop do podatkov,
- vsebuje točne podatke brez preobilja podatkov oz. odvečnih podvajanj,
- omogoča učinkovito delo,
- je prilagodljiva in
- zagotavlja varnost.

3.3.4. Računalniška platforma

Kot zadnji korak v fazi načrtovanja sistema je izdelava računalniške platforme, ki se nanaša na logičen in na fizičen model, strojno in programsko opremo, načrt in način procesiranja (Shelly, Cashman, Rosenblatt, 1998, str. 9.1-9.24). Pri zadnjem se odločamo med:

- **sprotno obdelavo** (ang. online processing), kjer se transakcije odvijajo v smeri od baze podatkov do uporabnika in podatki se obnovijo v trenutku, ko jih uporabnik na svojem terminalu spremeni in shrani. To je zelo pomembno v proizvodnem procesu za vodenje zalog v skladišču;
- **paketno obdelavo** (ang. batch processing), primerno za velike količine podatkov, ki se pošljejo v procesiranje naenkrat in v paketu. V tem primeru je sistem zelo obremenjen, obdelava pa se navadno izvaja, ko je sistem najmanj aktiven (ponoči, vikendi, prazniki). Tak način obdelave se uporablja za obračun plač ali obdelavo transakcij kreditnih kartic;
- **centraliziranim procesiranjem** (ang. centralized processing), to vključuje glavni računalnik in terminale, ki so locirani na enem mestu in preskrbujejo z informacijami celotno organizacijo. Slabost, predvsem zaradi slabega omrežja včasih, je bila ta, da je bil možen dostop do podatkov le preko terminalov v računalniškem centru. Predstavlja pa centraliziran sistem zopet trend razvoja⁷ in
- **porazdeljenim procesiranjem** (ang. distributed processing), ki omogoča procesiranje podatkov na osebnih računalnikih in za to ne potrebuje glavnega računalnika. Tako kot se izvaja porazdeljeno procesiranje, je možno, da so podatki shranjeni na več mestih. To omogoča porazdeljen sistem za upravljanje baz podatkov (ang. distributed database management system - DDBMS). Ta omogoča, da je baza podatkov bliže uporabniku in zato delo ne obremenjuje mreže, slabost pa je varnost podatkov.

Odločiti se je potrebno tudi o vrsti mrežne povezave. Krajevne računalniške mreže (ang. local area network – LAN) povezujejo računalnike ene stavbe ali ožjega kompleksa stavb. Široke računalniške mreže (ang. wide area network – WAN) pokrivajo širše geografsko območje. Sodobnejša tipa povezav, ki se uveljavljata danes, sta še brezžična zasebna mreža (ang. wireless private network – WPN) in navidezno zasebno omrežje (ang. virtual private network

⁷ To danes omogočajo hitri pretoki podatkov preko ADSL povezav, ki so tudi poceni, in pa vedno hitrejše računalniške mreže v podjetjih. Ta način dovoljuje uporabo tudi zastarelih računalnikov, kar je vsekakor prihranek podjetja. Prav tako pa je mogoč dostop do podatkov in izvajanje operacij preko Interneta. Citrix Systems je podjetje, ki je specializirano za nudenje svojih rešitev na ta način.

– VPN). Prvi je uporaben za podjetja, katerega zgradbe so narazen največ 30 km, med njimi pa ni večje naravne ovire. Deluje na frekvenci 2,4 in 5,8 GHz. VPN pa uporablja javno omrežje, to je internet, zakupljeni vodi. VPN je cenejši od WAN, ker podjetju ni potrebno zgraditi lastne komunikacijske mreže. Problem pa je lahko varnost. Logični računalniški model odjemalec-strežnik (ang. client-server) deluje v okviru takih mrež. Pri tem modelu se procesi na računalnikih v mreži odvijajo tako, da so računalniški viri čim bolj enakomerno obremenjeni, tako je delovanje mreže učinkovitejše. Kot pa sem zapisal zgoraj, je trend razvoja zopet v centraliziranemu procesiranju, kar omogočajo predvsem hitre povezave in zmogljivi procesorji v centralnih računalnikih oz. strežnikih.

V vsakem sistemu se pojavijo tudi nepredvideni dogodki, ki lahko povzročijo izgubo podatkov. Zato je potrebno te dogodke predvideti in najti rešitve zanje. Shelly, Cashman in Rosenblatt omenjajo 4 funkcije za podporo sistemu: izdelava varnostne kopije in obnova datotek (ang. backup and recovery), pridržek datoteke (ang. file retention), ponovni zagon (ang. restart) in zagonsko procesiranje (ang. start-up processing).

Načrtovanje sistema se konča s specifikacijo načrtovanja sistema (ang. system design specification) in s predstavitvijo projekta oddelku za informatiko in vodstvu organizacije, ki pa ga lahko še vedno prekine. Specifikacija načrtovanja sistema je dokument, v katerem je opisan celoten načrt IS z vsemi stroški, s potrebnim osebjem za upravljanje sistema in s časovnim načrtom izvedbe naslednje faze, to je implementacija. Ta dokument je namenjen predvsem programerjem, ki bodo izdelali potrebne programe (Shelly, Cashman, Rosenblatt, 1998, str. 9.20-9.24).

3.4. IMPLEMENTACIJA SISTEMA

V 4. fazi življenjskega cikla razvoja sistema se izvede razvoj aplikacij oz. programske opreme, njena namestitvev in vrednotenje.

3.4.1. Razvoj aplikacije

Pri razvoju aplikacije je vloga systemskega analitika v tem, da planira povezovanje programov in zagotavlja, da bodo delovali usklajeno ter zadostili organizacijskim potrebam. Programerju mora predstaviti jasne in transparentne specifikacije sistema, na podlagi katerih lahko izdelava, testira in dokumentira posamezne programske module.

Večina razvoja aplikacij se izvede s pristopom od zgoraj navzdol (ang. top-down approach), ki izhaja od splošne do podrobne strukture v posameznih logičnih korakih. S tem pristopom analitiki definirajo večje lastnosti sistema, ki jih potem razbijejo na podsisteme in module. Ta pristop se imenuje tudi modularno načrtovanje (ang. modular design). Modul je skupek programskih navodil, ki se lahko izvajajo kot skupina. Na ta način se lahko več področij razvoja izvaja istočasno (Shelly, Cashman, Rosenblatt, 1998, str. 10.6).

S testiranjem se odkrijejo napake, še preden bi lahko povzročile poslovno škodo, ko bi sistem v organizaciji dejansko uporabljali. Testiranje poteka sproti, na koncu pa je potrebno testirati celotni sistem. Testiranje sistema mora vključiti vse možne procese, ki se bodo izvajali v organizaciji, ter upoštevati količino podatkov, s katerimi bo sistem delal. Sem spada tudi prikaz delovanja informatikom in uporabnikom. Dodatno pa jim je potrebno pripraviti tudi programsko in sistemsko dokumentacijo ter dokumentacijo za uporabo.

3.4.2. Namestitvev in vrednotenje

Pred namestitvijo programske opreme se še enkrat preveri vsa strojna oprema, predvsem strežniki in mreža ter operacijski sistemi, ki bi lahko vplivali na samo delovanje IS pod predpostavko, da je sistem brezhibno implementiran in deluje. Izobraževanje je prav tako del faze implementacije, razdelimo pa ga na tri skupine. To je izobraževanje informatikov, uporabnikov in vodstvenih delavcev (Shelly, Cashman, Rosenblatt, 1998, str. 11.1-11.9).

Vrednotenje sistema pomeni, da se oceni doseganje kriterijev, ki so bili postavljeni v dosedanjih fazah razvoja. Ocenjevanje je primerno opraviti ne prej kot čez mesec in ne kasneje kot 6 mesecev od uvedbe novega sistema. V oceno naj bi vključili 9 kriterijev (Shelly, Cashman, Rosenblatt, 1998, str. 11.13-11.15):

- točnost, pravočasnost in celovitost IS,
- zadovoljstvo uporabnikov,
- zanesljivost in vzdržljivost,
- zadostnost kontrol in varnost,
- učinkovitost strojne opreme,
- učinek in kakovost razvojnega tima,
- učinkovitost baze podatkov,
- kvaliteta poročil in
- točnost predvidenih stroškov in časovne izvedbe.

Predzadnjo fazo končamo s končnim poročilom vodstvu organizacije. Poročilo naj bi vsebovalo vso dokumentacijo sistema, načrtovane in naknadne prilagoditve, primerjavo med planiranimi in dejanskimi stroški razvoja in časovnimi termini ter poročilo ocene implementacije sistema.

3.5. IZVAJANJE IN VZDRŽEVANJE SISTEMA

Da bo sistem zadostil stalnim poslovnim spremembam in zahtevam uporabnikov, ga je potrebno vzdrževati in izboljševati. Zadnja faza se začne, ko je sistem v celoti v uporabi, in konča, ko se ga zamenja z novim. V tej fazi se izvaja podpora (ang. support) tekočemu delu, in ga opravljajo strokovnjaki, ki skrbijo za brezhibno delovanje sistema s tehničnega vidika in pravilno ter učinkovito uporabo z vsebinskega vidika. Vzdrževanje (ang. maintenance) pa se

izvaja, dokler je sistem v uporabi, in predstavlja najdaljšo fazo. To pomeni, da se sistem prilagaja na spremembe ali pa na osnovi praktičnih izkušenj. Stroški vzdrževanja so po navadi podcenjeni, eden od vzrokov so hitre spremembe okolja, v katerem organizacija deluje. Vendar so največji razlog za velik porast stroškov v tej fazi napake, storjene v predhodnih fazah (Laudon, Laudon, 2000, str. 503-506).

V predhodni fazi se je izvajalo izobraževanje o delovanju sistema bolj na teoretični ravni in z večjim poudarkom na izobraževanju informacijske službe. Tukaj pa je potrebno uporabnike v celoti seznaniti z novim sistemom in jim nuditi vso podporo.

Vsak IS sčasoma zastara. Znak za to so nenadni veliki operativni stroški ali stroški vzdrževanja sistema. Ko je dosežen določen prag stroškov, je potrebno star IS zamenjati in življenjski cikel razvoja se začne znova.

4. ZAMENJAVA IS V PODJETJU ADRIA MOBIL, d.o.o.

4.1. PREDSTAVITEV ADRIE MOBIL, d.o.o.

Adria Mobil, d.o.o., iz Novega mesta je izvozno usmerjeno podjetje z lastno blagovno znamko in z osnovno dejavnostjo proizvodnje počitniških prikolic in avtodomov, ki ima že 35-letno tradicijo. Lani pa so začeli tudi proizvodnjo mobilnih stanovanjskih enot. Svoje proizvode tržijo pod blagovno znamko ADRIA in ADRIATIK.

Leta 1965 je podjetje Industrija motornih vozil Novo mesto (IMV) ustanovilo podjetje Adria, ki se je usmerilo v proizvodnjo počitniških prikolic. Podjetje se je nato preoblikovalo v Adrio Caravan, ki pa je leta 1995 šlo v stečaj. Nastalo je novo podjetje Adria Mobil, d.o.o. Poleg Adrie Mobil, ki je krovno podjetje, spadajo v poslovno skupino Adria še 4 podjetja (Priloga 1): Adria Caravan, ApS; Adria Plus, d.o.o.; Adria Caravan, d.o.o. in Podgorje, d.o.o. Lastništvo Adrie Mobil je v 100 odstotni lasti Slovenske razvojne družbe (SRD), v procesu pa je odkup večinskega deleža zaposlenih. Letos bo podjetje ustanovilo svojo peto hčerinsko družbo Adria Event, d.o.o., ki bo zadolžena za jahting program (interni podatki podjetja).

Počitniške prikolice proizvaja podjetje že od samega začetka in so po njih tudi najbolj prepoznavni. Izdelujejo 50 različnih modelov s 400 različicami, ki so razdeljene v 4 serije: A, B, C in Adiva, razlikujejo pa se po dolžini in opremljenosti. Avtodom je njihov drugi proizvod. S svojo blagovno znamko avtodomov so prišli na trg leta 1986, sedaj jih izdelujejo 20 različnih modelov v 33 različicah. Kot osnovno vozilo uporabljajo Fiatovo, Mercedesovo in Peugeotovo podvozje z različnimi motorji. Avtodome izdelujejo v 4 serijah: Adriatik Stargo, Adriatik Coral, Adriatik Serija 70 in Adria Van. Zadnjo serijo izdelujejo v 3 modelih in 10 različicah tudi za nemško podjetje Poeschl, ki pa jih prodaja pod svojim imenom. Mobilne stanovanjske enote Adria home so njihov tretji produkt. Z njihovo proizvodnjo so začeli konec leta 2002. Izdelujejo jih v velikosti 24,2 in 21,84 m² z različnimi tlorisi in z možnostjo razširitve s teraso. Občasno in po naročilu izdelujejo tudi prikolice-prodajalne.

V zadnjih letih se je podjetje prerinilo med 5 največjih evropskih proizvajalcev s tovrstnim programom. 98 odstotkov proizvodnje izvozijo v zahodno Evropo, kjer imajo 6,5-odstotni tržni delež pri prikolicah in 4-odstotnega pri avtomobilih. Za mobilne stanovanjske enote trenutno poteka intenzivno zbiranje naročil. Po Evropi imajo 23 pooblaščenih uvoznikov, ki z več kot 350 trgovci prodajajo njihove produkte. Na ostale trge zaenkrat še ne prodirajo, ker so stroški prevoza preveliki. Izjema je Japonska, tam imajo svoje predstavništvo od leta 1999.

Adria Mobil zaposluje 584 ljudi in spada med srednje velika podjetja. Na leto imajo 23 mrd SIT prihodkov in 1 mrd SIT dobička, ki ga namenijo za izobraževanje zaposlenih in za nove investicije. V letu 2002 so izdelali 6400 prikolic in 3360 avtomobov. V prihodnji letih pa pričakujejo od 5 do 10 odstotno letno rast (interni podatki podjetja). V Prilogi 2 so prikazani organizacijska struktura podjetja, diagram spremembe tržnih deležev, delež prodaje celotnega asortimana po državah in rast prodaje.

Letno sodelujejo na 14 sejnih po Evropi, med katerimi je največji v Düsseldorfu. Ker pa je trg zasičen, lahko povečajo svojo prodajo le z odvzemom tržnih deležev konkurentom. Osnovna trženjska cilja sta:

- kakovost, ki je enaka kakovosti konkurentov,
- butična proizvodnja – izdelajo veliko različic in se tako prilagajajo zahtevam kupcev iz posameznih držav.

Kakovost je eden od elementov, ki mu v Adrii posvečajo še posebno pozornost. V Sloveniji in tujini imajo preko 150 dobaviteljev, ki s tesnim sodelovanjem zagotavljajo zahtevano in pričakovano kvaliteto končnih produktov. Leta 1997 so uvedli sistem kakovosti po mednarodnem standardu ISO 9001. Letos pa nameravajo pridobiti še ekološki standard ISO 14001, kar kaže na moderno in okolju prijazno proizvodnjo. Lasten razvoj jim zagotavlja hiter odziv na spremembe na trgu ter hitro prilagajanje željam uporabnikov. Za oblikovanje produktov sodelujejo z zunanjim sodelavcem, ki sodeluje z njimi od ustanovitve Adrie Mobil.

Poslanstvo podjetja je usmeritev v ustvarjanje izdelkov za zadovoljevanje potreb odjemalcev vseh starostnih skupin, obeh spolov in kultur kjer koli na svetu, ki težijo h kvalitetnemu, predvsem pa aktivnemu preživljanju prostega časa. Pri viziji pa so si zadali cilj, da postanejo eden od vodilnih proizvajalcev proizvodov za prosti čas in uveljavijo blagovno znamko ADRIA in ADRIATIK kot sinonim za kakovost in prosti čas v Evropi in širše.

V nadaljevanju bom predstavil IS, ki je bil v uporabi pred zamenjavo, njegove pomanjkljivosti in vzroke za zamenjavo.

4.2. INFORMACIJSKI SISTEM PRED ZAMENJAVO

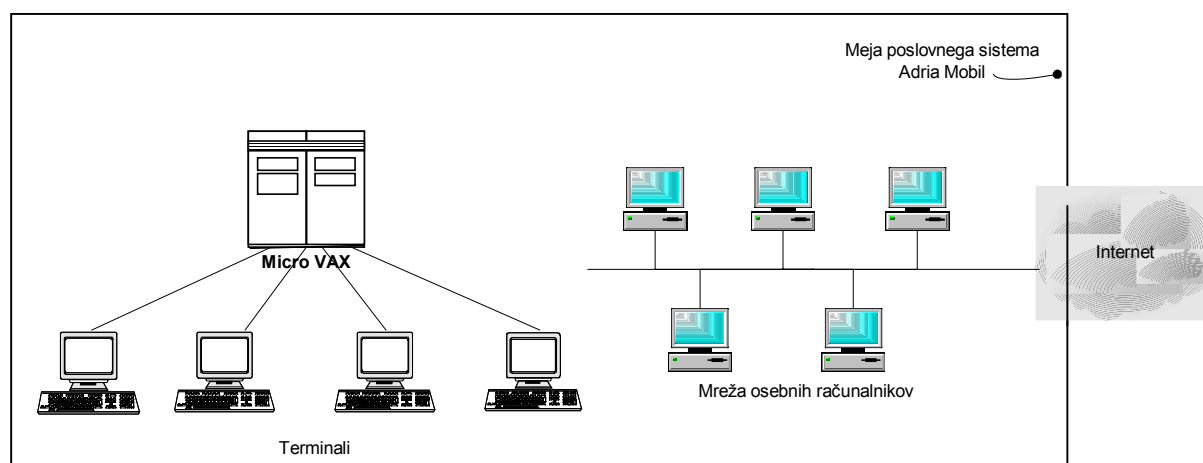
V letu 1985 in 1986 so v takratni Adrii Caravan, ki je delovala pod okriljem IMV, prvič postavili IS, imenovan PMPS – Proizvodno materialni poslovni sistem. Osnova je bil

operacijski sistem (OS) **Open VMS**, ki je bil nameščen na VAX računalnikih. To je bil tekstualni OS brez grafičnega vmesnika in je deloval na centralizirani računalniški mreži. Vse podatke je procesiral glavni računalnik, na katerem je bil nameščen celoten IS. Manipuliranje s podatki je potekalo preko terminalov, večinoma VT220.

OS Open VMS in VAX računalnike je leta 1975 začel izdelovati Digital Equipment Corp. (DEC), ki je bil v lasti podjetja Compaq, sedaj HP (Hewlett-Packard). OS so razvili posebej za računalnike VAX, ki so imeli za tiste čase veliko procesno moč in so spadali med miniračunalnike. Zaradi skupnega razvoja se imenujejo OpenVMS/VAX sistemi. Slabost teh sistemov je nezmožnost grafičnega prikaza in to, da na računalnikih VAX ne delujejo ostali OS in obratno. Te sisteme je Compaq prenehal izdelovati leta 2000. Vendar pa je letos HP predstavil nove večprocesorske strežnike, ki bodo delovali tudi z OS OpenVMS, kar bo morda omogočilo njegovo preživetje. Ta OS je drugače primeren za poslovne sisteme, ki morajo delovati neprestano in brez napak. Uporabljajo jih borze vrednostnih papirjev, bolnišnice, telekomunikacijska podjetja. OpenVMS/VAX sistem je izdelan tako, da deluje na enostavni platformi, kjer so možnosti napak nične. Kot je povedal Cliff Pederson, manager informacijske tehnologije v podjetju Sunoco: "*Te sisteme uporabljamo že leta, ne da bi morali računalnike kdaj ponovno zagnati*" (Vax to Alpha, 2003).

V Adrii so imeli na terminalih zaposleni na voljo podatke le v tekstovni in tabelarni obliki. Preko terminalov je lahko uporabnik vnašal, spreminjal, brisal, tiskal ali kako drugače oblikoval podatke, do katerih je imel dostop. Ta sistem je takrat pomenil pravo revolucijo, saj je racionaliziral delo uporabnikov, jih medsebojno povezal in jim praktično omogočil tisto, kar so do takrat delali ročno. Z leti in ob množični pojavnosti osebnih računalnikov, ki niso bili del enotne mreže in so uporabniku omogočali določeno svobodo, ustvarjalnost in prilagoditve, so se tako začele pri vsakodnevni uporabi odkrivati pomanjkljivosti, napake in slabosti celotne informacijske zasnove. Na sliki 7 je prikazana obstoječa mrežna arhitektura, ki so jo uporabljali v Adrii pred zamenjavo sistema, prikazuje pa terminalski OpenVMS/VAX sistem in ločeno mrežo osebnih računalnikov.

Slika 7: Mrežna arhitektura obstoječega sistema



Vir: Interni podatki podjetja.

Ob organizacijski spremembi in nastanku novega podjetja Adria Mobil leta 1995 je nastopil trenutek za zamenjavo PMPS. Zato je vodstvo podjetja naročilo izdelavo analize obstoječega stanja IS pri podjetju Infotehna iz Novega mesta. Istočasno pa so svoje mnenje podali tudi skrbniki aplikacij s posameznih funkcijskih področjih znotraj podjetja Adria Mobil.

4.2.1. Analiza Infotehne

Mnenje Infotehne je bilo, da obstoječa programska oprema pokriva le osnovne potrebe po posameznih poslovnih funkcijah in da je delo prilagojeno posebnostim poslovanja. Zaradi centralizirane računalniške mreže z glavnim računalnikom in s terminali nekateri bistveni elementi IS niso vključeni v samo poslovanje oz. se ne uporabljajo. V vseh 10 letih celotna zasnova OS, IS in same mrežne arhitekture ni sledila intenzivnemu vsebinskemu in tehnološkemu razvoju ter organizacijskim spremembam. Zaradi tega je bistveno manj učinkovita in prilagojena uporabi sodobnega poslovanja (mišljeno je predvsem komuniciranje med posameznimi organizacijskimi službami, ki je bilo zaradi tega počasnejše).

Po njihovem mnenju so prilagoditve potrebam in spremembam okoliščin poslovanja, dopolnitve in nadaljnji razvoj skoraj v celoti onemogočeni. To pa popolnoma zavira razvoj organiziranosti poslovnega sistema. Vzrok je odsotnost prvotnih avtorjev oz. skrbnikov IS ter pomanjkljiva sistemska in programska dokumentacija.

Vse zgoraj naštetu bistveno zmanjšuje uporabno vrednost in učinkovitost obstoječe opreme, kar vpliva tudi na organizacijski razvoj poslovnega sistema. Posledice so v bližnji in daljni prihodnosti v škodo podjetju, denarno jih je težko oceniti.

Po mnenju Infotehne bi bila vsaka izvedba dopolnitev in prilagoditev, ki so nujne, problematična in draga. Najslabše pa je, da bi ostala zasnova celotnega IS tehnološko in vsebinsko zastarela.

4.2.2. Mnenje skrbnikov aplikacij

Splošno mnenje vseh skrbnikov aplikacij je bilo, da obstoječa strojna in programska oprema ne omogoča učinkovitega in kvalitetnega dela. Iz lastnih izkušenj in pa s sodelovanjem uporabnikov na terminalih so pomanjkljivosti, napake in slabosti strnili v 4 točke:

- Programi so uporabniško neprijazni. Potreben je večkratni vnos istih podatkov v računalniški sistem. S tem pride do nekonsistentnosti podatkov. Različne službe v podjetju pa dobivajo zaradi napak pri vnosu, podvajanj in neažurnosti podatkov o istih stvareh različne informacije.
- Programi znotraj IS z zunanjimi niso povezani in niso dodelani. Nepovezanost programov je bila največkrat vzrok večkratnega vnosa.
- Osebni računalniki so fizično in tehnično zastareli. V podjetju so poleg terminalov obstajali tudi osebni računalniki, vendar pa s terminali niso bili povezani. Imeli so tudi

premalo spomina, majhno procesorsko moč in zastarelo programsko opremo. To pa je bil tudi vzrok za nezmožnost uporabe sodobnejših računalniških programov.

- Dokumenti se prepočasi tiskajo zaradi počasnih igličnih tiskalnikov. Zaradi že prej omenjene nepovezanosti programov je bilo potrebno veliko podatkov natisniti, jih vnesti v druge programe za podporo odločanju in na podlagi rezultatov odločati o operativnih in strateških odločitvah.

4.3. PROJEKT PRENOVE IS

V podjetju so po natančni analizi podaje mnenja Infotehne in Adriinih skrbnikov aplikacij leta 1997 začeli projekt prenove IS. Sprememba poslovnega IS vsekakor pomeni velik zagonski napor in stroške ob določenem tveganju. Sama nabava novih računalnikov in programske opreme ne more prinesiti bistvenih izboljšav, če se hkrati ne pristopi k oblikovanju novih organizacijskih rešitev v zvezi z delitvijo dela zaposlenih (znotraj in med sektorji). Le v primeru sprejetja primernih organizacijskih ukrepov se lahko pričakuje izboljšanje delovnih rezultatov.

Veliko pozornost so v podjetju namenili prihrankom. Zamenjava IS zahteva velike finančne stroške in prihranke je bilo potrebno najti, kjer je bilo to mogoče. Vendar pa je potrebno imeti v vidu resnični prihranek in ne le prihranek pri izbiri najcenejšega ponudnika strojne opreme in programske rešitve. Kasneje se lahko pojavi nekontrolirano preplačevanje zaradi slabega izbora dobaviteljev in neustreznosti programskih rešitev. Zato so v Adrii Mobil opredelili kritični področji pri projektu prenove IS:

- opredelitev ciljev projekta – katera stanja je potrebno kje in kdaj doseči ter
- opredelitev projektne organizacije – vodenje, nadzor in delitev pristojnosti za čas projekta.

Cilji projekta in organizacijska struktura podjetja so bili določeni do leta 1998, ko je podjetje začelo zbirati ponudbe izvajalcev za izdelavo novega IS. Za zunanjega izvajalca se je podjetje odločilo, ker informatiki v podjetju niso bili dovolj usposobljeni za tako obsežen projekt, kot je zamenjava IS. Po drugi strani pa štirje ljudje v informacijski službi poleg rednega dela tudi časovno ne bi zmogli tako obsežnega projekta. Zbiranje ponudb je potekalo na dveh področjih. Na eni strani so iskali ponudnika za nabavo strojne opreme in postavitev sodobne komunikacijske mreže, ki bi zagotavljala zanesljive in hitre pretoke podatkov v mreži osebnih računalnikov. Drugi del pa je bil najti izvajalca, ki bi bil cenovno ugoden in zmožen kvalitetno izdelati ter postaviti rešitev za računalniško podporo poslovanju.

V dokumentu **Projekt prenove IS** so v šestih fasciklih zbrane vse predhodne analize o trenutnem stanju, ki so jih opravili znotraj podjetja in v podjetju Infotehna. Dokument prav tako vsebuje vse bodoče zahteve in kritična področja, ki jih je potrebno obravnavati in ki so potrebna za uspešno izvedbo projekta prenove, ter vsi dogodki, ki so projekt prenove spremljali. Če na splošno povzamem in se naslonim na sliko 6, kjer je prikazan življenjski

cikel razvoja sistema, vsebuje dokument Projekt prenove IS naslednje: vso dokumentacijo od poročila predhodne raziskave do končnega poročila, ki vsebuje vso dokumentacijo sistema, naknadne prilagoditve ter oceno prednosti in slabosti sistema. Dokument pa se še vedno dopolnjuje s poročili o dodatnih zahtevah in naknadnih popravkih ter implementacijah.

4.3.1. Cilji projekta

Zaradi zastarele strojne opreme je bil prvi cilj postaviti sodobno računalniško platformo, ki bo enotna podlaga za uporabnike in razvijalce informatike. Vse to pa je potrebno podkrepiti s trdnim in robustnim jedrom aplikativnih programskih rešitev.

Poleg klasično opredeljenih ciljev, kot so povečanje produktivnosti, ekonomičnosti in rentabilnosti, so v cilje projekta vključili tudi (Projekt prenove IS, 2003):

- racionalizacijo poslovanja in doseganje prihrankov ter ustvarjanje poslovne prednosti;
- avtomatizacijo postopkov in pospešitev obravnave poslovnih dogodkov;
- povečanje deleža tržno cenjenega, kreativnega dela nasproti rutinski administraciji;
- povečanje poslovne sprejemljivosti Adrie Mobil za evropske partnerje;
- vzpostavitev konsistentne baze podatkov kot zanesljivega vira podatkov za oblikovanje informacij pri poslovnem odločanju;
- povečanje obvladljivega obsega poslovanja in
- zagotavljanje ustrezne skupne informacijske podlage za poslovne obdelave.

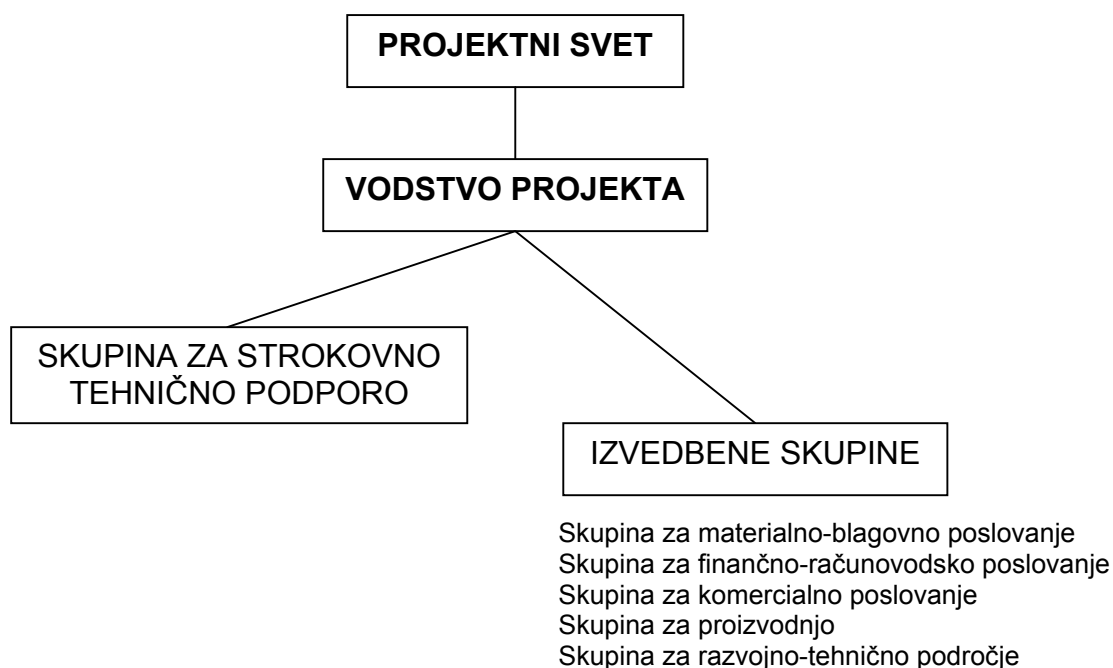
Operativni cilji, katerim projekt sledi, pa so:

- oblikovanje informatike po sodobnih konceptih,
- uvajanje aplikacij, realiziranih s 4-generacijskimi orodji,
- razvijanje znanja uporabe novih konceptov in tehnologij.

4.3.2. Opredelitev projektne organizacije

Zamenjava IS se je odvijala projektno, skladno s pravili projektne organizacije in vodenja projektov. V izvajanje projekta so udeleženi poslovni partnerji definirali skupno projektno organiziranost z združenim projektним vodenjem. Da je bilo takšno projektno vodenje mogoče, je informacijska služba izdelala organizacijsko strukturo projektnega tima (Slika 8). Hierarhično si sledijo: projektni svet, vodstvo projekta, skupina za tehnično podporo in 5 izvedbenih skupin. Smisel izvedbenih skupin izvira iz faznosti odvijanja projekta: dobava in namestitve potrebne strojne opreme ter postavitve razvojnega in uporabniškega aplikacijskega okolja.

Slika 8: Organizacijska struktura projektnega tima



Vir: Projekt prenove IS, 2003.

PROJEKTNI SVET

Predsednik sveta: gen. direktorica

Člani sveta:

- direktorji posameznih sektorjev,
- direktor izbranega podjetja za izdelavo IS,
- vodja projekta izbranega podjetja,
- vodja projekta Adrie Mobil.

Naloge:

- določi strategijo izdelave projekta,
- postavi projektno organiziranost,
- spremlja, usmerja in potrjuje izvajanja projekta,
- zagotavlja potrebne resurse za izvedbo projekta in
- določa financiranje projekta.

VODSTVO PROJEKTA

Člani:

- vodja projekta Adrie Mobil,
- vodja projekta izbranega podjetja in
- skrbniki aplikacij.

Naloge:

1. Planiranje projekta:
 - definiranje aktivnosti projekta,
 - definiranje tehnološko izvedbenega vrstnega reda odvijanja aktivnosti,
 - izdelava plana izvedbe,
 - vključevanje izvajalskih kapacitet,
 - plan kontrole izvajanja projekta,
 - izdelava plana vključevanja izvajalskih kapacitet in
 - izdelava plana kontrole izvajanja projekta.
2. Lansiranje izvajanja projekta:
 - izdelava predloga delovnih nalogov in izvedbenih pogodb,
 - lansiranje delovnih nalogov izvajalcem in
 - izdelava organizacijskih navodil.
3. Kontrola izvajanja projekta:
 - spremljanje poteka izvajanja projekta,
 - predlaganje potrebnih akcij v primeru motenj,
 - kontroliranje izvedbe ključnih dogodkov projekta,
 - kontroliranje kakovosti posameznih aktivnosti v smislu pogojevanja končne kakovosti projekta in
 - spremljanje vsebinskih odstopanj od pogodbeno dogovorjene vsebine.
4. Informiranje o poteku izvajanja projekta:
 - informiranje projektnega sveta in
 - informiranje uporabnikov.

SKUPINA ZA STROKOVNO TEHNIČNO PODPORO

Člani:

- tehnična podpora izbranega podjetja in
- tehnična podpora Adrie Mobil.

Naloge:

- tehnična pomoč pri reševanju vsebinskih vprašanj,
- informacijski inženiring,
- reševanje tehnično-tehnoloških problemov izdelave IS,
- komunikacije,
- namestitev sistemske in uporabniške programske opreme,
- prenos podatkov,
- standardi programiranja,
- standardi navodil in
- ostala strokovna pomoč.

IZVEDBENE SKUPINE

Izvedbene skupine so bile sestavljene iz strokovnjakov za posamezna področja in so bili zadolženi za neposredno izvedbo podprojektov. To so bile:

- skupina za materialno-blagovno poslovanje,
- skupina za finančno-računovodsko poslovanje,
- skupina za komercialno poslovanje,
- skupina za proizvodnjo in
- skupina za razvojno-tehnično področje.

Naloge:

- analiza potreb,
- prilagoditev programske opreme,
- testiranje,
- priprava osnutka organizacijskih navodil in
- uvajanje ter spremljanje uporabniških rešitev.

4.3.3. Komunikacijska mreža

Postavitev komunikacijske mreže so po planu razdelili na dva dela, upravno stavbo in proizvodno halo. V upravni stavbi, ki je v dveh nadstropjih, delujejo vodstvo in režijski delavci, v proizvodni hali s tremi 80m dolgimi tekočimi trakovi pa poteka celotna proizvodnja. Podjetje XY je postavilo celotno komunikacijsko mrežo. Tehnične zahteve za samo zasnovano mrežo podjetje, ki je bilo izbrano za izdelavo programske rešitve za računalniško podporo poslovanju, ni postavilo. Podali so le minimalne zahteve za strežnike in odjemalce.

4.3.4. Strojna oprema

Nova generacija aplikacij že v osnovi zahteva spremembo arhitekturne zasnove strojne opreme. Če se je z aplikacijami v znakovnem okolju (ang. character mode) govorilo o enem bistvenem strojnem členu, to je o zanesljivem in zmogljivem podatkovnem strežniku, se z uvedbo nove generacije aplikacij, ki zahtevajo način dela odjemalec-strežnik, te potrebe in zahteve razširijo tudi na delovne postaje in računalniško mrežo. Torej je podjetje moralo za novo generacijo aplikacij poskrbeti za kompleksnejšo strojno postavitve, kar v celoti predstavlja računalniško strojno osnovo celotnega IS.

Tehnologija novega IS ni bila vezana niti na določen tip računalnika niti na določenega proizvajalca strojne opreme in OS, zato so v podjetju pri izbiri upoštevali naslednja merila:

- cena glede na zmogljivost,
- zanesljivost sistema,
- odprtost in servisnost sistemov,
- "moč in pozicija" proizvajalca ter
- povezljivost proizvajalca z vodilnimi proizvajalci programske opreme.

4.3.5. Izbira ponudnika IS

Drugi del, izbira ponudnika, je bil mnogo bolj tvegan. Pri postavitvi mreže se cena določa predvsem po ceni dobavljene opreme z vsemi znanimi parametri (hitrost prenosa podatkov, tehnične specifikacije, možnost nadgradnje, življenjska doba, kvaliteta materialov itn.), ki so zahtevani po pogodbi. Odločitev o izbiri se tako v bistvu določa le preko cene. Tveganje je skoraj enako nič. Tudi pri nakupu standardnega paketa programske opreme kot je npr. MS Office, Corel Draw, Lotus Notes, je poslovno tveganje majhno, zmožnosti posameznega programa so znane in predvidljive, prav tako njihova kompatibilnost. Povsem drugače pa je, ko se podjetje odloči za izdelavo programske rešitve, ki bo izdelana posebej po potrebah poslovanja podjetja. Četudi bo takšna aplikacija v celoti izdelana po zahtevah in potrebah podjetja, obstaja še vedno tveganje, da se ne uresničijo vsa pričakovanja. Že same zahteve podjetja, ki jih poda pri izdelavi IS, so lahko napačne oz. jih sistemski analitik napačno interpretira. Tudi podjetje, ki bo projekt zamenjave IS izvedlo, mora dobiti zaupanje naročnika. Reference so tako eden od faktorjev, ki podjetju pomagajo pri izbiri izvajalca.

Ker IS vpliva na poslovanje podjetja, uporablja pa ga veliko število zaposlenih, je ena od možnosti, da se zaposleni negativno odzovejo na nov sistem, ker to prinaša spremembe pri njihovem vsakodnevem delu, s tem pa se sistem v celoti ne izkoristi. Potrebno je predvideti, da zamenjava zahteva velike stroške in veliko časa. V prihodnosti naj bi stroške IS v celoti povrnil in omogočal povečanje poslovanja, znižanje operativnih stroškov, večji nadzor kakovosti, boljše komunikacije med zaposlenimi, hitrejše odzivne čase itn.

V Sloveniji prevladujejo majhna in srednje velika podjetja. Ker je razvoj IS, ki bi bil od začetka do konca namenjen izključno zahtevam podjetja, zelo drag, se podjetja velikokrat odločajo za nakup celovitih IS (Kovačič et al., 2002, str. 199). Služba informatike je pričela pregled ponudnikov IS na slovenskem trgu. Bilo jih je veliko, različne so bile tudi osnove, na katerih so gradili svoje rešitve. Zajemanje celote in fleksibilnost opreme in izvajalskega podjetja je bila zaradi kasnejših dopolnjevanj in zaradi konsistentnosti s posameznimi bodočimi programskimi orodji, ki bodo nadgrajevala IS, tista lastnost, ki je podjetje ne bi smelo v nobeni fazi nastajanja prenove zanemariti.

Adria se je odločila, da bo celoten IS razdelila na dva dela, poslovni in proizvodni. Poslovnega so poimenovali Poslovno informacijski sistem (PIS), ki bo pokrival področje računovodstva in financ. Tega dela izbire, razvoj in implementacijo v diplomskem delu ne bom obravnaval. Drugi del, kateremu je namenjena naloga, pa se je nanašal na programsko rešitev, ki bo v celoti prilagojena proizvodnemu procesu, kot ga uporablja Adria Mobil. Izmed vseh ponudnikov, ki so se prijavili za rešitev proizvodne aplikacije, se je podjetje odločilo za podjetje Apros, ki je ponudilo celovit IS za upravljanje proizvodnje, imenovan UPRO.

Programska rešitev podjetja Apros je imela ob prvotni predstavitvi kar nekaj prednosti:

- ugodna lokacija,
- primernost variantnih kosovnic,
- izobraževanje uporabnikov,
- aplikacija v Windows okolju v celoti dokončana in
- izredno nizka cena.

Hkrati pa je podjetje imelo tudi pomanjkljivosti, ki so se kazale v:

- majhnem številu zaposlenih,
- nizkem kakovostnem razredu podatkovne baze ter
- neintegriranosti poslovnega in proizvodnega sistema.

Število zaposlenih izredno vpliva na zanesljivost IS. S podjetjem Apros je Adria sklenila pogodbo o najemu vzdrževalca sistema, ki je v osnovi zaposlen v podjetju Apros, fizično pa ima sedež na lokaciji Adrie Mobil, kjer izvaja dela za potrebe sistema. Adria je od podjetja zahtevala zagotovljeno 8-urno dnevno prisotnost vzdrževalca sistema. Če bi bila njegova odsotnost daljša od 3 dni, Apros poskrbi za njegovo zamenjavo. Vsi posegi in prilagoditve, ki bistveno posegajo v funkcionalnost programa, se rešujejo v sodelovanju s 5-članskim timom v podjetju Apros. Če te možnosti v začetku podjetje ne bi ponudilo, bi to pomenilo bistveno daljši čas uvajanja v začetni fazi.

Osnova izbranega programskega produkta je bila **baza podatkov**, ki je uporabljala sistem za upravljanje baze podatkov⁸ (SUBP) Paradox, kar na eni strani pomeni določen denarni prihranek, na drugi strani pa je lahko stabilnost in hitrost IS zelo vprašljiva. Kot rešitev

⁸ SUBP je zbirka programov, ki omogočajo tvorjenje, uporabo in vzdrževanje baz podatkov (Grad, Jaklič, 1996, str. 145).

problema so se s podjetjem Apros kasneje, ko je sistem že deloval, dogovorili za prehod na relacijsko bazo InterBase, ki je primerljiva z Oracle bazo in izredno stabilna. S tem prehodom so v Adrii prešli na višji kakovostni razred podatkovne baze. Več o tem v razdelku 6.8.

Seveda je **integriranost IS** cilj vsakega podjetja, vendar se pojavlja vprašanje, ali je Adria Mobil zmožna doseči ta cilj. Neintegracija obeh sistemov bi pomenila celo prednost, ker bi omogočala mehak prehod na nov IS. To se je kasneje tudi pokazalo kot pravilna odločitev. Uvajanje v dveh delih, poslovni in proizvodni IS, ni predstavljalo velikega napora ne za podjetje Apros ne za informatike in uporabnike v Adrii. Konec leta 2002 in v začetku leta 2003 pa so oba sistema povezali na ključnih delih, to je na področju prenosa faktur.

4.4. UPRO – Upravljanje PROizvodnje

UPRO je izdelek podjetja Apros, računalništvo, d.o.o. iz Novega mesta. To je sodoben interaktivni računalniški paket, ki je namenjen podpori vodenju proizvodnje v manjših in srednje velikih podjetjih. Posebej je primeren za lesno industrijo in kovinoplastiko, kar dokazujejo tudi reference podjetij, ki uspešno uporabljajo paket UPRO: TOMPlast iz Mirne na Dolenjskem, EKI iz Črnomlja, Modema iz Kočevja, Inplet iz Sevnice, Akripol iz Trebnjega, Paron iz Laškega, Alples iz Železnikov, Podgorje iz Šentjerneja, Akrapovič iz Ivančne Gorice, Vistim iz Črnomlja, Sito iz Cerknice in Mesojedec iz Ljubljane. Zaradi njegove univerzalnosti ga je mogoče uporabljati v proizvodnji po naročilu in tudi v maloserijski in velikoserijski proizvodnji. S svojimi rešitvami na področju planiranja, spremljanja proizvodnje in vodenja zalog je usmerjen v reševanje glavnega izziva proizvodne organizacije, ki se glasi: kako izboljšati racionalnost proizvodnega procesa (Domača stran Apros).

UPRO je napisan v okolju Delphi podjetja Borland in je izdelan po konceptih objektno orientiranih metod razvoja sistemov. Iz tega tudi izhaja velika prilagodljivost in možnost nadgradnje sistema po posameznih modulih in znotraj njih. Pri tem pa ne gre samo za nastavitve parametrov. Sistem so ravno za potrebe Adrie Mobil prilagodili do podrobnosti. Vsem zgoraj naštetim podjetjem je skupno to, da imajo serijsko proizvodnjo, kjer sledenje posameznim izdelkom ni potrebno. V nadaljevanju bom večje posege v sistemu opisal v razdelku 4.8.

Program deluje na različnih SUBP (Paradox, InterBase, SQL Server) odvisno od želja in zahtev kupca. Odlikujejo ga:

- enostavna uporaba in prijaznost paketa,
- visoka prilagodljivost s parametri,
- dobra zaščita podatkov in programov s prijavnimi gesli,
- interaktivna navodila za delo s programi,
- visoka stopnja integriranosti v Windows okolje.

Adria se je odločila za pokrivanje petih poslovnih področij, in sicer na področju tehnologije, proizvodnje, prodaje, nabave in UDOK. Vseh 5 modulov sem podrobneje predstavil v Prilogi 3.

Ker je UPRO programski paket, pomeni, da ima module že »sprogramirane«. Sem lahko štejemo obliko oken, posamezna vnosna polja oz. njihovo lego, interaktivno pomoč, medsebojno modularno povezanost, izvoz dokumentov ter podatkov v ostale programe, osnovno zasnovo podatkovne baze itn. Nastavitev parametrov in dodajanje novih rešitev, ki bodo služile popolni ustreznosti poslovanja Adrie, je bilo še potrebno razvijati. Zato je informacijska služba v sodelovanju z analitiki podjetja Apros naredila načrt aktivnosti aplikacijskega projekta. Določili so korake in aktivnosti, ki jih je bilo potrebno opraviti, da bi se UPRO prilagodil potrebam poslovanja na prej omenjenih petih področjih.

V nadaljevanju bom v 8 razdelkih predstavil, kako je potekala izvedba zamenjave IS od točke, ko je bila komunikacijska mreža postavljena, z izvajalcem za izdelavo programske rešitve za računalniško podporo poslovanju pa že podpisana pogodba o izvedbi. Ker nekatere aktivnosti po posameznih segmentih še vedno potekajo, prav tako pa se nekatera področja prekrivajo, posamezne točke po vsebini niso strogo ločene. Zato se nisem odločil za kronološki opis zamenjave, ampak sem se omejil na vsebinskega.

4.5. PREVERBA REŠITEV Z UPORABNIKI

To je faza strukturiranja potreb sistema in je ključnega pomena, ker bodo vsi nadaljnji koraki sloneli na analizah tega preliminarnega poizvedovanja. Njena glavna aktivnost je bila prepoznati vhode in izhode sistema. V sklopu priprave na podpis pogodbe je bilo potrebno v vsakem sektorju čim natančneje razdeliti projektno nalogo. Informacijska služba je s sodelovanjem podjetja Apros izvedla poglobljene predstavitve in usklajevanja z uporabniki na konkretnih primerih iz poslovanja. Na tem mestu so na podlagi dosedanjih izpisov in listin definirali nove, v tej fazi pa so definirali tudi aplikacije.

Rezultat je bil sestavljen zapisnik posebnih zahtev uporabnikov, ki je bil medsebojno usklajen in podpisan. Poleg vsake zahteve je bila navedena tudi časovna opredelitev izvedbe del, ki naj bi dajala informacijo o vrednosti dodatnih del na obstoječem programskem produktu.

Še preden se je končal zadnji korak izdelave zapisnika posebnih zahtev, so se razvijalci že ukvarjali s programskimi rešitvami. S tem so bistveno skrajšali čas izvedbe, saj so rešitve sprti prilagajali in popravljali. Če bi se ravnali po metodologiji življenjskega cikla, bi začeli izvajati programske rešitve šele po izvedbi 3. faze življenjskega cikla. To pa še vedno ne bi zagotavljalo nadaljnjega razvoja brez zapletov, kar bi pomenilo vnovično vračanje v predhodne faze razvoja, čas izvedbe pa bi se podaljšal.

4.6. PREVZEM PODATKOV IZ OBSTOJEČIH APLIKACIJSKIH BAZ PODATKOV

UPRO je bil v osnovi zasnovan z SUBP Paradox. Vse podatke, ki so jih imeli v podjetju v PMPS v stari podatkovni bazi, je bilo potrebno prenesti v Paradox. To ni pomenilo samo prepisa, ampak tudi prilagoditev novi organizaciji poslovanja, racionalizacijo in odpravo nepotrebnih dvojnih vnosov podatkov. Prenos je bil izveden za vse tiste podatke, ki so se nanašali na osnovne šifrate, medtem ko prometnih podatkov, predvsem materialnih dokumentov, niso prenašali. Med osnovne šifrate v podjetju štejejo stroškovna mesta, skladišča, vrste prometov⁹ in poslovne partnerje.

V Adrii so v času prenove imeli 30 stroškovnih mest, kar je veliko, to je bila posledica spreminjanja organizacije podjetja v daljšem časovnem obdobju. Z novim IS pa jim je uspelo njihovo število zmanjšati na 10 in jih prilagoditi trenutnim potrebam podjetja.

Skladišča, ki so se pojavljala v sistemu PMPS, so v veliki meri ustrezala takratnim zahtevam podjetja. Novost so predstavljala le skladišča gotovih izdelkov, ki so bolj podrobno razdeljena zaradi potreb knjigovodstva in spremljanja gibanja prikolic in avtomobilskega programa. Le-te je bilo potrebno dodati novemu sistemu šifrantov.

Število vrst prometov materialnih dokumentov, ki so se pojavljale v sistemu PMPS, je bilo izredno veliko, približno¹⁰ 115. Takšno število je tudi odsev spreminjanja organizacije podjetja v daljšem časovnem obdobju. Ko so izvedli analizo, katere vrste prometa se dejansko uporabljajo, so ugotovili, da je takšnih le približno 40. Analiza je bila izvedena na podlagi celotnega gibanja materiala, rezervnih delov in ostalega materiala. Ker so spremembe smiselne ob prehodu na nov IS, se je ta posodobitev tudi izvedla. Prav tako pa so izvedli tudi posodobitev šifrantov poslovnih partnerjev.

Zmanjšanje stroškovnih mest in števila vrst prometov je redek primer, kjer so poslovni proces prilagodili novemu IS. Bilo je še nekaj manjših, ki pa jih v Adrii niso beležili, saj so se zgodili sprti in niso bistveno vplivali na organizacijsko poslovanje.

4.7. NAMESTITEV PROTOTIPNE RAZVOJNE APLIKACIJE

Namen prototipne razvojne aplikacije je bil preveriti delovanje sistema na dejanskih podatkih in v izvajanju delovnega poslovnega procesa. Prototipna aplikacija je bila nameščena 2 meseca pred dokončno zamenjavo IS in je bila implementirana v celoti ter dostopna vsem uporabnikom. Intenzivno testiranje je potekalo v službi informatike, ki je povezala ključne uporabnike posameznih sektorjev in na testnih podatkih izvajala aktivnosti bodočega

⁹ Promet je v proizvodnji šifrant različnih dokumentov in opisuje, kako se naj različni dokumenti v poslovnem procesu obnašajo. V podjetju jih uporabljajo za vodenje stroškovnega računovodstva.

¹⁰ Približno sem uporabil zato, ker število prometov med poslovanjem variira.

poslovanja. Hkrati je potekalo tudi testiranje zmogljivosti sistema in odzivnosti programa glede hitrosti. Pri samem delovanju ni bilo ugotovljenih nepravilnosti, zato so nov IS v celoti aktivirali in ga začeli uporabljati 1. septembra 1999.

Kot pišem kasneje v razdelku 6.12., so se nekatere težave pokazale šele pri polni uporabi sistema. Zakaj se nekatere napake niso ugotovile v testni fazi, je več razlogov. Eden je vsekakor to, da je sistem testiralo premalo uporabnikov, le okoli 20 odstotkov, in logično je, da se vsi problemi v testni fazi niso mogli pokazati. Druga stvar pa je čas izvedbe testne faze. Julij in avgust za test nista bila najprimernejša zaradi dopustov, zato je bilo majhno število testnih uporabnikov. Po drugi strani pa je ravno zaradi zmanjšane proizvodnje v poletnih mesecih lažje posegati v delovanje IS, testnega in delujočega. Zaradi zadnjega se je podjetje tudi odločilo izvesti fazo namestitve prototipne razvojne aplikacije v poletnih mesecih.

V razdelku 2.5. sem opisal nekatere glavne značilnosti prototipnega pristopa razvoja sistema. Adriin primer prototipne razvojne aplikacije lahko uvrščamo v drugo skupino prototipa, kjer se uporabljen prototip ne zavrže, ampak se sistem, ki se uporablja v testni fazi, uporabi kot osnova bodočega operativnega IS. V tem primeru je bilo to smiselno, saj se v predhodni testni uporabi niso pojavljale napake, ki bi zmanjšale uporabno in funkcionalno vrednost samega sistema. Prav tako pa je bila prototipna aplikacija razvita zaradi nadaljnje uporabe.

4.8. IZVEDBA POSEBNIH OZ. DODATNIH ZAHTEV V APLIKACIJAH

Spisek dodatnih zahtev je obsegal večje število manjših prilagoditev, ki so pomenile predvsem drugačen način pristopa obravnavanja podatkov in racionalizacije vnosa. Ker je v podjetju stalno prisoten programer podjetja Apros, imajo uporabniki to prednost, da se vse dodatne zahteve izvedejo v najkrajšem možnem času. Sem podjetje šteje razne manjše napake, ki se lahko pojavijo ob specifičnih dogodkih in operacijah, spremembe izpisov, podatkov ipd.

Program UPRO nudi programerju to možnost, da si v primeru, ko se napaka pojavi, pomaga s takojšnjim prikazom programske kode, ki izvaja nek ukaz. Če ugotovi napako, jo lahko takoj popravi. Na sliki 9 je primer SQL stavka, ki napolni tabelo v ozadju. V primeru napake tega ukaza ni potrebno iskati v programski kodi, tako se ne izgublja čas, zato se lahko manjši problemi rešujejo vsak dan sproti. Manjših prilagoditev (te se rešujejo dnevno) podjetje ni beležilo, večjih posegov je bilo 8. Te je podjetje evidentiralo v Dokumentu sprememb in dopolnitev (Priloga 4).

Slika 9: Program UPRO

MP13 - Vnos dokumenta

Dokument: PS300448

Dok: PS Leto: 2003 Številka: 448 Skladišče: [200300] Vrsta prometa: [3010] Datum zapisa: 05.03.2

Poz	St	Šifra	Naziv
1	03	5000 87642 00	NALEPKA 2L
2	03	200 43018	OBLOGA TALNA TERM. MOBIL RUSALKA
3	03	9021 87212 00	NALEPKA 800X63 L
4	03	5000 87291 00	NALEPKA STR. VELIKA LEVA
5	03	5000 87643 00	NALEPKA 2D
6	03	5000 87283 00	NALEPKA NAPIS "ADRIA"
7	03	5000 87661 00	NALEPKA "ADRIA"-TISK NA FOLIJI
8	03	5000 87267 00	NALEPKA CELNE ST. 400D
9	03	5000 87632 00	NALEPKA ZAKLJ. LUČI-LEVA
10	03	5000 87633 00	NALEPKA ZAKLJ. LUČI-DESNA
11	03	90014 016 00	TEKAČ ZA ZAVESE - RIŽEK
12	03	5000 79658 108	ZAVESA PROSOJNA 950X770

```

SELECT VDOK_SIFVD
, VDOK_SIFLE
, VDOK_SIFST
, VDOK_POZIC
, VDOK_STATS
, VDOK_MATER
, VDOK_VEZDO
, VDOK_VDPOZ
, VDOK_SKLAD
, VDOK_ORGAN
, VDOK_SARZA
, VDOK_PKOLI
, VDOK_DKOLI
, VDOK_DVRED
, VDOK_VRVAL
, VDOK_DADOK
, VDOK_MESOB
, VDOK_DANOB
, VDOK_ZAPST
, VDOK_UZIDO
, VDOK_STVOZ
, VDOK_NADGR
, VDOK_UZIDN
, VDOK_NAKLK
, VDOK_DANZA
, VDOK_DOKU1
, VDOK_DOKU2
, VDOK_OCENA
, VDOK_CDOMA
, ARTI_NAZIV
, ARTI_ENMER
, ARTI_DELME
, ARTI_DELM1
, ARTI_DELM2
, ARTI_ORGAN
, ARTI_LOKAC
, ARTI_PKP
, ARTI_ABCD
FROM VDOK
join arti on vdok_mater = arti_sifra
where
vdok_sifvd = :mdok_sifvd and
vdok_sifle = :mdok_sifle and
vdok_sifst = :mdok_sifst
order by vdok_pozic
    
```

EM	Datum dok.	Datum zap.	Vezni dok.
00	KOS	05.03.2003	05.03.2003
00	M2	05.03.2003	05.03.2003
00	KOS	05.03.2003	05.03.2003
00	KOS	05.03.2003	05.03.2003
00	KOS	05.03.2003	05.03.2003
00	KOS	05.03.2003	05.03.2003
00	KOS	05.03.2003	05.03.2003
00	KOS	05.03.2003	05.03.2003
00	KOS	05.03.2003	05.03.2003
00	KOS	05.03.2003	05.03.2003
00	KOS	05.03.2003	05.03.2003
00	KOS	05.03.2003	05.03.2003

Vir: Programski paket UPRO.

Večja prilagoditev, ki je bistveno posegla v koncept izgradnje IS, se nanaša na vodenje gotovih izdelkov po šasijskih številkah. Programski paket UPRO je bil izdelan za serijsko proizvodnjo, kjer so enaki izdelki vodeni pod enotno šifro. Adriin način spremljanja gotovih izdelkov je nekoliko drugačen in primerljiv z avtomobilsko industrijo, kjer se pojavljajo posebne zahteve za vodenje in sledenje posameznih produktov. Zato se je ena izmed večjih prilagoditev programa UPRO nanašala na vodenje gotovih izdelkov v kombinaciji z vodenjem po kataloških in šasijskih številkah. To je pomenilo, da bo vsak končni izdelek, ki bo zapustil tovarno ali čakal v skladišču, voden pod enovito šifro, po kateri ga bo mogoče izslediti.

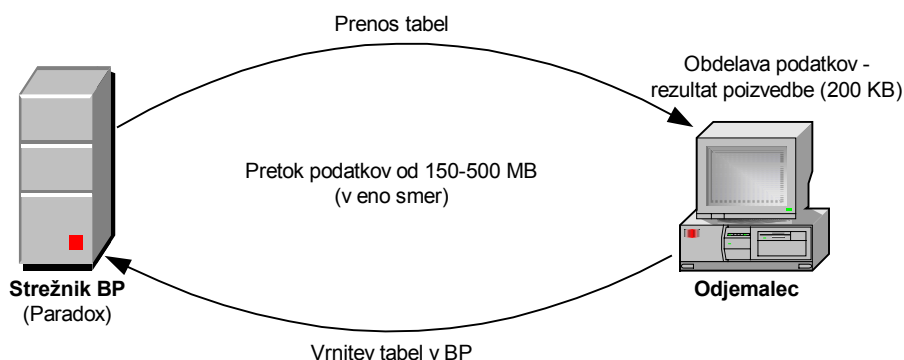
Ko se je izvajala ta faza, je bil nov sistem že v uporabi. V točki, ko je bila izvedena prilagoditev sistema vodenje izdelkov po šasijskih številkah, je sistem uporabljal Paradoxov SUBP. Specifičnost Adriinega poslovanja je velika podatkovna baza¹¹. To pomeni, da ko se

¹¹ Ob prenosu leta 1999 je bila baza podatkov velika 500 MB (ne vključuje prenosa prometnih podatkov – glej razdelek 4.6.). V 3 letih je narasla na 2 GB, kar pomeni nekaj milijonov zapisov. V naslednjih 5 letih predvidevajo, da bo narasla na od 6 do 7 GB podatkov. vendar pa baza ne narašča konstantno, ampak sečasni podatki v njej tudi brišejo, zato njena velikost v krajšem obdobju niha.

sproži ukaz za poizvedovanje podatkov (ang. data query), se začne obdelava oz. procesiranje podatkov, ki so v podatkovni bazi.

Z uporabo SUBP Paradox (Slika 10) procesiranje izvaja odjemalec. Ko poda uporabnik zahtevo po poizvedbi (SQL stavek), ki je iz več tabel, se po računalniški mreži prenesejo potrebni podatki od strežnika do odjemalca. Ena tabela s spremljajočimi datotekami je lahko velika tudi do 350 MB. Konkretno¹² to pomeni: 153 MB za samo tabelo in 176 MB za spremljevalne datoteke, ki so ključi, validacije in indeksi. Rezultat poizvedbe pa je navadno velik okoli 200 KB. Preko mreže je bila najprej odposlana tabela. Ker pa mrežo uporabljajo še drugi uporabniki, je zaradi obremenitve lahko obstala. Drugi del pretoka podatkov pa je bil vrnitev tabel v bazo, kar mrežo zopet obremeni. Tako se je dogajalo, da ko je strežnik poslal še zadnjo datoteko, ki je del tabele, je preteklo že preveč časa in strežnik je tak tok podatkov prekinil ter javil, da je prišlo do napake.

Slika 10: Obremenitev mreže pri uporabi SUBP Paradox



Slika 11: SUBP InterBase



Pri relacijskih SUBP, kot je InterBase (Slika 11), pa se procesiranje izvaja na strani strežnika, odjemalec dobi le rezultat. Na ta način se obremenitev mreže zelo zmanjša, s prejšnjih 500 MB na zgolj 200 KB. Procesiranje na strežniku omogoča zgradba baze, ki ločeno vsebuje

¹² Tabela Delavci.

diagrame, uporabnike, tabele, vloge, pravila. Relacijska baza tudi ne potrebuje več začasnih tabel, kar je bilo pri Paradox prej pravilo kot izjema. Čas, ki je bil potreben za pridobitev poizvedbe s Paradox, so merili v minutah, pri InterBase se je čas poizvedbe skrajšal na nekaj sekund.

Oba SUBP imata dobre in slabe lastnosti. Paradox omogoča uporabo manj zmogljivih strežnikov, na drugi strani pa so potrebni zmogljivejši odjemalci. Mrežo obremenjuje bolj, za svoje delovanje pa uporablja začasne tabele, ki se kopičijo in zasedajo prostor. InterBase bolje izkoristi procesno moč strežnika, razbremeni mrežo in odjemalce. Z uporabo SPROC¹³ in prehodnih tabel se odpravi uporaba začasnih tabel. Ne gre pa razumeti, da je SUBP Paradox neuporaben. Zaradi velike količine podatkov, za Adrio ni primeren ga pa uporabljajo v vseh naštetih podjetjih v razdelka 4.4. in deluje brez zapletov.

Predvsem upočasnitev delovanja omrežja, ko se izvaja procesiranje podatkov za več uporabnikov hkrati, je bila vzrok za prehod s Paradox na relacijsko bazo InterBase. Ta prehod je trajal približno 6 mesecev in je bil tudi največji poseg v že delujočo bazo podatkov. Na delovanje sistema ni bistveno vplival, ker so se spremembe delale vzporedno, v »živo« pa so jih vključili naenkrat.

4.9. USPOSABLJANJE UPORABNIKOV

Usposabljanje uporabnikov je korak, ki ga je potrebno začeti izvajati, še preden sistem stopi v veljavo. V Adrii Mobil se zavedajo, da je to eden od pomembnejših segmentov za uspešno izvajanje sistema. Začetno usposabljanje oz. izobraževanje uporabnikov za nov IS je izvedlo podjetje Apros še pred uvedbo novega sistema. Nadaljnja usposabljanja so in še vedno potekajo na internem nivoju, kjer imajo vsi zaposleni, ki se pri svojem delu srečujejo s programom UPRO, možnost udeležbe izobraževanja, ki so splošna ali pa področno usmerjena. Sedaj je IS v uporabi že tretje leto. V teh dveh letih se je sistem dopolnjeval in izpopolnjeval in tako postal obsežnejši. Zaradi raznih dopolnitev in sprememb imajo v podjetju organizirano usposabljanje tudi na letnem nivoju.

S sprotnim usposabljanjem so odpravili klice uporabnikov, ki imajo vprašanja v zvezi s samim programom in delovanjem sistema. Ti klici so bili v začetku pogosti, sedaj pa so se znižali na najnižjo možno raven, kar pomeni v povprečju manj kot en klic na dan. To tudi dokazuje, da je sistem zanesljiv in uporabniku prijazen.

4.10. UVAJANJE IN NAMESTITEV PROGRAMSKEGA PAKETA

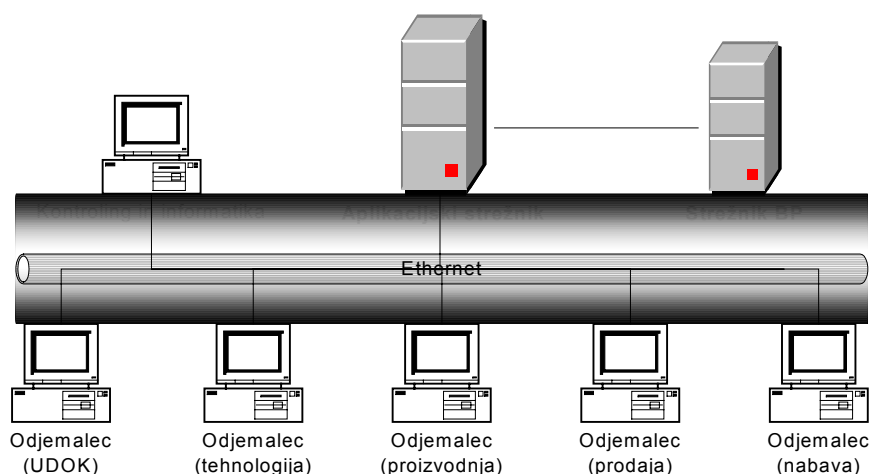
Programski paket so uvajali postopoma, v več sklopih (modulih). Vrstni red uvajanja posameznih sklopov je bil pogojen z medsebojnimi odvisnostmi med sklopi in s problematiko

¹³ Stored PROCedures – skripta, kjer je že napisana programska koda poizvedb z SQL stavki.

posameznih aplikacij. Vsak programski sklop zajema namestitvev, prepis podatkov, uvajanje uporabnikov, testiranje in nastavitvev parametrov. V uvajanje, ki se tudi prekriva z usposabljanjem, so bili vključeni vsi uporabniki IS in skrbniki aplikacij. V fazi uvajanja sta Adria in podjetje Apros definirala tudi morebitne prilagoditve zaradi specifičnih zahtev poslovanja.

Namestitev na strežnik je bila potrjena, ko se je izvedel funkcionalni preizkus delovanja in pregledala pripadajoča uporabniška dokumentacija. Na sliki 12 je prikazan diagram razvrstitve, ki prikazuje glavni aplikacijski strežnik, ki je povezan s strežnikom baze podatkov, do katerega je mogoč dostop le preko aplikacijskega strežnika. V računalniško mrežo Ethernet¹⁴ so priključeni vsi odjemalci in tudi služba kontrolinga in informatike.

Slika 12: Diagram razvrstitve računalniške platforme za Adrio Mobil



4.11. ZAVAROVANJE PODATKOV

Zavarovanje podatkov je eno izmed najpomembnejših opravil. S posebno organizacijo podatkov na diskih strežnika podatkov, ki ima možnost zrcaljenja (ang. mirroring), in izdelavo dvojnikov na tračni enoti lahko zagotovijo vzpostavitev stanja podatkov do trenutka okvare tudi v primeru, da se disk fizično poškoduje. V primeru izpada električne energije je IS pred izgubo podatkov varovan tudi z UPS sistemom (ang. uninterruptible power supply).

Na področju zaščite podatkov pred nepooblaščenim dostopom se za vstop v program uporablja uporabniško ime in geslo, znotraj program pa so pravice za uporabo posameznih programov dodelijo vsakemu uporabniku posebej. Uporabniško ime in geslo se zaposlenemu dodeli v začetku. Geslo si lahko uporabniki kasneje po želji sami spremenijo. Gesla delujejo na vseh računalnikih v podjetju, kar pomeni, da ima uporabnik s svojo prijavo povsod enake

¹⁴ Ethernet je vrsta računalniške mreže in predstavlja standard za sodobna lokalna omrežja. Podatki se prenašajo po žici, posebnost pa je ta, da vodi UTP kabel (kabel s 4 prepletenimi paricami) do vsakega računalnika ločeno. V podjetju uporabljajo trenutno hitrost prenosa podatkov 100Mb/s.

pravice in nastavitve v aplikaciji (Priloga 3, b). Služba informatike ima možnost, da izvaja nadzor nad tem kdo, kdaj in kaj se je vneslo, izbrisalo ali kako drugače spremenilo v bazi. Ta prilagoditev je bila izdelana naknadno, ker se je večkrat dogajalo, da so uporabniki za izgubo podatkov krivili nepravilno delovanje sistema in tako vnašali zmedo med ostale uporabnike. Sistem sedaj omogoča večjo varnost in nadzor nad manipulacijami podatkov, s predpostavko, da so za izgubo podatkov odgovorni izključno uporabniki.

UPRO ima prav tako močno kontrolo vnosa. Analitiki so jo poimenovali primarna in sekundarna kontrola vnosa podatkov (ang. check constraints). Primarna deluje in se izvaja na strani strežnika, vklopi pa se, ko želi uporabnik podatke shraniti. Sekundarna se izvaja na strani odjemalca v sami aplikaciji. Več poudarka so dali primarni kontroli. Zahtevnejši uporabniki lahko dostopajo do baze tudi preko programov za administriranje baze podatkov, kot so Enterprise Manager, SQL Query Analyzer, IB Console ipd, takrat seveda kontrole vnosa na strani odjemalca v aplikacijah ni. V teh primerih uporabnika primarna kontrola opozori ob shranitvi podatkov, če le-ti ne ustrezajo.

4.12. PREDNOSTI IN SLABOSTI NOVEGA IS

O prednostih sem napisal že nekaj v razdelku 4.4., kjer sem opisal UPRO. Tukaj pa bom navedel prednosti in slabosti, ki so se pokazale pri uporabi sistema v Adrii Mobil.

Po dobrih treh letih je podjetje v grobem ocenilo nov IS. Denarno je vrednost, ki jo je prinesel sistem, težko oceniti. Kljub temu da se je z leti poslovanje Adrie vidno izboljšalo, so na dobre rezultate vplivali tudi ostali notranji in zunanji dejavniki. Prednosti uvedbe novega IS so po mnenju informacijske službe naslednje:

- racionalizacija poslovanja, pretok dokumentov je hiter in natančen;
- uresničili so cilj popolne avtomatizacije postopkov, ki ga omogoča dober programski paket;
- povečal se je obvladljivi obseg poslovanja, saj so skrajšali časovne omejitve na minimum;
- v letu in pol so vzpostavili kvalitetno bazo podatkov, ki zagotavlja zanesljiv vir informacij in
- uresničili so tudi cilj minimalnega števila zaposlenih v informacijski službi, trenutno skrbijo 4 zaposleni za celotno informacijsko arhitekturo, pomoč uporabnikom in kontroling.
-

Prihranki so se pri organizaciji poslovanja, zajemu podatkov in delovanju celotnega poslovanja povečali, povečala pa se je tudi količina uporabnega-aktivnega dela. Zelo pomembno je tudi, da so se uporabniki po prvotnih težavah uvajanja pozitivno odzvali na spremembe in prilagoditve v poslovanju, ki ga je prinesla uvedba novega IS. To pa gre pripisati tudi dobri izvedbi izobraževanja zaposlenih, saj na ta način izkoriščajo vse prednosti sistema.

Ena od slabosti novega sistema je bil napačen izbor SUBP, kar je imelo za posledico obremenitev mreže in počasno obdelavo podatkov. Ta slabost se je pokazala, ko je bil UPRO že implementiran. Rešitev problema je bila zamenjava z novo relacijsko bazo InterBase, ki je bistveno bolj stabilna. Drugih slabosti uporabniki in služba informatike niso opazili.

Po mojem mnenju je ena ključnih prednosti, ki jo ima celoten projekt zamenjave IS v Adrii ta, da je v podjetju stalno prisoten vzdrževalec podjetja Apros. Kljub temu da so stroški delovanja sistema nekoliko višji, pa imajo na drugi strani v podjetju stalno podporo delujočemu sistemu. Na ta način se lahko problemi, ki so v bistvu malenkostni, odpravljajo dnevno, uporabniki pa sploh ne občutijo napak v sistemu. Če vzdrževalca v podjetju ne bi bilo, bi se napake odpravljale v paketu (ko se jih nekaj nabere) na lokaciji podjetja, ki je program prodalo. Rešitve za napake bi se tako izvajale daljše časovno obdobje, prav tako pa bi bila komunikacija med programerji in uporabniki slabša.

5. SKLEP

Adria Mobil se je odločila za nakup celovitega IS za vodenje in upravljanje proizvodnje s čim večjo možnostjo prilagoditve. Pri tem odločanju sta imeli velik vpliv cena in pripravljenost ponudnika za čim večjo prilagoditev že razvitega sistema. Veliki izdelovalci, kot so Oracle, SAP in Navision, se informatikom v Adrii Mobil niso zdeli primerni. Razloga sta bila predvsem cena in pomanjkanje zanimanja za prilagoditve podjetju, ki je za njihove kriterije majhno. Med slovenskimi ponudniki pa sta bila glavna faktorja bližina, predvsem zaradi hitrega posredovanja ob nepravilnem delovanju sistema, in možnost izobraževanja uporabnikov. Adria se je odločila za programski paket UPRO, podjetja Apros iz Novega mesta. Programski paket je razvit s sodobnimi 4-generacijskimi orodji (program je napisan v okolju Delphi). Za to pa niso uporabili nobene določene metodologije za razvoj sistemov. Kljub temu, kot je razvidno iz zadnjega razdelka diplomske naloge, pa je nastal končni produkt, s katerim so poslovanje Adrie izboljšali in dosegli želeni učinek (hiter in natančen pretok dokumentov, popolna avtomatizacija postopkov, skrajšanje časovnih omejitev na minimum, kvalitetna baza podatkov, minimalno število zaposlenih v informacijski službi).

Bistvo naloge je bila predstavitev zamenjave IS od točke odločanja o potrebi zamenjave, do samega ovrednotenja zamenjave. Tudi v tem delu niso razvijalci sledili nobeni metodologiji. Iz same vsebine praktičnega dela pa lahko razberemo, da so zajeti vsi potrebni koraki, ki sem jih predstavil v metodologiji življenjskega cikla razvoja sistema in prototipnega pristopa.

V podjetju so se odločili za čim večjo prilagoditev programske rešitve poslovnim procesom. Z IS jim je uspelo pokriti več kot 90 % že vpeljanih poslovnih rešitev, v sedanji fazi pa še vedno potekajo manjše prilagoditve in popravki sistema. Kljub temu da je sedanji trend pokrivanja informacijskih potreb ob nakupu celovitih rešitev največ 70 %, podjetje ni hotelo tvegati.

Zaključim lahko z ugotovitvijo, da so v Adrii Mobil z dobrim strateškim načrtom prenove IS ter z dobrim sodelovanjem in komunikacijo znotraj projektnega tima razvili in implementirali programsko rešitev, ki omogoča podjetju nemoteno poslovanje, hkrati pa je uporabniku prijazna. Podjetju je uspelo priti v tretjino uspešnih implementacij celovitih rešitev (glej poglavje 2), prav tako pa zamenjava IS v Adrii odstopa od nekaterih trendov:

- programska rešitev pokriva več kot 90 % informacijskih potreb,
- pri razvoju niso uporabili nobene določene metodologije,
- prototip je bil osnova bodočemu operativnemu sistemu,
- programer podjetja, ki je razvilo programsko rešitev, je stalno prisoten v Adrii Mobil.

6. LITERATURA

1. Avison E. David, Fitzgerald Guy: Information Systems Development: Methodologies, Techniques and Tools, 2nd ed. London: McGraw-Hill, 1996. 505 str.
2. Booch Grady, Rumbaugh James, Jacobson Ivar: The Unified Modeling Language User Guide. Harlow, England: Addison Wesley Longman, 1999. 482 str.
3. Booch Grady: Object-Oriented Analysis and Design. Santa Clara, California: Benjamin Publishing Company, 1994. 589 str.
4. Grad Janez, Jaklič Jurij: Baze podatkov. Ljubljana: Ekonomska fakulteta, 1996. 254 str.
5. Gradišar Miro, Resinovič Gortan: Informatika v poslovnem okolju. Ljubljana: Ekonomska fakulteta, 2001. 508 str.
6. Hammer Michael, Champy James: Preurejanje podjetja. Ljubljana: Gospodarski vestnik, 1995. 223 str.
7. Indihar Štemberger Mojca: Zlepki v okolju objektnih baz podatkov. Doktorska disertacija. Ljubljana: Ekonomska fakulteta, 2000. 177 str.
8. Kovačič Andrej: Informatizacija in uspešnost poslovanja. Zbornik posvetovanja, Dnevi slovenske informatike 2000, II. Ljubljana: Slovensko društvo INFORMATIKA, 2000, str. 712-715.
9. Kovačič Andrej: Kakšne uporabniške programske rešitve potrebujemo?: Uporabna informatika, Ljubljana, V (1997), 1, str. 8-15.
10. Kovačič Andrej: Podatkovni prototipni pristop h gradnji informatike. Doktorska disertacija. Ljubljana: Ekonomska fakulteta, 1992. 193 str.
11. Kovačič Andrej et al.: Celovite rešitve. Uporabna informatika, Ljubljana, X (2002), 4, str. 185-250.
12. Kovačič Andrej, Vintar Mirko: Načrtovanje in gradnja informacijskih sistemov. Ljubljana: DZS, 1994. 316 str.
13. Laudon C. Kenneth, Laudon P. Jane: Management Information Systems; Sixth Edition. London: Prentice Hall International, 2000. 588 str.

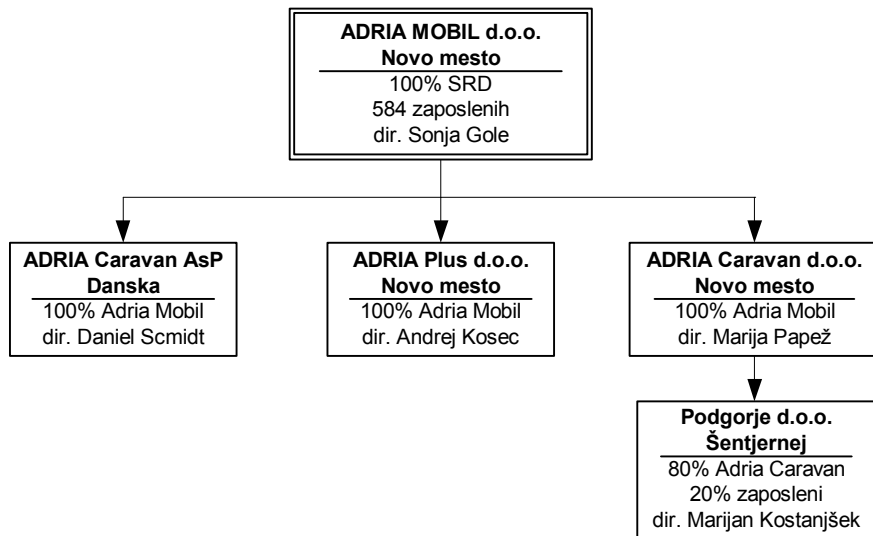
14. Sedej Andrej: Adria Mobil. Caravaning – priloga revije Horizont in Motorist, Maribor, september 2002. 14 str.
15. Shelly, Cashman, Rosenblatt: Systems Analysis and Design; Third Edition. Cambridge: an International Thomson Publishing company, 1998. 450 str.
16. Skukan Katjuša et al.: Stroški, vrednost in cena razvoja programske opreme. Zbornik posvetovanja, Dnevi slovenske informatike 2000, II. Ljubljana: Slovensko društvo INFORMATIKA, 2000, str. 723-729.
17. Veljović Alempije: Osnove objektnog modeliranja UML. Čačak: Kompjuter biblioteka, 2002. 250 str.

7. VIRI

1. Angleško-slovenski slovar. Ljubljana: DZS, 1994. CD-rom.
2. Dokument Projekt prenove IS. Interni podatki Adrie Mobil, d.o.o. 2003.
3. Domača stran Adrie Mobil, d.o.o. [<http://www.adria-mobil.si/>], 20. 12. 2002.
4. Domača stran Aprosa, d.o.o. [<http://www.apros.si>], 1. 2. 2003.
5. Domača stran Citrix Systems, Inc. [<http://citrix.com>], 15. 2. 2003.
6. Domača stran predmeta Razvoj informacijskih sistemov (RIS), Gradiva. DFD.doc. [<http://www.ef.uni-lj.si/predmeti/ris-ekon/>], 14. 10. 2002.
7. Encyclopedia Britannica CD 99: Industrial and Organizational Relations. Chicago: Encyclopedia Britannica, 1999. 3 CD-romi.
8. Interni podatki podjetja Adrie Mobil, d.o.o. 2002.
9. Sistem – priloga revije Monitor: Zunanji strokovnjaki – da ali ne?, Mobilni VPN. Ljubljana: Mladina, d. d., letnik 13, št. 5, 2003 Str. 8-9, str. 38-39.
10. Slovar slovenskega knjižnega jezika (SSKJ). Ljubljana: DZS, 1994. CD-rom.
11. UML Dictionary. [<http://www.softdocwiz.com/UML.htm>], 8. 1. 2003.
12. VAX to Alpha – Passing the torch. [http://h18002.www1.hp.com/alphaserver/vax/timeline/vax_article.html], 1. 2. 2003.

PRILOGE

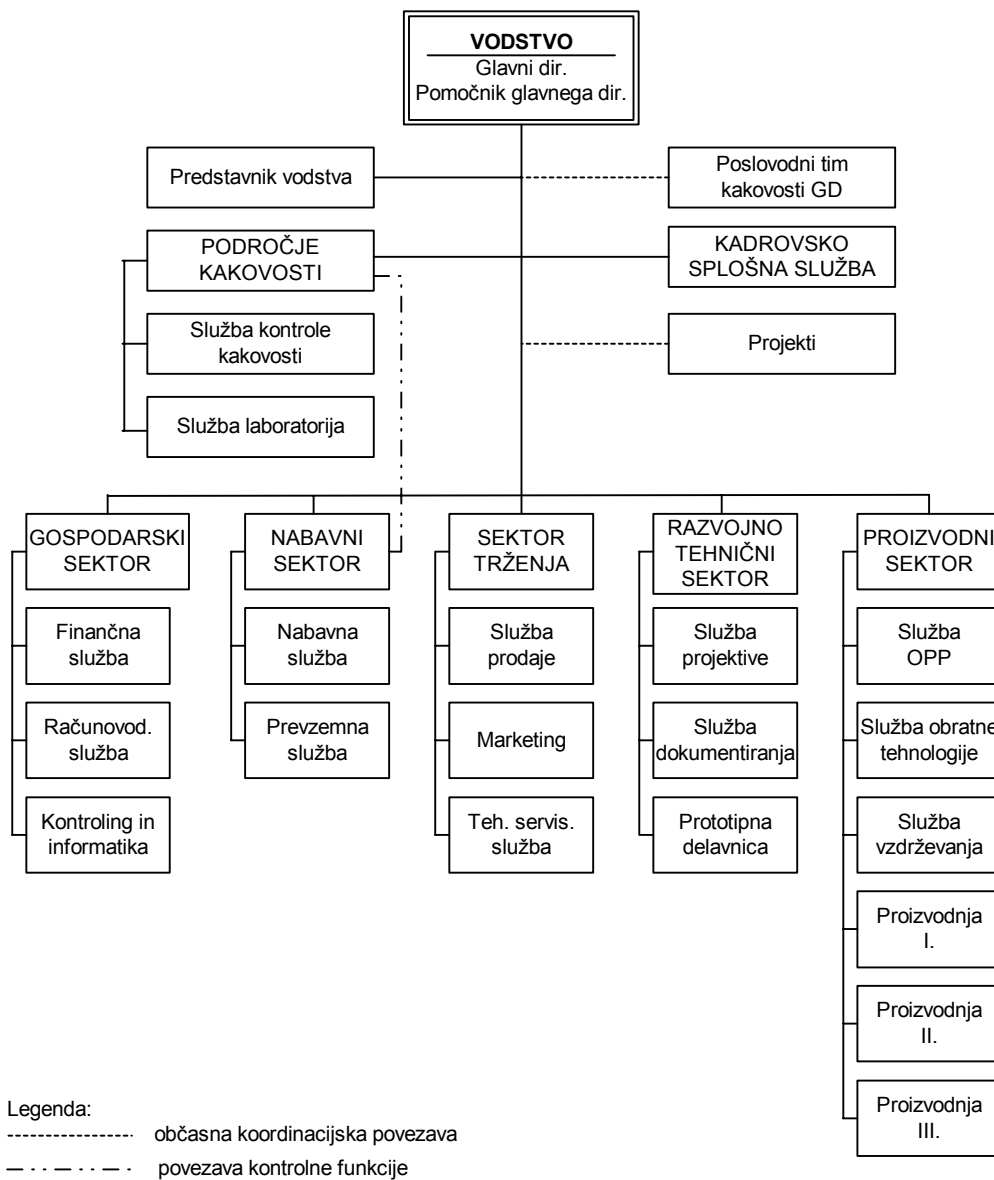
Priloga 1: Poslovne in kapitalske povezave skupine Adria



Vir: Interni podatki podjetja.

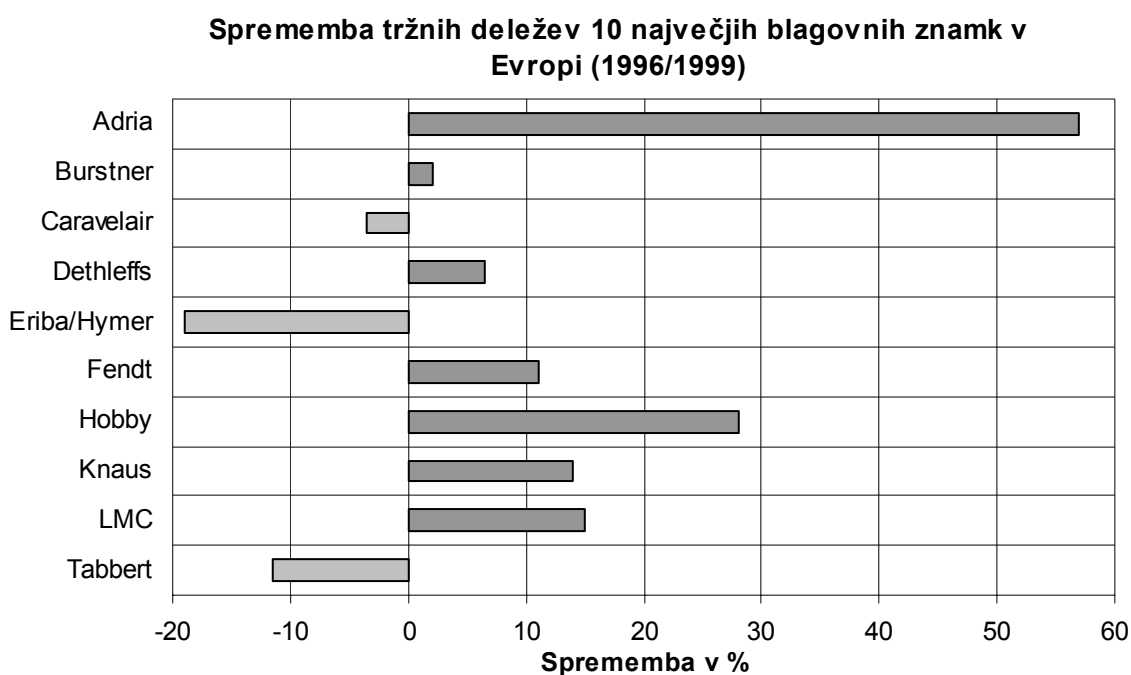
Priloga 2: Poslovanje Adrie Mobil, d.o.o.

a. Organizacijska struktura Adrie Mobil



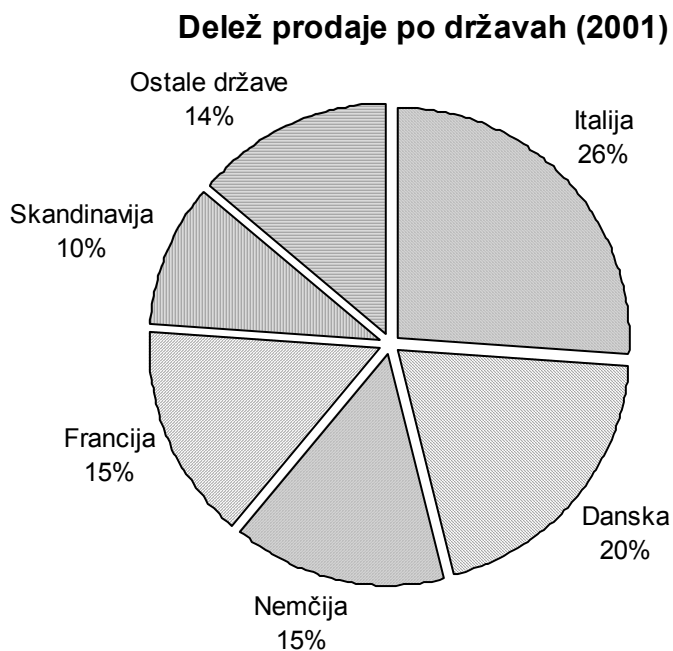
Vir: Interni podatki podjetja.

b. Diagram spremembe tržnih deležev



Vir: Interni podatki podjetja.

c. Delež prodaje celotnega asortimana po državah

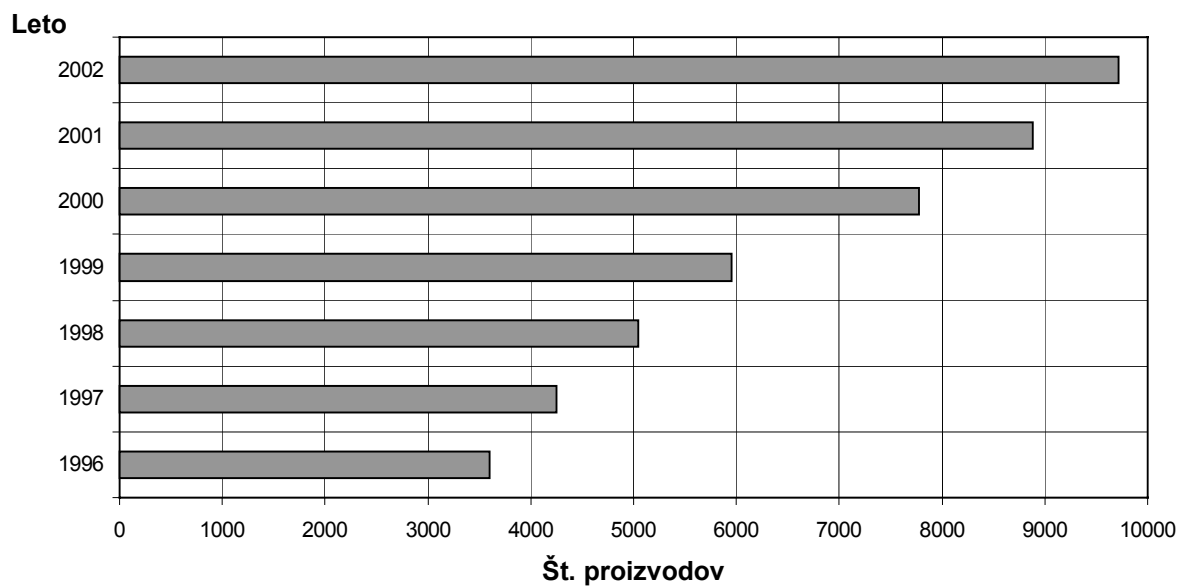


Vir: Interni podatki podjetja.

Pri ostalih državah imajo največje deleže, Nizozemska, Belgija, Avstrija in Švica.

d. Rast prodaje

Trend rasti prodaje proizvodov



Vir: Interni podatki podjetja.

Priloga 3: Moduli UPRO

Programski paket UPRO ima na voljo 7 modulov na področjih tehnologije, proizvodnje, prodaje, nabave, UDOK, M.I.S. in kontrola kvalitete. Adria se je odločila, da bo v svoj sistem vgradila prvih 5, podjetje Apros pa jih bo prilagodilo poslovanju Adrie. V nadaljevanju sem posamezen modul predstavil z alineami, ki najbolje opišejo njegove lastnosti (Domača stran Aprosa).

TEHNOLOGIJA:

- različni načini zapisa sestavnic (klasična večnivojska, najemanje, variantna sestavnica, receptura);
- možnost priprave tehnološke sestavnice (materiali po delovnih mestih);
- možnost priprave tehničnih postopkov za izdelek v več variantah;
- možnost kopiranja sestavnic ali postopkov;
- avtomatično vodenje sprememb na sestavnicah in tehnoloških postopkih (ISO);
- poljuben izbor "časovnega sistema" (šestdesetiški, stotiški);
- simuliranje kalkulacij izdelkov;
- možnost blokiranja opuščenih izdelkov;
- možnost klasificiranja izdelkov – 8-nivojski klasifikacijski sistem;
- možnost systemske zamenjave materialov v celem proizvodnem programu in
- možnost systemske spremembe časovnih normativov v celem proizvodnem programu.

PROIZVODNJA:

- neomejen planski horizont v planiranju;
- različni sistemi planiranja materialov (tedenski, signalne zaloge);
- simulacije plana proizvodnje;
- možnost pokrivanja različnih tipov proizvodnje (za znanega kupca ali na zalogo);
- možnost avtomatičnega prenosa naročila kupca na delovni nalog;
- prijazen nadzor nad potrebnimi proizvodnimi resursi (materiali, delovna mesta zaposlenih);
- spremljanje proizvodnje po kontrolnih točkah;
- 3-cenovno vrednotenje zalog in prometov (planska, nabavna, povprečna);
- možnost pokalkulacije po delovnih nalogih;
- sledljivost posla (naročilo – delovni nalog – dobava – račun ISO);
- možnost priprave etiket s črtno kodo;
- avtomatična evidenca zalog pomožnih materialov v proizvodnji;
- možnost vodenja zalog materialov ali izdelkov po saržah;
- možnost vodenja ur delavcev in
- analize doseganja normativov.

PRODAJA:

- možnost priprave cenikov po kupcih (verzije cenikov);
- možnost oblikovanja dokumentov (dobavnica, račun...) za vsakega kupca posebej;
- cene in računi v kateri koli valuti;
- razporejanje (klasificiranje) kupcev po trgih;
- 4 oblike prodaje (klasična, konsignacijska, predračuni, distribucija);
- različne oblike fakturiranja (posamezna dobava, več dobav skupaj, storitve...);
- možnost zapisa bremepisov in dobropisov;
- možnost priprave izjave o poreklu vgrajenih materialov;
- pripravljene podatki za obračun DDV v saldakontih;
- možnost spremljanja realizacije po komercialistih;
- direktno pošiljanje dokumentov (dobavnica, faktura...) po FAX-u;
- priprava eksternih dokumentov v katerem koli jeziku;
- možnost priprave avansnih računov in zapiranje le teh in
- možnost prenosov analiz in pregledov v Excelu.

NABAVA:

- možnost priprave cenikov po dobaviteljih (variate cenikov);
- možnost avtomatične priprave novega cenika (preračun s faktorjem);
- možnost izpisa plana nabavnih sredstev;
- možnost naročanja storitev;
- prenos vseh poročil in analiz v Excel;
- možnost avtomatičnega prenosa potrebnih materialov iz plana proizvodnje v naročilo;
- možnost oblikovanja dokumentov za vsakega dobavitelja posebej;
- cene v kateri koli valuti;
- 2 obliki nabave (klasična, kooperacijska);
- direktno pošiljanje dokumentov (naročilo...) po FAXu;
- priprava dokumentov v katerem koli jeziku (v šifrantu partnerja izbran jezik, po katerem se dokument pripravi) in
- spremljanje zadnjih nabavnih cen.

NADZOR IN PRIPRAVA ORGANIZACIJSKE DOKUMENTACIJE – UDOK:

- integriranost z Wordom in Excelom ter zmožnost izvoza tudi v formatih *.pdf, *.tif, *.html, *.jpg, *.bmp, *.txt;
- lasten izbor šifranta dokumentov;
- možnost pregleda po avtorjih, prejemnikih, ključnih besedah, procesih...;
- evidenca sprememb na dokumentih in
- elektronsko dodeljevanje dokumentov prejemnikom.

MANAGERSKI INFORMACIJSKI SISTEM – M.I.S.:

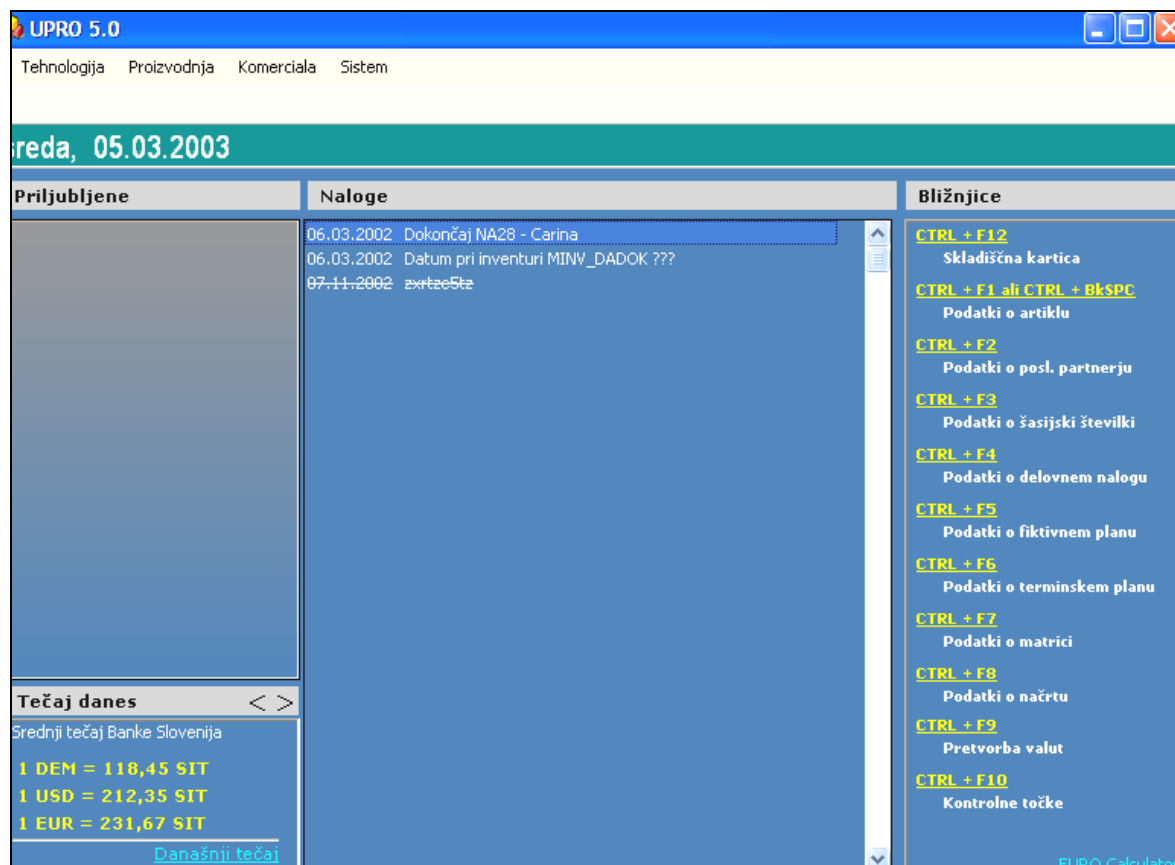
- spremljanje realizacije po mesecih na vseh segmentih (nabava, prodaja, proizvodnja...) in glede na plan;
- diagrami vseh oblik;
- simulacije in planiranje;
- preprosta uporaba;
- enostaven prenos grafov v Word;
- prenosi vseh podatkov v Excel;
- izračuni trendov in povprečij;
- prikazi napovedi (glede na trend);
- priprava letnih planov na podlagi doseženih rezultatov in
- analiza rentabilnosti proizvodnega programa.

KONTROLA KVALITETE:

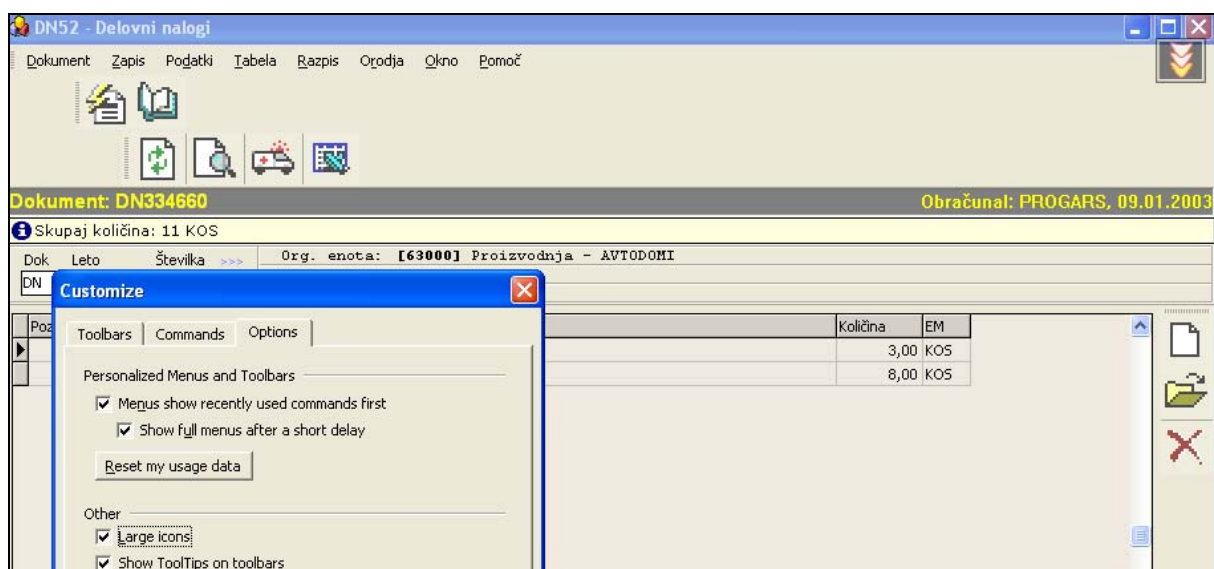
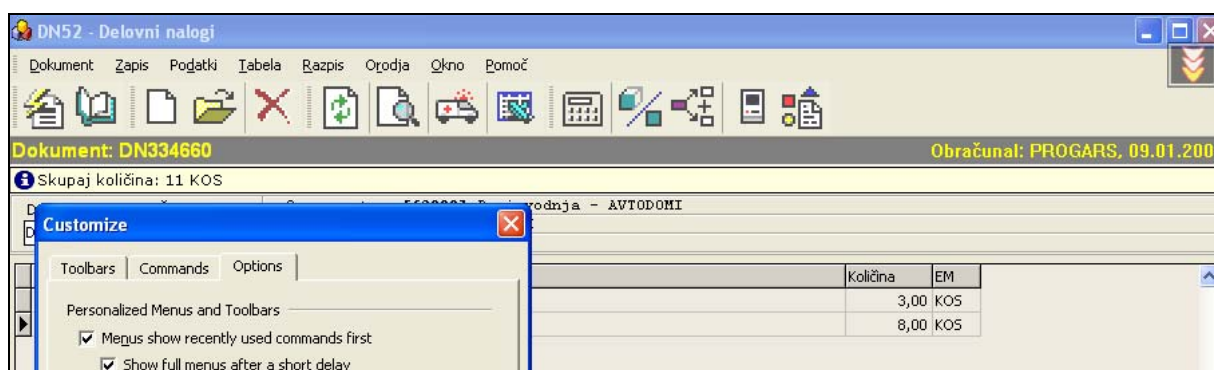
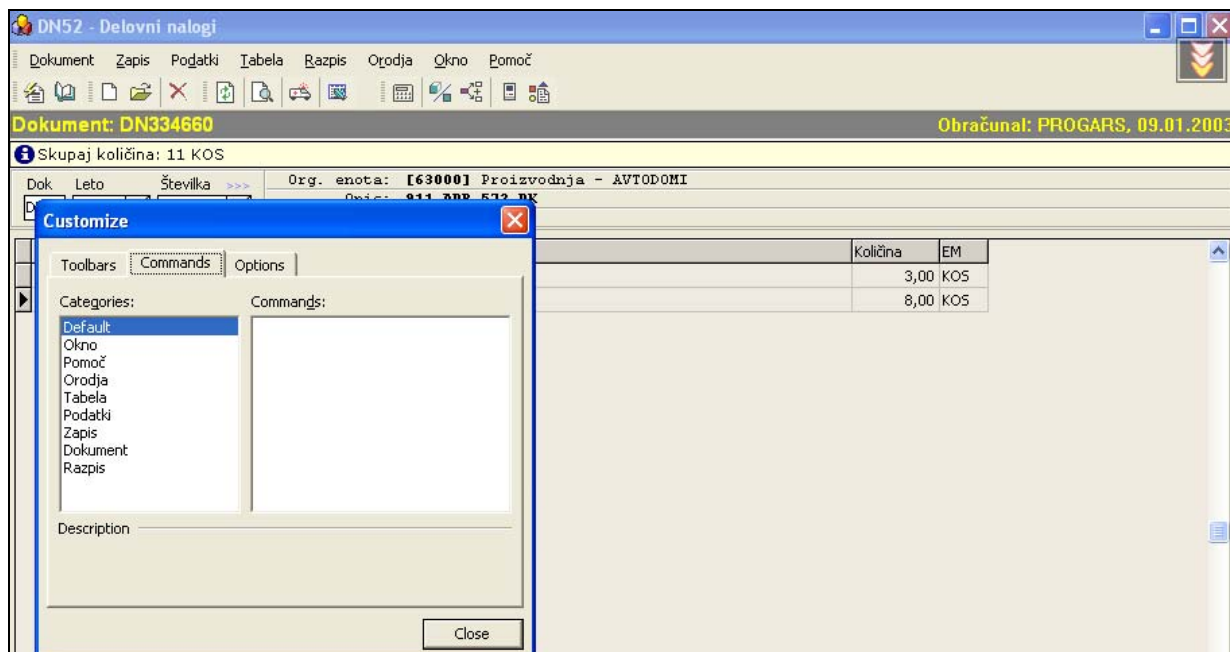
- priprava kontrolnih kartonov za posamezne skupine proizvodov;
- analize po proizvodih, napakah;
- spremljanje doseganja zahtevanih vrednosti kontrolnih parametrov in
- priprava prevzemnih dokumentov v skladišče.

Naslednje slike prikazujejo delovno okolje programa UPRO.

a) Osnovni meni programa UPRO



b) Prilagoditev ukazne in orodne vrstice po željah uporabnikov



Vir: Programski paket UPRO

Priloga 4: Dokument sprememb in dopolnitev

V prilogi sta dva primera sprememb in dopolnitev iz Dokumenta sprememb in dopolnitev, ki so jih izvedli v Adrii Mobil po implementaciji IS UPRO, to je 1. septembra 1999.

a) Dokument sprememb in dopolnitev št. 3

Firma: **ADRIA Mobil d.o.o.**

Enota . **Nabava**

Št: 3

Sprememba / dopolnitve

Opis:

Ceniki

- a) Pripraviti možnost »različnih verzij« cenika. Aktualna verzija je označena pri poslovnem partnerju.
- b) Omogočiti postavitvev prioritet dobaviteljev na artikel.
- c) Na ceniku dodati tudi pakirno enoto.
- d) Omogočiti možnost kopiranja cenika v novo verzijo in sprememba cene s faktorjem.
- e) Omogočiti preračun cenika v izbrano valuto.

Spremembe v katalogu:

Nova polja

DOMA_VERI C(5) Verzija cenika

Nove datoteke:

POPK Podatki specialno za kupca

POPNI Podatki specialni za dobavitelja

Reorganizacija datoteke poslovnih partnerjev!!!

Spremembe na programih:

NA01 Zapis cenika (popolnoma na novo)

OR08 Zapis poslovnih partnerjev (popolnoma na novo)

NA02 Priprava naročila dobavitelju

Novi programi:

Opombe

Prilagoditi in testirati vse programe v nabavnem modulu.

Predvideni čas:

Začetek:

Stroški:

Konec:

b) Dokument sprememb in dopolnitev št. 4

Firma: **ADRIA Mobil d.o.o.**

Enota . **Proizvodni sektor**

Št: 4

Sprememba / dopolnitve

Opis:

Linija »ŠASIJSKA ŠTEVILKA«

Spremembe v katalogu:

Nova datoteka

STVO Datoteka s šasijskimi številkami in z vsemi podatki, ki se nanašajo na to številko.

Spremembe na programih:

Novi programi:

- a) Program za odpiranje številke pri razpisu DN
- b) Program za spreminjanje vozila skozi proizvodnjo
- c) Program za pripravo dobavnice (po vozilih)
- d) Program za pregled vozil po posameznih fazah (stanjih)
- e) Program za prikaz sledljivosti po podatkih za izbrano vozilo

Opombe

Popolnoma nov sklop programov za proizvodnjo vozil (spremljanje vozil od lansiranja do opreme vozila in fakturiranja).

Predvideni čas:

Začetek:

Stroški:

Konec:

Vir: Interni podatki podjetja.

Priloga 5: Slovarček slovenskih prevodov tujih izrazov

- ADSL (asymmetric digital subscriber line) – asimetrična digitalna naročniška linija. ADSL tehnologija omogoča prenos podatkov do 8.192 kbit/s do uporabnika in do 768 kbit/s od uporabnika.
- backup – varnostna kopija
- batch processing – paketna obdelava
- business process reengineering (BPR) – prenova poslovnih procesov
- check constraints – kontrola omejitev vnosa podatkov
- client-server – stranka-strežnik
- computer aided software engineering (CASE) – računalniško podprta gradnja informacijskega sistema
- data flow diagrams (DFD) – diagrami toka podatkov (DTP)
- database – baza podatkov
- database management system (DBMS) - sistem za upravljanje baz podatkov
- distributed database management system (DDBMS) - porazdeljen sistem za upravljanje baz podatkov
- distributed processing – porazdeljeno procesiranje
- downsizing – sestopanje
- FAQs (frequently asked questions) – pogosto zastavljena vprašanja
- file – datoteka
- file retention – pridržek datoteke. To pomeni, da se v primeru izpada sistema, podatki oz. datoteka, ki bi morali biti shranjeni, kljub izpadu sistema ohranijo in se ob ponovnem zagonu shranijo na disk.
- garbage in, garbage out (GIGO) – smeti noter, smeti ven
- hardware – strojna oprema
- help – pomoč
- help desk – način pomoči preko ekrana
- implementation – implementacija, namestitvev
- information system – informacijski system
- input – vhod
- local area network (LAN) – krajevna računalniška mreža
- maintenance – vzdrževanje
- modular design – modularno načrtovanje
- node – vozlišče je fizični element, ki predstavlja računalniški vir, v osnovi ima vsaj nekaj spomina in procesne moči. Izvajalni elementi so lahko na vozliščih.
- online processing – sprotna obdelava
- output – izhod
- outsourcing – organizacijsko izločanje
- processing method – način procesiranja
- query – poizvedovanje
- recovery – obnova datotek

- restart – ponovni zagon
- software – programska oprema
- SQL (Structured Query Language) – strukturiran jezik za poizvedovanje po bazi podatkov
- start-up processing – zagonsko procesiranje
- state transition diagrams – diagrami prehajanja stanj
- structured english – strukturirana angleščina
- support – podpora
- system architecture – računalniška platforma
- system design specification – specifikacija načrtovanja sistema
- systems analysis – sistemska analiza
- systems design – načrtovanje sistema
- systems development life cycle (SDLC) – življenjski cikel razvoja sistema
- systems implementation – implementacija sistema, namestitev sistema
- systems operation and support – delovanje in vzdrževanje sistema
- systems planning – načrtovanje projekta
- tabular application development (TAD) – tabelarni razvoj aplikacije
- top-down approach – pristop od zgoraj navzdol
- unified modeling language (UML) – poenoten jezik modeliranja
- whole/part – nesimetrična povezava tipa celota/del
- wide area network (WAN) – široka računalniška mreža

1 Mb (megabit) = 1/8 MB = 125 KB (kilobajtov)

1 MB (megabyte) = 1024*1024 = 1.048.576 bajtov (ang. byte)