

METODA PROJEKTOVANJA I SPECIFIKACIJE WEB SERVISA

METHOD OF DESIGN AND SPECIFICATION OF WEB SERVICES

Sadržaj - U radu će biti prezentovan model za planiranje i specifikaciju web servisa koji se bazira na korišćenju dokumentacije sistema kvaliteta. Na osnovu modela specifikacije web servisa prezentovan je originalno razvijeni CASE alat za projektovanje i specifikaciju web servisa i generisanje WSDL koda.

Abstract – Model for planning and specification of web services, based on documentation of quality system will be presented in this paper. Based on that model, CASE tool for design and specification of web service and WSDL code generator will be presented.

1. UVOD

U samom početku web servisi su uglavnom bili usmereni na komunikaciju između preduzeća i distribuiranih jedinica (u integraciji lanaca snabdevanja između različitih preduzeća), međutim, glavna snaga web servisa će biti iskazana u implementaciji unutar jednog preduzeća, bilo da se pomoću njih rešava pitanje integracije nasleđenih aplikacija, bilo da je reč o podršci novim procesima [1]. Rastuća potreba za korišćenjem web servisa je pokrenula potrebu rešavanja problema vezanih za projektovanje, razvoj, implementaciju, sigurnost i upravljanje web servisima [1].

U radu će biti prezentovan originalni model za planiranje i specifikaciju web servisa koji se bazira na korišćenju dokumentacije sistema kvaliteta, a na osnovu tog modela je prezentovan CASE alat.

2. SPECIFIKACIJA I IMPLEMETACIJA WEB SERVISA

Prilikom procesa implementacije i korišćenja web servisa javljaju se različiti problemi. Postoji veliki broj tehnika i pristupa za prevazilaženje postojećih problema. Koncept CBDi (*Component Based Development and Integration*) koji je koristio IBM sa pojavom web servisa izbio je u pravi plan. Problemi vezani za implementaciju web servisa definisani su kao [1] :

- organizacioni (projekt menadžment, obuka, promene u upravljanju),
- metodološki (potrebe da se postojeće metodologije, dorade i profilišu tako da najbolje podrže SOA),
- problemi vezani za arhitekturu (najbolja praksa i obrasci za kreiranje čvrste ali i fleksibilne arhitekture za podršku web servisima),
- problemi vezani za tehnologiju implementacije (izbor tehnologije implementacije) i
- infrastrukturni i problemi nedostatka softverskih alata (standardi; alati koji su potrebni za servisno-orijentisanu paradigmu).

Drugi autori [2, 3] su kao značajne probleme u specifikaciji i implementaciji označili:

- činjenicu da su pojedinačni servisi dizajnirani bez potrebne analize u vezi poslovnih procesa na koje se oni odnose i veze njihove pozicije u kompletnoj arhitekturi informacionog i poslovnog sistema,
- usklađivanje funkcionisanja web servisa iako su u tom pravcu činjeni mnogobrojni koraci kao definisanje BPEL4WS - *Business Process Execution Language for Web Services* (spajanjem XLANG - Microsoft i WSFL - *Web Services Flow Language* - IBM) i BPML (*Business Process Modeling Language*) sa WSCL (*Web Service Composition Languages*) (Sun, SAP),

Ovde će se razmatrati problemi specifikacije i modeliranja web servisa. Fansel i Busler (2002) [4] su predložili WSMF (*Web Service Modeling Framework*), koji obezbeđuje konceptualni model za

razvoj i opis web servisa. WSMF se sastoji od četiri glavna elementa: ontologije, Fanel (2001) (formalna eksplicitna specifikacija) - koja obezbeđuje terminologiju koju koriste ostali elementi; ciljnog repozitorijuma - koji daje opis problema koji treba rešiti pomoću web servisa; opisa web servisa - gde se definišu različiti aspekti web servisa, i medijatora - koji rešavaju problem interoperabilnosti web servisa. Iz ovog okvira je evoluirao i projekat SWWS (*Semantic Web Enabled Web Service*) koji ima zadatak da definiše automatizovanu, *ad-hoc* interoperabilnost između sistema, a u cilju ostvarivanja poslovnih zadataka.

Iz okvira za razvoj web servisa proistekle su različite metode kreiranja web servisa. Jedan od pristupa za projektovanje i specifikaciju web servisa je i *Event Based Approach* [2]. Ovaj pristup se direktno oslanja na OOA i OOD metodologiju MERODE [2]. Takođe postoji i prilazi koji se zasnivaju na AOCE (*Aspect Oriented Component Engineering*) u kreiranju web servisa [5] ili metoda modeliranja web servisa zasnovana na CBD (*Component Based Development*) [6]. Postoje pristupi implementaciji web servisa koji se oslanjaju na paterne (obrasce), i gde se zagovara primena POAD (*Pattern-Oriented Analysis and Design*) [3]. Sopstvene grupe web servis *pattern-a* formulisao je IBM (*Business, Integration, Composition, Custom, Application, Runtime patterns*) [7], postoje i obrasci definisani od strane J2EE i Microsoft-a. Takođe, mogu se koristiti i drugi prilazi specifikaciji ili implementaciji web servisa kao i softverski alati za podršku, kao što su: IBM Web Services Toolkit, MS Web Service Toolkit itd.

Fokus ovog rada će biti usmeren na definisanje modela za planiranje i specifikaciju web servisa koji bi se bazirao na korišćenju dokumentacije sistema kvaliteta. Na osnovu modela specifikacije web servisa biće razvijen CASE alat za projektovanje i specifikaciju web servisa i generisanje WSDL koda.

3. METODA KREIRANJA WEB SERVISA

U okviru CIM centra Mašinskog fakulteta u Kragujevcu razvijena je metoda specifikacije web servisa koja se oslanja na korišćenje dokumentacije sistema kvaliteta. Osnovu čine polazne pretpostavke da se servisno orijentisana arhitektura upravo zasniva na poslovnim servisima, a da poslovni servisi najčešće odgovaraju dokumentaciji koja se razmenjuje u realnom poslovnom sistemu. Pri tome su dokumenta QS identifikovana kao dokumenta koji na dobar način opisuju procese u poslovnom sistemu i identifikuju razmenjenu dokumentaciju sa okruženjem.

Kreiranje web servisa se odvija kroz sledeće korake:

1. Kreiranje matrice dokumentacije sistema kvaliteta. Dokumenta sistema kvaliteta možemo podeliti na procedure, uputstva, instrukcije, obrasce i zapise. Matrica dokumentacije sistema kvaliteta dovodi u vezu procedure i zapise (odnosno procese i informacije koje su ulaz ili izlaz iz tih procesa). Procedure se u prvom koraku generalizuju kao web servisi, a ulazno izlazni zapisi kao poruke koje taj web servis razmenjuje sa okruženjem ili kao interfejsi.
2. Kreiranje matrice komunikacije. Utvrđuje se između kojih se entiteta vrši razmena dokumentacije. U ovom i prethodnom koraku najbolji polaz je postojanje razvijenih dijagrama aktivnosti. Iz prethodne matrice se izdvaja izabrana procedura i posmatra se kretanje obrazaca i zapisa ka toj proceduri sa aspekta učesnika u komunikaciji. Iz matrice komunikacije se dobijaju dijagrami sekvenci za web servise, sa definisanim scenarijom za web servise (na primer *Request/Response*), operacijama i tipom poruke.
3. Kreiranje matrice podataka, odnosno za konkretan obrazac/zapis koji se razmenjuje između entiteta u komunikaciji, a na osnovu prethodne matrice, utvrđuje se koji podaci se razmenjuju i ko i na

kakav načinu učestvuje u kreiranju, modifikaciji tih podataka. Na osnovu ove matrice generišu se operacije, specificiraju podaci i definišu tipovi podataka.

4. Poslednji korak je implementacija i kreiranje sumarnog opisa web servisa.

3. 1 Kreiranje matrice dokumentacije sistema kvaliteta

Modeliranje sistema kvaliteta zasniva se na osnovnim konceptima procesa i informacija i vezama između njih na raznim nivoima dekompozicije. Veze među procesima i informacijama predstavljaju tokove informacija u sistemu i određene su ulogom koju neka informacija ima u nekom procesu. Zbog toga, dokumentacija sistema kvaliteta, procedure, uputstva, zapisi i obrasci predstavljaju polaznu osnovu za uočavanje razmenjenih informacija između procesa što može da poslužiti za kreiranje web servisa.

Na osnovu sistema kvaliteta i dokumentacije sistema kvaliteta mogu se kreirati asocijativne matrice procedura i zapisa, koja omogućava da se definiše logička arhitektura sistema, koju čine podsistemi i veze između njih i da se ona zatim prevede u servise.

Podsisteme čine grupe međusobno povezanih procesa i informacija koje imaju relativno slabe veze sa drugim podsistemima. Procesi i informacije u asocijativnoj matrici poređani su imajući u vidu informacione tokove i prirodan redosled odvijanja procesa povezanih tim tokovima tako da se većina asocijacija grupiše u pravougaonicima oko glavne dijagonale.

Svrha ove matrice je da:

- definiše potencijalne kandidate za web servise (Svaka pojedinačna procedura je kandidat za servis ukoliko je moguće identifikovati postojanje dokumentacije koja u nju "ulazi" iz okruženja ili koja iz nje "izlazi" u okruženje);
- definiše potencijalne interfejsse između jedostavnijih web servisa (Ukoliko su procedure susedne, onda zapis koji proističe kao rezultat aktivnosti jedne i koristi se u drugoj proceduri služi za definisanje interfejsa između jednostavnijih servisa, ili se vrši objedinjavanje tih aktivnosti u jedan servis);
- definiše potencijalne interfejsse prema nasleđenim sistemima, odnosno nasleđenim bazama podataka (Ukoliko se podaci iz nekog zapisa nalaze u bazi podataka ili ukoliko se koriste podaci iz DB za formiranje nekog zapisa, onda se na osnovu tog zapisa definišu paramteri interfejsa ka nasleđenom sistemu);
- definiše postojanje dijaloga između entiteta u poslovnoj komunikaciji ili između funkcionalnih delova sistema.

3. 2 Kreiranje matrice komunikacije

Sledeći korak je kreiranje matrice komunikacije (tabela 1) gde se unose entiteti u komunikaciji i navode vrste događaja i vrste učešća entiteta u tim događajima

Prethodna matrica nam omogućava da identifikujemo koja komunikacija postoji i na koji način tu komunikaciju možemo podržati web servisima, odnosno u realnom sistemu o kojim je procesima reč i kako ti procesi mogu biti informatički podržani. Ova matrica praktično služi za definisanje slučajeva korišćenja, odnosno na osnovu nje se kreira dijagram koji opisuje dinamiku sistema u pogledu slučajeva korišćenja.

Tabela 1 - Matrica komunikacije

| Razmenjeni dokumenti | Entiteti u komunikaciji | | | |
|------------------------------|-------------------------|---|---|----------|
| | Klijent | Sektor marketing / komercijale i prodaje | | |
| Zahtev kupca za katalog | K,M | O - Qp MS 01 (prijem zahteva za katalogom), B, K- Qz MI 22, | | |
| Katalog proizvoda (Qz MI 22) | Zahtev, B | K,M,B | | |
| Zahtev kupca | K,M,B | O, Qp MS 01 (prijem zahteva i analiza kompetnosti), B | | |
| .. | ... | .. | | |
| Legenda | K | Kreira | O | Obraduje |
| | M | Modifikuje | B | Briše |

Ova matrica, odnosno model baziran na DQS (*Documentation of Quality System*), omogućava da se prateći menjanje statusa dokumenata prate slučajevi korišćenja. Ukoliko su poslovne funkcije strane u komunikaciji, onda razmenjena poruka služi za definisanje tipa interfejsa, a na osnovu slučajeva korišćenja definišu se metode, u suprotnom se definišu interfejsi ka okruženju. Ukoliko neki zapis "proističe" iz nasleđenog sistema, ili nasleđene baze podataka onda se na osnovu tog zapisa kreira interfejs prema nasleđenom sistemu.

U ovom i prethodnom koraku neke procedure su postale kandidati za servise, a neki zapisi ili interfejsi i poruke za razmenu.

3.3 Kreiranje matrice podataka

Poslednji korak predstavlja definisanje operacija, odnosno kreiranje matrice podataka (tabela 2) koja će ukazati na to koji se podaci nalaze u razmenjenim dokumentima, odnosno omogućiće da se na osnovu tih podataka specificiraju podaci u okviru web servisa i da se na osnovu matrice komunikacije ustanovi gde se pojedini podaci kreiraju ili koriste.

Tabela 2 - Matrica podataka

| Operacija | Scenario | Parametar | Tip |
|---|---------------|-------------|--|
| Katalog proizvoda (Qz MI 22) Katalog | KatalogZahtev | nema | |
| | KatalogOdovor | Informacija | anyURL, boolean,float, String, DateTime, |

Na osnovu ove i prethodne matrice dobijaju se dijagram sekvenci scenarija koji prikazuje zahteve za web servisom, odnosno ispunjenje tog zahteva . Na osnovu ovih podataka kompletiraju se informacije potrebne za specificiranje web servisa.

3.4 Implementacija

Na osnovu podataka iz prethodna tri koraka, kao i inforamcija koje se tiču mrežnih portova, povezivanja i protokola kompletira se web servis i daje sumarni prikaz web servisa. Zatim se koristeći neki pristup implementacije definiše scenario implementacije i bira odgovarajuća tehnologija.

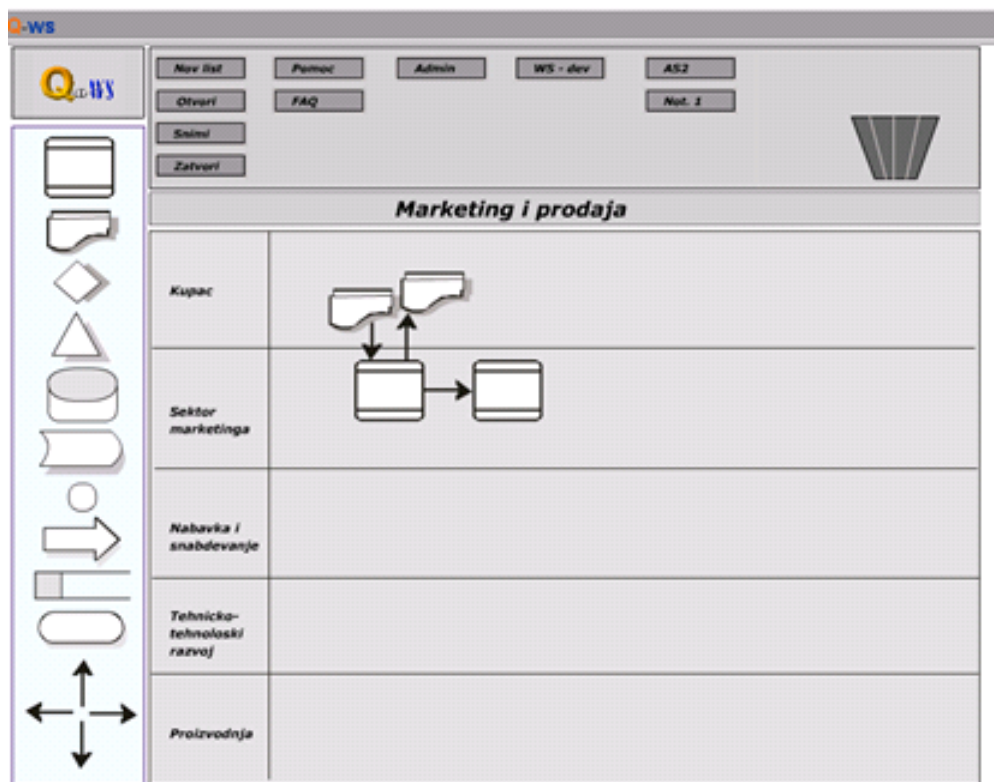
4. CASE ALAT Q-WS KAO PODRŠKA ZA KREIRANJE WEB SERVISA

Osnovna zamisao vezana za ovaj alat je da se korišćenjem dokumenata sistema kvaliteta i dijagrama aktivnosti kreiraju servisi i generiše deo koda web servisa. Alat se zasniva na prethodno razvijenom modelu specifikacije web servisa na osnovu dokumentacije QS

Na osnovu AS2 metode, osnovna dokumentacija može biti: procedura, instrukcija ili uputstvo, što zavisi od kompleksnosti aktivnosti i organizacionog nivoa na kome se aktivnosti sprovode i definišu. Prethodne entitete možemo nazvati zajedničkim imenom postupci. Iz jednog postupka može da proistekne jedan ili više obrazaca. Postupci i obrasci su sastavni deo jednog scenarija na primer: razvoja proizvoda ili marketinga i prodaje. Na realizaciji jednog scenarija je uključeno više funkcija iz preduzeća. Sledeći korak je definisanje faza i funkcija posle koga dolazi na red definisanje elemenata scenarija i unos podataka o elementima scenarija. Prilikom rasporeda funkcija po fazama potrebno je definisati aktivnosti, dokumenta, uneti pomoćne elemente i definisati veze između elementa kao i obezbediti unos osnovnih podataka. Po unosu podataka softver predlaže nove web servise i generiše deo WSDL koda.

4.1 Definisanje scenarija

Na samo početku rada sa ovim CASE alatom moguće je ili kreirati novi projekat sa novim scenarijima ili koristiti stari projekat i modifikovati postojeće scenarije, naravno ukoliko nam korisnička prava dozvoljavaju takvu akciju. Slika 1 pokazuje način kreiranja scenarija. Moguće je kreirati novi scenario ili iskoristiti stari scenario, kao i dodavati nove funkcije za svaki pojedinačni scenario. Ukoliko je izabrala opcija kreiranja novog scenarija, unosi se naziv scenarija i selektuju potrebne funkcije.



Slika 1- Definisanje toka i izgled procedura i zapisa na scenariju Marketinga i prodaje

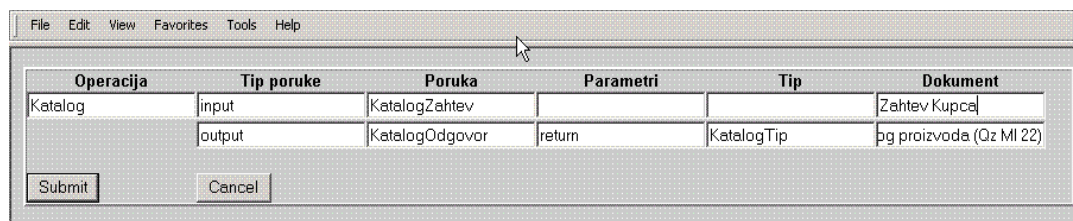
Nakon izbora scenarija potrebno je uneti tokove podataka koji su označeni korišćenjem notacije AS2. Korišćenjem okruženja koje je predstavljeno na slici 1 i izborom simbola koji se nalaze u meniju sa desne strane formira se dijagram. Jednostavnom "prevuci - pusti" (*drag and drop*) metodom raspoređuju se simboli po radnoj površini i uspostavlja veza između njih (slika 1).

Sledeći korak je definisanje parametara za svaku proceduru, odnosno za svaki zapis. Na slici 2 se vidi način definisanja pojedinih atributa procedure. Na ovom dijagramu je prikazana Procedura QpMI01 "Reklama i propaganda" koja definiše da se po zahtevu kupca šalje katalog proizvoda. Sam servis nazvan je WSProdaja i ima operacije Katalog i Naručivanje. Obe ove operacije imaju input i output poruke. U operaciji Katalog to su zapisi "Zahtev Kupca" i "Katalog proizvoda (Qz MI 22)", dok je u operaciji Naručivanje to "Naručivanje od strane kupca".

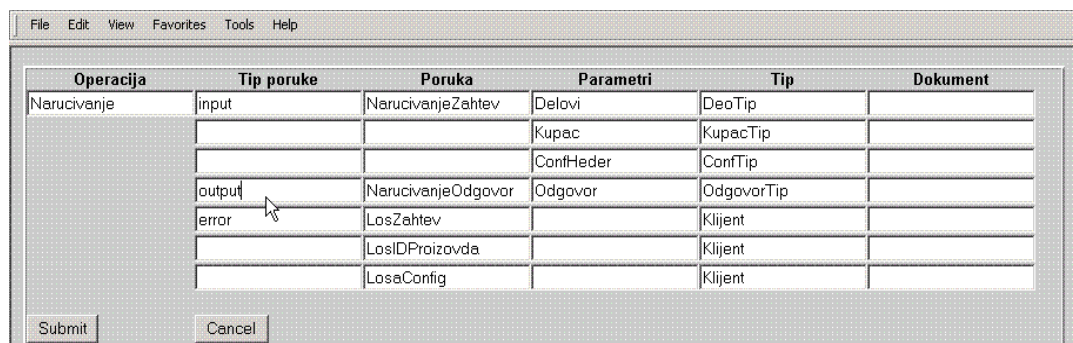
Posle ovih aktivnosti moguć je povratak na izradu novog scenarija unutar projekta ili pak definisanje parametara koji vode kreiranju web servisa i generisanju WSDL koda.

4.3 Kreiranje web servisa

Nakon definisanje potrebnih parametara procedura i zapisa, prelazi se na deo kada CASE alat autonomno identifikuje moguće kandidate za servis i daje potrebne parametre. Na ovaj način ovaj alat iskazuje potencijal ne samo da bude generator WSDL koda i da olakšava kreiranje web servisa već i pruža podršku za SOA arhitekturu i omogućava upravljanje dokumentacijom sistema kvaliteta. Na osnovu razmene dokumenata između dva entiteta, u ovom slučaju to su kupac i funkcija marketinga, softver daje predlog za kreiranje web servisa koji bi imao poruke koje bi suštinski predstavljale zapise ili informacije koje se razmenjuju između kupca i marketinga. Ukoliko je potrebno mogu se naknadno izvršiti korekcije spajanja odnosno deljenja pojedinih servisa i/ili operacija, pri čemu bi razmenjena dokumentacija između dve procedure poslužila kao osnova za definisanje interfejsa između servisa ili nasleđenog sistema. Polaz za generisanje je izbor uređenih parova ili poruka koje se razmenjuju između entiteta, pri ovome zapisi postaju *request/respons* poruke, zapisi postaju Operacije, a spajanje susednih procedura između kojih nema razmene strukturiranog dokumenta produkuje web servis.



**Slika 2 -
Definsanje naziva i osnovnih parametra dokumenta sistema kvaliteta**



**Slika 3 -
WSProdaja Operacija Narucivanje**

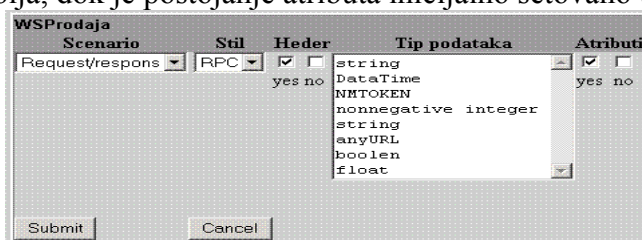
Samo funkcionisanje alata zasniva se na modelu koji je predstavljen u prethodnom tekstu, a koji se, pak, zasniva na definisanju matrica dokumentacije, komunikacije i podataka. U ovom primeru dobijen

je jedan web servis koji je nazvan WSProdaja koji se sastoji iz dve operacije koje su nastale od odgovarajućih procedura, i poruka koje su nastale od pripadajućih strukturiranih ili slobodnih zapisa. Na slici 3 je prikazana Operacija Katalog u WS Prodaja, dok je na slici 4 prikazana operacija Naručivanje.

Svi predloženi podaci mogu se ručno korigovati i može im se promeniti vrednost. Završetkom ovog koraka definisani su svi potrebni parametri koji definišu web servis. Na ovaj način imamo svojevrsnu mapu web servisa. Naredni korak u razvoju ovog alata predstavljala bi vizuelizacija servisa na odgovarajući način. Na ovom nivou svi servisi i operacije dati su tabelarno uređeni.

4.4 Generisanje WSDL koda

Sledeći korak predstavlja sumarna prezentacija WSProdaja, inicijalno su setovane vrednosti scenarija (*Request/respons*), a može se izvršiti izbor stila, hedera i tipova podataka koji se mogu selektovati iz izabranog polja, dok je postojanje atributa inicijalno setovano u ovom primeru slika 4.



Slika 4 - Sumarni prikaz WSProdaja

U ovom koraku, na osnovu unetih i postojećih parametara, generiše se deo WSDL koda. Na ovaj način izvršena je automatizacija procesa kreiranja web servisa u domenu razvoja koda i smanjeni su ili eliminisani uzroci nastajanje grešaka u kodiranju.

Ovaj alat je u mogućnosti da generiše hedere poruka kao i same delove poruka koristeći prethodno određene nazive polja promenljivih, stilova i drugih parametara U ovom delu se definiše input poruka sa delovima hedera i parametrima koji definišu telo poruke, a isto to važi i za output poruku. Nakon definisanih parametara moguće je generisati WSDL kod koji koj je zatim moguće menjati i doterivati. Sam kod ostaje asocijativno vezan za naziv web servisa.

4.5 Karakteristike i ograničenja CASE alata

Ovaj alat ima izvesnih prednosti jer omogućuje specifikaciju web servisa korišćenjem dokumentacije sistema kvaliteta. Na ovaj način standardno opisani procesi bivaju preslikani u web servise. Sem ovoga ovaj alat omogućava znatno skraćenje postupka kreiranja WSDL koda i praktično generiše mnoge standardne komponente koda, a time smanjuje mogućnost grešaka. Alat nam omogućava da rešimo neke od glavnih problema koja nastaju u definisanju SOA, a to su:

1. razumevanje i prikaz postojećih servisa,
2. prikaz informacionih resursa koji su potrebni,
3. definisanje potrebnih interfejsa i prikaz postojećih interfejsa,
4. detaljan prikaz procesa i njihovo razumevanje,
5. definisanje novih servisa i informacija za povezivanje sa nasleđenim sistemom,
6. definisanje ili promena procesa pod uticajem novih servisa,
7. olakšava implementaciju novih procesa,
8. olakšava selektovanje seta tehnologija i
9. omogućava testiranje i ocenjivanje.

Osnovni dalji pravci razvoja ovog alata bili bi usmereni ka omogućavanju grafičke prezentacije dijagrama sekvenci i dijagrama aktivnosti za web servisa, zatim proširivanja i poboljšanja postojećih mogućnosti i integracije sa razvojnim okruženjima i nasleđenim sistemima.

5. UPOREDNA ANALIZA CASE ALATA Q-WS I DOSTUPNIH SRODNIH ALATA

Softverski alati, i softveri uopšte mogu biti analizirani na više načina (prema performansama, kvalitetu, ceni i sl). Prema (Bohem 1989) ocenjivanje nivoa softverskog alata može se izvršiti prema tabeli 2. Softverski alati su rangirani počev od onih koji imaju samo mogućnosti kodiranja, editovanja i ispravljanja greški, pa do onih koji potpuno podržavaju odvijanje procesa ili životnog ciklusa i integrišu se u kompletno razvojno okruženje. Svaki sledeći nivo podržava opcije sa prethodnog nivoa.

Korišćenjem ovog pristupa ocenjivanja softverskih alata napravljeno je poređenje između predloženog Q-WS alata i dostupnih CASE alata u segmentima u kojima oni služe za kreiranje i podršku web servisima (tabela 2). Poređenje je izvršeno sa alatima koji su ponuđeni od strane najvećih softverskih kuća ili predstavljaju lidere u ovoj oblasti. Naravno, sem ovih, postoji još veliki broj drugih alata koji uglavnom podržavaju parcijalne potrebe i funkcije Svi navedeni, u ovom momentu dostupni alati, omogućavaju kodiranje, editovanje, ispravljanje greški WSDL koda. Map Point, IBM Web Service Toolkit i XML Spy omogućavaju importovanje i eksportovanje šema kao i validaciju, verifikaciju i dokumentaciju web servisa.

Tabela 2 - Upporedna analiza CASE alata

| SOFTVERSKI ALATI za web Service i WSDL | Vrlo nizak | Nizak | Normalan | Visko | Vrlo visok |
|--|--|---|---|--|--|
| | Kodiranje, editovanje, ispravljanje greški | Jednostavan <i>front end</i> , <i>back end</i> , malo integracije | Osnove životnog ciklusa, srednji nivo integracije | Jako podržan životni ciklus i srednji nivo integracije | Jako podržan, proaktivni životni ciklus i visoka integracija |
| <i>Q-WS</i> | X | X- | | | |
| <i>MapPoint Web Service SDK</i> , | X | X | X | X | |
| <i>IBM Web Services ToolKit</i> | X | X | X | X | |
| <i>XMLSpy® 2005</i> | X | X | X- | | |
| <i>Stylus Studio 6</i> | X | X | | | |
| <i>Panacea Software</i> | X | X- | | | |
| <i>Gde oznaka X ukazuje na postojanje a X- na delimično postojanje</i> | | | | | |

Ovi alati podržavaju vezu ka XML, XML/XSLT, SOAP.

Mesto alata Q-WS se može videti iz tabele 2. Mora se napomenuti da je alat Q-WS na nižem stepenu doradenosti, na nižem nivou kvaliteta korisničkog interfejsa, da poseduje niži nivo vizuelizacije i da mu nedostaju elementi verifikacije i dokumentacije web servisa. Svi ovi nedostaci su prevashodno posledica činjenice da je Q-WS alat razvijen od strane autora, dok su ostali alati komercijalni proizvodi velikih softverskih kuća.

Alat ima tu relativnu prednost jer daje mogućnost integrisanja sa drugim web aplikacijama i lak za *on-line* testiranje. Međutim, ako se pogledaju neke karakteristike softvera bitne za web servise kao što je: detaljan prikaz procesa u preduzeću i njihovo razumevanja, prikaz postojećih informacionih resursa, definisanje potrebnih resursa za kreiranje web servisa, planiranje i projektovanje web servisa onda Q-WS ima izvesne prednosti. Prednosti se pre svega ogledaju u pogledu planiranja i projektovanja web servisa, gde se zahvaljujući primenjenom postupku omogućuje kreiranje novih web servisa, a zatim se vrši automatsko generisanje koda za izabrani web servis.

Prema navedenom može se zaključiti da alat Q-WS ima jednu prednost u odnosu na ostale alate, a to je podrška procesu planiranja i projektovanja web servisa, dok u kompletnosti rešenja i doradenosti detalja zaostaje za komercijalno dostupnim CASE alatima.

ZAKLJUČAK

Ono što predstavlja najznačajniji originalni rezultat perzentovan u ovom radu jeste predlog modela kreiranja i specifikacije web servisa zasnovanog na dokumentaciji sistema kvaliteta. Kreiranje web servisa se ostvaruje u tri koraka pri čemu se formiraju matrice dokumentacije sistema kvaliteta, matrice komunikacije i matrice podataka. Osnovu čine polazne pretpostavke da se servisno orijentisana arhitektura upravo zasniva na poslovnim servisima, a da poslovni servisi najčešće odgovaraju dokumentaciji koja se razmenjuje u realnom poslovnom sistemu. Pri tome su dokumenta QS identifikovani kao dokumenti koji na dobar način opisuju procese u poslovnom sistemu i identifikuju razmenjenu dokumentaciju sa okruženjem.

Sam model se razlikuje od drugih postojećih modela pošto se zasniva na dokumentaciji QS. Ovaj model ima izvesnih sličnosti sa "*Event Based Approach*" koji se oslanja na izdvajanje osnovnih događaja (*events*) i planiranja, specifikacije i projektovanja web servisa na osnovu tih događaja i metoda vezanih za konkretni događaja (pri čemu se javlja problem u definisanju tih događaja), ali se razlikuje u tome što ovaj model zasniva na razmeni dokumenata (DQS) koji u principu predstavljaju elektronske ekvivalente realnih dokumenata. Poslovna dokumenta i dokumenta sistema kvaliteta predstavljaju poslovnu realnost i lakši su za razumevanje. Za razliku od CBDi modela, koji polazi od poslovnih procesa i na osnovu njih gradi komponente, predloženi model olakšava identifikaciju poslovnih servisa koje je moguće prevesti u procese pošto se zasniva na analizi protoka dokumentacije, koja inače i prati odvijanje poslovnih procesa.

Na osnovu predloženog modela razvijen je softverski CASE alat Q-WS. Ovaj CASE alat, na osnovu predloženog modela, kreira web servise i vrši generisanje WSDL koda koji je podložan kasnijim intervencijama. Razvijeni CASE alat je upoređen sa reprezentativnim softverskim alatima koji se tiču web servisa i WSLD. Mora se napomenuti da je alat Q-WS na nižem stepenu doradenosti, na nižem nivou kvaliteta korisničkog interfejsa i da poseduje niži nivo vizuelizacije, da mu nedostaju elementi verifikacije i dokumentacije web servise u odnosu na alate koji su paralelno prezentirani. Svi ovi nedostaci su prevashodno posledica činjenice da je Q-WS alat razvijen od strane autora, dok su ostali alati komercijalni proizvodi velikih softverskih kuća.

LITERATURA

- [1] Arsanjani A., Hailpern B., Martin J., Tarr L. P., "*IBM Research Report - Web Services: Promises and Compromises*", Computer Science, RC22494 (W0206-107) June 20, 2002

- [2] Wilfried L., Snoeck M. Michiels C., Goethals F., "An Event Based Approach to Web Service Design and Interaction", APWeb 2003
- [3] Buhler P. A., Starr C., Schroder H. W., Vidal M. J., "Preparing for Service-Oriented Computing: a composite design pattern for stubless Web service invocation", University of South Carolina, Computer Science and Engineering, Columbia, 2004.
- [4] Fensel D., Busler C., "The Web Service Modeling Framework WSMF", <http://informatik.uibk.ac.at/c70385/wese/>
- [5] Santokh S., Grundy J., Hosking J., "Developing .NET Web Service-based Applications with Aspect-Oriented Component Engineering", Department of Computer Science1 and Department of Electrical and Computer Engineering, University of Auckland, 2004
- [6] Fu, X., Bultan T., Su J., "A Top-Down Approach to Modeling Global Behaviors of Web Services", University of California at Santa Barbara, REOS 03.
- [7] IBM Corporation 2004. "Navigating you to a new generation of e-business applications", <http://www-106.ibm.com/developerworks/patterns/>
- [8] Casati F., Sayal M., Shan M. C., "Developing E-Services for Composing E-Services", CAiSE 2001



Dr MILADIN STEFANOVIĆ, docent

Naučne oblasti: *Informacioni sistemi, CIM sistemi, Upravljanje razvojem.*

Mašinski fakultet Kragujevac, Katedra za industrijski inženjering
Sestre Janjić 6
34000 Kragujevac

e-mail: miladin@kg.ac.yu



Dr SLAVKO ARSOVSKI, redovni profesor

Naučne oblasti: *Proizvodni sistemi, Menadžment kvalitetom, Menadžment tehnologijama i Inženjerska ekonomija.*

Mašinski fakultet Kragujevac, Katedra za industrijski inženjering
Sestre Janjić 6
34000 Kragujevac

e-mail: cqm@kg.ac.yu



Dr MILAN MATIJEVIĆ, docent

Naučne oblasti: *Automatsko upravljanje: Digitalno upravljanje, Projektovanje SAU, Računarski podržano merenje i upravljanje.*

Mašinski fakultet Kragujevac, Katedra za primenjenu mehaniku i automatsko upravljanje

Sestre Janjić 6
34000 Kragujevac

e-mail: mmatijevic@ptt.yu, control@kg.ac.yu

