

Izvrješćivanje u SOA okruženju

Ivan Pogarčić, Miro Frančić, Vlatka Davidović
Poslovni odjel, Studij informatike, Veleučilište u Rijeci, Trpimirova 2/V,
51000 Rijeka, Hrvatska
{ pogarcic ,mfrancic, v david}@veleri.hr

Sažetak. Informacijski sustav poslovnog sustava se može promatrati kao konačan produkt niza pažljivo planiranih aktivnosti objedinjenih u metode, metodologije i tehnike primijenjene u strateški pažljivo smišljen plan njegovog razvoja. Na spomenute metode, principe i tehnologije koje se koriste utječe više faktora od kojih su razvoj hardware i software sigurno najznačajniji. Kako je poslovni sustav realizacija korisnički ciljanih aktivnosti tako je i informacijski sustav koji prati te aktivnosti određen potrebama za informacijama koje iz tih aktivnosti proizlaze. Prezentiranje i promatranje strukture i organizacije sustava na spomenuti način u recentno vrijeme je poznato kao service oriented architecture (SOA). Međutim, bez obzira na pristup u razmatranju na najvišem nivou apstrakcije informacijski sustav promatramo kao sustav koji, između ostalog, obrađuje ulazne informacije i proizvodi izlazne podatke i informacije – izvješća (reporting). Oblik, struktura i sadržaj tih izlaza varira u skladu s nizom različitih faktora. U okvirima SOA ta izvješća nerijetko postaju ulazne informacije i podatci pa je problem njihovog oblikovanja i isporuke korisnicima od izuzetne važnosti. Radom se nastoji ukazati na okolnosti i utjecaje koji mogu strože determinirati ta izvješća i ukazati na potrebu i put ka određenim standardima.

Ključne riječi: izvješće, SOA, izlazne informacije, korisnik, prezentiranje informacija

Reporting in the SOA environment

Ivan Pogarcic, Miro Francic , Vlatka Davidović
Business Dept, Study of Information Systems, Polytechnic of Rijeka, Trpimirova 2/V,
51000 Rijeka, Croatia
{ pogarcic ,mfrancic, v david}@veleri.hr

Abstract. Information system of a concrete business system can be treated as the final product of a series carefully planned activities in a unified methods, methodologies and techniques strategically applied in a carefully premeditated plan of development. To the methods, principles and technologies that are used affect several factors. Hardware and software development are certainly the most important factor. Business system is the realization of user-targeted activities and information system monitors these activities and is determined by the needs of the information provided from these activities arise. Presentation and observing the structure and organization systems to mention a way in recent times is known as service oriented architecture (SOA). However, regardless of the approach in considering the higher-level abstraction system is viewed as a system which, among other things, handles input and produces output data and information - report (reporting). Form, structure and content of the output vary in accordance with a number of different factors. In the SOA these reports often become the input data and so are the problem of their design and delivery to users is extremely important. Paper tries to point out the circumstances and influences that can strongly determine these reports and point to the need and time to certain standards.

Keywords: Report, SOA, output information, user, reporting

1. UVOD

Brzina razvoja i usavršavanja tehničkih i tehnoloških rješenja u području informacijsko komunikacijskih tehnologija nameće obvezu stalnog praćenja i prilagođavanja istima. Hardverska i softverska rješenja su sve sofisticiranija. Sedamdesetak godina od prvog računala do danas nije značajan vremenski period ali je napredak impozantan. Softverska rješenja su pratila taj razvoj na primjeren način. Danas su računala prisutna u svim djelatnostima i skoro je nezamislivo organiziranje bilo kakve aktivnosti bez primjerenog učešća računala. Planiranje i razvoj odgovarajućeg sustava za podršku cjelokupnom poslovanju, analiziranju poslovanja i donošenju valjanih i

kvalitetnih odluka zahtijeva odgovarajući pristup i primjenu primjerene tehnologije i metodologije. Koja i kakva će tehnologija izrade i metodologija pri tome biti primijenjena ovisi o više faktora. Podrazumijevano, financijski faktor igra veoma značajnu ulogu ali nije i najvažniji.

Činjenica je da informacijski sustav općenito i konkretno kao aplikativno rješenje nije samostojan jer kao takav ne bi imao smisla. Sinergija poslovnog sustava i pratećeg informacijskog sustava su uvjet njegove egzistencije tj. poslovni procesi su nužno praćeni odgovarajućim informacijskim procesima. Tako vezani mogu egzistirati ali ne nužno i funkcionirati. Funkcija nekog sustava je sadržana u procesnim realizacijama istog, kako na nivou poslovnih procesa tako i na nivou pratećih informacijskih procesa. Međutim, svaki proces ima i pragmatično značenje koje je određeno korisnikom, odnosno njegovom ulogom, pozicijom u sustavu, njegovim potrebama i zahtjevima prema sustavu koji proizlaze iz njegovog ukupnog određenja. Sinergija poslovnog i pripadajućeg informacijskog sustava su taj način spuštene na najznačajniji nivo – nivo korisnika. Dakle, sustav ima smisla teka kad je u njemu korisnik. Ili nije tako?

Na toj poziciji i u tom trenutku se pojavljuje problem koji je određivao, u povijesnom smislu, ono što će se poslije često zvati paradigma. Prihvatimo za potrebe rasprave da je paradigma pristup i način gledanja na problem i njegovo rješenje. Stroži oblik će zahtijevati da prihvatimo paradigmu „kao način života“.

Što je osnova za ovakvu tvrdnju? Osnova je definicija korisnika sustava. Projektiranje, razvoj i implementacija aplikativnih rješenja kao osnovnog dijela informacijskog sustava zahtijeva timski pristup. Pri tome svaki član tima ima jednak značaj, ali je najvažnije da je istovremeno i korisnik sustava, u skladu sa svojom ulogom u poslu i u timu. Posljedično, važeća ili primijenjena paradigma zahtijeva nijansiranje u definiciji ili praktično gledano uvažavanje specifičnosti položaja. S tog aspekta i potreba ovog razmatranja nužno se je referencirati na uobičajenu definiciju informacijskog sustava, posebice na onaj dio koji govori i spremanju, dijeljenju i isporučivanju informacija pozicijama i osobama kojima su te informacije nužne. Pozicija korisnika u tom trenutku iskazuje sve svoje specifičnosti koje moraju biti uzete u obzir. Uobičajeno programski moduli kojima se priprema izvješćivanje prema različitim korisnicima čine između polovine i dvije trećine, ako ne i više, ukupnog broja. Konačno, niti jedno aplikativno rješenje nema statični karakter u svim svojim dijelovima, a naročito ne u dijelu namijenjenom izvješćivanju. To znači da će tijekom vremena biti zahtijevane modifikacije u različitim dijelovima. Izvješćivanje kao dio ukupne funkcije informacijskog sustava je dinamično kako po broju izvješća tako i po potrebnom opsegu izvješćivanja. Svrha ovog razmatranja je ukazivanje na probleme izvješćivanja iz/za informacijskog sustava i njihovu povezanost za korisnika i primjenjivane pristupe odnosno paradigme.

2. Informacijski sustav, korisnik i paradigme

Za svaku je znanost nužno prihvatanje određenih postavki na nivou postulata i aksioma. Te postavke moraju imati snagu aksioma i postulata jer će se sve naknadne teorijske postavke temeljiti na njima. Također je za znanost, iz istih razloga, važna povezanost s drugim znanostima ili znanstvenim granama. Informatika se kao znanost oblikovala u kratkom vremenu. Povezanost s drugim znanostima (matematika, ekonomija, ..) je očigledna kao i involviranost u pojedinim bazičnim, tehničkim i primijenjenim znanostima i znanstvenim granama. Spomenuta povezanost će utjecati na formiranje mišljenja i oblikovanje teorijskih postavki, metodičkih, metodoloških i tehnoloških obrazaca. Tako je u središte informatike kao znanosti postavljena informacija kao osnovni pojam i predmet istraživanja i posljedično sve što može biti vezano uz nju. Jasno da je nužno jednoznačno i precizno definicijsko određenje pojma informacija, zatim posljedično i ostalih temeljnih pojmova. Tu je u prvom redu potrebna točna definicija sustava, a zatim informacijskog sustava. Nužno je inzistirati na rigoroznom određenju. Međutim, ovakav se stav vrlo često ublažava na način da se definiranje tih pojmova oslabljuje popuštajući pragmatičnim stavovima u određenoj situaciji i definiciji konačnog cilja koji se želi postići. Za potrebe ovog razmatranja se treba odrediti prema pojmovima informacija, sustav, informacijski sustav i korisnik. Sve ostalo što bude potrebno proizlaziti će iz definicija tih pojmova i njihovih međusobnih odnosa.

Neka je informacija definirana kao sve ono što smanjuje neizvjesnost i neodređenost (Shannon, 1948), odnosno entropija je mjera vrijednosti informacije u stohastičkom sustavu. Neka se sustav se definira kao grupa nezavisnih, ali međusobno povezanih elementa sastavljenih u jedinstvenu cjelinu sa zajedničkom funkcijom i zajedničkom ciljem (wordnetweb.princeton.edu/perl/webwn?s=system, 10.V.2009). Nadalje, neka je informacijski sustav definiran kao sustav koji prikuplja informacije i podatke, obrađuje ih, pohranjuje, čuva i stavlja na raspolaganje ovlaštenim korisnicima. U literaturi se ovakva definicija može naći u različitim varijantama i s različitim dodatcima što ovisi u prvom redu o kutu gledanja i stanovištu razmatranja. Ako je kostur definicije isti onda su dodatci samo nužnost potrebna u takvim okolnostima. Potrebno je naglasiti još jednu notornu činjenicu, a to je brkanje pojma informacijski sustav i elektronički sustav. Danas je primjena računala u prikupljanju, obradi, čuvanju i dijeljenju informacija podrazumijevana ali se ta dva pojma ne smiju miješati tj. informacijski sustav može egzistirati i bez računala. Zbog naglašene uloge tehnologije i potreba ovog rada neka informacijski sustav bude definiran kao: kombinacija baze podataka, ljudskih i tehničkih izvora koji u zajedničkim organizacijskim okvirima proizvode informacije potrebne za podršku određenim ekonomskim aktivnostima, upravljanju resursima i/ili procedurama donošenja upravljačkih i drugih odluka. Ovim se naglašava djelatni aspekt informacijskog sustava (costg9.plan.aau.dk/Bremen2001/Rados/RadosGlossary.htm, 10.V.2009.).

Kako je sustav realiziran u potpunosti tek aktivnim uključivanjem korisnika potrebna je i definicija korisnika. Iz palete definicija se može odabrati slijedeća: korisnik je osoba koja koristi stvari; netko tko koristi nešto ili je zaposlen oko toga. Tako se u računalnom okruženju korisnik može definirati kao netko tko koristi računalni sustav

programer ili krajnji korisnik (www.netc.org/openoptions/appendices/glossary.html, 10.V.2009.). Važno je napomenuti da korisnik nije nužno osoba, pojedinac već može biti i sustav ali se kao takav nužno personalizira kroz pojedinca ili grupu pojedinaca. U literaturi se može sresti tvrdnja da je informacijski sustav dio poslovnog sustava. Sam po sebi informacijski sustav ne postoji već je nužno sinergijski vezan uz poslovni sustav kojemu služi kao podrška za kvalitetno realiziranje poslovnih procesa i poslovne funkcije u cijelosti. U recentnoj se literaturi može naći mišljenje da je informacijski sustav podatkovna slika procesa iz poslovnog sustava realizirana kroz modele podataka, modele procesa i modele korisnika. Odnos između ta dva fenomena ima svojstva simbioze pa se informacijski sustav treba definirati kao viša projekcija poslovnog sustava pri čemu informacijski procesi odražavaju informacijsku dimenziju poslovnih procesa.

Jasno je da tamo gdje postoje razlike u definiciji osnovnih pojmova, pa čak i kad je u pitanju uvažavanje specifičnosti, nema potpunog suglasja oko razumijevanja, shvaćanja i prihvaćanja načina rješavanja problema. Obično se kao rezultat javljaju različite metodologije, tehnike i tehnologije u planiranju, razmatranju i rješavanju problema. Niti jedna znanost nije imuna na ovakve situacije pa prema tome ni informatika nije izuzetak. Međutim informatika je, povijesno gledano, bila određivana upravo korisnikom, njegovim položajem u sustavu i njegovim potrebama, ali i mogućnostima. Korisnik izvodi aktivnosti kojima se realiziraju poslovni procesi i cjelokupna poslovna funkcija sustava na osnovu informacija dobivenih kroz informacijske procese koji prate i podržavaju spomenute poslovne procese. Ako je korisnik u mogućnosti koristiti računala i odgovarajuća aplikativna rješenja vrlo često time nisu u potpunosti pokrivena sve njegove potrebe za informacijama.

Poslovni sustav nije statičan ni po funkciji niti po strukturi pa se tijekom vremena posljedično mijenjaju potrebe za informacijama i samim time za modifikacijama aplikativnih rješenja. Brzina mijenjanja i potrebnih prilagodbi postaje na taj način značajan ograničavajući faktor u planiranju i razvoju informacijskih sustava. Podrazumijevano je nužan strateški plan, ali upravo on je najizloženiji modifikacijama u vremenu uzrokovanim brzinom promjena i u hardveru i u softveru. Ostali nivoi taktički, operativni i transakcijski se zbog vremena za koje su pripremljeni lakše i jednostavnije prilagođavaju novonastalim situacijama.

Modeli razvoja aplikativnog softvera su, povijesno gledano, bili uvjetovani, određivani i definirani različitim faktorima i okolnostima. Najznačajniji su ipak trenutni tehnološki nivo hardvera te potreba njegove komercijalizacije što se očitovalo u različitostima istih. Drugi važan faktor je uloga i pozicija korisnika sustava.

Tako se u ranim fazama razvoja aplikativnih sustava rješavalo pojedinačne poslovne funkcije npr. glavnu knjigu u računovodstvu dok su ostale funkcije ako su i bile riješene zahtijevale integraciju u jedan zajednički sustav. To je doba lokaliziranih konfiguracija i modularnih rješenja ili tzv. monolitnih host-centričnih modela – period šezdesetih i sedamdesetih godina prošlog stoljeća. Vrijeme programskih jezika treće generacije. Međutim činjenica je da se ovakvi sustavi koriste i danas u nekim poslovima posebno kad se moraju podržavati tzv. rizični poslovni procesi ili procesi u kojim je potrebna obrada većih količina podataka. Već u ovom periodu korisnik sustava može biti krajnji korisnik iz poslovnog okruženja i korisnik koji razvija aplikativno rješenje. Njihov pristup informacijama je različit na djelatnom nivou pa samim time i oblikovanje informacija.

Pojava osobnih računala omogućila je razvijanje klijent/server strukture i značajniji pomak prema korisniku kao središtu sustava. Mogućnosti grafičkog dizajniranja korisničkog sučelja i izlazak iz karakter orijentiranog okruženja su postavile korisnika u izraženiji položaj. Značajan je napredak u informacijskoj osposobljenosti krajnjeg korisnika. Također je evidentno povećanje zahtjeva prema informacijskom sustavu uz osjetni porast sofisticiranosti istih. To je zahtijevalo i sofisticiranija rješenja. Model iz ovog perioda je poznat i kao dvoslojni (2-tier) model razvoja aplikacije – baza je rezidentna na serveru a verzija aplikacije se izvodi na poziciji klijenta. Ipak ovisno o veličini sustava moguće su dvije osnovne vrste organizacija „fat“ klijent okruženje gdje se veći dio aplikacije izvodi na klijent strani i „thin“ klijent okruženje gdje se veći dio izvodi na serveru. (Watt et al., 2002).

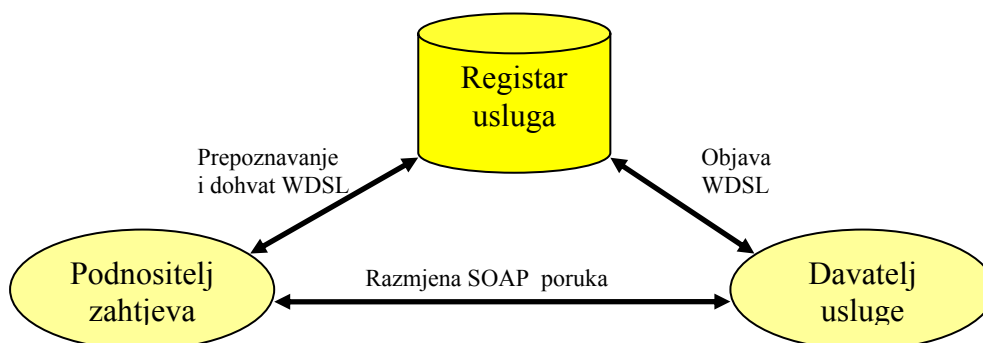
Ovaj model je proširen na 3-slojni model razdvajanjem prezentacijske komponente i modela poslovne logike od modela podataka. Pri tome svali sloj može biti na svojoj platformi. Logika razvoja ovog modela je primjenom interneta razvijena u n-slojni oblik. Prednosti su se očitovala u lakšoj i jeftinijoj izgradnji i organizaciji što je razlog širokoj zastupljenosti u razvoju distribuiranih aplikativnih rješenja podržanih na webu. (Mitchell, 2006). Moderni poslovni sustavi odnosno poslovni procesi su dinamični i vrlo često zahtijevaju prilagodbe novim okolnostima u poslovanju. Također je često potrebno povezivanje proces ili povezivanje s poslovnim partnerima. Posljedično za informacijski sustav to znači osiguranje načina za praćenje svih tih promjena i osiguranje pravovremenih i pravovaljanih informacija. Osiguranje i ispunjenje spomenutih uvjeta je moguće kroz primjenu različitih sredstava postojećih i novih aplikacija organiziranih kroz poslovni sustav. Nekako istovremeno se počinje u planiranju i razvoju informacijskih sustava spominjati pojam paradigma. Najčešće prihvaćena definicija paradigme je definicija Thomasa Kuhna, koji opisuje paradigmu kao skup zajedničkih uvjerenja i dogovora prihvaćenih između znanstvenika/stručnjaka o tome kako bi trebalo shvatiti i riješiti nastali problem. (Kuhn, 1996). Istovremeno se nastoji smanjiti troškove planiranja i razvoja informacijskih sustava. Spomenuto vrijeme je vrijeme objektno orijentiranog pristupa i primjene paradigmi na kojima je zasnovan. Na području Case alata dolazi do jače standardizacije. Pojava UML i sličnih alata značajnije pomiče žarište prema korisniku. Bazirajući se na snimanju zahtjeva i opisivanju slučajeva korištenja (use case) korisnik se postavlja u poziciju u kojoj izrazitije odlučuje o načinu organizacije i izgradnje informacijskog sustava.

Konkurentnost u poslovnom okruženju i potreba brze prilagodbe automatiziranih poslovnih procesa kao odgovor na vanjske i/ili interne utjecaje zahtjeva i brze prilagodbe aplikativnih rješenja. Trenutna aplikativna rješenja su često razvijana u okvirima pojedinačnih odjela i ne prate potrebe poslovnih procesa na adekvatan način. Niti korisnik više nije statičan po svojoj ulozi. Potrebne prilagodbe se reflektiraju preko izmjena procesa i na korisnika. Paradigme objektno orijentiranog pristupa zadovoljavaju djelomično. Potrebe promjena zahtijevaju zahvate ne više u strukturi nego u arhitekturi sustava odnosno načinu organizacije poslovnog a time i informacijskog sustava.

SOA paradigma predstavlja modularni pristup u kojem, na pristupačan, transparentan, implementacijski neovisan, interakcijski operativan način uz višekratnu uporabu komponenti se obavljaju usluge koje se mogu pozivati s udaljenih lokacija konzumirati samostalno ili uz druge programe ili kombinirati sa drugim uslugama (Stencil, 2002). SOA omogućuje izbor ekonomične i učinkovite alternative kroz integraciju heterogenih aplikacija za podršku poslovnih procesa unutar i/ili preko granica sustava (Janssen, et al., 2006). SOA je arhitektura koja podržava organizaciju poslovnih procesa kroz korištenje rješenja iz sastava distribuiranih usluga, gdje je usluga je dobro definiran, frekventan poslovni zadatak koji može biti preoblikovan u računalni program (Uleman, 2006). Usluga se obično provodi kroz neko softversko rješenje koje samostalno egzistira i kroz interakciju s drugim uslugama koristeći labavo spregnute poruke (loose coupling) unutar određenog modela komunikacije (Brown et al., 2005). Primjetno je nastojanje većine autora da SOA pristup definira kao paradigmu. Za spomenute okolnosti tako koncipirana paradigma ima strateški značaj. Service-Oriented Architecture (SOA), omogućava organizaciju i vođenje poslovnih procesa u skladu s informacijskim procesima koristeći raspoložive informacijske tehnologije (IT). SOA je arhitektonski stil koji podržava integraciju poslovnih procesa i vezanih usluga koje mogu biti potrebne u različitim uvjetima kao kad se npr. pristupa sustavu preko mreže. S programerskog aspekta, SOA je skup modela i programskih alata za izgradnju, izvođenje i pristupanje uslugama koji implementira poslovni dizajn.

3. SOA kao paradigma

Evidentna je činjenica da SOA nije revolucionarni preokret već je nastala evolucijom ranijih paradigmi. Međutim u okvirima ICT pojam evolucija ima malo izmijenjeno značenje. Napuštanje jedne paradigma i pomicanje u drugu često izaziva promjene koje imaju revolucionarni karakter. Iako velikim dijelom nasljeđuje objektno orijentirane paradigme SOA kao paradigma čini pomak koji se spominje često u raspravama a smatra se skoro pa podrazumijevanim. To je pomak sa strukture sustava kao najvažnije činjenice ka arhitekturi sustava. Struktura je temeljni, ne nužno, materijalni pojam koji obuhvaća prepoznavanje entiteta od kojih je sačinjen uzorak, promatranje njihove prirode i odnosa te stabilnosti uzorka. Jednako se struktura može promatrati kroz povezanost tih entiteta i usluga koje mogu pružiti. Naravno da to ne znači da je struktura zanemarena, ali je potrebno jasno naglasiti razliku u pristupu. Arhitektura je podložna promjenama koje su često i zahtijevane dok se struktura može promatrati na različitim nivoima u poslovnom sustavu, a posebno u pratećem informacijskom sustavu. Kao proces arhitektura jest skup aktivnosti dizajniranja i konstruiranja neke fizičke ili virtualne strukture ali uključuje dokumentiranost nastanka.



Slika 1. Bazični SOA koncept

Premda se govori o arhitekturi SOA se može promatrati i kao politika, praksa i okvir koji omogućuje primjenu funkcionalnosti koje treba osigurati, kroz skup usluga. Usluge se stavljaju na raspolaganje podnosiocu zahtjeva za uslugom implementirane pomoću standardiziranog sučelja. (CORE.gov, 2005). Potrebno je skrenuti pažnju na pojam podnositelj zahtjeva kojim je uglavnom zamijenjen pojam korisnik. Podnositelj zahtjeva jest korisnik usluge koju sustav pruža, ali je posredno naglašeno da je on pokretač aktivnosti kojima se realizira usluge. Ključni aspekti primjene SOA paradigme su osigurani kroz:

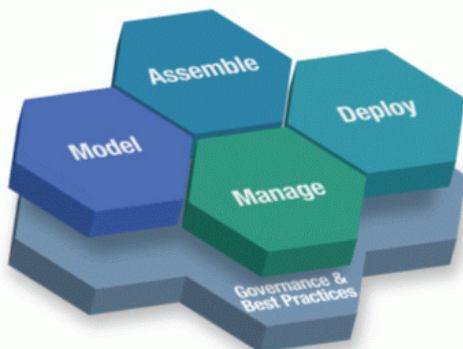
- održavanje određene povezanost usluga uz minimalnu međuzavisnost i stalnu prepoznatljivost,
- ugovor o pružanju uslug(e)a, načinu komuniciranja i obliku dokumentiranosti – izvješćivanju,
- autonomnost i kontrola nad logikom izvedbe koju enkapsuliraju,
- apstraktnost i enkapsuliranost izvan onog što je dogovoreno u ugovoru o pružanju usluga
- reuporabljivost, mogućnost ponovnog – višestrukog korištenja usluge,

- kompozitnost – mogućnost koordiniranja usluga i kompletiranja usluga u kolekcije,
- nezavisnost usluge uz čuvanje informacija o djelatnosti za koju su vezane,
- prepoznatljivost – vanjska deskripcija usluge i mehanizam pomoću kojeg se aktivira.

U navedenim načelima se lako prepoznaju objektno orijentirane paradigme. Primitivni bazični SOA koncept je prikazan na slici 1. Suvremeni oblici SOA se grade po uzoru na primitivni SOA koncept razvijajući ga uz pomoć tehnologije. Tehnika razvoja može varirati, ali varijante moraju imati zajednička svojstva. (Earl, 2005). SOA treba biti tretirana kao evolucijski oblik, ali i taj proces nije dovršen. Ako se tretira kao ideal onda treba biti dostižan ideal. Suvremene SOA arhitekture trebaju biti takve da osiguraju svojstva, odnosno da SOA :

- čini srž(core) računalne platforme za podršku uslugama
- povećava kvalitetu usluga
- potpuno je autonomna
- zasnovan na otvorenim standardima
- podržava raznolikost
- nastoji osigurati interaktivnu operativnost
- promiče aktualna postignuća
- promiče suvremena SOA udruženja
- promiče suvremenu arhitektonsku kompozitnost
- njeguje svojstvenu reuporabljivost
- osigurava i potiče ekstenzivnost
- podržava uslugama orijentiranu paradigmu poslovnog modeliranja
- implementira nivoe apstrakcije
- promiče suvremenu organizacijsku agilnost
- je gradbeni element poslovnog sustava

Zamjetne su i razlike u pristupu poznatih proizvođača softvera (IBM, HP, itd.). IBM model razvoja SOA poznat po akronimu MDAM zapravo opisuje životni ciklus SOA kroz četiri faze (slika 2). Faze čine logički slijed aktivnosti i sve su podržane pažljivo odabranim softverom iz šire palete IBM softvera.



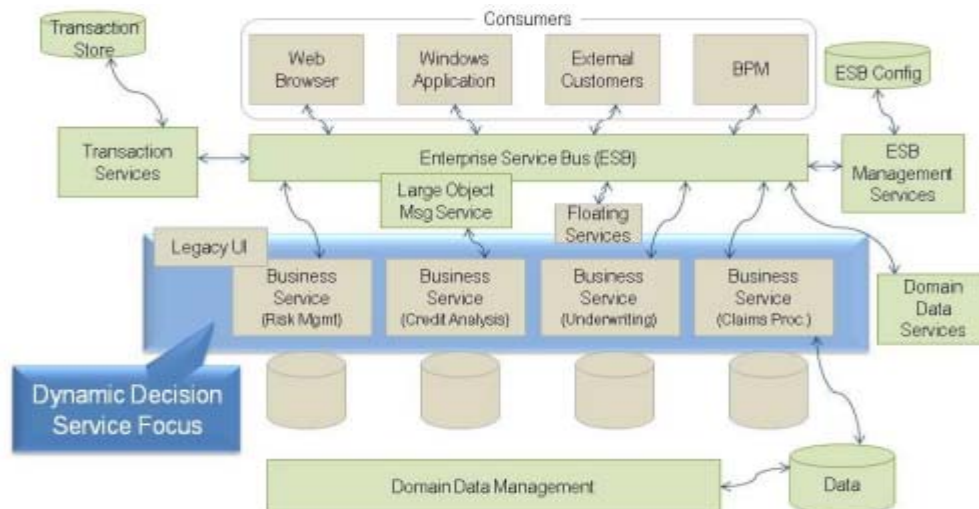
Slika 2. IBM SOA - Osnovni životni ciklus

Životni ciklus svaka usluga počinje s modeliranjem na osnovu zahtjeva za uslugom (model). Slijedi dizajn usluge, njezin razvoj, uklapanje i vezivanje s drugim uslugama i definiranje načina izvođenja – scenarij (assemble). Usluga se raspoređuje i određuje vrijeme izvođenja usluge i vrijeme rada posredničkog sustava (deploy). Faza upravljanja definira okruženje za izvršenje usluga, procese kroz koje će se usluge realizirati te mehanizme za nadzor i upravljanje uslugom (governance&best practices).

Međutim, svaki životni ciklus zahtijeva plan i strategiju provođenja plana u svim njegovim fazama od ideje do implementacije. Složenost strukture poslovnog sustava, poslovnih procesa, informacijskih tijekova i različitost kompetencija korisnika zahtijevaju primjerenu složenost planiranja i dizajna. Da bi se spomenuto moglo promatrati u cjelini nužno je uvesti određenu hijerarhiju u dizajniranje, prvenstveno zbog precizne definicije uloga i odgovornosti učesnika/korisnika te dinamiku njihovih interakcija u postizanju poslovnih ciljeva. Bez obzira na specifičnosti kod planiranja strateški pristup u planiranju i izgradnji SOA treba voditi računa o:

- Prepoznavanju i razumijevanju dinamičkih poslovnih aplikacija i potrebe dinamičkog odlučivanja. Potrebno je graditi aplikaciju koja će brzo i učinkovito prihvatiti promjene u poslovnim procesima. Dinamičnost mora biti na isti način ugrađena u sustav odlučivanja.
- Osigurati prikladnost projekta za dinamički pristup. Ne trebaju sve poslovne aplikacije biti dinamične. Prikladnost se može odrediti promatrajući učestalost mijenjanja poslovne logike, osjetljivost na vremenske termine, proširenost u svim dijelovima poslovnog sustava, utjecaj na podnositelja ili grupu podnositelja. Strateško razmatranje a ne ograničenje na trenutnu situaciju.

- Shvatiti da odjel za IT to ne može sam učiniti. Promjene u poslovanju nisu u nadležnosti IT pa je nužna uključenost zaposlenika iz sfere poslovanja u kreiranju i donošenju odluka.
- Naći načina za izoliranje mjesta s učestalim izmjenama logike i pokušati ih izolirati. Stručnjaci iz poslovnog okruženja mogu pomoći da se logika i pravila odlučivanja učine promjenljivima prema potrebama, kompleksna koliko to zahtjeva poslovanje i vidljiva svima koji imaju prava i obveze koje proizlaze iz poslovanja.



Slika 3. Prototip SOA implementacije (izvor: Chomko,R.:6 strategies for building dynamic business applications with SOA and rules technology, <http://TechRepublic.com.com>, 15.IV.2008.)

- Razumjeti SOA kao paradigmu i ulogu koju igra u čitavom sustavu. SOA kao pristup podrazumijeva razmatranje arhitekture sustava tj. načina slaganja njegovih dijelova i ostvarenja određene funkcije kroz tu kombinaciju elemenata. Cilj je ostvarenje visokog nivoa fleksibilnosti i višestruke uporabljivosti sustava, kontinuiranosti u razvoju, primjeni i održavanju, smanjenju poslova integracije i povećanju osjetljivosti IT sektora na zahtjeve iz poslovnog okruženja. Prototip SOA implementacije je prikazan na slici 3.
- Provesti/primijeniti pravila dinamičkog odlučivanja kao usluge sustava.

Cjelokupna strategija će morati uvažiti i specifičnosti koje bude pokazivao poslovni sustav u bilo kojem segmentu poslovanja i bilo kojem segmentu odlučivanja.

4. Izvješćivanje kao dio i kao funkcija informacijskog sustava

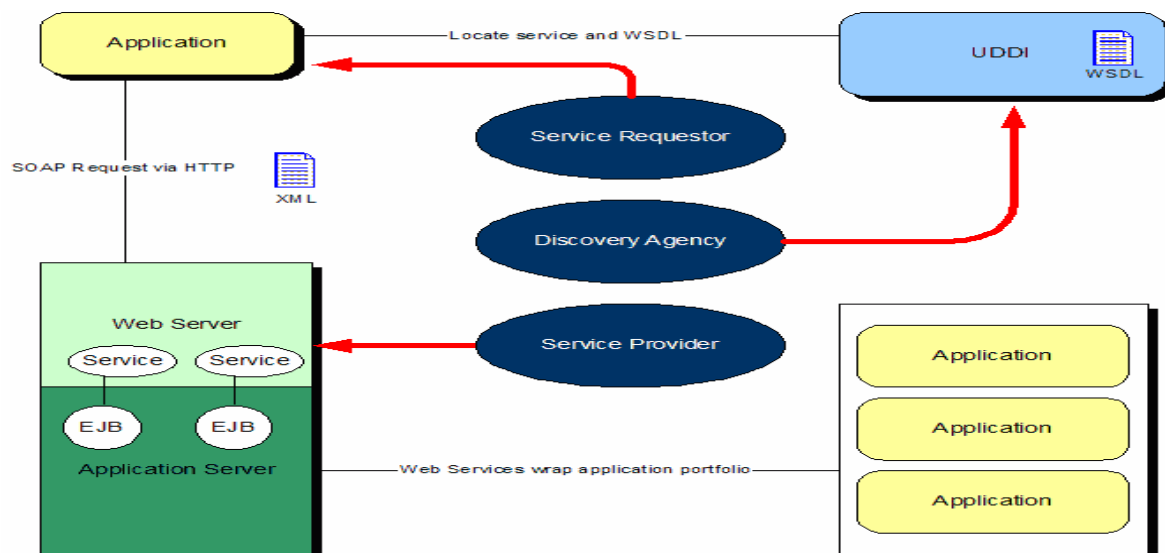
Aplikativno rješenje kao realizacija informacijskog sustava se može promatrati kao skup međusobno povezanih modula – implementacija programskih rješenja. U najgrubljoj podjeli to su moduli za korisničko sučeljevanje s bazom podataka. Pojam sučeljevanje neka predstavlja manipuliranje podacima i informacijama u smislu pohranjivanja, mijenjanja te ponovnog spremanja i/ili brisanja u bazu bez obzira na vrstu baze i njezinu organizaciju. Sama baza po strukturi nema statični karakter ali nema niti preveliku dinamičnost, posebno složene relacijske baze. Veću dinamičnost u promjenama, radi lakšeg dodavanja ili brisanja je moguće ostvariti u objektno orijentiranim bazama podataka. Stoga uobičajena aplikativna rješenja obično imaju jedan na barem dva ako ne i više preostalih modula.

Pregledavanje baze, formiranje različitih izvješća je češći oblik zahtjeva prema bazi podataka. Zbog toga su izvješća pretežiti dio aplikativnog rješenja. Pod izvješćem se za potrebe ovog razmatranja podrazumijeva bilo kakav oblik prikaza postojećih podataka napravljen s određenom svrhom i namjenom. Svako izvješće je odgovor na konkretan zahtjev kojeg postavlja korisnik informacijskog sustava bez obzira na njegovu ulogu i poziciju u poslovnom sustavu. To može biti djelatni učesnik u poslovnom procesu, bilo da je u poslovnom sustavu ili izvan njega (kao suradnik) ili stručnjak iz IT sektora kao netko tko održava informacijski sustav.

Izvješće se može promatrati i kao određena/ciljana kombinacija podataka iz baze. Teorijski je to matematička kombinacija broja podataka ali praktično se broj kombinacija svodi na broj kombinacija koji ima smisla i realnu uporabljivost. Karakteristično za izvješće kao modularni oblik je uobičajena predefiniranost tj. oblici izvješća se formiraju prema zahtjevima već u fazi analize zahtjeva i dizajna sustava. Kao takvi imaju statični karakter i zadovoljavaju svojstva reuporabljivosti. Ponovna uporaba je obično određena vremenskom frekvencijom (dnevno, tjedno, godišnje) ili na zahtjev. Drugi dio izvješća čine ad hoc izvješća koja nemaju predefiniranu vrijednost i

obično nastaju zbog specifičnog zahtjeva iz poslovnog sustava ili njegove okoline npr. različiti zahtjevi za kontrolom i revizijom poslovanja ili informacijskog sustava kao takvog. Promatrano s ovog aspekta izvješće predstavlja oblik sučeljavanja korisnika ss sustavom ili preciznije s bazom podataka.

Izvješće može biti u funkciji sučeljavanja dva ili više sustava ili dva ili više podsustava unutar istog sustava informacijskog sustava. Uobičajeno je da izvješće predstavlja izlazne podatke ili pruža informacije o ishodu određene obrade podataka npr. u funkciji odlučivanja pripremljeni oblik informacija za donošenje odluke u poslovanju. S aspekta formalnog oblika izvješće može biti pripremljeno u elektroničkom ili tzv. tvrdom obliku.



Slika 4. SOA - implementacija u Web okruženju

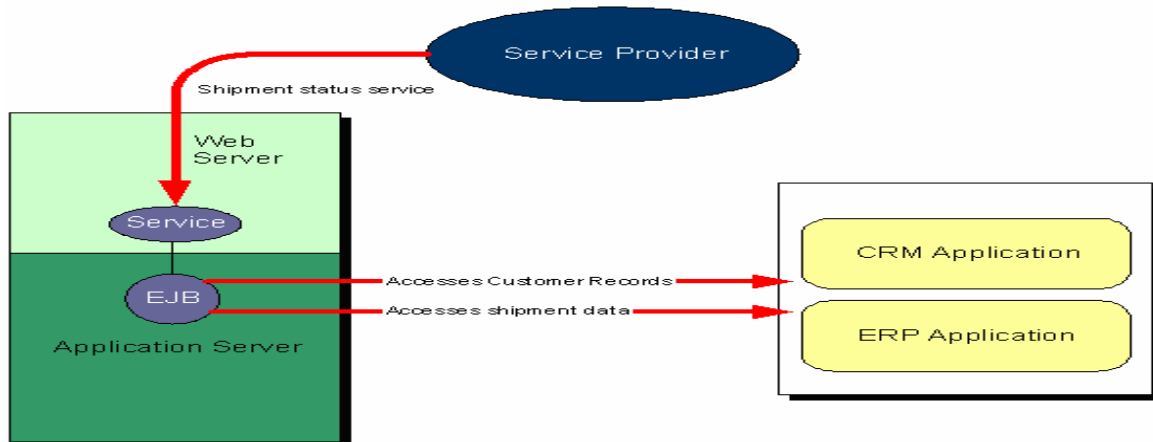


Figure 5. Primjer uključenja usluge kao aplikacijske funkcionalnosti



Slika 6. Pristup podnositelja zahtjeva za uslugom WSDL-u

Što se u SOA okruženju po tom pitanju mijenja, ako se mijenja. SOA se definira kao arhitektonski stilu, čiji je cilj postići labav spoj između softverskih agenata – osoba koje za nečiji račun izvršavaju određenu uslugu. Kako je usluga elementarni oblik djelovanja njezina realizacija treba zadovoljiti obojicu tj. i davatelja usluga i naručitelja usluge. Rezultat je često prikazan u obliku koji se može poistovjetiti s uobičajenim oblikom izvješća. Bez obzira na fizičku realizaciju i okruženje izvješće mora ispuniti osnovnu funkciju i zadovoljiti ciljeve zbog kojih je pripremljeno. SOA pretpostavlja, premda ne nužno, web okruženje pa se time i način pripreme izvješća može realizirati raspoloživim sredstvima dostupnim u tom okruženju. Na slikama 4.-6. su prikazani neki od nivoa implementacije u SOA okruženju. Iz slika je očito da je potreba izvješćivanja izrazita i da zahtijeva primjereno oblikovanje.

5. Problem pripreme, čitanja i razumijevanja izvješća

SOA promovira tzv. labavu povezanost diljem poslovnog sustava. Pri tome je labava povezanost zastupljena na elementarnom nivou u odnosu dvojca zahtjevatelj usluga - davatelj usluga. Premda se ovo svojstvo SOE prezentira kao prednost (samostalnost, odgovornost, prilagodljivost itd.) kad je u pitanju izvješćivanje iskazuje se kao velik nedostatak. Iako se usluge pripremaju za ciljanu populaciju teško je predvidjeti tko će zahtijevati uslugu. Problem je jako izražen ako je ciljana populacija heterogena ili se radi o aplikaciji široke namjene. Ne može se procijeniti informatičke kompetencije onoga koji postavlja zahtjev. Vrlo lako njemu i web može biti relativno nepoznat a pogotovo pojmovi kao XML, WDSL, SOAP, UDDI, EDJ i sl. Dapače poznavanje spomenutog se ne treba niti očekivati. Ali time se dovodi u pitanje trend približavanja informatičkih alata krajnjim korisnicima – onima koji trebaju usluge.

Kad je u pitanju sučeljavanje sustava ili podsustava izvješće je bi-funkcionalno tj. jednom sustavu je u funkciji izlaza, a u drugom može biti ulaz. Zahtjevatelj može imati potrebu predefiniranog formatiranog izvješća koje će moći primijeniti na nekom drugom mjestu bez dodatnih modifikacija što za njega može značiti financijske izdatke i gubitak vremena. Mogućnost izravnog preslikavanja izvješća uz to može biti i prednost za SOA okruženje. U situaciji tzv. tvrdog oblika izvješća mogu biti postavljeni dodatni uvjeti koje treba zadovoljiti kao što su estetski izgled izvješća ili prilagođenost uređaju za tiskanje. Elektronički oblik u web okruženju bi podrazumijevano trebao zadovoljavati principe poznate kao WYSIWYG (What You See Is What You Get – Imaš to što vidiš).

SOA arhitektonski opisuje aplikaciju ili sustav kao skup međuzavisnih usluga koje se mogu uporabiti više puta i brzo implementirati u novonastale potrebe posla koristeći postojeću opremu. Taj skup usluga mora osigurati funkcionalni odgovor jednog dijela sustava ostalim dijelovima. Konačno, politika, praksa i okvir u kojem biti osigurana funkcionalnost skupa usluga treba biti objavljena (u fazi pripreme apstraktnim) zahtjevateljima kroz standardne oblike formi i sučelja. Upravo tu problemi vezani za izvješćivanje kao formu usluge za zadovoljavanje potreba korisnika dolaze do punog izražaja.

Djelomično rješenje je u definiranju ograničenog broja sučelja unutar aplikativnog rješenja. Sučelja trebaju biti prilagođena svim tipovima zahtjevatelja. Sučelja trebaju imati generički karakter i biti dostupna i zahtjevateljima i davateljima usluga. Sučelja trebaju biti podržana porukama za pomoć u okviru ponuđene forme. Rječnik treba biti ograničen ali struktura poruka jasna. Broj aktivnosti u ponašanju sustava treba biti što manji ali optimalan. Shema usluga treba biti tako organizirana da se nova verzija ili vrsta usluge može jednostavno ugraditi bez razbijanja sheme postojećih usluga.

Na tržištu postoji više ponuđenih rješenja. Jedno od njih je Crystal Reports (Crystal Reports inc., Seagate Technology). Alat je komercijalno proklamiran s mogućnostima virtualne konekcije bilo kojem izvoru podataka, dizajniranja i formatiranja interaktivnih izvješća i podjele istih unutar i izvan sustava. Mogu se koristiti i Microsoft Office dokumenti. (<http://www.sap.com/services/bysubject/index.epx>, 10.V.2009.). Među reklamiranim prednostima se ističu: izbalansiranost cijene i stručnosti, mogućnosti on-report sortiranja i filtriranja, minimiziranje napora IT kadra, razvijanje mogućnosti kombiniranja podataka, ušteda vremena dizajna izvješća, moguća ugradnja u Java i .NET aplikacije, kreiranje vlastitih rješenja za nadzor izvještavanja i pregledavanje izvješća. Spomenuto čini srž ponude koju nude i ostali proizvođači.

6. Zaključak

Krajnji korisnik će često imati rezerviran, ako ne odbojan stav, prema promjenama posebno prema čestim promjenama. Stav će biti to rigidniji što se bude više angažiranosti od njega zahtijevalo. Uteg koji može nivelirati takav stav je intenzitet potreba za kvalitetnim informiranjem bilo kad i bilo gdje. Mediji i oblici u kojima se informacije stavljaju na raspolaganje igraju veliku ulogu u procesu prihvaćanja određene paradigme od strane korisnika. Skoro sve rasprave i definicije o SOA naglašavaju fleksibilnost, modularnost, pokrivenost poslovnih aktivnosti povećanu dostupnost i jednostavnije upravljanje uslugama. Jasno je da se može povjerovati u navedeno tek po realizaciji i dokazanosti u stvarnosti. Kako je SOA zasnovana na značajnoj angažiranosti tehnologije može se lako pretvoriti u svoju suprotnost tj. tehnologija može postati sama sebi svrhom ukoliko se ne koristi pravilno. (Liebow, 2005)

Statična i finalna arhitektura nije cilj niti izbor. Arhitektura je realnost s ograničenim trajanjem. Kad se treba organizirati ljude, poslovne procese, podatke, informacije i sustave rezultat je često nepredvidiv i neodgovarajući. U procesu izgradnje se često izgubi iz vida konačni cilj ili bude zamagljen sporednim činjenicama koje ometaju proces. U uvjetima brzih izmjena u poslovanju ipak je potreban primjeren odgovor. SOA to bez sumnje jest, ali to ne znači da je oslobođena poteškoća u primjeni. Posebno poteškoća u dijelu izvješćivanja.

Realizacija usluga je ključni cilj SOA. Usluga je atom sustava, može biti jednostavna ali može biti i složena ili kombinirana. SOA dozvoljava višeslojnu organizaciju automatiziranosti. Na svim nivoima je nužno izvješćivanje bez obzira na akter u realizaciji usluge. Odnosno usluga je često i predstavljena u obliku izvješća. Kad zahtjeva pokretanje niza određenih aktivnosti ponovo u dijelu realizacije sadrži neki oblik izvješćivanja. Plan realizacije SOA je baziran na četiri komponente: modelu, prikupljanju i korištenju infrastrukture, rasporedu pri uporabi i modelu upravljanja.

Problem izvješćivanja u SOA okruženju proizlazi iz situiranosti arhitekture u web okruženju. Iz sigurnosnih razloga ne može se zahtjevatelja pustiti na bazu podataka gdje je najlakše napraviti izvješće po ukusu i mjeri trenutnih potreba. To i jest glavni problem koji se treba riješiti a da se pri tome korisniku dopusti sloboda i autonomnost u radu. Informatičke kompetencije zahtjevatelja usluga jesu problem SOA realizacije ali ne mogu ograničavati niti jednu od preostalih funkcija pa niti sami izvješćivanje. Dinamičnost i ekstenzivnost koju pruža SOA mogu ublažiti ako ne riješiti taj problem.

Prelazak na SOA organizaciju može biti kompliciran. Možda slijedeća prispodoba može najbolje ilustrirati stav prema SOA unutar IT sektora. Na upit jednog IT CEO da li njegov tim gradi SOA drugi odgovara: „Moj arhitekt misli da je SOA orijentiran, projektant inzistira na objektnoj orijentaciji a sistem analitičar želi biti orijentiran poslovnim procesima. Sve što mogu reći je da ono što imam nije isto što je bilo prije i čini mi se da smo počeli graditi web usluge” (Erl, 2005)

Literatura:

1. Sommerville I., Sawyer P.: Requirements engineering, John Willey & Sons Ltd., 1997.
2. Kuhn, Thomas. (1996) The Structure of Scientific Revolutions, 3rd Edition, The University of Chicago Press, Chicago.
3. Watt, E.R., Denoncourt, D., Lee, S., Stevens, R., and Cancilla, B. (2002). "Understanding e-Business Application Integration." Double Oak, Texas: MC Press, LLC.
4. Mitchell, R.L. (2006). "Morphing the Mainframe." Computerworld, Vol. 30, No. 5, pp. 29-31.
5. Janssen, M., Gortmaker, J., and Wagenaar, R.W. (2006). "Web Service Orchestration in Public Administration: Challenges, Roles, and Growth Stages." Information Systems Management, Spring, Vol. 23, No. 2, pp. 44-55.
6. Uleman, R. (2006). "Service Oriented Architecture Unveiled." Geospatial Solutions, Vol. 16, No. 6, pp. 30-33.
7. Brown, A.W., Delbaere, M., Eeles, P., Johnston, S., and Weaver, R. (2005). "Realizing Service-oriented Solutions with the IBM Rational Software Development Platform." IBM Systems Journal, Vol. 44, No. 4, pp. 727-752.
8. Stencil (2002). "The Laws of Evolution: A Pragmatic Analysis of the Emerging Web Services Market." The Stencil Group: <http://www.stencilgroup.com>.
9. Erl, T.: Service-Oriented Architecture: Concepts, Technology, and Design, Prentice Hall PTR, August 04, 2005
10. Schaffner, B.: Implement the components of an SOA with Web services, | Sep 25, 2003 7:00:00 AM, TecRepublic
11. Liebow, M.: Do customers really want SOA?, IBM, TechRepublic, ZDNet News: Feb 10, 2005
12. CORE.gov Governance Process: A Joint Proposal of the: Architecture and Infrastructure Committee, Version 1.01, November 2005 Federal Enterprise Architecture Program Management Office, USA

Podaci o autorima:

	
1. First / Middle / Family Name: Ivan, Pogarčić	
2. Titles: MSc in Information System	
3. Position / Since: Head of Education	
4. Institution: Polytechnic of Rijeka	
5. Place and Date of Birth : 1953-07-18	
6. Nationality / Citizenship: Croatian/Croatia	
7. Field of interests (key words): Database, Object-Oriented Analysis and Design , Developement of Information System, Project management ,e-learning	
8. Hobbies: Skiing, music	
9. E-mail address: pogarcic@veleri.hr	
10. Site: www.veleri.hr/~pogarcic	
11. Phone & Fax #: +385 98 456 065 & +385 51 673 529	
12. Postal address: Marinici Mucici 46 a, 51216 Viskovo, Croatia	

(picture place)	
1. First / Middle / Family Name: Miro, Frančić	
2. Titles: Bsc. of mathematical science	
3. Position / Since: Lecturer, since 2003.	
4. Institution: Polytechnic of Rijeka	
5. Place and Date of Birth : 1953-02-16	
6. Nationality / Citizenship: Croatian/Croatia	
7. Field of interests (key words): Strategic planning and development of information systems, Modelling, IT management, Information system quality, Project management	
8. Hobbies: Hiking, Gardening	
9. E-mail address: mfrancic@veleri.hr	
10. Site: www.veleri.hr/~francic	
11. Phone & Fax #: +385 91 53 03 666 & +385 51 353 777	
12. Postal address: Porečka 4, 51000 Rijeka, Croatia	

(picture here)	
1. First / Middle / Family Name: Vlatka, Davidović	
2. Titles: B.Sc.	
3. Position / Since: Asistent	
4. Institution: Polytechnic of Rijeka	
5. Place and Date of Birth : Rijeka, 1971-03-03	
6. Nationality / Citizenship: Croatian/Croatia	
7. Field of interests (key words): Object-Oriented Analysis and Design	
8. Hobbies: Lego, mount climbing	
9. E-mail address: vlatka.davidovic@veleri.hr	
10. Site: http://www.veleri.hr/~vdavid/	
11. Phone & Fax #: +385 91 253 7260	
12. Postal address: Trinajstici 21, 51215 Kastav, Croatia	