UNIVERZITET SINGIDUNUM
BEOGRAD
DEPARTMAN ZA POSLEDIPLOMSKE STUDIJE
I MEĐUNARODNU SARADNJU

DOKTORSKA DISERTACIJA

MODEL INTEROPERABILNOSTI SISTEMA ELEKTRONSKOG POSLOVANJA ZASNOVAN NA SERVISNO ORIJENTISANOM RAZVOJU SOFTVERA

Mentor
Prof. dr Milan Milosavljević

Kandidat
mr Snežana Popović, dipl. inž.

Beograd, 2012. godina

Mentor:

Prof. dr Milan Milosavljević
Univerzitet Singidunum Beograd

Članovi komisije:

Prof. dr Angelina Njeguš
Univerzitet Singidunum Beograd

Prof. dr Božidar Radenković
Fakultet organizacionih nauka
Univerzitet u Beogradu

Model interoperabilnosti sistema elektronskog poslovanja zasnovan na servisno orijentisanom razvoju softvera

Rezime

Predmet disertacije je istraživanje modela interoperabilnosti u sistemima elektronskog poslovanja, zasnovanih na servisno orijentisanom razvoju softvera, posmatrano iz različitih perspektiva. U okviru istraživanja, razmatrani su: standardi; okviri za interoperabilnost i arhitekturu informacionih sistema preduzeća; semantičke tehnologije, zasnovane na ontologijama; kao i servisna orijentacija.
Uvidom u trenutno stanje, najbolju praksu i trendove, disertacija doprinosi rešavanju problema interoperabilnosti uvođenjem: metodološkog okvira za povezivanje na ad-hoc principu u sistemima elektronskog poslovanja; generičkog meta modela za obezbeđenje semantičke konzistentnosti sistema i opisivanje i otkrivanje komponenti u sistemima elektronskog poslovanja; modela servisno orijentisane arhitekture za semantičku interoperabilnost poslovnih procesa na ad-hoc principu.
Verifikacija obavljenih istraživanja sprovedena je uspostavljanjem interoperabilnosti i unapređenjem izvršavanja aktivnosti poslovnih procesa u oblasti delatnosti ustanova za zaštitu kulturnih dobara. Ostvareni rezultati doprineli su efikasnijem korišćenju raspoloživih informacija, dokumenata i ilustrativnih materijala o različitim vrstama kulturnih dobara.

Ključne reči: interoperabilnost; servisno orijentisana arhitektura; elektronsko poslovanje.

E-business systems interoperability model based on service-oriented software development

Abstract

The subject of the dissertation is the research of e-business systems interoperability models, based on service-oriented software development, considered from different perspectives. The study considered: standards, frameworks for interoperability and enterprise information systems architecture, semantic technologies based on ontologies, and service orientation.
Reviewing the current state, best practices and trends, dissertation contributes to solve interoperability problems by introducing: a methodological framework for connecting e-business systems on the ad-hoc basis, a generic meta model for providing semantic consistency of the system as well as description and discovery of components in e-business systems; a model of service-oriented architecture for semantic interoperability of business processes on an ad-hoc basis.
Research verification has been done by establishing interoperability and improvement of business process execution in the areas of the cultural heritage protection. Achieved results have improved the use of available information, documents and illustrative material referring the different types of cultural artifacts.

Keywords: interoperability; service oriented architecture; e-business.

Sadržaj

1 Uvod ... 1
1.1 Predmet i cilj disertacije ... 1
1.2 Naučne hipoteze ... 2
1.3 Značaj i naučno-stručni doprinos teze ... 3
1.4 Metodologija istraživanja ... 4
1.5 Struktura teze ... 5
2 Servisno orijentisani sistemi elektronskog poslovanja ... 8
2.1 Elektronsko poslovanje ... 8
2.2 Metode i tehnike modelovanja sistema elektronskog poslovanja ... 9
2.2.1 Modelovanje poslovanja ... 12
2.2.2 Modelovanje poslovnih procesa ... 16
2.3 Servisno orijentisani razvoj softvera ... 30
2.3.1 Servisi i servisna orijentacija ... 30
2.3.2 Servisno orijentisana arhitektura ... 35
2.3.3 Veb servisi ... 41
3 Interoperabilnost kod servisno orijentisanog elektronskog poslovanja ... 51
3.1 Koncept interoperabilnosti ... 51
3.1.1 Definicije interoperabilnosti ... 51
3.1.2 Nivoi interoperabilnosti ... 53
3.2 Pristupi rešavanju problema interoperabilnosti ... 56
3.2.1 Standardi i interoperabilnost ... 57
3.2.2 Okviri za arhitekturu informacionih sistema preduzeća ... 59
3.2.3 Okviri za interoperabilnost ... 68
3.2.4 Semantička interoperabilnost ... 73
4 Modelovanje interoperabilnosti sistema elektronskog poslovanja ... 88
4.1 Razvoj okvira za interoperabilnost ... 89
4.1.1 Metodologija i proces modelovanja okvira ... 89
4.1.2 Korišćenje SOFI okvira ... 99
4.1.3 Poređenje SOFI okvira sa drugim izvedenim okvirima ... 111
4.2 Razvoj generičkog meta modela za interoperabilnost ... 112
4.2.1 Meta modelovanje ... 113
4.2.2 Generički meta model ... 115
5 Verifikacija razvijenog metodološkog okvira ... 118
5.1 Primena SOFI metodološkog okvira ... 118
5.2 Opis Portala i ciljne grupe ... 119
5.2.1 Opis sistema – priroda i prezentacija kulturnih dobara ... 119
5.2.2 Ciljne grupe i primeri korišćenja sistema ... 122
5.3 Implementacija metodološkog okvira ... 125
5.3.1 Opis Portala i zahtevi ... 125
5.3.2 Poslovni model ... 128
5.3.3 Model sistema ... 138
5.3.4 Tehnološki model ... 150
5.3.5 Diskusija implementacije metodološkog okvira ... 151

6 Naučni i stručni doprinosi 153
7 Dalji pravci rada i istraživanja 158
8 Zaključak 159
Literatura 160
Spisak slika 167
Spisak tabela 170

1 Uvod

Istraživanja u ovoj tezi okrenuta su konvergenciji dva važna i aktuelna pravca razvoja softverskih sistema.

Na jednoj strani su koncepti servisno orijentisanog razvoja softvera, koji bi trebalo da reše pitanja distribuiranih i slabo povezanih sistema. Dosadašnja praksa, pretežno zasnivana na razvoju komponenti, poslednjih godina značajno je podržana servisno orijentisanom arhitekturom (SOA). Pokazalo se da su veb servisi, jedna od čestih implementacija SOA koncepta, primenljivi i efikasni u slučajevima softverskog okruženja, koje karakteriše trend sve izraženije distribucije i sve slabije povezanost softverskih sistema. Uključenjem veb servisa u sistem, stvaraju se preduslovi za pristup softverskim komponentama, koje su donedavno bile iza "barijere".

Na drugoj strani je trend sveobuhvatnog izlaganja informacija, prvenstveno posredstvom Internet servisa. Količina dostupnih informacija uslovila je da se problem "nedostatka informacija" preformuliše u problem: "Kako pronaći odgovarajuće informacije?". Jedno od ponuđenih rešenja, na kome se radi, je uvođenje semantičkih standarda i tehnologija. Poslovne sisteme dodatno interesuje pristup informacijama, funkcionalnostima i resursima, koje žele da integrišu u svoje softverske sisteme. Posmatrano iz ugla poslovnog sistema, pitanja su: "Kako pronaći i integrisati u naš softverski sistem informacije, koje nam nedostaju?" i "Kako partnerima isporučiti potrebne informacije?".

U pokušaju da odgovori na neka od postavljenih pitanja, teza se bavi modelovanjem sistema elektronskog poslovanja, sa ciljem da se uspostavi interoperabilnost između sastavnih elemenata sistema, kao i između sistema i okruženja u kome se sistem nalazi. Ciljevi i zadaci istraživanja okrenuti su pre svega suštinskim i logičkim rešenjima, koja bi trebalo da se uklope u navedene trendove razvoja softvera.

1.1 Predmet i cilj disertacije

Sistemi elektronskog poslovanja najčešće objedinjuju više poslovnih procesa i podržavaju relativno veliki broj korisnika, koji često postavljaju specifične zahteve. Problemi mogu nastati usled heterogenosti obuhvaćenih poslovnih procesa, uključenih podataka i različitih tehnoloških rešenja. Delimično rešenje problema nude tehnička i sintaksna interoperabilnost. Međutim, teškoće i dalje postoje pri izvršavanju poslovnih procesa, zasnovanih na ad-hoc povezivanju aktivnosti unutar sistema i aktivnosti, koje se izvršavaju u sistemima okruženja.

Cilj aktuelnih istraživanja je da obezbede tehnologije i metodološke okvire za dinamičko izvršavanje poslovnih procesa, uz potrebna povezivanja, kao i da omoguće neophodno semantičko razumevanje, između aktivnosti procesa i potrebnih komponenti za izvršavanje aktivnosti.

Osnovni cilj teze je istraživanje modela interoperabilnosti u sistemima elektronskog poslovanja, zasnovanih na servisno orijentisanom razvoju softvera. Istraživanje obuhvata tehnike i metode za kreiranje modela poslovnih procesa, pregled rešenih i nerešenih pitanja interoperabilnosti, načine za semantičko označavanje i otkrivanje servisa, kao i mogućnost razvoja okvira za interoperabilnost. Istraživanja u ovoj tezi obuhvataju:

- istraživanje postojećih rešenja za integraciju poslovnih sistema, klasifikovanih prema načinu povezivanja i nivoima na kojima se vrši povezivanje i njihova primena u sistemima elektronskog poslovanja;
- istraživanje postojećih metoda interoperabilnosti i istraživanje mogućnosti njihove primene u sistemima elektronskog poslovanja, zasnovanih na servisno orijentisanom razvoju softvera;
- istraživanje mogućnosti razvoja modela, neophodnih za: nalaženje; pristup; integraciju i prihvatanje informacija iz distribuiranih, heterogenih izvora podataka;
- modelovanje poslovnih procesa, arhitekture zasnovane na servisima i komponovanje servisnih komponenti u izvršne poslovne procese;
- obezbeđenje semantičke konzistentnosti sistema, koji se modeluje, primenom generičkog meta modela;
- istraživanje mogućnosti semantičkog opisivanja i otkrivanja servisnih komponenti, primenom generičkog meta modela;
- prikaz i analiza koncepata i standarda, vezanih za razvoj i semantičko označavanje veb servisa.

S obzirom na postavljene ciljeve, zadaci istraživanja su:

- istraživanje glavnih standarda vezanih za modelovanje poslovnih procesa, sa osvrtom na oblasti, koje nisu u potpunosti pokrivene postojećim standardima i specifikacijama;
- istraživanje postojećih modela interoperabilnosti;
- mogućnosti primene standardnog i semantički označenog veb servisa;
- razvoj generičkog meta modela za obezbeđenje semantičke konzistentnosti sistema i semantičko opisivanje i otkrivanje veb servisa;
- razvoj okvira za interoperabilnost sistema elektronskog poslovanja;
- verifikacija razvijenog okvira za interoperabilnost i primena okvira za semantičko opisivanje i otkrivanje veb servisa u sistemu za pretraživanje informacija, dokumenata i ilustrativnih materijala o različitim vrstama kulturnih dobara.

1.2 Naučne hipoteze

Predmet istraživanja, postavljeni ciljevi i proistekli zadaci istraživanja bave se problematikom proširenja postojećih okvira za uspostavljanje interoperabilnosti tokom izvršenja aktivnosti poslovnih procesa. Potrebe korisnika sistema uslovljavaju dinamički izbor servisnih komponenti, koje će podržati postavljene zahteve. Dinamičko uspostavljanje veze "aktivnost poslovnog procesa - servisna komponenta" moguće je, ukoliko se semantički opišu servisne komponente i ako aktivnosti poslovnog procesa prepoznaju opisane servisne komponente. U slučajevima, kojima se bavi ova teza, potrebna rešenja za uspostavljanje interoperabilnosti u sistemima elektronskog poslovanja,

dinamičko pronalaženje servisnih komponenti i konačno, zadovoljenje zahteva korisnika sistema, moguće je zasnovati na generičkom meta modelu.

Polazeći ponovo od predmeta istraživanja, postavljenih ciljeva i zadataka, može se postaviti sledeća hipoteza:

Modelovanje interoperabilnosti sistema elektronskog poslovanja moguće je unaprediti razvojem metodološkog okvira i razvojem semantički orijentisanog, generičkog meta modela, koji će obezbediti semantičku konzistentnost modela sistema i koji će se koristiti za kreiranje opisa servisnih komponenti i za njihovo dinamičko pozivanje preko aktivnosti poslovnog procesa.

Detaljnije, postavljena hipoteza uključuje glavne elemente, na kojima se zasniva rešenje postavljenog problema:

- Modelovanje sistema elektronskog poslovanja mora se zasnivati na jasno određenom i dosledno primenjenom metodološkom okviru. Složenost i dinamika procesa, koji se odvijaju u sistemu, kao i trendove razvoja sistema elektronskog poslovanja, inicirali su razvoj više metodologija. Postojanje i korišćenje različitih metodologija ukazuje da nije moguće izabrati "najbolju", već samo "odgovarajuću", ili "prihvatljivu", a polazeći od klase problema koja se rešava.
- Promena poslovnog sistema, uslovljena potrebama i zahtevima korisnika, statičke modele sistema čini neefikasnim i neaktuelnim, u slučajevima kada se model sistema projektuje na osnovu zatečenog, trenutnog stanja sistema. Iz navedenih razloga, željena funkcionalnost može se zadovoljiti primenom opštijeg modela sistema. U ovom radu, kao jedno od mogućih rešenja, predlaže se kreiranje generičkog meta modela.
- Na osnovu neposredno postavljenih zahteva sistema, a po šemi generičkog meta modela, omogućiće se modelovanje dinamičkog izbora servisnih komponenti. Primenom konzistentnog semantičkog opisivanja, moguće je uspostaviti interoperabilnost pri izvršavanju aktivnosti poslovnog procesa, zasnovanog na servisima, i omogućiti njihovo povezivanje po ad-hoc principu.
- Razvijeni metodološki okvir i izvedeni generički meta model poslužiće za modelovanje interoperabilnih, servisno orijentisanih sistema elektronskog poslovanja.

Uspešnost razvijenog metodološkog okvira i semantički orijentisanog meta modela biće verifikovana na konkretnoj, užoj klasi poslovnih sistema. Ukoliko se planirani sistem uspostavi, a kriterijumi verifikacije budu zadovoljeni, realno je očekivati da će razvijeni metodološki okvir moći da ponudi rešenja i za određene šire klase poslovnih sistema.

1.3 Značaj i naučno-stručni doprinos teze

Rešavanje problema interoperabilnosti izuzetno je značajno za unapređenje razmene informacija u svakodnevnom životu, uključujući i oblast poslovanja, pa je samim tim, važan i aktuelan zadatak IT stručnjaka.

Sa namerom da ponudi jedno od mogućih rešenja problema interoperabilnosti, autor ove teze je, saglasno ciljevima i zadacima teze, uradio određena istraživanja. Sumirajući rezultate, osnovni doprinosi ovog istraživanja su:

- Razvoj metodološkog okvira za povezivanje na ad-hoc principu u sistemima elektronskog poslovanja.
- Razvoj generičkog meta modela za obezbeđenje semantičke konzistentnosti sistema i opisivanje i otkrivanje komponenti u sistemima elektronskog poslovanja, zasnovanih na servisno orijentisanoj arhitekturi.
- Predlog arhitekture za semantičku interoperabilnost poslovnih procesa na ad-hoc principu.
- Verifikacija razvijenog okvira, na primeru poslovnog sistema ustanova kulture, a u cilju dinamičkog izvršavanja procesa za pretraživanje i prezentovanje informacija, dokumenata i ilustrativnih materijala o različitim vrstama kulturnih dobara.

Ostvareni, postojeći nivo interoperabilnosti, u oblasti elektronskog poslovanja zasnovanog na servisno orijentisanom razvoju softvera, ne obezbeđuje ad-hoc pronalaženje i pozivanje servisnih komponenti, neophodnih za kompletno izvršenje poslovnog procesa. Tokom izvršavanja poslovnog procesa, pozivaju se unapred dodeljene i opisane servisne komponente. Nasuprot tome, predloženi metodološki okvir omogućiće da se, na osnovu generičkog meta modela, semantički dopuni model poslovnog procesa i model servisnih komponenti i uspostavi interoperabilnost na ad-hoc principu.

Metodološki okvir, predložen u ovoj disertaciji, može se realizovati u nekoliko koraka. Kreiranjem generičkog meta modela, moguće je ad-hoc komponovanje aktivnosti poslovnog procesa i njihovo semantičko označavanje. Time se stvaraju preduslovi za dinamički izbor potrebnih servisnih komponenti. Za kompletiranje izvršenja poslovnih procesa elektronskog poslovanja, neophodno je uspostaviti i odgovarajuću, semantičku interoperabilnost.

Verifikacija rezultata istraživanja biće ilustrovana na primeru planiranja organizacijski složenog sistema ustanova za zaštitu različitih vrsta kulturnih dobara.

1.4 Metodologija istraživanja

Teza bi trebalo da odgovori na aktuelna pitanja u oblasti modelovanja interoperabilnosti servisno orijentisanih sistema elektronskog poslovanja. Kako bi se dostigao postavljeni cilj, primenjena je kombinacija metodologija: konstruktivnog pristupa istraživanju i istraživanje dizajniranja informacionih sistema.

Konstruktivni pristup istraživanju predstavlja suštinski značajnu analizu problema i prikaz rezultata u obliku kreiranih: tekstova; modela; dijagrama i planova. Ovaj pristup često se koristi za istraživanje u računarskim naukama i pogodan je za korišćenje u oblasti primenjenih istraživanja [1].

Istraživanje dizajniranja informacionih sistema uključuje dizajn novih, ili inovativnih artifakata i analizu njihovog korišćenja i performansi, sa ciljem unapređenja i razumevanja aspekata ponašanja sistema koji se dizajnira. IT artifakti definišu se kao [2]: konstrukcije (rečnici i simboli); modeli (apstrakcije i reprezentacije); metode (redosled aktivnosti) i kreiranje instanci (implementacija i prototip sistema). Očekuje se da artifakti budu funkcionalno opisani, kako bi se omogućila njihova primena na odgovarajući domen.

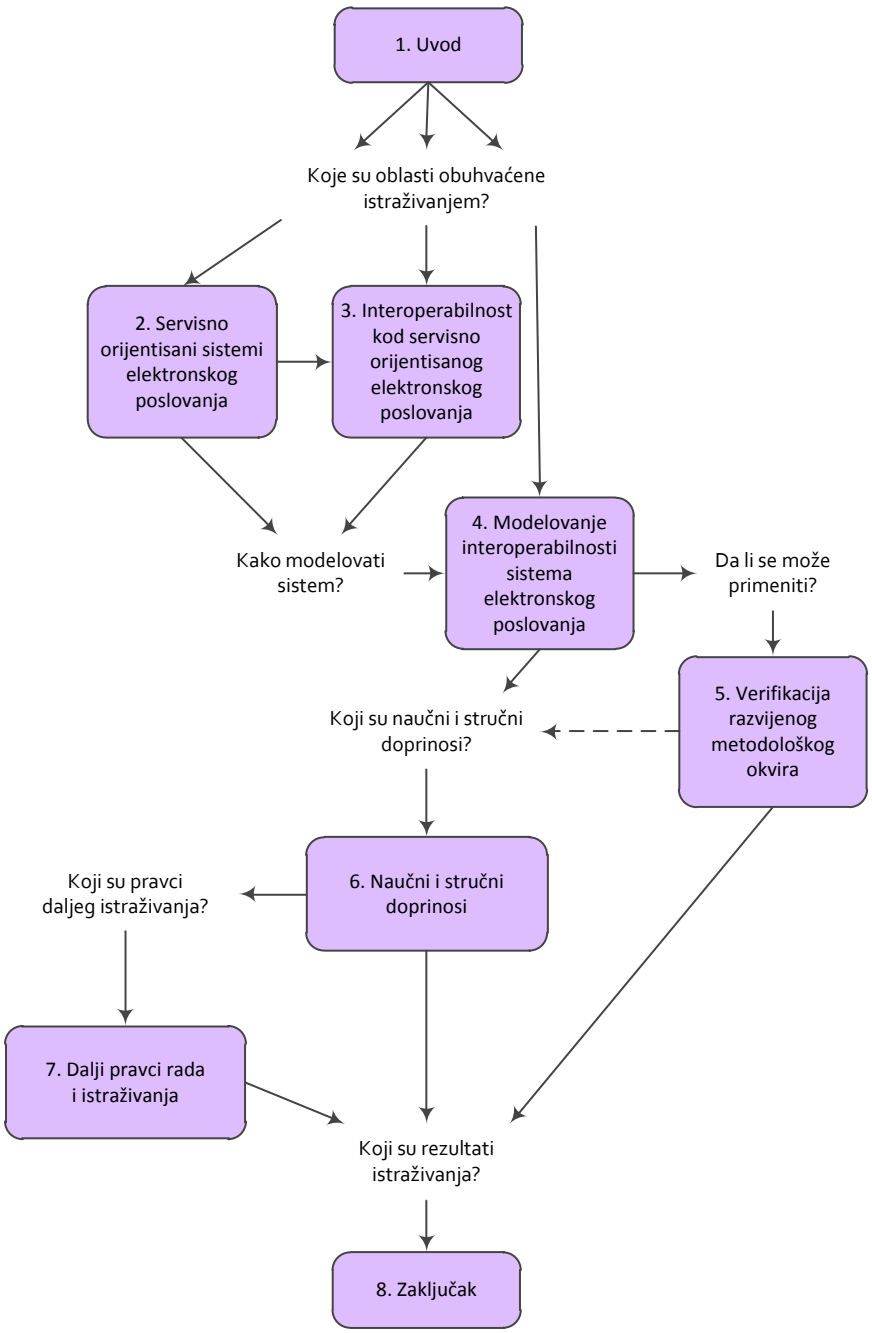
Kombinacijom obe metodologije, istraživanje obavljeno u ovoj tezi realizovano je u okviru sledećih koraka:

1. prikupljanje i proučavanje postojeće literature i modela;
2. otkrivanje problema zasnovano na aktuelnosti, trendovima i pretpostavci da je moguće doći do inovativnih rešenja, koja će unaprediti performanse ili primenu u domenu, koji se istražuje;
3. razumevanje i postavka problema;
4. predlog rešenja;
5. izgradnja inovativnih rešenja;
6. prikaz veza između teorije i doprinosa istraživanja;
7. procena uspešnosti i primenljivosti rezultata istraživanja i
8. zaključak.

1.5 Struktura teze

Spisak i tematska povezanost poglavlja, koja čine tezu, prikazana su na slici 1.1.

Poglavlja 2. i 3. izlažu teoretsku osnovu istraživanja. Poglavlje 4. koristi teoretske postavke za modelovanje interoperabilnosti sistema elektronskog poslovanja, opisane prethodnim poglavljima. Verifikacija razvijenog metodološkog okvira prikazana je u poglavlju 5. Završna poglavlja opisuju naučne i stručne doprinose teze i ukazuju na dalje pravce rada i istraživanja.



Slika 1.1: Grafički prikaz strukture teze

U nastavku dat je pregled pojedinačnih poglavlja.

- Poglavlje 1. uvodi predmet, ciljeve i zadatke istraživanja. Istraživanje je motivisano potrebom da se unapredi modelovanje interoperabilnosti servisno orijentisanih sistema elektronskog poslovanja i pretpostavkom da se unapređenje može postići razvojem metodološkog okvira i generičkog meta modela.

- Poglavlje 2. uvodi i objašnjava pojam elektronskog poslovanja i klasifikuje metode i tehnike modelovanja sistema elektronskog poslovanja, koristeći Open-EDI referentni model. U poglavlju je ukazano na različite pristupe modelovanju elektronskog poslovanja i

klasifikovane su postojeće metodologije i tehnologije u ovoj oblasti. Osim toga, objašnjen je način, na koji se različite specifikacije međusobno uklapaju i navedene su mogućnosti za prevazilaženje postojećih razlika. Poglavlje uvodi koncept servisne orijentacije, čija primena omogućava izgradnju distribuiranih sistema u formi servisa, i opisana je SOA, kao najčešće korišćeni arhitektonski stil za kreiranje servisno orijentisanih sistema. Daljim istraživanjem obuhvaćen je odnos modelovanja poslovnih procesa i SOA. Za realizaciju servisno orijentisanih sistema preporučeno je korišćenje veb servisa. Preporuka je zasnovana na činjenici da veb servisi podržavaju: slabo povezivanje; ugovore; apstrakciju i mogućnost kompozicije. Navedene karakteristike veb servisa obrazložene su u poglavlju opštim opisom samih veb servisa, opisom njihove arhitekture, platformi za izvršavanje i načinom njihovog komponovanja.

- Poglavlje 3. predstavlja koncept interoperabilnosti kod servisno orijentisanog razvoja softvera, kojim se bavi istraživanje i opisuje pristupe za rešavanje problema interoperabilnosti. U ovom poglavlju naglašena je važnost primene standarda pri rešavanju problema interoperabilnosti. Takođe su opisani okviri za arhitekturu informacionih sistema preduzeća i interoperabilnost. Polazeći od činjenice, da bi korisnici i kreatori servisa trebalo da razmenjuju informacije, koje će obe strane razumeti, u poglavlju je posebno razmatrana semantička interoperabilnost i njen značaj za primenu kod SOA.

- Poglavlje 4. predstavlja glavni naučni doprinos. Poglavlje opisuje postupak za prilagođavanja Zahmanovog okvira potrebama modelovanja sistema elektronskog poslovanja. Polazeći od originalnog Zahmanovog okvira, a imajući u vidu zahteve i karakteristike servisno orijentisanog sistema, koji se modeluje i koji je potrebno razviti, prikazan je algoritam za punjenje ćelija Zahmanovog okvira. Na osnovu analize popunjenosti ćelija Zahmanovog okvira, izveden je i opisan novi okvir, koji bi trebalo da usmeri i prati razvoj interoperabilnih, servisno orijentisanih sistema elektronskog poslovanja. U drugom delu poglavlja predstavljen je razvijeni generički meta model. Prikazani generički meta model namenjen je za kreiranje i proveru semantičke konzistentnosti kolekcija artifakata, kojima se popunjavaju ćelije razvijenog okvira.

- Poglavlje 5. opisuje verifikaciju razvijenog metodološkog okvira, tokom svih faza razvoja sistema za pretraživanje i prezentaciju KD, počev od definisanja i analize zahteva, zaključno sa analizom dobijenih rezultata i mogućnosti.

- Poglavlje 6. navodi i obrazlaže naučne i stručne doprinose teze.

- Poglavlje 7. ukazuje na moguće pravce rada i istraživanja u oblasti modelovanja interoperabilnosti sistema elektronskog poslovanja, zasnovanog na servisno orijentisanom razvoju softvera.

2 Servisno orijentisani sistemi elektronskog poslovanja

Poglavlje uvodi i objašnjava pojam elektronskog poslovanja i klasifikuje metode i tehnike modelovanja sistema elektronskog poslovanja, koristeći Open-EDI referentni model. U poglavlju su opisani različiti pristupi modelovanju poslovanja i modelovanju poslovnih procesa i navedeni su problemi, koje bi u budućnosti trebalo prevazići, a odnose se na povezivanje modela na višim i nižim nivoima. Takođe, u poglavlju su objašnjeni pojmovi "servis" i "servisna orijentacija". Zatim, opisani su servisno orijentisana arhitektura i veb servisi, kao najviše korišćen način za implementaciju servisno orijentisanih sistema.

2.1 Elektronsko poslovanje

Pojam elektronskog poslovanja prvi put je definisan 1996. godine, kroz marketinšku kampanju kompanije IBM. Elektronsko poslovanje označeno je kao bilo koja vrsta razmene poslovnih transakcija, u kojoj strane komuniciraju elektronskim putem, preko računarskih mreža, umesto razmenom klasičnih dokumenata ili direktnim fizičkim kontaktima. Elektronsko poslovanje rezultat je potrebe za promenama i primene novih tehnologija, pri čemu se suštinski menja način na koji se vodi poslovanje [3].

Verovatno najpotpuniji opis pojma elektronskog poslovanja dat je u radu [4], po kojem se elektronsko poslovanje odnosi na širu definiciju elektronske trgovine, ne samo na kupovinu i prodaju roba i usluga, već i na servisiranje kupaca, saradnju sa poslovnim partnerima i obavljanje elektronskih transakcija unutar preduzeća.

U najširem smislu reči, bilo koji oblik savremenog poslovanja, koji uključuje informaciono-komunikacione tehnologije za podršku poslovnih aktivnosti, može se nazvati elektronskim poslovanjem.

Različite inicijative (re-inženjering, outsourcing, upravljanje lancem vrednosti, virtuelne organizacije, kao i samo unapređenje elektronskog poslovanja) doprinele su da se originalni koncept elektronskog poslovanja, nastao krajem 1990-ih, podigne na viši nivo: elektronsko upravljanje poslovnim procesima, koje se usredsređuje na korišćenje informaciono-komunikacionih tehnologija za vođenje i unapređenje poslovnih procesa [5].

Kako bi se iskoristila prednost Interneta i novi načini kreiranja vrednosti, i odgovorilo na očekivanja klijenata u današnjoj ekonomiji, elektronsko poslovanje treba da uključi re-inženjering i redizajniranje poslovnih procesa i na nivou preduzeća i na nivou lanaca snabdevanja [6] [7].

Sistemi elektronskog poslovanja podržavaju definisanu poslovnu funkcionalnost, nizom poslovnih procesa, koji se izvršavaju po zadatom redosledu. Poslovni procesi se modeluju kao skup aktivnosti,

čije se izvršavanje definiše kontrolama toka. Da bi se procesi izvršili, sistemi elektronskog poslovanja uobičajeno se implementiraju preko posredničkog softvera, kao što su na primer, Distributed Component Object Model (DCOM), Common Object Request Broker (CORBA) i Java Remote Method Invocation (RMI), ili, u novije vreme, primenom sistema zasnovanih na servisno orijentisanoj arhitekturi.

U praksi se pokazalo da korišćenje posredničkog softvera, za implementaciju sistema elektronskog poslovanja, stvara probleme. Na primer, kod DCOM i RMI uočeni su problemi interoperabilnosti, vezani za operativne sisteme i korišćene programske jezike, pa je prednost data CORBA tehnologiji. Međutim, određeni problemi uočeni su i kod CORBA tehnologije, na primer [8]:

- implementacija CORBA tehnologije je skupa;
- nedovoljan broj iskusnih CORBA programera;
- dugo vreme razvoja i visok procenat napravljenih grešaka;
- nekriptovani saobraćaj, posebno preko Interneta, pri korišćenju COBRA tehnologije ugrožava bezbednost;
- verzionisanje nije kompatibilno unazad.

Navedeni problemi primene posredničkog softvera usmerili su pažnju na servisno orijentisanu arhitekturu i doprineli njenom ubrzanom usvajanju.

Poslednjih nekoliko godina trend je da se svaka aktivnost u okviru poslovnog procesa poveže sa servisom, koji će preuzeti na sebe izvršenje konkretnog zadatka. Sistemi elektronskog poslovanja, zasnovani na servisima, usmeravaju se na: korišćenje postojećih aplikacija i sistema; primenu efikasne interoperabilnosti; integraciju aplikacija i kompoziciju poslovnih procesa. Oslanjanjem na servise, stvara se preduslov da se verovatne, očekivane ili neočekivane, buduće promene karakteristika procesa, mogu podržati na relativno jednostavan i lak način.

Uvođenje sistema elektronskog poslovanja predstavlja značajnu promenu u poslovanju svakog preduzeća. Očekuje se da novi sistem zadrži funkcionalnosti i prednosti tradicionalnog poslovanja i da, istovremeno, preduzeću obezbedi nove vrednosti. Preduslov za uspešnu realizaciju je kvalitetan model budućeg sistema. Jedan od prvih koraka, ka izradi dobrog poslovnog modela, je izbor odgovarajućih metoda i tehnika za modelovanje. U narednom poglavlju predstavljen je određeni broj metoda i tehnika, koji se češće koriste.

2.2 Metode i tehnike modelovanja sistema elektronskog poslovanja

Uspešna realizacija softverskih rešenja za podršku sistemima elektronskog poslovanja uslovljena je kvalitetom modela sistema. Cilj modelovanja je da se, na različitim nivoima realizacije sistema obezbede odgovarajući modeli, koji će prikazati poslovanje iz različitih perspektiva. Fleksibilan dizajn i implementacija sistema elektronskog poslovanja zahteva definisanje modela poslovanja, na osnovu modela poslovanja definisanje poslovnih procesa i na kraju, realizaciju poslovnih procesa primenom informacionih tehnologija.

Model predstavlja prezentaciju ili apstrakciju realnosti. Model je najčešće pojednostavljen prikaz realnosti, s obzirom da je realnost suviše kompleksna da bi se tačno opisala. Pri rešavanju određenog problema, veći deo kompleksnosti u suštini je irelevantan. Pri izradi modela, stepen

pojednostavljenja i apstrakcije zavisiće od ugla iz koga se realnost posmatra i potrebnog nivoa detaljnosti.

Prema stepenu apstrakcije, modeli se mogu klasifikovati u tri kategorije [9]: jonski (na pr. prototipovi); analogni (grafikoni, dijagrami) i matematički (na pr. algoritmi). U ovoj tezi dat je fokus na analogne modele, koji predstavljaju simboličku prezentaciju realnosti. Često se umesto termina "analogni", koristi termin "konceptualni" model. Za konceptualno modelovanje koriste se:

- Jezici za modelovanje
  Sadrže sintaksu za predstavljanje koncepata i relacija između njih.
- Procedure za modelovanje
  Definišu izvore na osnovu kojih je model izveden. Na primer: tekstualni dokumenti; intervjui; prepoznavanje ključnih reči; šabloni.
- Referentni modeli
  Predstavljaju predefinisane konceptualne modele, koji sadrže znanja i koji omogućavaju timu za modelovanje sistema, da prepozna koncepte sistema koji modeluje. Na primer: šabloni ili strukture, kao što su petlje i grananja; primeri najboljih praksi za određenu oblast.
- Definicije povezanih tehnika za modelovanje
  Obezbeđuju modelovanje iz različitih perspektiva. Na primer, Unified Modeling Language (UML).

Metod konceptualnog modelovanja može biti posmatran kao sistem povezanih zadataka, koje je potrebno obaviti, da bi se kreirao konceptualni model određenog sistema, koristeći pri tome tehnike konceptualnog modelovanja.

Pojam "preduzeće", koji se koristi u radu, po definiciji iz Vikipedije [10] uključuje:

- organizacije iz privatnog i javnog sektora;
- kompanije;
- deo veće kompanije, kao što je poslovna jedinica;
- sistem nekoliko organizacija.

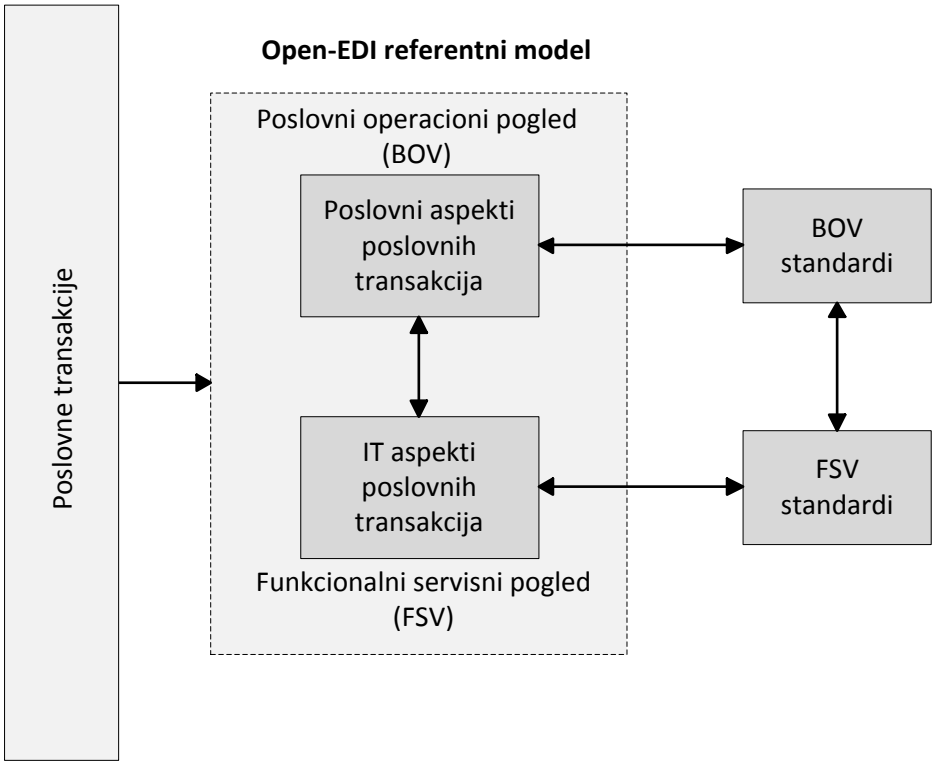
Takođe, pojam preduzeća uključuje i ceo kompleksni, socio-tehnološki sistem: ljude; informacije; tehnologiju; posao, na primer operacije.

Uloga modelovanja u elektronskom poslovanju je da obezbedi jasno definisan okvir za podršku prezentaciji, dizajniranju, implementaciji i unapređenju poslovnih procesa. Bez obzira na činjenicu da je veliki broj preduzeća usvojio elektronsko poslovanje, kao jedan od savremenih načina za obavljanje poslovanja, projekti elektronskog poslovanja ne donose uvek očekivana poboljšanja. Iskustvo je pokazalo da uspešno uvođenje elektronskog poslovanja zahteva usredsređivanje i na poslovne procese i na sveobuhvatan i sistematičan pristup transformaciji elektronskog poslovanja [5]. Preduslov je da analitičari i menadžeri razumeju buduće, verovatno, ponašanje elektronskih poslovnih procesa. Proveru ponašanja potrebno je uraditi na modelu, a pre bilo kakve implementacije.

Svrha modela elektronskog poslovanja je da prikaže njegovo funkcionisanje iz ekonomske perspektive [11]. Modeli elektronskog poslovanja važni su, s obzirom da obezbeđuju okvir za

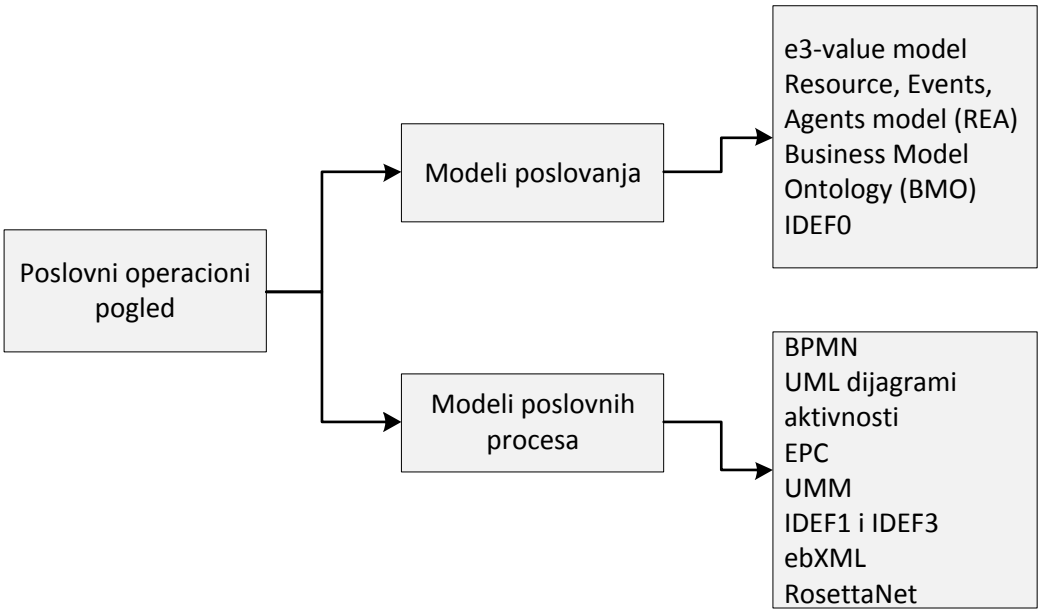
poslovne zahteve, koje moraju da zadovolje IT sistemima i mrežama. Uobičajeno se predstavljaju kombinacijom neformalnog teksta, verbalno i grafičkim prezentacijama. Ovakav način predstavljanja poslovnih zahteva ne pomaže uvek da, često suprotstavljene, zainteresovane strane u potpunosti razumeju sve aspekte elektronskog poslovanja.

Tokom godina razvijeni su brojni načini modelovanja elektronskog poslovanja. U ovoj tezi koristi se Open-EDI referentni model ISO organizacije [12] za klasifikaciju standarda pri elektronskoj obradi poslovnih transakcija između više preduzeća, u okviru i između različitih sektora. Primeri mogućih, obuhvaćenih sektora su: javni, privatni, industrijski, geografski sektor, itd. Open-EDI referentni model, prikazan na slici 2.1, koristi dva pogleda da opiše relevantne aspekte poslovnih transakcija: poslovni operacioni pogled i funkcionalni servisni pogled. Poslovni operacioni pogled obezbeđuje definisanje modela poslovanja i modela poslovnog procesa. Funkcionalni servisni pogled prikazuje poslovne transakcije iz ugla informacionih tehnologija.



Slika 2.1: Open-EDI referentni model

Na slici 2.2 navedene su tehnike i metode, koje se najčešće koriste za opis poslovnog operacionog pogleda [12]. Navedene tehnike i metode, na apstraktnom nivou, opisuju razmenu vrednosti između poslovnih partnera, kao i način kreiranja vrednosti, koje se razmenjuju.



Slika 2.2: Pregled tehnika i metoda za poslovni operacioni pogled
prilagođeno po [12]

Navedeni pogledi biće osnova za pregled tehnika i metoda, koje se koriste ili ih je moguće koristiti, u sistemima elektronskog poslovanja.

2.2.1 Modelovanje poslovanja

Model poslovanja preduzeća predstavlja pojednostavljenu prezentaciju poslovne logike i opisuje: šta preduzeće nudi svojim klijentima, kako ih pronalazi i kakvi su odnosi sa njima; preko kojih resursa, aktivnosti i partnera to postiže, i na koji način ostvaruje prihode. Model poslovanja pokazuje, sa stanovišta učesnika u poslovanju, osnovni način obavljanja poslovanja preko kreiranja i razmene vrednosti. Ukratko, koncept modelovanja elektronskog poslovanja koncentriše se oko pojma vrednosti.

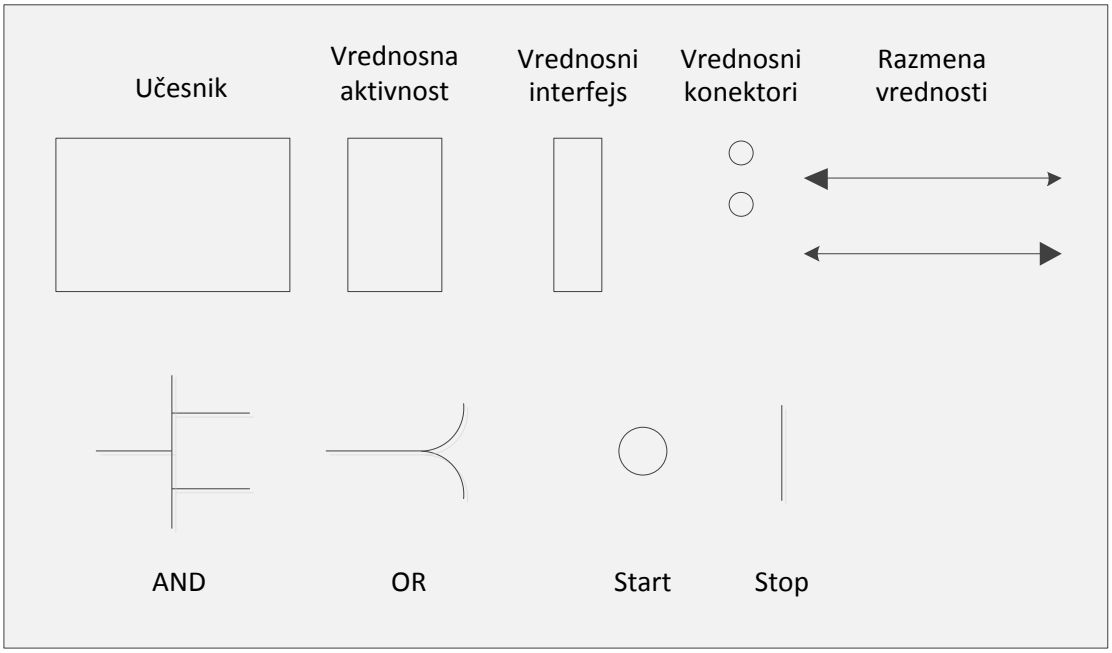
Dva aktuelna pristupa za modelovanje poslovanja su poslovna ontologija i modelovanje vrednosti. Poslovna ontologija obezbeđuje apstraktni opis preduzeća u njegovom poslovnom kontekstu, fokusirajući se na ono što je potrebno za kreiranje i prenošenje vrednosti. Modelovanje vrednosti predstavlja pristup, koji razmatra vrednost objekata koji se razmenjuju u poslovnim mrežama. Istraživanja poslovne ontologije i modelovanja vrednosti sprovode se korišćenjem instrumenata, kao što su: e^3-value, REA, BMO i IDEF0.

2.2.1.1 e^3-value model

e^3-value metodologija za modelovanje i dizajniranje poslovnih modela elektronskog poslovanja zasnovana je na ontologiji.

Ključni aspekt modela je oslanjanje na vrednosti, s obzirom da definiše kreiranje ekonomske vrednosti u okviru mreže učesnika. e^3-value model obezbeđuje: bolju komunikaciju i donošenje odluka o osnovama modela elektronskog poslovanja među zainteresovanim učesnicima, kao i potpunije razumevanje operacija i zahteva elektronskog poslovanja, a kroz analizu i kvantifikaciju

[11]. e^3-value pristup sadrži ograničeni skup međusobno povezanih osnovnih koncepata (ontologija). Ontologije i pridruženi grafički elementi koriste se za definiciju i analizu scenarija elektronskog poslovanja. e^3-value pristup ne uzima u obzir vreme i ne prikazuje redosled transfera vrednosti u okviru modela.



Slika 2.3: e^3-value koncepti

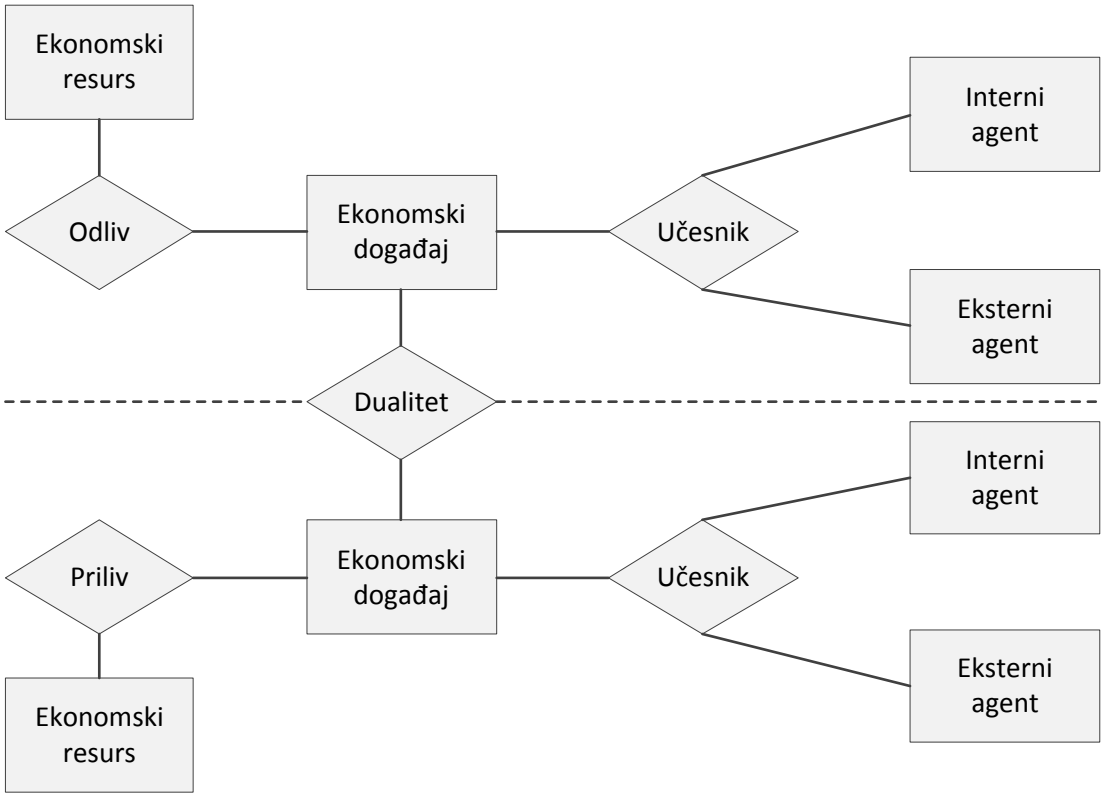
Grafički elementi osnovnog koncepta e^3-value modela, prikazani na slici 2.3, su i opisuju [13]:

- učesnik (*actor*) - predstavlja ekonomski nezavistan entitet, koji je uglavnom, ali ne i obavezno, pravno lice (primer: preduzeće, krajnji kupci);
- vrednost objekta (*object value*) - je nešto što obezbeđuje ekonomsku vrednost najmanje jednom učesniku (kola, internet pristup, knjiga);
- vrednosni konektor (*value port*) - koristi učesnik kako bi obezbedio vrednosne objekte drugim učesnicima, ili preuzeo vrednosne objekte od drugih učesnika (primanje robe, obavljanje plaćanja);
- vrednosni interfejs (*value interface*) - sastoji se od ulaznih i izlaznih konektora, koji pripadaju istom učesniku;
- razmena vrednosti (*value exchange*) - predstavlja par vrednosnih konektora, suprotnih smerova, koji pripadaju različitim učesnicima;
- vrednosna aktivnost (*value activity*) - je operacija koja bi mogla da donese profit najmanje jednom učesniku.

2.2.1.2 Resource, Events, Agents model (REA)

REA model, za dizajniranje poslovnih aplikacija, koristi koncepte: resurs (proizvodi, usluge ili novac); agent (ljudi ili druge kompanije) i događaj (poslovne transakcije ili dogovori, koji utiču na resurse). Iako je REA model prvobitno predstavljen kao teoretski model u oblasti računovodstva [14], može se primeniti i na druge poslovne domene.

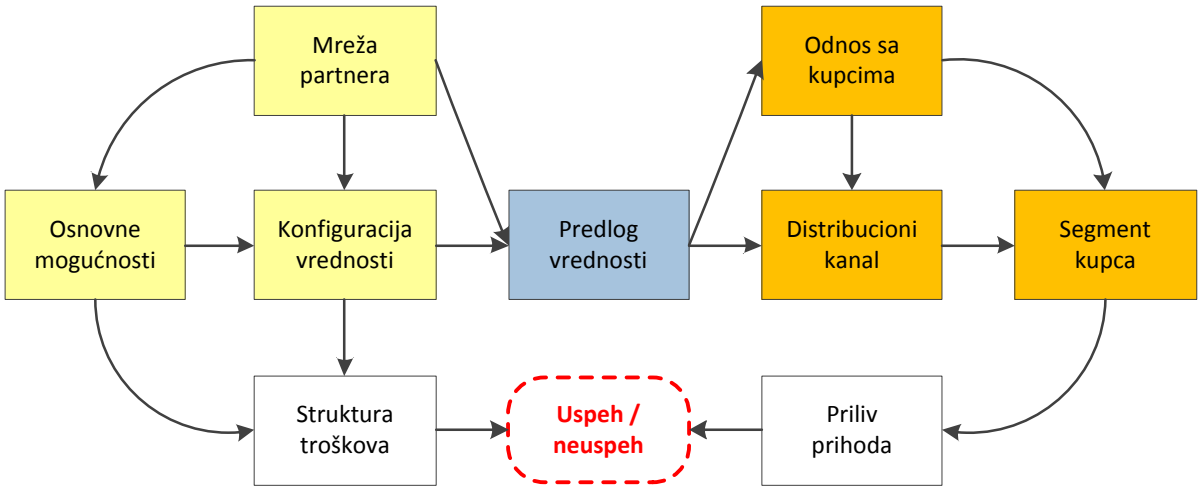
REA sistemi uobičajeno se modeluju dijagramima entiteti-relacije, kao relacione baze podataka, sa ciljem da prikažu osnovne koncepte i njihov međusobni odnos. Koristeći REA model, korisnik može da sagleda skup podataka, potreban za obavljanje određenog posla.



Slika 2.4: REA šablon
(prilagođeno po [14])

2.2.1.3 Business Model Ontology (BMO)

Business Model Ontology (BMO) razvijena je da omogući precizan i detaljan opis modela poslovanja preduzeća. BMO opisuje modele poslovanja korišćenjem četiri elementa: proizvodi i usluge koje preduzeće nudi; infrastruktura i mreža partnera, neophodna da se kreira vrednost i održe dobri odnosi sa kupcima; odnosi sa kupcima, koje preduzeće kreira i održava sa kupcima; finansijski aspekt. BMO se fokusira na određenog učesnika, njegovu poziciju u poslovnoj mreži i način na koji učesnik može da ostvari profit [15]. Po ovom modelu, inovacija proizvoda uglavnom se odnosi na vrednost, koju učesnik donosi određenom ciljanom segmentu kupaca, dok se element odnosa sa kupcima odnosi na različite kanale distribucije za ispostavljanje kreirane vrednosti. Upravljanje elementom infrastrukture odnosi se na način kreiranja vrednosti resursa učesnika i njegove partnerske mreže. Element finansijskog aspekta je pod uticajem prethodna tri elementa i određuje učesnikov model prihoda i troškova.

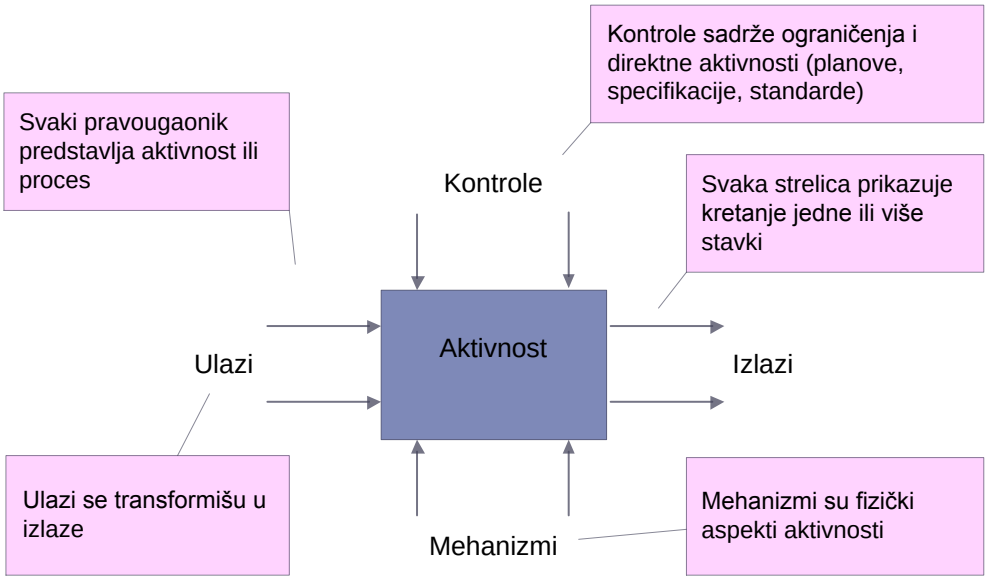


Slika 2.5: BMO elementi

2.2.1.4 Integration Definition for Function Modeling (IDEF0)

IDEF0 metod [16] omogućava modelovanje odluka, akcija i aktivnosti preduzeća, ili sistema. Kreirao ga je 1993. godine National Institute of Standards and Technology (NIST). Zasnovan je na strukturnoj analizi i tehnici modelovanja - Structured Analysis and Design Technique (SADT). IDEF0 model koristan je pri definisanju opsega analize, posebno u delu funkcionalne analize. Kao alat za analizu, IDEF0 pomaže da se odredi: šta funkcija obavlja, šta je potrebno za obavljanje tih funkcija, šta sadašnji sistem radi dobro, a šta loše.

IDEF0 model aktivnosti omogućava modelovanje zadataka, koje izvršava jedno preduzeće. Uključeni su ulazi, kontrole, izlazi i mehanizam svakog zadatka (Input, Control, Output, Mechanism - ICOM). Proces se sastoji od više ICOM. Hijerarhijska dekompozicija aktivnosti prikazuje šta preduzeće radi.



Slika 2.6: IDEF0 model aktivnosti

2.2.1.5 Modeli poslovanja - neki pristupi

Svrha kreiranja modela elektronskog poslovanja je da prikaže obavljanje posla iz ekonomske perspektive. Kreiranje modela poslovanje omogućava definisanje poslovnih zahteva, koji se realizuju pomoću IT sistema i mreža. Sa ciljem da se različitim zainteresovanim stranama obezbedi jasan prikaz, elektronsko poslovanje se konceptualno modeluje. Za izradu konceptualnog modela najčešće se koriste: e^3-value, BMO, REA i IDEF0 modeli.

U radu [17] porede se modeli poslovanja e^3-value i BMO. Navedene su zajedničke karakteristike i razlike, a sa ciljem njihove integracije u budućnosti. Stav autora je da se integracijom različitih ontologija mogu objediniti prednosti različitih modela. Osnovna razlika između dva modela je njihov fokus. e^3-value model usredsređuje se na mrežu poslovnih partnera i razmenu vrednosti. BMO odnosi se na određeno preduzeće i opis njegovih poslovnih interakcija, ukazujući na ponudu i kupce.

U radu [18] predložena je jedinstvena ontologija modela poslovanja, zasnovana na tri postojeća modela: e^3-value, BMO i REA. Polazeći od činjenice da postoje velike razlike između rezultata navedenih modela, tokom izrade jedinstvene ontologije, autori su pratili koncepte komunikacije i razmene vrednosti između poslovnih učesnika.

U radu [19] predloženi su načini za mapiranje e^3-value modela preko UML notacije, uz obrazloženje da su rezultujući UML dijagrami lako razumljivi i poslovnim ekspertima sa ograničenim znanjem UML. Autori polaze od konstatacije da se poslovni modeli najčešće opisuju nekom od sopstvenih notacija, koje su nekompatibilne sa UML. Sa ciljem da se obezbedi jedinstven proces modelovanja, počev od modela poslovanja, preko modela poslovnih procesa, pa do softverskih rešenja, poželjno je koristiti jedinstven pristup. Jedna od mogućnosti je korišćenje UML u svim fazama modelovanja. Mogućnost uniformnog modelovanja ilustruju primerima kreiranih e^3-value koncepata UML notacijom.

Korišćenje predloženih modela u praksi uslovljeno je postojanjem alata, koji omogućavaju primenu modela. Zahvaljujući postojećoj podršci alata, za modelovanje poslovanja pretežno se koristi IDEF0, mada se, u naučnim krugovima, više piše i raspravlja o drugim prikazanim modelima.

2.2.2 Modelovanje poslovnih procesa

Model poslovanja obezbeđuje pogled odozgo na aktivnosti, koje se odigravaju u elektronskom poslovanju. Model procesa, sa druge strane, fokusira se na operacione i proceduralne aspekte poslovne komunikacije, odnosno na način izvršavanja poslovne aktivnosti. Iako različiti, modeli poslovanja i modeli poslovnih procesa međusobno su povezani, jer ispravnost i kvalitet modela poslovnih procesa zavisi od njegove sposobnosti da podrži model poslovanja.

Osnovna razlika između modela poslovanja i modela poslovnog procesa proističe iz činjenice da se model poslovnog procesa zasniva na načinu obavljanja posla, a ne na izvršiocu. Model poslovnih procesa trebalo bi da opiše izvršavanje aktivnosti, koje kreiraju, dodaju i razmenjuju vrednosti između učesnika.

Činjenica je da još uvek ne postoji jedinstven pristup za neposredno povezivanje modela poslovanja sa modelima poslovnih procesa, odnosno, još uvek postoji diskontinuitet između različitih nivoa modelovanja sistema elektronskog poslovanja.

Tabela 2.1: Razlike između modela poslovanja i modela poslovnog procesa
(prilagođeno po [20])

Model poslovanja	**Model procesa**
Govori u kojoj meri učesnici dodaju vrednost i da li su spremni da razmene objekte vrednosti sa drugima.	Pokazuje koje aktivnosti treba izvršiti, po kom redosledu i koje objekte (i kojim redosledom) treba razmeniti.
Učesnik dodaje vrednost.	Učesnik izvršava aktivnost u okviru procesa.
Objekat predstavlja određenu vrednost učesniku.	Objekti preuzimaju ulogu ulaza i izlaza za aktivnosti i mogu se koristiti za upravljanje izvršavanjem procesa.
Učesnici mogu koristiti osobine objekta da bi odredili vrednost objekta.	Osobine objekta se koriste da odrede stanje u kojem se objekat nalazi.
Vrednost koja se razmenjuje predstavlja prenos vlasništva.	Tok informacija, ili dobara utiče na promenu stanja.
Postoji pojam "jedno dobro delo zaslužuje drugo", što se konceptualizuje pomoću vrednosti interfejsa.	Pojam "dobro delo" ne postoji u modelu procesa.
Bitne su jedino aktivnosti, koje omogućavaju dodavanje vrednosti. Dekompozicijom takvih aktivnosti često se otkrivaju manje aktivnosti, koje još uvek dodaju vrednost. Otkrivanje ovih aktivnosti može dovesti do preraspodele aktivnosti između učesnika.	Cilj dekompozicije je objašnjenje toka, ili prikaz dodele aktivnosti učesnicima.

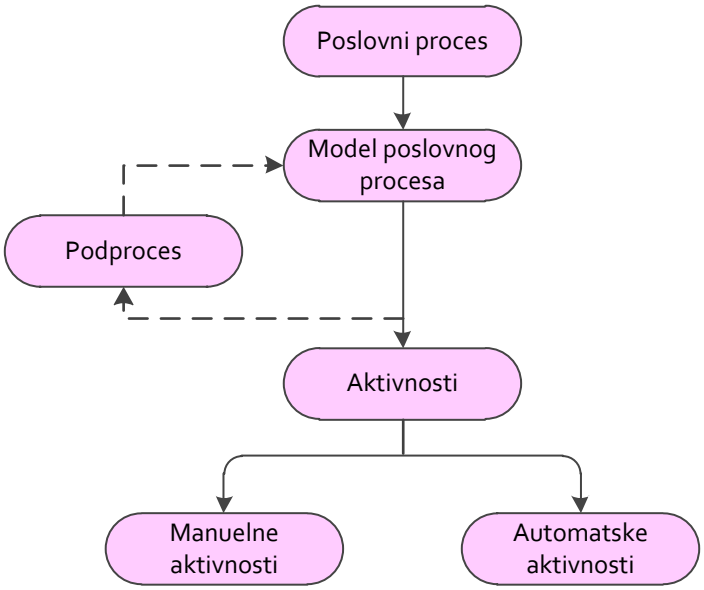
Osnovna definicija poslovnog procesa data je u radu [21]. Po definiciji, poslovni proces je skup logički povezanih zadataka, koji se izvršavaju da bi se ostvario poslovni rezultat. Skup poslova formira poslovni sistem, u kome poslovne jedinice, ili skup jedinica, obavljaju svoj posao. Procesi poseduju dve važne karakteristike: procesi imaju kupce, koji predstavljaju primaoce poslovnih rezultata, i procesi prelaze organizacione granice i nezavisni su od organizacione strukture.

Više autora proširilo je navedenu definiciju poslovnog procesa. U radu [22] smatra se da je osnovnu definiciju potrebno proširiti i dopuniti u slučaju kad je reč o procesima, zasnovanim na IT i u slučaju mrežnih procesa. Tako]e se navodi da bi definicija poslovnog procesa trebalo da ukazuje na početno i krajnje stanje procesa. Savremena definicija trebalo bi da obuhvati sposobnost za komunikaciju i interakciju sa drugim procesima i da, kao rezultat, iskaže vrednost.

Određeni broj autora pravi razliku između tradicionalnog poslovnog procesa i poslovnog procesa koji se izvršava korišćenjem Interneta, a koji nazivaju: e-proces. U radu [23] autori smatraju da je e-proces kolekcija poslovnih pravila, koja se mogu primeniti pri korišćenju Interneta i WWW, prilikom svakog ponovljenog zahteva (ulaz). U tim slučajevima, definisana kolekcija poslovnih pravila koordinira mrežu interakcija (procesi/dodatni ulazi) preko vrednosne mreže: kupac - preduzeće - dobavljači, kako bi se isporučila jedinstvena vrednost (izlaz) svim zainteresovanim učesnicima.

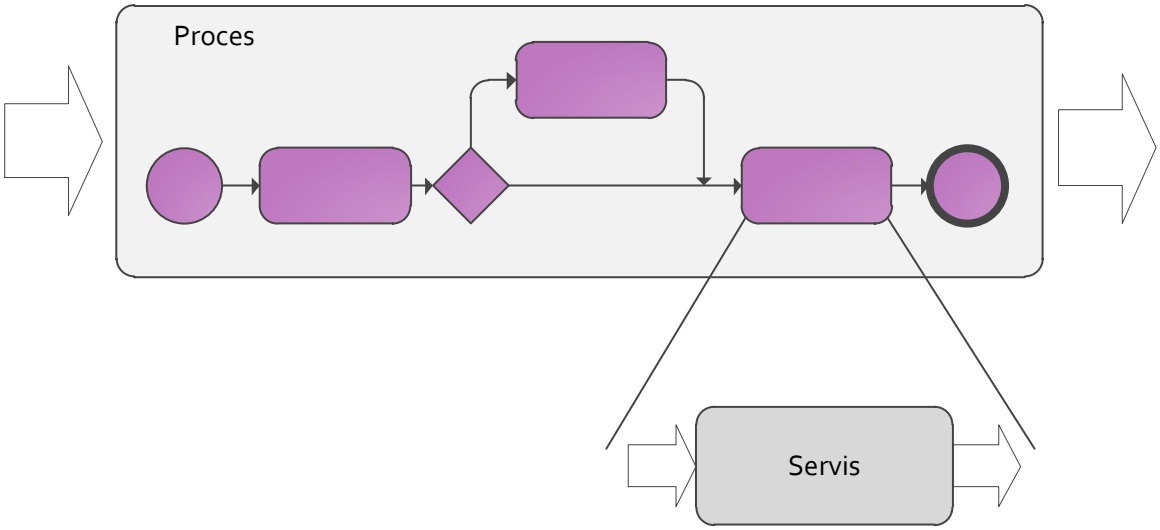
Struktura poslovnog procesa može biti veoma složena. Često je uključeno više učesnika (ljudi, preduzeća i sistemi), koji izvršavaju više aktivnosti. Da bi se poslovni proces obavio u celini, učesnici moraju koordinirano da izvršavaju definisane aktivnosti, koje su najčešće grupisane u podprocese. U nekim situacijama, podprocesi se izvršavaju paralelno, u drugim su sekvencijalni. Neki procesi

zahtevaju ponovno izvršavanje podprocesa. Većina procesa sadrži tačke odluke, koje dovode do grananja toka, u zavisnosti od postavljenih, zadovoljenih ili nezadovoljenih uslova. U okviru nekih procesa, učesnici moraju da prosleđuju određene informacije. Prenos informacija može imati ulogu okidača za otpočinjanje nove aktivnosti. Definisane aktivnosti mogu se izvršavati automatski ili manuelno.



Slika 2.7: Definicija poslovnog procesa

U ovoj tezi pojam "poslovni proces" označava proces sastavljen od više pojedinačnih aktivnosti, koje se izvršavaju određenim redosledom, korišćenjem servisa. Pojedinačne aktivnosti komuniciraju među sobom koristeći Internet i servise Interneta. Poslovni proces, koji koristi servise za izvršavanje poslovnih aktivnosti, naziva se servisno orijentisani proces i grafički je predstavljen na slici 2.8.



Slika 2.8: Relacija između procesa i servisa

Modelovanje poslovnih procesa uobičajeno ima širi opseg u odnosu na modelovanje pojedinačnog softverskog sistema. Cilj modelovanja je da poslovni analitičar jasno predstavi opseg sistema, koji se modeluje i koji će biti, na osnovu modela, implementiran.

U radu [24] razvijen je skup kriterijuma za izbor metodologije za modelovanje poslovnih procesa:

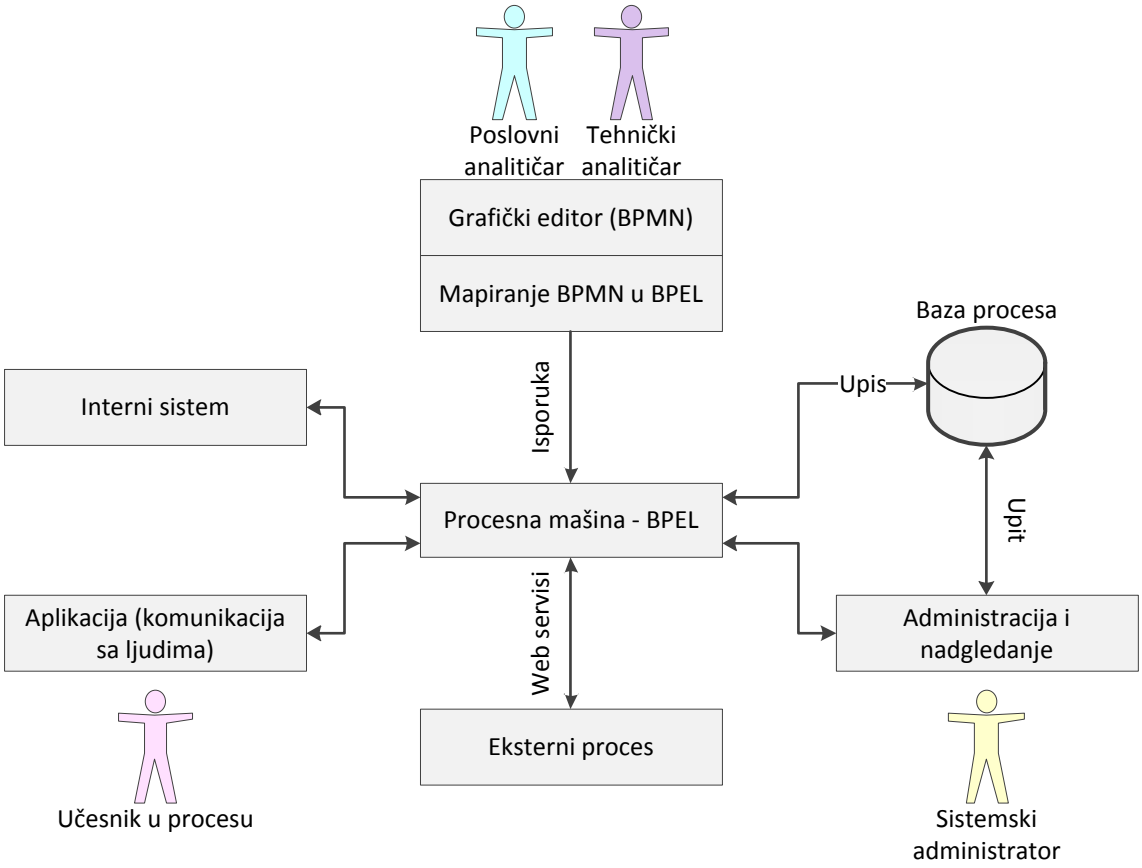
1. Sposobnost da modeluje sve složene situacije vezane za poslovni proces: sekvence; grananja; petlje; konkurentne elemente - paralelne grane i sinhronizaciju; kontrolu vremena izvršavanja; krajnji rok za izvršavanje; kontrolu grešaka i agregaciju.
2. Mogućnost za razlikovanje uloga, kao i mogućnost da se ulogama dodeljuju različiti zadaci.
3. Da postoji nedvosmislena, grafička prezentacija jezika.
4. Da postoji podrška za transakcioni model, koji opisuje stanja, u slučaju da se proces mora vratiti u prethodno dobro stanje.
5. Da je moguće definisanje načina na koji će instance procesa biti pokrenute i identifikovane tokom izvršavanja.
6. Mogućnost za specificiranje karakteristika poslovnog procesa, koje su važne za eksterne korisnike (što uključuje kvalitet usluga i cene).
7. Da jezik ne sadrži detalje komunikacionog protokola.

U radu [25] grupišu se pristupi modelovanju procesa u četiri kategorije:

- Pristupi zasnovani na aktivnostima
  Definišu poslovni proces kao određeni redosled aktivnosti. Ovaj pristup se pretežno koristi tokom modelovanja procesa.
- Objektno zasnovani pristupi
  Vezuju se za objektnu orijentaciju, kao što su enkapsulacija, nasleđivanje i specijalizacija.
- Pristupi zasnovani na ulogama
  Ukazuju na uloge, uključene u skup poslovnih aktivnosti, a po osnovu svojih odgovornosti.
- Pristupi zasnovani na govoru
  Posmatraju proces iz jezičke perspektive, kao komunikacioni proces, koji se izvršava kroz četvorofaznu petlju: predlog, dogovor, odigravanje i zadovoljstvo.

Modelovanjem poslovnih procesa bavi se posebna disciplina, na engleskom jeziku: Business Process Modeling (BPM). BPM posmatra poslovni proces kao tok aktivnosti, a u model uključuje i: osobe, interne sisteme, ili procese druge kompanije, čija je uloga ostvarenje poslovnog cilja. Tokom godina, područje BPM je prošireno: polazeći od izvornog predstavljanja "toka posla", došlo se do discipline Business Process Management. Business Process Management bavi se dizajniranjem, upravljanjem, izvršavanjem i nadgledanjem poslovnog procesa.

Business Process Management bavi se upravljanjem poslovnim procesima i predstavlja pristup, koji objedinjuje: metode, alate i servise za modelovanje, izvršavanje i optimizaciju poslovnih procesa u preduzeću, kao i procesa koji se izvršavaju kroz komunikaciju sa spoljnim preduzećima i poslovnim partnerima [26].



Slika 2.9: BPM arhitektura
(prilagođeno po [26])

Iz ugla prikazane BPM arhitekture, najbitniji deo sistema je procesna mašina, koja izvršava poslovni proces napisan u odgovarajućem jeziku. Najčešće se koristi BPEL. Poslovni i tehnički analitičari dizajniraju procese koristeći grafički editor, koji podržava grafičku notaciju - najčešće BPMN. Očekuje se da editor omogućava kreiranje BPEL koda na osnovu BPMN dijagrama.

Na izvršavanje procesa mogu uticati komunikacije sa ljudima i aplikacijama. Za povezivanje sa procesnom mašinom, ljudi, koji učestvuju u procesu, koriste aplikacije dijagrama toka. Dijagram toka omogućava pregled i izvršavanje aktivnosti, koje čekaju izvršenje. Postoje dva tipa interakcije sa aplikacijama: interna i eksterna.

Internim aplikacijama, koje se nalaze u okviru mreže preduzeća, ali van adresnog prostora procesne mašine, pristupa se pomoću neke od integracionih tehnologija, kao što su: veb servisi, J2EE, C#. Najčešći format poruka je XML.

Eksterne aplikacije su uobičajeno veb servisi, sastavni delovi poslovnih procesa drugih kompanija.

Za administriranje i nadgledanje procesa, koji se izvršavaju, BPM sistemski administratori koriste grafičke konzole. Konzole sa procesnim mašinama komuniciraju preko jezika za upravljanje. Procesne mašine čuvaju stanje procesa u bazi podataka. Da bi izvršili neki upit, sistemski administratori pristupaju bazi procesa.

Događa se da različiti učesnici isti proces opisuju drugačije. Zato je potrebno da se metodologijom za modelovanje poslovnih procesa mogu prezentovati različiti aspekti opisa procesa. Prihvatljiva metodologija trebalo bi da omogući predstavljanje procesa na način jednostavan za prebacivanje u oblik prepoznatljiv učesniku. Poslovnim analitičarima BPM služi da sagledaju preduzeće preko skupa procesa: koji se mogu definisati; kojima se može upravljati i koji se mogu optimizovati. Umesto tradicionalnog pristupa, po kome se poslovi analiziraju unutar organizacionih celina, BPM se usmerava isključivo na poslove, bez obzira u kojoj se organizacionoj jedinici izvršavaju. Za tehničko osoblje, BPM predstavlja grupu tehnologija fokusiranih na definisanje, izvršavanje i nadgledanje procesne logike. Bez obzira na različite poglede, i poslovni analitičari i tehničko osoblje, imaju za cilj unapređenje poslovnih procesa.

Modelovanje poslovnih procesa trebalo bi koristiti samo u slučaju procesno orijentisanih aplikacija. U tom slučaju, aplikacije bi trebalo da poseduju bar neke od sledećih karakteristika [26]:

- dugo se izvršavaju: više sati, dana, nedelja ili meseci;
- čuvaju stanje u bazama podataka;
- veći deo vremena čekaju na akciju, koja će pokrenuti narednu aktivnost;
- procesi aplikacije odgovorni su za upravljanje i koordinaciju komunikacija između različitih uloga sistema i ljudi.

Modelovanje poslovnih procesa obezbeđuje da se postojeći procesi u preduzeću formalizuju, bolje razumeju i po potrebi poboljšaju. Modelovani procesi mogu se efikasno izvršavati korišćenjem BPM softvera, što doprinosi povećanju produktivnosti i smanjenju učešća ljudi u izvršavanju procesa. Takođe, modelovanje omogućava ljudima da brže i tačnije reše komplikovane probleme, pri čemu je praćenje propisa i poslovnih pravila jednostavnije.

Sa ciljem da se smanji mogućnost pojave grešaka u modelu procesa i poveća razumljivost modela, prilikom modelovanja potrebno je poštovati sledeće preporuke [27]:

1. Koristiti što je moguće manje elemenata u modelu.
2. Minimizovati putanje do i od elemenata.
3. Koristiti samo jedan početni i samo jedan završni događaj.
4. Struktuirati model koliko god je to moguće, odnosno, obezbediti da svaki razdvajajući (*split)* konektor ima odgovarajući spajajući (*join)* konektor.
5. Izbegavati *OR* elemente.
6. Za označavanje aktivnosti koristiti glagole.
7. Dekomponovati model, ukoliko ima više od 50 elemenata.

Tokom vremena, razvijeno je više pristupa za modelovanje poslovnih procesa, koji uglavnom koriste jezike i notacije za grafičko modelovanje. Mogućnosti ovih jezika i notacija su različite. Kreću se od jednostavnih tehnika za crtanje dijagrama toka do naprednih jezika, koji mogu da predstave informacije potrebne za simulaciju i izvršavanje procesa. U radu će biti ukratko predstavljeni sledeće tehnike i jezici za modelovanje poslovnih procesa: BPMN, BPEL, UML dijagrami aktivnosti, EPC, UMM, IDEF1 i IDEF3, ebXML, RosettaNet.

2.2.2.1 Business Process Modeling Notation (BPMN)

Prva verzija BPMN specifikacije razvijena je 2004. godine od strane Business Process Management Initiative (BPMI). Značajnu ulogu u razvoju imao je Vajt (White) [28] iz IBM kompanje. BPMI se kasnije pridružuje organizaciji Object Management Group (OMG), grupi poznatoj po standardizaciji Unified Modeling Language (UML). 2006. godine zajedno kreiraju finalnu verziju BPMN 1.0.

BPMN je razvijen sa ciljem da obezbedi grafičku notaciju za definisanje poslovnih procesa, lako razumljivu svim poslovnim korisnicima: od poslovnih analitičara, koji bi trebalo da skiciraju procese, do tehničkih korisnika, koji su odgovorni za implementaciju tehnologije pomoću koje se izvršavaju procesi, i na kraju do poslovnih izvršilaca, koji će upravljati i nadgledati poslovne procese. BPMN omogućava da se premosti praznina između dizajniranja poslovnog procesa i njegove implementacije [29]. BPMN je napravljen sa idejom da bude intuitivna notacija, koja se lako usvaja i koristi.

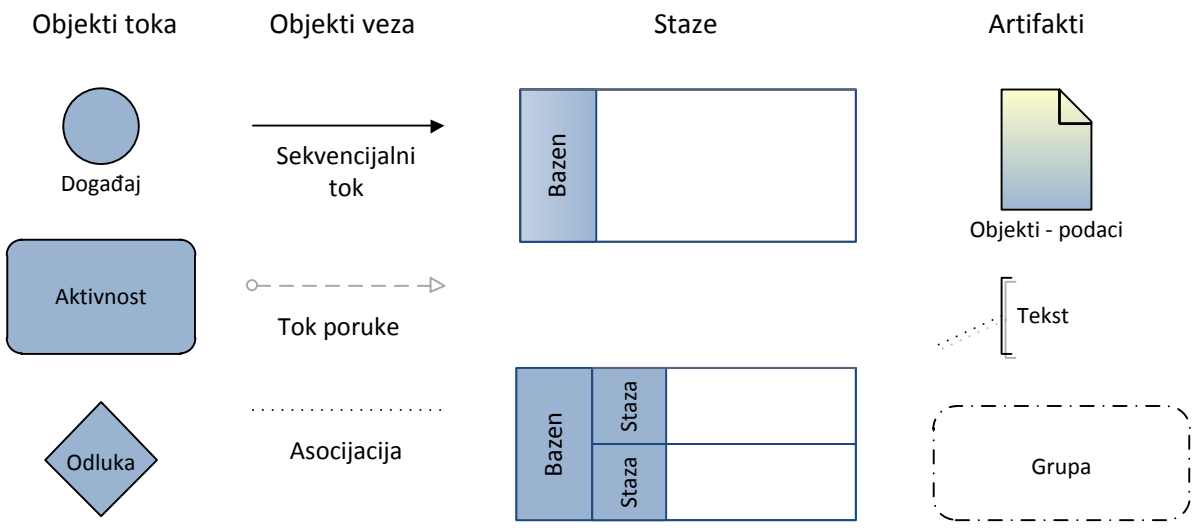
Trenutno postoji nekoliko desetina BPMN implementacija, koje omogućavaju kreiranje poslovnih procesa korišćenjem BMPN specifikacije [29], što pokazuje da je BPMN postao vodeći grafički standard za modelovanje procesa.

BPMN se definiše preko Business Process dijagrama (BPD), koji je baziran na tehnici blok-dijagrama i koji je prilagođen za kreiranje grafičkih modela aktivnosti poslovnih procesa. Business Process Model (BPM) predstavlja mrežu grafičkih objekata, odnosno aktivnosti i kontrola, kojima se definiše njihov redosled izvršavanja.

BPMN dijagram sastavljen je od skupa grafičkih elemenata. Elementi omogućavaju lak razvoj jednostavnih dijagrama i bliski su većini poslovnih analitičara. Na primer (slika 2.10), aktivnosti su predstavljene kao pravougaonici, a odluke kao rombovi.

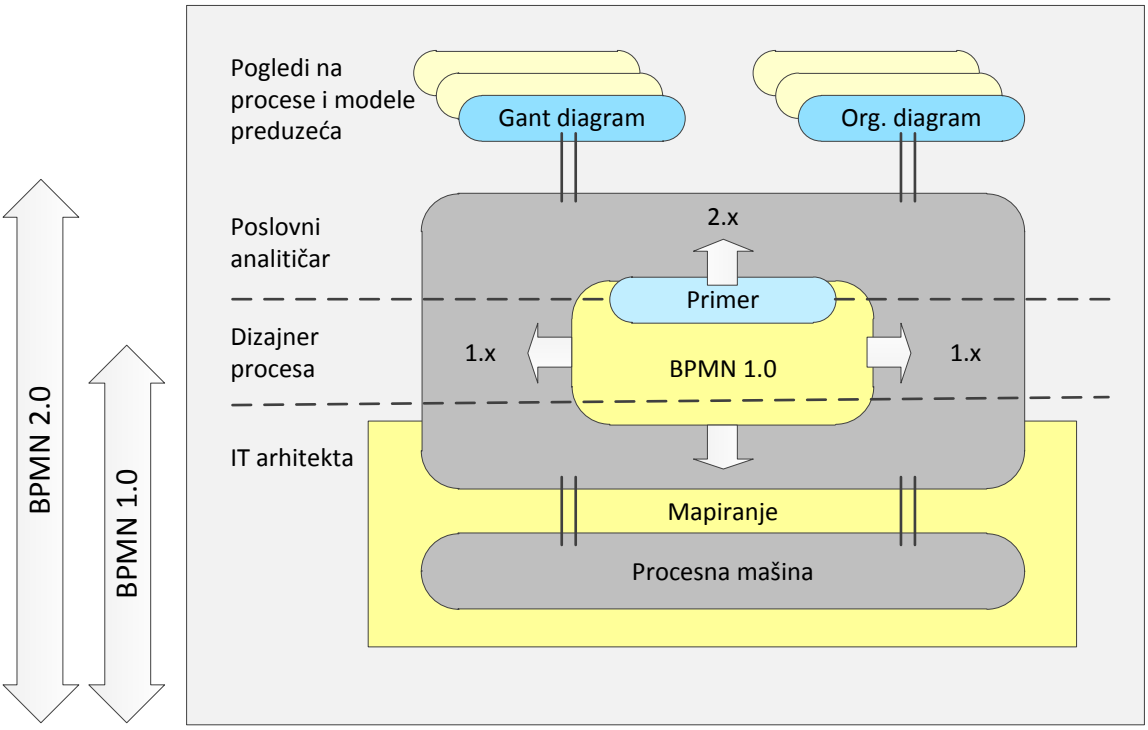
BPMN specifikacija grupiše elemente u četiri osnovne kategorije:

- objekti toka (osnovni grafički elementi za definisanje ponašanja poslovnih procesa: Event, Activity i Gateway);
- objekti veza (omogućavaju povezivanje objekata toka preko: prikaza redosleda izvršavanja aktivnosti, prikaza toka poruka i prikaza ulaza i izlaza iz aktivnosti);
- staze (omogućavaju razdvajanje grupa aktivnosti, sa ciljem njihove organizacije i kategorizacije);
- artifakti (obezbeđuju mogućnost proširenja osnovne notacije dodavanjem konteksta, koji odgovara situaciji koja se modeluje).



Slika 2.10: Osnovni skup BPMN elemenata

BPMI organizacija, odgovorna za kreiranje BPMN i drugih standarda vezanih za modelovanje poslovnih procesa, smatra da je BPMN 1.0 samo prvi korak u razvoju ove notacije. Početkom 2011. godine izašla je nova verzija BPMN, verzija 2.0. Nova verzija dopunjena je dodatnim definicijama, vezanim za meta model i formate za razmenu poruka, i predstavlja odgovor na kritike prethodne verzije BPMN, po kojima BPMN ne sadrži formalnu specifikaciju meta modela [30]. Funkcionalni obim i odnos BPMN 1.0 i BPMN 2.0 prikazan je na slici 2.11.



Slika 2.11: Funkcionalni obim i odnos BPMN 1.0 i BPMN 2.0

2.2.2.2 Business Process Execution Language (BPEL)

BPEL je jezik za izvršavanje poslovnih procesa i zasniva se na XML. BPEL su kreirali IBM, Microsoft i BEA. Sada je u nadležnosti neprofitnog konzorcijuma OASIS (Organization for the Advancement of Structured Information Standards), koji razvija, održava i promoviše standarde za elektronsko poslovanje, uključujući ebXML, SGML, UDDI, PKI i BPEL.

BPEL definiše model i gramatiku za opis ponašanja poslovnog procesa, zasnovanog na međusobnom uticaju procesa i njegovih partnera. Interakcija sa svakim partnerom izvršava se korišćenjem interfejsa veb servisa. Struktura relacija, na nivou interfejsa, prikazuje se elementom *partner link*. BPEL proces definiše i opisuje koordinaciju servisa sa partnerima, radi postizanja poslovnog cilja, kao i stanje i logiku, neophodnu za koordinaciju. Takođe, BPEL uspostavlja sistematski mehanizam za ponašanje u slučaju dešavanja poslovnih izuzetaka i obradu greške. BPEL predstavlja mehanizam za definisanje načina na koji, pojedinačne ili kompozitne aktivnosti u okviru procesa, mogu biti kompenzovane, kada se pojave izuzeci, ili promene zahtevi partnera. [31]

BPEL poslovni proces reprezentuje dugotrajnu interakciju, koja: nadgleda stanje; ima početak, trajanje i kraj. Kreiranje instance BPEL procesa je uvek implicitno. Aktivnosti, koje primaju poruke (<receive> i <pick>), mogu biti označene i kao inicijatori novog poslovnog procesa.

Da bi bio instanciran, poslovni proces mora da ima barem jednu startnu aktivnost. To mora biti inicijalna aktivnost, pri čemu ne sme da postoji osnovna aktivnost, koja logički prethodi.

Poslovni proces može da čuva stanje i najčešće se dugo izvršava. Pri svakom pozivu poslovnog procesa, kreira se njegova instanca. Iz toga proizilazi da poslovni proces ima ulogu šablona za kreiranje poslovnih instanci. Sličan odnos uspostavlja se u slučaju veze klasa-objekat, u kojoj klasa predstavlja šablon za kreiranje objekta u vreme izvršavanja.

Poslovni proces se završava na jedan od sledećih načina:

- Kada se aktivnost, koja definiše ponašanje procesa, u potpunosti izvrši. U ovom slučaju, kraj poslovnog procesa završava se normalno.
- Kada se desi greška tokom izvršavanja procesa, koja može biti obrađena ili neobrađena. U ovom slučaju, kraj poslovnog procesa završava se nenormalno.
- Kada se proces eksplicitno prekine, korišćenjem *terminate* aktivnosti. U ovom slučaju, kraj poslovnog procesa završava se nenormalno.
- Ako je *compensation handler* definisan za ceo poslovni proces, instanca poslovnog procesa može biti kompenzovana posle normalnog završetka, što se omogućava postavljanjem *enableInstanceCompensation* atributa na "yes".

Najvažniji elementi BPEL jezika odnose se na pozivanje veb servisa. BPEL omogućava da se poziv veb servisa obavi na jednostavan način, kako sinhrono, tako i asinhrono. Operacije se mogu pozivati i sekvencijalno i paralelno. Takođe, BPEL obezbeđuje: opis povratnih funkcija, upravljanje greškama, podršku procesima koji se dugo izvršavaju, kao i kompenzaciju, kojom se stornira posao koji je odradio proces, a koji se nije završio uspešno. BPEL jezik omogućava [32]:

- opisivanje logike poslovnog procesa kroz kompoziciju servisa;
- sastavljanje većih poslovnih procesa od manjih procesa i servisa;

- upravljanje sinhronim i asinhronim pozivanjem operacija servisa, kao i upravljanje povratnim funkcijama, koje se izvršavaju kasnije;
- pozivanje operacija servisa sekvencijalno ili paralelno;
- selektivno kompenzovanje neuspešno završene aktivnosti;
- održavanje višestruke, dugotrajne transakcione aktivnosti;
- nastavak prekinutih, ili neuspešno završenih aktivnosti, što minimizuje potrebu za njihovim ponovnim izvršavanjem;
- usmeravanje dolazećih poruka u odgovarajuće procese i aktivnosti;
- uzajamno povezivanje zahteva u okviru, ili van poslovnog procesa;
- planiranje izvršavanja aktivnosti, zasnovano na vremenu izvršavanja i definisanje njihovog redosleda izvršavanja;
- izvršavanje aktivnosti paralelno i definisanje sinhronizacije paralelnih tokova u odnosu na uslove sinhronizacije;
- struktuiranje poslovnih procesa u više oblasti;
- upravljanje događajima, zasnovanim na porukama i vremenu.

Neka od ograničenja BPEL jezika navode se u članku [33]:

- BPEL nema standardno rešenje za pristup softveru koji nije veb servis;
- BPEL jezik ne definiše standardni način za pristup relacionim podacima, ili drugim informacijama, koje nisu XML;
- BPEL ne podržava pristup procesa lokalnim objektima;
- BPEL ne definiše mehanizme za interakciju sa ljudima.

2.2.2.3 Mapiranje BPMN u BPEL

BPMN i BPEL su de facto postali standardi za opis i izvršavanje poslovnih procesa, čemu je doprinelo i postojanje brojnih alata za njihovo korišćenje. U narednom delu biće navedene mogućnosti mapiranja BPMN dijagrama u BPEL izvršni jezik.

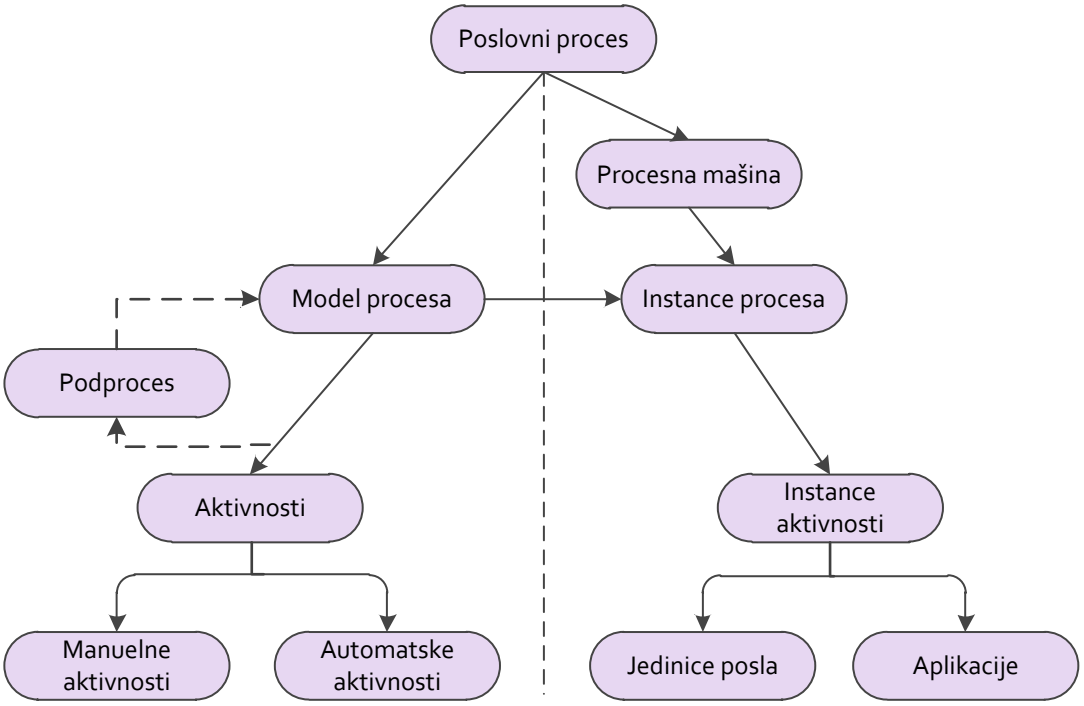
Svrha kreiranja BPMN dijagrama je dvostruka: trebalo bi da omogući poslovni pogled na poslovni proces i da omogući generisanje izvršnog koda pomoću BPEL jezika.

Bez obzira na naziv - Business Process Execution Language, BPEL se realno ne bavi poslovnim procesima, bar ne na način kako se procesima bave BPM stručnjaci. Na primer:

- Sa BPM stanovišta, poslovni proces je sastavljen od različitih podprocesa, koji mogu biti ugnježdeni ili ulančani međusobno.
- Svaki podproces može biti vlasništvo, ili ga mogu izvršiti različite poslovne jedinice.
- BPEL ne poznaje koncept podprocesa, koji je osnovna jedinica za ponovno korišćenje u BPM. To ne znači da se korišćenjem BPEL ne može kreirati proces, sastavljen od ugnježdenih i ulančanih komponenti.
- BPEL ne podržava koncepte ljudi, učesnika identifikovanih preko uloga, grupa, ili korisničkih imena.
- U BPEL, aktivnosti procesa se dodeljuju završnim tačkama (endpoint) servisa, koje se razrešavaju preko URL. Ne postoji odvojena završna tačka servisa za svaku ulogu, grupu, ili

imenovanog korisnika. Umesto toga, postoji jedna završna tačka, koja rutira zadatak prema odgovarajućem korisniku, ili prema redu čekanja. [34]

BPMN specifikacija sadrži više od 50 strana, koje prikazuju mapiranje BPMN u BPEL jezik. Cilj je da se uspostavi veza između grafičkog dizajna procesa i izvršavanja procesa. Sam proces mapiranja nije jednoznačan, pre svega zbog postojanja različitog broja elemenata. Osim toga, s obzirom da je mapiranje samo opisano u BPMN specifikaciji, ono predstavlja predmet interpretacija. Uopšteno govoreći, u BPMN specifikaciji za mapiranje u BPEL jeziku postoji mnogo dvosmislenosti [35].



Slika 2.12: Relacija između modelovanja i izvršavanja poslovnog procesa

Problem nemogućnosti jednoznačnog mapiranja BPMN notacije u BPEL izvršni jezik, proizvođači softvera rešavaju na neki od sledećih načina:

- Grafički predstavljaju BPEL proces i nazivaju ga BPMN.
- Koriste jedan alat za kreiranje BPMN dijagrama, a drugi za formiranje BPEL procesa.
- Na osnovu kreiranog BPMN dijagrama kreira se BPEL izvršni kod, prilagođen određenoj procesnoj mašini, pri čemu se uobičajeno zahteva da BPMN ne sadrži elemente, koji se ne mogu direktno mapirati.

U ovom momentu, da bi se pomoću bilo kod dostupnog alata generisao BPEL od BPMN, potrebno je da se reše dva problema:

1. Pre izvoza BPMN u BPEL potrebno je dodati sve detalje vezane za implementaciju, što često dovodi do očekivanih pitanja: "Da li se očekuje da će poslovni analitičar postati programer?, i obrnuto: "Da li se očekuje da će programer postati poslovni analitičar?";
2. Zbog loše struktuiranog BPMN grafikona, izvoz ponekad uopšte nije moguć, ili se dobija nevažeći BPEL kod.

Problemi, koji se javljaju pri generisanju BPEL koda na osnovu BPMN dijagrama, karakteristični su kod svih modela na visokom nivou. Modele na visokom nivou potrebno je, na odgovarajući način, sinhronizovati sa modelima na nižem nivou. To svakako nije jednostavno, osim u retkim situacijama, kada je moguće obaviti mapiranje jedan-na-jedan.

2.2.2.4 Drugi načini za modelovanje poslovnih procesa

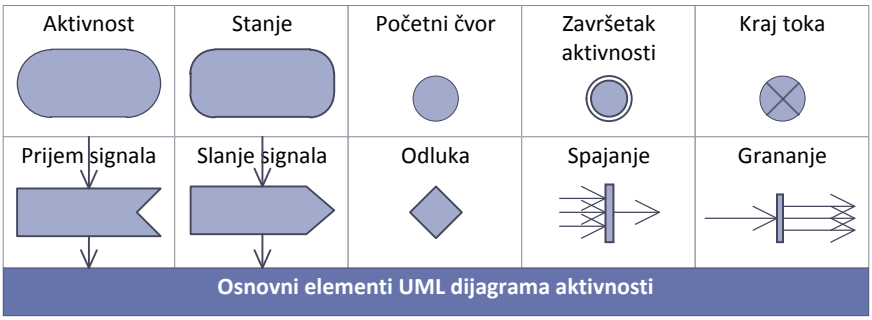
Pored opisanog BPMN, koriste se i drugi načini za modelovanje poslovnih procesa. Najčešće su zastupljeni: UML dijagrami aktivnosti, EPC, UMM, IDEF1 i IDEF3, ebXML i RosettaNet .

UML dijagrami aktivnosti

Unified Modeling Language (UML) predstavlja jezik za specifikaciju, vizuelizaciju, konstrukciju i dokumentovanje proizvoda softverskog sistema, kao i za modelovanje poslovnih procesa i drugih nesoftverskih sistema. UML predstavlja skup najboljih, praktično primenjenih, inženjerskih modela, koji se koriste u razvoju velikih i složenih sistema. UML je postao standardni jezik za modelovanje, između ostalog, i zbog činjenice da je nezavistan od programskih jezika.

UML podržava bogat skup grafičkih elemenata. Raspoloživi grafički elementi omogućavaju da se na razumljiv način opišu: klase i komponente; čvorovi; aktivnosti i dijagrami toka; slučajevi korišćenja, objekti i stanja, kao i da se modeluju relacije između navedenih elemenata. UML podržava i mogućnost proširenja preko stereotipnih elemenata.

UML dijagrami aktivnosti (UML AD) najčešće se koriste za grafički prikaz modela poslovnih procesa. UML AD obezbeđuju prikaz toka aktivnosti uključenih u proces, kao i način na koji te aktivnosti zavise jedna od druge. Osnovni skup elemenata UML dijagrama aktivnosti je prikazan na slici 2.13.



Slika 2.13: Osnovni skup elemenata UML dijagrama aktivnosti

EPC

Event-driven Process Chain (EPC) predstavlja tip dijagrama toka, koji se koristi za modelovanje poslovnih procesa. EPS predstavlja direktni graf događaja i funkcija. Obezbeđuje različite konektore za prikaz alternativnog i paralelnog izvršavanja procesa. Koristi logičke operatore OR, AND i XOR. Prednosti EPC zasnovane su na jednostavnoj notaciji, što je doprinelo da EPS bude široko prihvaćena tehnika za označavanje poslovnih procesa [36].

UMM

UN/CEFACT's Modeling Methodology [37] zasnovana je na UML pristupu i koristi se za modelovanje poslovnih procesa. UMM obezbeđuje prikaz poslovnih zahteva interorganizacionih poslovnih procesa, nezavisno od implementacione tehnologije. Cilj je da se predstavi opšta koreografija za saradnju između poslovnih partnera, koji imaju svoje posebne koreografije i koje komuniciraju sa koreografijama poslovnih partnera korišćenjem servisno-orijentisane arhitekture.

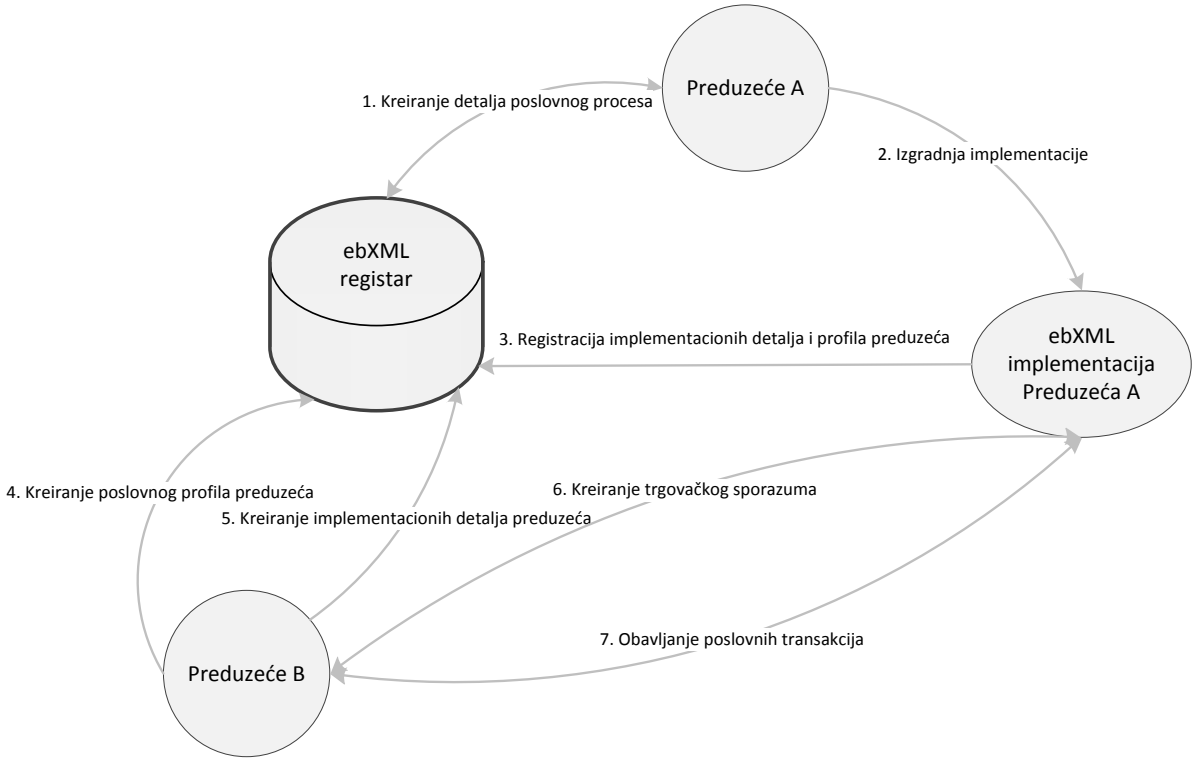
IDEF1 i IDEF3

IDEF1 i IDEF3, kao i IDEF0, pripadaju klasi jezika za modelovanje preduzeća.

IDEF1 koristi jednostavne grafičke elemente (slično Entity-Relation dijagramu) da prikaže: objekte; fizičke ili apstraktne asocijacije, koje postoje između objekata; informacije o objektima i strukturu podataka, koja predstavlja informaciju. IDEF3 metod za modelovanje poslovnih procesa je komplementaran IDEF1 metodu. IDEF3 omogućava predstavljanje opisa toka procesa, kao i promene stanja objekata.

ebXML

electronic business using eXtensible Markup Language (ebXML) kreirali su UN-CEFACT (United Nations Centre for Trade Facilitation and Electronic Business) i OASIS (Organization for the Advancement of Structured Information Standards), sa ciljem da podrži zajedničko elektronsko tržište na svetskom nivou. ebXML sadrži specifikaciju za definisanje standarda poslovnih procesa, razmenu poslovnih poruka i realizaciju trgovinskih sporazuma. ebXML najčešće se opisuje kao horizontalni B2B standard. Uopšten je i odnosi se na sve sektore poslovanja.



Slika 2.14: ebXML arhitektura

RosettaNet

RosettaNet je konzorcijum proizvođača i potrošača, koji su definisali standardne poruke u XML formatu, namenjene razmeni podataka i procesa u okviru elektronskog poslovanja. Najveća vrednost primene RosettaNet je opšta saglasnost u vezi odgovora na pitanje: "Koji procesi šta i gde rade?". RosettaNet je vertikalni standard. Podržava potrebe specifičnih industrijskih grana, uključujući automatizaciju i optimizaciju poslovnih procesa.

2.2.2.5 Modelovanje poslovnih procesa - uporedni prikaz

Činjenica da postoji više različitih grafičkih notacija i standarda za modelovanje poslovnih procesa govori da je teško odabrati i opredeliti se za "najbolji". Pre se može govoriti o izboru, koji je primeren problemu koji se rešava. Pri tome, mogu se uzeti u obzir neki od sledećih kriterijuma:

- Da li zadovoljava potrebe ciljne grupe kojoj je namenjen?
- Da li je jednostavan za korišćenje?
- Ima li dovoljno grafičkih elemenata za jednoznačan prikaz modela?
- Da li obezbeđuje neophodan nivo razumljivosti za sve učesnike u procesu modelovanja?
- Da li je jednostavan za kreiranje i ažuriranje grafičkih modela?
- Da li postoji dovoljno dobrih alata za kreiranje i dokumentovanje modela?

Predložene metode modelovanja poslovnih procesa zastupaju različite pristupe, sa istim ciljem: izraditi model poslovnog procesa. Tokom godina, kao dominantan pristup modelovanju poslovnih procesa, nametnuo se pristup zasnovan na aktivnostima. Karakterističan predstavnik je BPMN notacija, dok se, kao jezik za opis izvršavanja poslovnog procesa, izdvojio BPEL. U poglavlju je diskutovano o njihovom međusobnom mapiranju, kako bi se uspostavio kontinuitet između grafičkog dizajna procesa i izvršavanja procesa.

Poslovni analitičari i analitičari sistema često dovode u pitanje svrsishodnost postojanja velikog broja notacija i jezika za modelovanje poslovnih procesa, kao i pogodnosti njihove primene. Ovo se, pre svega, odnosi na BPMN, BPEL i UML dijagrame aktivnosti, koji se u praksi najčešće koriste za modelovanje i opis izvršavanja poslovnih procesa.

U radu [38], autori razmatraju primenu UML dijagrama aktivnosti za prikaz poslovnih procesa. Zaključak je da UML dijagrami aktivnosti nisu pogodni za modelovanje svih aspekata poslovnih procesa. UML AD pružaju sveobuhvatnu podršku za kontrolu toka podataka, ali nisu pogodni za modelovanje organizacionih resursa.

UML podržava faze analize i dizajna u okviru procesa razvoja softverskih sistema. Faza analize obuhvata prikupljanje zahteva korisnika, a formalizuje se pomoću UML slučaja korišćenja i dijagrama aktivnosti, koji zajedno predstavljaju poslovnu logiku softverske aplikacije. U okviru faze dizajna, poslovna logika se mapira u komponente, koje čine arhitekturu aplikacije. Mapiranje se vrši ručno, a posledice su: neefikasnost, nekonzistentnost i nefleksibilnost. UML do sada nije obezbedio mogućnost da na odgovarajući način mapira slučajeve korišćenja i dijagrame aktivnosti u jezik za izvršavanje procesa [39]. Zato je neophodno da se izabere drugi jezik za programiranje poslovnih procesa. Navedeni problem rešava se korišćenjem BPMN dijagrama i BPEL.

Drugi autori [40], porede složenost BPMN i UML i zaključuju da su oba načina za modelovanje veoma kompleksna. Takođe smatraju da su, imajući u vidu veoma široku primenu, stručnjaci našli načina da se izbore sa njihovom kompleksnošću. Uočeno je da stručnjaci uglavnom ne koriste sve mogućnosti jezika, već samo manji skup jezičkih konstrukcija, neophodan za rešavanje konkretnih problema, što je potrebno uzeti u obzir tokom procesa revizije i proširenja BPMN i UML.

Ukratko, odgovor na pitanje da li postoje razlozi za postojanje BPMN notacije i BPEL jezika, pored UML jezika, su sledeći:

- namenjeni su različitim ciljnim grupama:
    - UML je namenjen softverskim analitičarima, dizajnerima i diveloperima, jer nudi specifikaciju, vizuelizaciju i mogućnost dokumentovanja artifakta za računarski zasnovane sisteme;
    - BPMN i BPEL namenjeni su poslovnim analitičarima, arhitektama sistema i softverskim inženjerima, jer nude specifikaciju, vizuelizaciju i mogućnost dokumentovanja artifakta vezanih za poslovni domen.
- zasnivaju se na različitim paradigmama:
    - UML je zasnovan na objektno orijentisanom pristupu;
    - BPMN i BPEL zasnovani su na procesnom pristupu.
- različitost u nivou implementacije:
    - UML jeziku nedostaje mogućnost implementacije poslovnih modela, koji su visoko konceptualni.

2.3 Servisno orijentisani razvoj softvera

Servisna orijentacija omogućava izgradnju distribuiranih softverskih sistema korišćenjem servisa. Potrebni servisi identifikuju se tokom procesa dekompozicije i kreiraju se u obliku samostalnih funkcionalnih jedinica softvera, koje se mogu ponovno koristiti i po potrebi komponovati sa drugim jedinicama softvera.

2.3.1 Servisi i servisna orijentacija

Servis predstavlja osnovnu jedinicu, ili element, servisno orijentisane arhitekture. Međutim, reč "servis" je jedan od mnogih, nedovoljno određenih termina, koji imaju različita značenja, a u zavisnosti od konteksta u koji se smeštaju.

U ovoj tezi termin "servis" označava koncept softverskog servisa. U navedenom kontekstu, servis predstavlja IT funkcionalnost, koju jedan računarski sistem obezbeđuje drugom. Osim toga, servis je nezavisna jedinica poslovne logike, koja ima sposobnost komunikacije sa drugim servisima, sa ciljem postizanja određenog poslovnog cilja.

2.3.1.1 Servisi

Servisi mogu biti čvrsto ili labavo povezani, a komunikacija među njima obavlja se posredstvom poruka. Poruka je jednostavan način za slanje informacija jednog učesnika, drugom učesniku. Poruke olakšavaju labavo povezivanje, jer dozvoljavaju razdvajanje strane, koja je kreirala servis, od strane koja koristi servis.

Svaki servis trebalo bi da poseduje interfejs, koji definiše ugovor između vlasnika i korisnika servisa. Interfejsi servisa mogu, ali ne moraju, biti zasnovani na standardima. S obzirom da uključenje standarda znatno olakšava obezbeđenje fleksibilne arhitekture, koja podržava nepredviđene promene, potrebno je da se interfejsi servisa i razmena poruka između servisa bazira na standardima. Interakcije između servisa na Internetu opisuju standardi veb servisa, koji specificiraju jezike za definiciju i interakciju veb servisa. Međutim, treba obratiti pažnju da nisu svi servisi veb servisi, niti su svi veb servisi obavezno labavo povezani. O veb servisima i standardima vezanim za veb servise, biće više reči u sledećem delu poglavlja.

Osnovna karakteristika modela servisa je razdvojenost njegovog interfejsa od implementacije. Za uspešno pozivanje servisa neophodno je samo razumevanje interfejsa. Vremenom, implementacija servisa može se razvijati bez znanja klijenata servisa. Takođe, isti interfejs može biti rezultat različitih implementacija, što obezbeđuje ključnu prednost servisnoj orijentaciji – apstrakciju mogućnosti od načina za implementaciju te mogućnosti.

Klasifikacija servisa

Cilj klasifikacije servisa u literaturi je da se odredi najbolja praksa za njihovu implementaciju i upravljanje.

Jedan od klasifikacionih pristupa, zasnovan je na sadržaju servisa [41].

- **Servisi entiteta**
  Sadrže poslovne entitete kao što su: kupac, račun, zaposleni, ugovor, i slično. Obezbeđuju pristup do osnovnih elemenata podataka, posredstvom CRUD operacija (osnovne operacije pristupa, kao što su kreiranje, čitanje, ažuriranje i brisanje) i drugih, logičkih operacija, a na osnovu atributa entiteta. Servisi entiteta uglavnom su agnostički poslovni procesi. Primeri su servisi podataka, koji obezbeđuju pristup podacima u paketnim aplikacijama.
- **Servisi zadataka**
  Obavljaju deo poslovnih procesa, ili sve poslovne procese i aktivnosti i zauzvrat mogu pristupiti različitim entitetima, ili drugim servisima zadataka. Primer servisa zadatka je servis za ažuriranje plaćanja, koji uključuje entitete, kao što su kupac i račun.
- **Pomoćni (Utility) servisi**
  Povezuju servise entiteta i zadataka sa aplikacijama (na primer, servisi veza), posreduju prilikom pristupa infrastrukturi, koja obezbeđuje sigurnost sistema, ili olakšavaju realizaciju aktivnosti (na primer, logovanje).

Druga podela servisa predložena je u radu [42]**.** Servise svrstava u tri grupe, koje suštinski predstavljaju osnovne vrste apstrakcije u SOA.

- **Poslovni servisi**
  Poslovni servisi moraju da budu: procesno orijentisani (da podržavaju poslovne procese u okviru preduzeća); labavo povezani (i klijent i servis mogu imati nezavisne životne cikluse); distribuirani (dostupni od strane bilo kog klijenta, iz bilo koje tačke); i pozivaju se bez predaje stanja (servisni zahtevi ne zavise jedan od drugog).

- **Integracioni servisi**
  Često se ova vrsta servisa tretira poput poslovnog servisa i pored činjenice da su veoma različiti. Integracioni servisi vezuju se uobičajeno za veb servise, namenjene da pojednostave integraciona rešenja u preduzeću.
- **Infrastrukturni servisi**
  Prezentuju zajedničke IT artifakte, u cilju podrške razvoju infrastrukture.

Po [43], postoje četiri vrste servisa:

- **Osnovni servisi**
  Predstavljaju osnovu servisno orijentisane arhitekture. Dele se na servise usredsređene ka podacima i servise usredsređene ka logici. Prva podvrsta servisa uključuje upravljanje podacima: skladištenje i pronalaženje podataka; zaključavanje podataka; upravljanje transakcijama. U osnovi su veoma slični sloju za pristup podacima kod tradicionalnih aplikacija. Druga podvrsta obuhvata servise usredsređene ka logici i sadrži algoritme za kompleksna izračunavanja, ili implementaciju poslovnih pravila. Slični su komponentama poslovnog sloja, koje se sreću kod tradicionalnih aplikacija.
- **Procesno orijentisani servisi**
  Obezbeđuju izvršavanje poslovnih procesa u preduzeću. Najčešće su složeni za implementaciju.
- **Posrednički servisi**
  Koriste se da premoste tehnološke nedoslednosti, ili konceptualne praznine.
- **Javni servisi**
  Obezbeđuju interfejse za komunikaciju sa partnerima i drugim preduzećima. Uobičajeno obezbeđuju mehanizme za sigurnost, plaćanje i asinhronu komunikaciju.

Iako se terminologija predstavljenih klasifikacija servisa razlikuje, uočava se suštinska sličnost između opisa pojedinih kategorija servisa. Na osnovu toga, može se izvesti opšta klasifikacija servisa na: osnovne (servise entiteta), poslovne (procesne) i infrastrukturne (pomoćne, posredničke) servise.

2.3.1.2 Servisna orijentacija

Servisna orijentacija, kao način za izgradnju distribuiranih sistema u formi servisa, može se posmatrati iz različitih uglova. Većina autora posmatra servisnu orijentaciju iz poslovnog i tehničkog ugla.

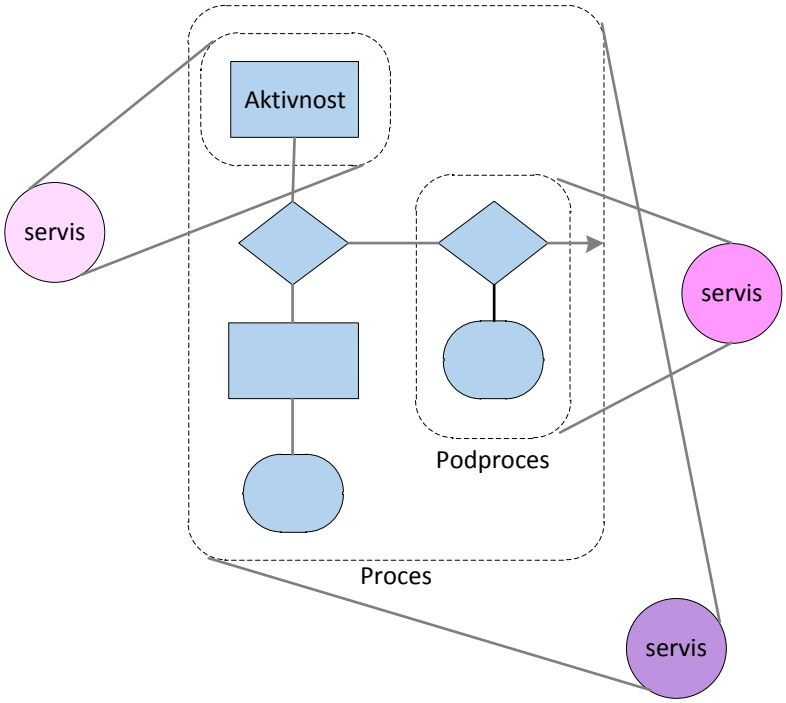
Po [44], servisna orijentacija predstavlja poslovni pristup, koji posmatra servise kao resurse, koje kompanija može da kreira, otkrije i kombinuje i kojima može, po potrebi, da pristupi. Od ključnog značaja je način stavljanja servisa u poslovni kontekst. Dok je, u tehničkom smislu, servis ugovoreni interfejs, u poslovnom smislu, servis je prezentacija resursa dostupnih poslovanju. Drugim rečima, servisi se u servisno orijentisanim organizacijama mogu smatrati resursima, poput ljudskih i fizičkih resursa.

Po [45], sa poslovne tačke gledišta, servisno orijentisani sistemi su način izlaganja funkcionalnosti udaljenim klijentima. Cilj je da se implementiraju novi modeli poslovnih procesa, korišćenjem postojećih softvera ili softvera nezavisnih proizvođača, uz smanjenje ukupnih IT troškova i povećanje inovativnih potencijala, investiranjem u softver. Iz tehničkog ugla, servisno orijentisani sistemi

predstavljaju pristup razvoju softvera, kod koga servisi obezbeđuju višestruku upotrebnu funkcionalnost, sa dobro definisanim interfejsima. Servisna infrastruktura omogućava otkrivanje, kompoziciju i pozivanje servisa, kao i kreiranje softvera korišćenjem dostupnih servisa.

Nezavisno od navedenih gledišta i početnog sporog prihvatanja standarda, koji ih podržavaju, servisno orijentisani sistemi postali su često korišćen pristup, za povezivanje poslovnih modela sa softverskom infrastrukturom. Osim toga, servisno orijentisani sistemi pružaju fleksibilnu podršku u slučajevima promena u sistemu.

Servisna orijentacija, kao sredstvo za izgradnju distribuiranih sistema, na najapstraktnijem nivou, uključuje sve vrste softvera, koji mogu imati ulogu servis provajdera i koji izlažu svoje mogućnosti preko interfejsa. Servisno orijentisana arhitektura mapira izložene mogućnosti i interfejse, omogućavajući orkestriranje u procese. Zahvaljujući toj osobini, može se reći da je servisni model u suštini fraktalni: novoformirani procesi su servisi za sebe, koji potencijalno stvaraju nove, agregatne mogućnosti. Odnos servisa i poslovne logike prikazan je na slici 2.15.



Slika 2.15: Odnos servisa i poslovne logike
(prilagođeno po [41])

Još uvek ne postoji potpuna saglasnost o funkcionalnom značenju pojma "servisna orijentacija". Jedan od razloga je učešće različitih stručnjaka u kreiranju i korišćenju, na ovim principima zasnovanih sistema, s obzirom na njihove različite poglede. Analitičari sistema i ostali učesnici u razvoju, servisnu orijentaciju posmatraju kao sredstvo za kreiranje dinamičkih aplikacija za saradnju. Servisna orijentacija, u vreme izvršavanja, podržava mogućnost izbora isporučioca servisa, dozvoljava aplikacijama da se prilagode sadržaju i kontekstu specifičnih poslovnih procesa, kao i da se tokom vremena uključe i novi isporučioci servisa. Za IT menadžere, servisna orijentacija je sredstvo za efikasnu integraciju različitih sistema. Zahvaljujući modelu za agregaciju podataka i objedinjavanje poslovne logike, više sistema u jedan interfejs, servisna orijentacija omogućava da se različitim

sistemima pristupa preko zajedničkog, povezanog skupa interfejsa. Rukovodioci informacionih centara, u primeni servisne orijentacije vide način za zaštitu postojećih IT investicija, bez sprečavanja isporuka novih mogućnosti. Učaurenjem (eng. encapsulate) poslovne aplikacije na mogućnostima zasnovanih interfejsa, servisni model dozvoljava kontrolisani pristup važnim aplikacijama i stvaranje preduslova za kontinuirano unapređenje implementacije, koja se nalazi iza interfejsa. Servisna orijentacija štiti poslovanje preduzeća od promena, koje mogu nastati vremenom. Za poslovne analitičare, servisna orijentacija je sredstvo za usaglašavanje ulaganja u informacionu tehnologiju sa poslovnom strategijom. Objedinjenim mapiranjem: zaposlenih, mogućnosti spoljnih provajdera i mogućnosti za automatizaciju sistema u jedinstven model, analitičar može bolje da razume troškove ulaganja u ljude, sisteme i druge resurse.

Don Boks među prvima je definisao skup smernica za ostvarenje servisne orijentacije [46]:

- granice su eksplicitne - servisi su nezavisne aplikacije i komuniciraju slanjem poruka preko granica sistema, što obezbeđuje implementaciju nezavisnu od platforme, posrednika ili programskog jezika;
- servisi su autonomni - servisi poseduju sopstvenu funkcionalnost i ponašaju se kao nezavisni entiteti;
- servisi dele šemu i ugovor, a ne dele klase - struktura servisa se izražava preko šeme, a ponašanje preko ugovora koji, kao i sam servis, potpuno opisuje strukturu poruka;
- kompatibilnost servisa zasniva se na pravilima - i provajderi servisa i korisnici servisa imaju pravila za interakciju preko granica koja, druga strana, mora da poštuje u međusobnoj komunikaciji.

Vremenom, različite organizacije objavile su svoje verzije principa servisne orijentacije. Kao rezultat, pojavile su se mnoge varijacije principa, koje se uglavnom odnose na: labavu povezanost; autonomiju; mogućnost otkrivanja; mogućnost komponovanja; ugovore servisa, apstrakciju i neposedovanje stanja. U literaturi je najčešće citirano osam principa, potrebnih da se ostvari koncept servisne orijentacije, a uveo ih je [41]:

- Servisi se mogu ponovo koristiti.
  Bez obzira da li mogućnosti za ponovnom upotrebom odmah postoje, servisi su dizajnirani da podrže višestruku upotrebu.
- Servisi dele formalni ugovor.
  Da bi servisi stupali u interakciju, potrebno je da dele samo formalni ugovor, koji opisuje svaki servis i definiše uslove za razmenu informacija.
- Servisi su labavo povezani.
  Servisi moraju biti dizajnirani tako da mogu da komuniciraju bez potrebe za čvrstim, unakrsnim zavisnostima između njih.
- Servisi ne izlažu postojeću logiku.
  Jedini deo jednog servisa, koji je vidljiv drugim servisima, izložen je u ugovoru servisa. Osnovna logika, izvan onog što je prikazano u opisima koji čine ugovor, nevidljiva je i nebitna za one koji će koristiti servis.
- Servisi se mogu komponovati.

Servisi mogu da čine druge servise. Ovo omogućavaju logika, predstavljanja na različitim nivoima detaljnosti, i promocija mogućnosti ponovnog korišćenja i kreiranja apstraktnih slojeva.

- Servisi su autonomni.
  Servisima upravljanja logika, koja se nalazi iza eksplicitne granice. Servis ima kontrolu u okviru svojih granica i ne zavisi od drugih servisa, pri izvršavanju svoje funkcionalne namene.
- Servisi ne čuvaju stanje.
  Servisi ne bi trebali da čuvaju informacije vezane za stanje. Čuvanje stanja moglo bi da degradira osobinu labave povezanosti. Servise bi trebalo dizajnirati s ciljem da se minimalizuje čuvanje stanja, uključujući mogućnost da se stanja čuvaju na drugim lokacijama.
- Servisi se mogu otkriti.
  Servisi treba da dozvole otkrivanje svojih opisa, razumevanje od strane ljudi i korisnika koji, pri tome, mogu da koriste sopstvenu logiku.

2.3.2 Servisno orijentisana arhitektura

Servisno orijentisana arhitektura (SOA) je arhitektonski stil za izgradnju rešenja preduzeća na osnovu servisa. Najveću vrednost SOA rešenja postižu u slučajevima u kojima se servisi ponovno koriste i kombinuju, prilagođavajući se dinamičkom poslovnom okruženju.

SOA pristupa izgradnji softverskih aplikacija koristeći kolekcije autonomnih servisa, koji stupaju u interakciju, bez obzira na njihove platforme, strukture podataka, ili interne algoritme. Pristup nije nov i pre SOA korišćen je u slučajevima dizajniranja klijent/server softvera. Slično kao kod SOA, distribuirane arhitekture i tehnologije, koje se koriste kod udaljenog pozivanja metoda (DCOM ili CORBA tehnologije), takođe koriste pojam "servis". Navedene tehnologije znatno su uticale na SOA. Međutim, za razliku od njih, SOA je prevazišla problem nedostatka standarda za razmenu poruka.

Principi SOA nastali su na osnovu rada sa distribuiranim objektima:

- Iskustvo u radu sa tehnologijama za distribuirane objekte (na primer, DCOM i CORBA) dovelo je do skupa principa dizajna, smernica i najbolje prakse.
- Postalo je popularno da se upućuje na distribuirane objekte kao na "usluge", odnosno servise.

SOA je originalno opisana kao servisno orijentisani stil arhitekture u tehničkoj sferi znanja. Termin SOA upotrebio je 1996. godine Natis, istraživač u Gartneru. Natis [47] je SOA definisao kao softversku arhitekturu, koja počinje definicijom interfejsa i gradi čitavu topologiju aplikacije, sačinjene od interfejsa, implementacije interfejsa i poziva interfejsa. Imajući u vidu navedene karakteristike, Natis zaključuje da je bolji naziv za SOA: interfejs-orijentisana arhitektura.

U međuvremenu, predloženo je više definicija SOA. Definicije su slične u mnogim aspektima, ali uočavaju se i razlike. U principu, postoje dva glavna pristupa SOA. Na jednoj strani je tehnička definicija SOA, koja se fokusira na automatizaciju poslovnih procesa preko orkestracije, uglavnom veb servisa, za sprovođenje procesa. U tom scenariju, servis se uobičajeno opisuje preko BPMN notacije, na osnovu koje se može izvesti izvršni opis komponovanih servisa preko BPEL jezika.

Iz poslovne perspektive, SOA se prihvata kao upravljački koncept u kompaniji. Ovakva interpretacija SOA je novija, i poslednjih godina, u poslovnim krugovima, dobija na popularnosti. Različiti entiteti u kompaniji, koji stvaraju vrednost, struktuirani su u servise, koji se mogu kombinovati na fleksibilan način, kako bi odgovorili poslovnim potrebama. Servisi ne moraju biti ni veb servisi, niti bilo koja druga vrsta tehnologije, jer smatra se da je SOA nezavisna od informacionih tehnologija.

Neki autori [48] ističu da je moguća primena koncepta SOA analogno u poslovnom domenu, kako bi se poboljšali poslovni modeli, i pored činjenice da većina publikacija definiše SOA u odnosu na tehnički domen. Po definiciji, koja je primenljiva i na poslovni i na tehnički nivo, SOA je šema za organizovanje i korišćenje distribuiranih mogućnosti, koje mogu biti pod kontrolom različitih vlasničkih domena. Definicija je navedena u SOA referentnom modelu Organizacije za unapređenje i strukturiranje informacionih standarda (Organization for the Advancement of Structured Information Standards - OASIS).

U daljem tekstu navedeno je još nekoliko definicija SOA, koje se najčešće mogu sresti u literaturi. Na izbor je uticala tehnička orijentacija autora teze.

Erl [41], značajan autor u oblasti servisno orijentisane arhitekture, definiše SOA na sledeći način: SOA je oblik distribuirane arhitekture, koja se pridržava principa servisne orijentacije. Kada se realizuje preko veb servisa, SOA može da podrži i promoviše servisno orijentisane principe preko poslovnih procesa i automatizacije poslovanja preduzeća.

Po [49], SOA predstavlja klijent-server pristup dizajnu softvera, u kojem se aplikacija sastoji od softverskih servisa i korisnika softverskih servisa. Karakteristika SOA, a razlika u odnosu na klijent-server model, je međusobno labavo povezivanje softverskih komponenti.

Po [50], u softverskom inženjerstvu, SOA predstavlja skup principa i metodologija za dizajniranje i razvoj softvera u formi interoperabilnih servisa. Servisi predstavljaju dobro definisane poslovne funkcionalnosti, koje su ugrađene u softverske komponente. Servisi se mogu ponovo koristiti za različite namene. Principi SOA dizajna koriste se tokom faze razvoja i integracije sistema.

Može se zaključiti da većina definicija, koje se sreću u literaturi, navodi jednu, ili više karakteristika SOA :

- samostalni servisi;
- autonomni servisi;
- labava povezanost;
- standardni protokoli;
- izvršavanje poslovnog procesa;
- poslovno orijentisana arhitektura;
- standardizovani interfejsi;
- nezavisnost i mogućnost saradnje;
- dinamička integracija;
- mogućnost ponovnog korišćenja;
- kompozicija servisa.

Međutim, nijedna definicija ne sadrži sve prethodno navedene karakteristike, što ukazuje na različite aspekte posmatranja SOA koncepta, kao i da se SOA ne može posmatrati samo iz ugla jednog učesnika.

Trendovi, koji su najviše uticali na poslovanje preduzeća do 2010. godine, svrstani su u šest oblasti [45]. Za sve navedene oblasti karakterističan je značajan uticaj SOA, koji je u narednom tekstu označen *iskošenim fontom*:

- **Globalna integracija** - Potrebno je da preduzeća brzo reaguju na globalne mogućnosti, što znači da bi zaposleni trebalo da koriste globalni, intelektualni kapital u preduzeću, kao i van preduzeća i da otkriju prethodno nepoznate mogućnosti. Ova sposobnost omogućava brzo reagovanje na tržištu, pre konkurencije.
  Realizuje se kreiranjem, otkrivanjem i višestrukim korišćenjem poslovnih servisa.
- **Participativni Internet** - Ideja je da se obuhvate i ponovo koriste informacije o korisnicima na Internetu, prati njihovo ponašanje i da se izvedeni zaključci primene na buduće interakcije sa njima. Dobri primeri su: Google rangiranje strana, procena prodavaca na eBay portalu, prikaz i ocena proizvoda na Amazonu.
  Ovaj trend zahteva kreiranje jedinstvenog pogleda na klijente i mogućnost da se kombinuju i analiziraju interakcije sa njima u okviru preduzeća.
- **Demografija radne snage** - Zamena zaposlenih novim radnicima podrazumeva preuzimanje postojećih podataka.
  Od vitalnog značaja za ovaj proces je da se obezbedi da podaci preduzeća budu dostupni preko servisa.
- **Softver kao servis** - Korišćenje Software as a Service (SaaS) aplikacija i podataka je jeftinije, jednostavnije za održavanje i unapređenje, jednostavno za integraciju i podršku i dostupno na bilo kom mestu. Postojeće aplikacije najčešće nisu jednostavne za održavanje, unapređenje i integraciju. Vlasničke aplikacije, koje se uobičajeno koriste, skupe su za rad i održavanje i složene za integraciju sa ostalim aplikacijama u preduzeću.
  Korišćenje SaaS aplikacija obezbeđuje konkurentnu prednost.
- **Virtuelizovani podaci i uređaji** - Aplikacije i podaci se premeštaju na mrežne uređaje i skladišta, obezbeđujući da se, sa bilo koje platforme, podacima pristupa pomoću raspoloživih veb brausera.
  Sposobnost da se obezbede potrebni podaci zainteresovanim učesnicima, bez obzira gde se nalaze, donosi konkurentne prednosti preduzeću.
- **Jednostavnost dizajna** - Složenost okruženja otežava komunikaciju sa zaposlenima i klijentima, čime se uvećava potreba da se aktivnosti obavljaju u jednostavno dizajniranom okruženju.
  Potrebe korisnika mogu se zadovoljiti jednostavnim metodama interakcije.

Bez obzira na velike mogućnosti primene SOA, postoje situacije kada SOA ne bi trebalo koristiti [44]**:**

- kada postoji homogeno IT okruženje;
- kada su kritične performanse aplikacija, koje se izvršavaju u realnom vremenu;
- kada se objekti poslovanja ne menjaju;
- kada je čvrsto povezivanje prednost, a ne nedostatak.

2.3.2.1 Pristupi dizajniranju SOA sistema

S obzirom da SOA predstavlja arhitektonski stil za izgradnju rešenja preduzeća na osnovu servisa, identifikovanje servisa trebalo bi da bude jedna od prvih aktivnosti tokom procesa modelovanja servisno orijentisanih sistema. Postoji više pristupa za određivanje servisa, i bez obzira koji se pristup ili kombinacija prihvati, važno je pridržavati se osnovnih principa, potrebnih da se ostvari koncept servisne orijentacije, koje je uveo Erl [41].

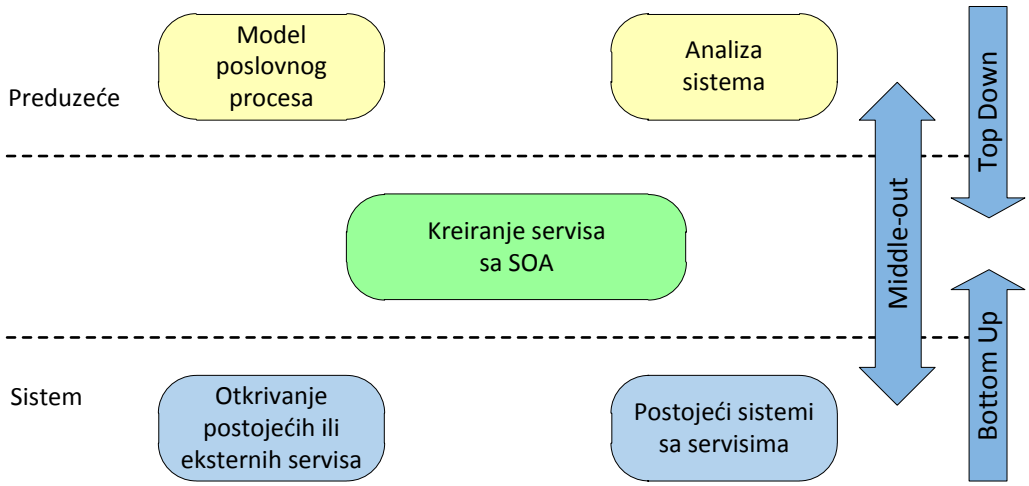
Projektovanje rešenja retko počinje sa potpuno poznatim problemom i direktnom implementacijom kompletnog rešenja. Samim tim, nisu unapred poznati IT resursi, potrebni za izgradnju servisa. Da bi se došlo do potrebnih rešenja, važno je opredeliti se za odgovarajuću metodologiju, koja će omogućiti dekompoziciju procesa, definisanje aktivnosti i servisa. Ovako identifikovane i kreirane servise moguće je naknadno komponovati u procese.

Pristup dizajniranju SOA sistema može da bude "top-down" i "bottom-up". U tezi opisani pristupi predstavljeni su slikom 2.16.

Pristup "top-down" koristi se za dekompoziciju poslovnih procesa, identifikaciju i kreiranje servisa. U okviru dekompozicije polazi se od poslovnih procesa, koji se rastavljaju na podprocese, ili se dekomponuju u zadatke i aktivnosti. Najniži nivo dekompozicije sadrži logičke jedinice posla, koje treba podržati servisima. Ovako identifikovani servisi, uklapaju se u funkcionalne potrebe sistema, koji se kreira. Glavna prednost ovog pristupa je istovremeni razvoj sistema i njegovih integracionih mehanizama.

Pristup "bottom-up" koristi postojeće funkcionalnosti kao servise, sa ciljem da se komponuju u poslovne procese. Za postojeće aplikacije, kreiraju se novi interfejsi, koji odgovaraju veb servisima. Pristup je pogodan za okruženja, koja obuhvataju heterogene tehnologije i platforme.

U članku [44] razmatra se primena navedenih principa za dizajniranje SOA sistema. Smatra, da posledica primene samo "top-down" pristupa, može usloviti tehnički tešku i složenu implementaciju. Osim toga, isključiva upotreba "bottom-up" pristupa može dovesti do korišćenja servisa, koji nisu potrebni, ili ne odgovaraju zahtevima poslovanja.



Slika 2.16: Pristupi dizajniranju SOA sistema
(prilagođeno po [51])

"Middle-out" je treći, kompromisni, pristup dizajniranju SOA. Kombinuje "top-down" i "bottom-up" pristupe. Ovaj pristup omogućava izgradnju informacione i poslovne arhitekture na najvišem nivou, kao i kreiranje isporučivih servisa. "Middle-out" pristup primenljiv je zahvaljujući SOA referentnoj arhitekturi, koja obuhvata: definiciju servisa i tipove servisa; njihov odnos, relacije i implementaciju [51]. Tokom razvoja sistema, referentna arhitektura neprekidno uzima u obzir kontekst preduzeća pri sprovođenju projektnih aktivnosti i kreiranju projektnih artifakata.

Do sada je razvijeno više metodologija za modelovanje servisno orijentisanih sistema. Većina metodologija koristi neki od navedenih pristupa. U radu [52], dat je uporedni pregled trideset pristupa za identifikaciju i analizu poslovnih i softverskih servisa. Smatra da su poslovni servisi određeni skup akcija, koje organizacija izvršava, a zadatak softverskih servisa je da podrže izvršavanje poslovnih servisa. Kriterijumi za evaluaciju pristupa zasnovani su na: SOA konceptu (podrška za poslovne i softverske servise); vrsti pristupa (top-down, bottom-up, middle-out); pokrivanju životnog ciklusa; postojanju smernica, ili procesa za razvoj servisa; pristupnosti i validnosti.

Iz uporednog prikaza može se videti da su, kod većine analiziranih metodologija, poslovni servisi preovlađujući SOA koncept, što uslovljava korišćenje "top-down" pristupa za dizajniranje servisa. Pokrivenost životnog ciklusa servisa značajno se razlikuje, jer se u literaturi ne sreće jedinstven, standardizovan životni ciklus servisa. Većina metodologija za identifikaciju servisa ne daje smernice za izvođenje servisa, ili su smernice uopštene.

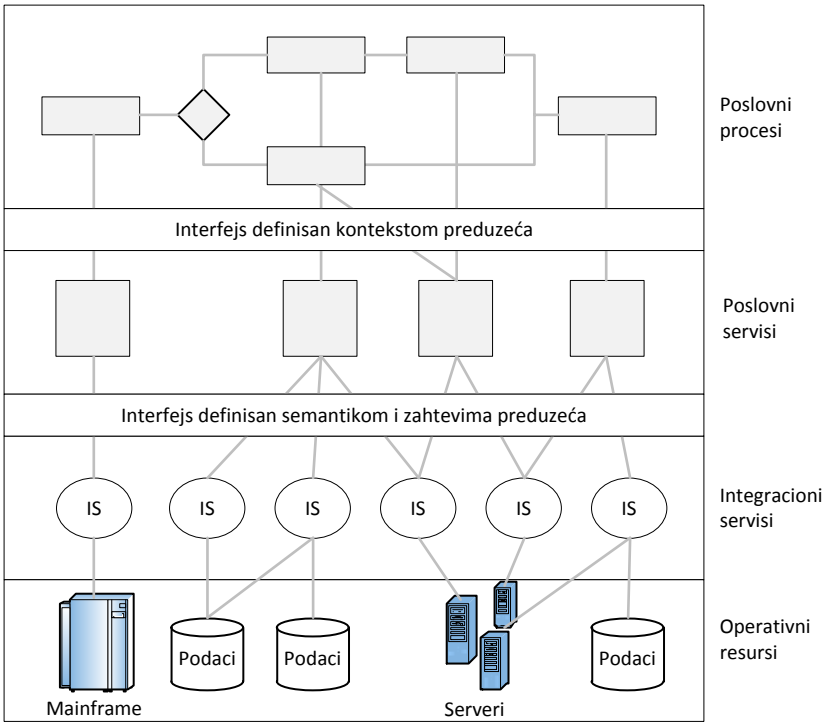
2.3.2.2 Odnos SOA i BPM

Predmet primene BPM i SOA su analiziranje poslovnih procesa preduzeća, dekomponovanje poslovnih procesa na aktivnosti i identifikovanje servisa, kojima se aktivnosti mogu implementirati u poslovni sistem preduzeća. Zbog činjenice da je predmet modelovanja isti, ne postoje sasvim jasne granice između BPM i SOA modela. BPM modele kreiraju poslovni analitičari. Dobijenu arhitekturu sistema preuzimaju sistem analitičari, čiji je zadatak da BPM model prevedu u SOA model i da, kao krajnji rezultat, definišu aplikacioni nivo sistema.

Korišćenje BPM omogućava poslovnim analitičarima da kreiraju poslovne procese preduzeća, nadgledaju njihove performanse i utiču na efikasnost izvršavanja. Svaki poslovni proces modelovan je preko skupa individualnih aktivnosti, koje se mogu implementirani kao servisi u okviru preduzeća. BPM sistem obezbeđuje skup alata, koji poslovnom analitičaru omogućava: da kreira modele procesa, koristeći notacije poput BPMN; da obavlja automatizaciju poslovnih procesa, ili izvršava model, pozivajući servise, uobičajeno koristeći BPEL jezik. Dodatno, BPM sistem može da podrži nadgledanje i upravljanje poslovnim procesom. Na osnovu navedenih karakteristika, BPM se može posmatrati preko tri različita koncepta: BPM kao disciplina; BPM kao sistem i BPM kao paket alata za implementaciju BPM sistema.

Za razliku od BPM, SOA predstavlja arhitektonski pristup za kreiranje distribuiranih sistema, pomoću skupa servisa. SOA obezbeđuje aplikacionu platformu, koja povezuje poslovne procese i operativne resurse, kao što je prikazano na slici 2.17. Na nivou poslovnog procesa, SOA obezbeđuje interfejse, koji direktno podržavaju izvršavanje poslovnih zadataka. Interfejsi se definišu da podrže konzistentnost i mogućnost ponovnog korišćenja. Na nivou operacionih resursa, SOA podržava postojeće mogućnosti, u obliku integracionih servisa. Podrška se ne realizuje direktnim mapiranjem

postojeće aplikacije u servise, već obezbeđenjem interfejsa za novi servis, koji se zasniva na semantici preduzeća i funkcionalnim zahtevima i koji se mapira u postojeće sisteme. Aplikacioni nivo kreira se primenom kompozicije servisa, kojom se povezuju najviši i najniži nivoi.



Slika 2.17: BPM i SOA
(prilagođeno po [51])

U praksi, zajednička primena BPM i SOA obuhvata nekoliko scenarija:

- Određeni broj preduzeća posle primene SOA uvodi BPM, i postojeće nadležnosti SOA direktno prenosi na BPM.
- Sa druge strane, postoje preduzeća koja već koriste BPM kada odluče da uvedu SOA, pri čemu se uobičajeno koristi postojeća analiza poslovanja i integraciona infrastruktura, kao osnova za SOA primenu.
- Deo preduzeća se oslanja na gotova rešenja kupovinom vlasničkih paketa za BPM/SOA, koji svoja postojeća rešenja proširuju primenom jednog od prethodna dva scenarija.

BPM i SOA zajedno obezbeđuju prilično dobru osnovu za viši stepen automatizacije elektronskog poslovanja preduzeća. BPM obezbeđuje na visokom nivou apstrakciju za definisanje poslovnih procesa, kao i mogućnost nadgledanja i upravljanja procesima. Servisi obezbeđuju funkcije, koje podržavaju te procese. SOA obezbeđuje mogućnost kombinovanja servisa i podršku da se kreira fleksibilno preduzeće. BPM, bez SOA modela, može se iskoristiti za kreiranje aplikacija, ali je takve aplikacije teško proširiti. SOA model, bez BPM, uvodi višestruko korišćenje i konzistentnost servisa, ali bez sposobnosti da se ti servisi uključe u aktivno, konkurentno preduzeće.

2.3.3 Veb servisi

Poslovni procesi, u servisno orijentisanim okruženjima, automatizuju se i izvršavaju pomoću servisa. Servisi moraju biti prilagođeni specifičnim potrebama preduzeća i konfigurisani za izvršavanje u određenim situacijama. Pored toga, nezavisne servise trebalo bi međusobno povezati, sa ciljem da zadovolje različite poslovne ciljeve. Veb servisi su najčešći oblik servisa, koji koriste za implementiranje servisno orijentisane arhitekture.

Najčešće se navodi i koristi nekoliko sličnih definicija.

Veb servisi su softverski objekti, koji obavljaju funkcije, ili izvršavaju poslovne procese i kojima se može pristupati preko Interneta, korišćenjem standardnih protokola [53].

Po definiciji World Wide Web konzorcijuma [54] veb servis je softverski sistem, dizajniran da podrži interoperabilnu interakciju mašina-mašina preko mreže. Interfejs veb servisa opisan je u mašinski obradivom formatu (konkretno Web Service Description Language - WSDL). Drugi sistemi komuniciraju sa veb servisima na način propisan njegovim opisom, korišćenjem SOAP poruka, a uobičajeno preko HTTP protokola.

Korišćenjem veb servisa može se implementirati automatizaciona logika, preko koje se primenjuje servisna orijentacija. Mogu se graditi servisno orijentisani EJB ili .NET sklopovi, koji se mogu koristiti kao veb servisi [41].

S obzirom da prirodno podržavaju labavo povezivanje, ugovore, apstrakciju i mogućnost kompozicije, veb servisi su preporučen izbor za realizaciju servisno orijentisanih sistema.

Neke od ključnih osobina veb servisa su [55]:

- **Veb servisi su samostalni.**
  Na klijentskoj strani nije potreban dodatni softver. Dovoljan je samo programski jezik sa XML i HTTP podrškom. Na serverskoj strani, potreban je veb server.
- **Veb servisi su samoopisujući.**
  Definicija formata poruke nalazi se u samoj poruci. Nisu potrebna nikakva dodatna skladišta za meta podatke, niti alati za njihovo kreiranje.
- **Veb servisi su modularni.**
  Veb servisi predstavljaju tehnologiju za isporučivanje i obezbeđivanje pristupa poslovnim funkcijama preko veba, J2EE, CORBA i drugih standarda, koji predstavljaju tehnologije za primenu veb servisa.
- **Veb servisi mogu biti publikovani, locirani i pokrenuti preko veba.**
  Da bi se to realizovalo, neophodni su sledeći standardi:
  - Simple Object Access Protocol (SOAP)
  - Web Service Description Language (WSDL)
  - Universal Description, Discovery, and Integration (UDDI)
- **Veb servisi su nezavisni od jezika i interoperabilni su.**
  Interakcija veb servisa i klijenta, koji mu pristupa, dizajnirana je da bude nezavisna od platforme i jezika. Njihova interakcija zahteva WSDL dokument za definiciju interfejsa i opis servisa, zajedno sa mrežnim protokolom. S obzirom da veb servis ne zna na kojoj platformi i u kom jeziku su klijenti kreirani, i obrnuto, proističe da korišćenje veb servisa omogućava interoperabilnost.

- **Veb servisi su zasnovani na otvorenim standardima.**
  Tehnička osnova veb servisa su XML i HTTP. Veliki deo tehnologije veb servisa kreiran je korišćenjem Open Source projekata.
- **Veb servisi su dinamički.**
  Opis i pronalaženje veb servisa mogu se automatizovati preko UDDI i WSDL.

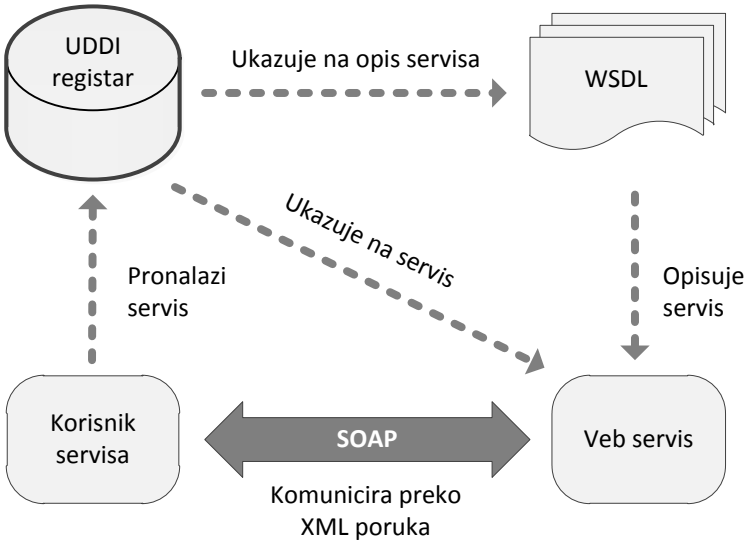
Karakteristike servisno orijentisanih aplikacija i veb servisa nisu potpuna novost. Slično ranijim, distribuiranim tehnologijama i arhitekturama, njihova glavna svrha je da se omoguće pozivi ugrađenih funkcionalnosti, slično kao što JEE posrednik omogućava da klijentske aplikacije pozivaju metode JEE komponenti.

Suštinska razlika, u odnosu na JEE, je usredsređenost servisno orijentisanog modela i podržanih tehnologija na interoperabilnost i rešavanje praktičnih problema, koji se javljaju zbog razlika u platformama i programskim jezicima. Iako je moguće dizajnirati i izgraditi servisno orijentisani sistem, koristeći bilo kog posrednika za distribuirano računarstvo i integraciju, samo tehnologije veb servisa mogu da zadovolje aktuelni uslov za uspostavljanje interoperabilnosti, koji čini veoma važan deo servisno orijentisane vizije.

2.3.3.1 Arhitektura veb servisa

Popularna interpretacija veb servisa zasniva se na IBM-ovoj arhitekturi [55], koju čine tri elementa:

- **Korisnik servisa (Service Consumer)**: Predstavlja aplikaciju, softverski modul, ili drugi veb servis, koji inicira otkrivanje veb servisa u registru, povezuje se sa veb servisom i izvršava veb servis operaciju.
- **Isporučilac servisa (Service Provider)**: Predstavlja mrežno-adresni entitet, koji prihvata i izvršava zahteve korisnika servisa. Isporučilac servisa publikuje opis svojih servisa u registar servisa, sa ciljem da ga potencijalni korisnici mogu pronaći.
- **Registar servisa (UDDI Service Registry)**: Obezbeđuje otkrivanje i pronalaženje veb servisa. Sadrži skladište dostupnih veb servisa i dozvoljava zainteresovanim, potencijalnim korisnicima pretraživanje informacija, koje sadrži.



Slika 2.18: Standardni scenario veb servisa

Standardne operacije arhitekture veb servisa, predstavljene na slici 2.18, obuhvataju:

- **Objavljivanje (Publish)**: Da bi bio dostupan, veb servis mora biti objavljen, što omogućava zainteresovanim korisnicima da ga otkriju i pozovu na izvršavanje.
- **Pronalaženje (Find)**: Zainteresovani, potencijalni korisnik otkriva veb servis pretraživanjem registra servisa po zadatom kriterijumu.
- **Povezivanje i pozivanje (Bind and Invoke)**: Posle pronalaženja opisa veb servisa, korisnik se povezuje sa veb servisom i poziva ga na izvršavanje, a u skladu sa informacijama, koje se nalaze u opisu veb servisa.

Sadržani artifakti u arhitekturi veb servisa su:

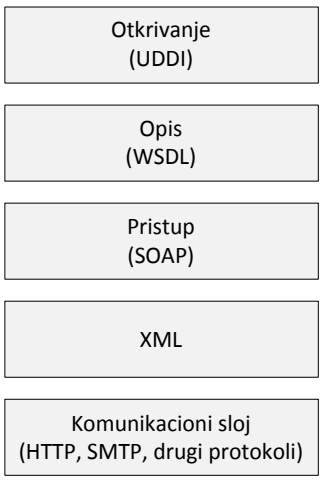
- **Servis**: stavlja se na raspolaganje za korišćenje, preko objavljenog interfejsa.
- **Opis servisa**: specificira način interakcije sa isporučiocem servisa i opisuje format zahteva i odgovor servisa.

2.3.3.2 Platforma veb servisa

Veb servisi predstavljaju tehnologiju, koja je relativno brzo i široko prihvaćena, kao način za integraciju aplikacija. Razlog leži u činjenici da veb servisi obezbeđuju distribuirani pristup za integraciju veoma različitih aplikacija preko Interneta. Specifikacija veb servisa potpuno je nezavisna od programskog jezika, operativnog sistema i hardvera. Obezbeđuje labavu povezanost između veb servisa i sistema, koji ga koristi. Veb servisi zasnovani su na otvorenim standardima i specifikacijama:

- eXtensible Markup Language (XML),
- SOAP,
- Representational State Transfer (REST),
- Web Services Description Language (WSDL),
- Universal Description, Discovery and Integration (UDDI).

XML definiše skup standarda za označavanje strukture i sadržaja dokumenta, koji omogućavaju njihovu čitljivost na različitim platformama i u okviru različitih aplikacija.



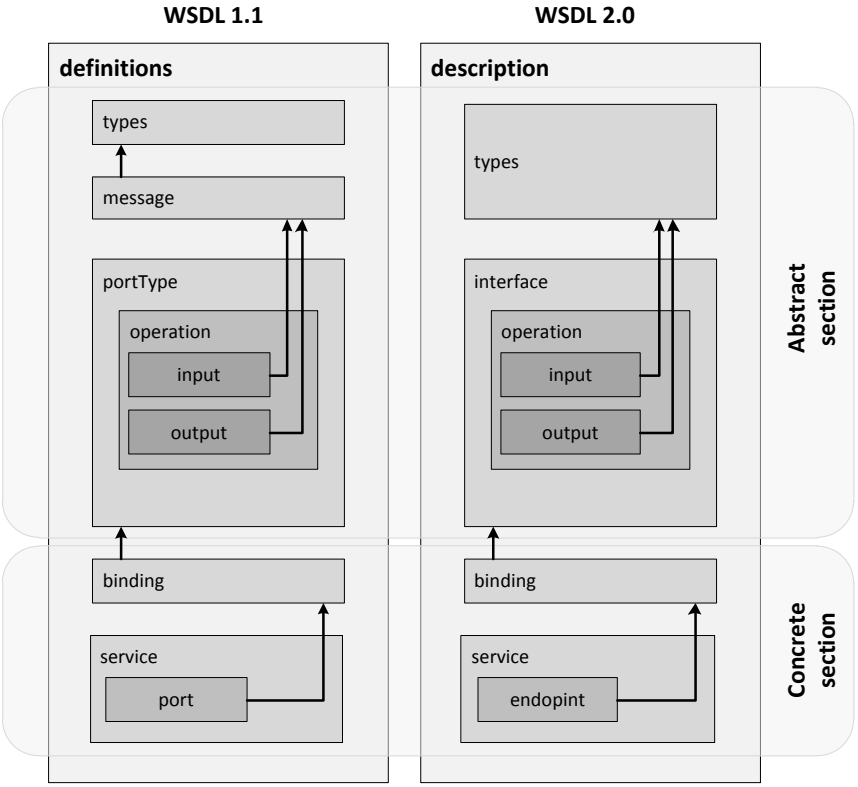
Slika 2.19: Osnovni slojevi platforme veb servisa

SOAP je protokol kreiran da omogući transfer XML poruka. SOAP predstavlja komunikacioni mehanizam za povezivanje veb servisa. XML i SOAP održava W3C konzorcijum, čiji je cilj razvoj interoperabilnih specifikacija i uputstava. U vreme pisanja ovog rada, aktuelna verzija je SOAP 1.2. Do pojave poslednje verzije, slova u nazivu SOAP najčešće su tumačena kao skraćenica od: Simple Object Access Protocol. Pojavom poslednje SOAP verzije, slova u nazivu nemaju više neko posebno značenje.

REST predstavlja skup arhitektonskih principa, čijim se korišćenjem mogu dizajnirati veb servisi, koji se fokusiraju na resurse sistema. Principi uključuju i način pristupa, posredstvom HTTP protokola, velikog broja klijenata resursu, čiji je kod napisan korišćenjem različitih jezika. RESTful veb servisi su vrsta veb servisa, koji su implementirani korišćenjem HTTP protokola i koji su zasnovani na REST principima.

WSDL je jezik za opis veb servisa i zasniva se na XML jeziku. WSDL dokument opisuje šta veb servis radi, kako se sa njim može komunicirati i gde se može pronaći. Delovi opisa veb servisa mogu se nalaziti u više dokumenata, što obezbeđuje veću fleksibilnost i povećava mogućnost ponovnog korišćenja. Prilikom pretraživanja registra veb servisa, ukazuje se na njegov opis, koji se nalazi u WSDL dokumentu.

WSDL ima sličnu ulogu, poput uloge IDL (Interface Definition Language) u slučaju konvencionalnih posredničkih (middleware) platformi: opisuje servis i može se iskoristiti za automatsko generisanje koda za poziv servisa. Međutim, WSDL, kao i IDL jezici, ne sadrži informacije o: semantici i poslovnim protokolima i komunikaciji.



Slika 2.20: WSDL 1.1 i WSDL 2.0 dokumenti i relacije između elemenata
(prilagođeno po http://en.wikipedia.org/wiki/File:WSDL_11vs20.png)

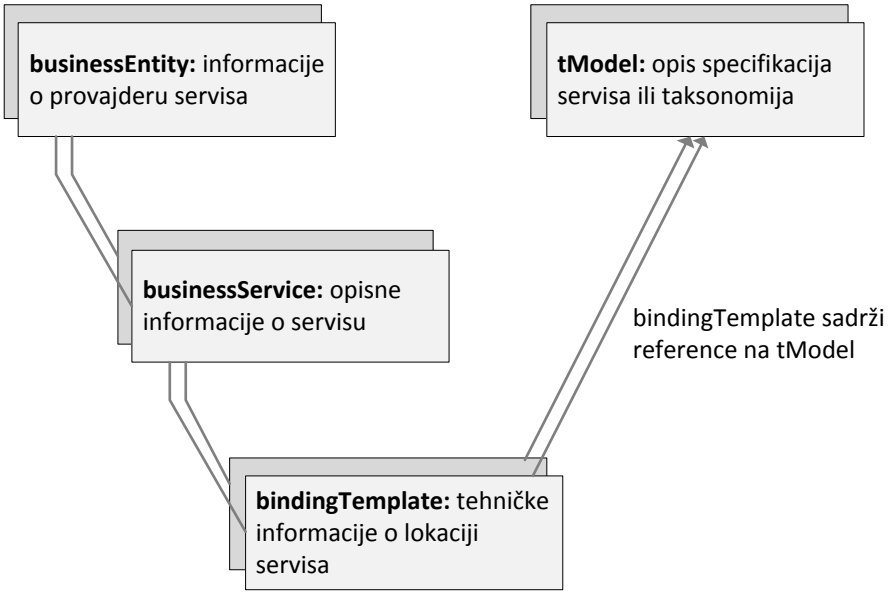
UDDI obezbeđuje mehanizam za oglašavanje i otkrivanje veb servisa. UDDI predstavlja registar za veb servise, pri čemu i sam UDDI može biti veb servis. Osnovna razlika, između UDDI registra i drugih registara i direktorijuma, je u činjenici da UDDI obezbeđuje mehanizam za kategorizaciju poslova i servisa, korišćenjem više klasifikacionih šema (taksonomija). Taksonomije pomažu korisnicima veb servisa da pronađu odgovarajući servis, koji odgovara njihovim zahtevima.

UDDI specifikacija opisuje:

- UDDI model podataka, koji se koristi za opis poslovanja i veb servisa, i
- UDDI API, koji opisuje SOAP poruke za pretraživanje i publikovanje UDDI podataka.

UDDI model podataka, prikazan na slici 2.21, definiše se pomoću XML šema dokumenata i sastoji se od četiri glavna elementa:

- businessEntity: Element obezbeđuje informacije o provajderu servisa i uključuje: ime provajdera, njegov opis i detalje o kontaktima. Uobičajeno je da se svaki poslovni entitet opisuje preko reference na odgovarajući poslovni katalog, odnosno taksonomiju.
- businessService: Element daje informacije o veb servisu, ili grupi servisa i uključuje: naziv servisa, opis i, opciono, listu bindingTemplate.
- bindingTemplate: Element obezbeđuje informacije o lokaciji na kojoj se može pronaći i pristupiti određenom veb servisu.
- tModel (technical Model): Predstavlja pokazivač na eksterne, tehničke specifikacije i opise servisa, pre svega na WSDL dokument.



Slika 2.21: UDDI model podataka

Otkrivanje veb servisa preko UDDI registra izvršava se manuelno (uključuje pretraživanje preko veb formi), ili programski (korišćenjem UDDI API). Oba načina imaju određene nedostatke, a posledica je sprečavanje metoda za otkrivanje veb servisa da pronađu najrelevantnije servise. Neki od uočenih nedostataka su [56]:

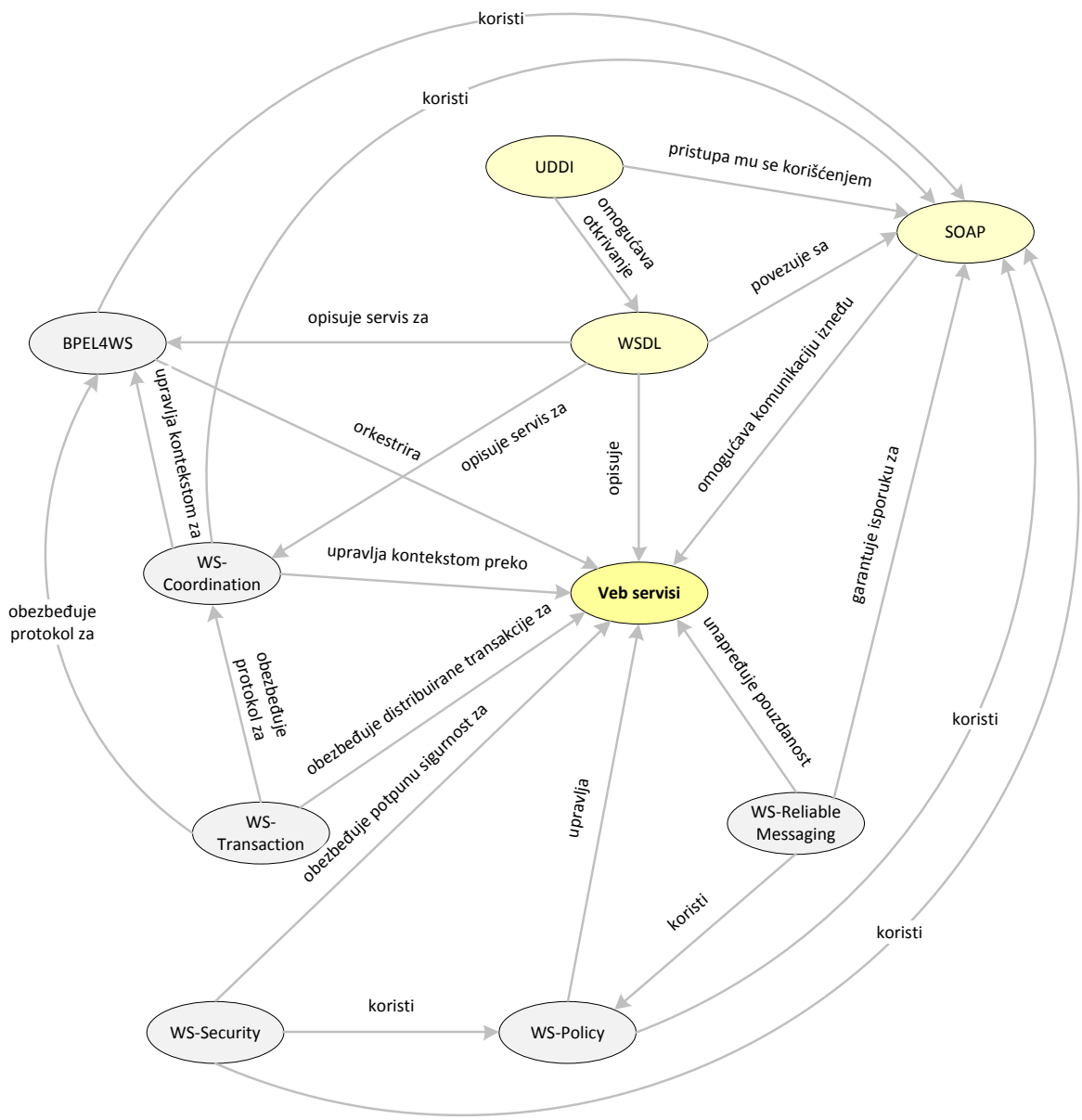
- Neformalni opis mogućnosti i funkcionalnosti servisa - servisi su najčešće opisani korišćenjem prirodnog jezika, bez postojanja zajedničkog rečnika.
- Nekompletan opis mogućnosti i funkcionalnosti servisa – nivo detaljnosti opisa servisa određuje svaki provajder servisa, s obzirom da ne postoje predefinisana i obavezna polja za opis servisa.
- Sintaksna relevantnost - pronalaženje servisa, na osnovu traženih pojmova, uobičajeno ne daje dobre rezultate, jer je servis pronađen samo ukoliko se poklapaju ključne reči, pri čemu se ne uzimaju u razmatranje razlike između jednine i množine reči, padeža i slično.
- Nedostatak specifikacije sa ograničenjima - provajder servisa trebalo bi da obezbedi specifikaciju, koja se odnosi na preduslove i ograničenja bitna za uspešno izvršavanje servisa, a koja može značajno uticati na početni izbor servisa.
- Ograničena izražajnost domenskih klasifikacionih šema - za opis funkcionalnosti servisa najčešće se koriste standardni rečnici, kao što su *North American Industry Classification System* (*NAICS*) i *United Nations Standard Products and Services Code (UNSPSC)*, koji su znatno ograničeni i nefleksibilni.
- Nepostojanje podrške za indirektno uparivanje - u situacijama, u kojima je nemoguće direktno pronalaženje odgovarajućeg servisa po zahtevu, metode pretraživanja UDDI ne omogućavaju indirektno uparivanje, što predstavlja ozbiljno ograničenje.

Pored navedenih nedostataka, bitni nedostaci opisane, standardne veb servis platforme, odnose se na: bezbednost na nivou poruke i na transakcije između servisa i pouzdanost pri razmeni poruka. Ove nedostatke, proširenjem, rešava druga generacija platforme za veb servise. Proširenja uobičajeno imaju prefiks WS- i u oblastima tehnologije i dizajna nude bogat skup složenijih funkcija.

Neke od najznačajnijih WS - * specifikacija su:

- WS-Security;
- WS-Coordination, WS-AtomicTransaction, WS-BusinessActivity;
- WS-ReliableMessaging;
- WS-Policy;
- WS-Addressing.

Grafički prikaz prve i druge generacije tehnologija veb servisa predstavljen je na slici 2.22.

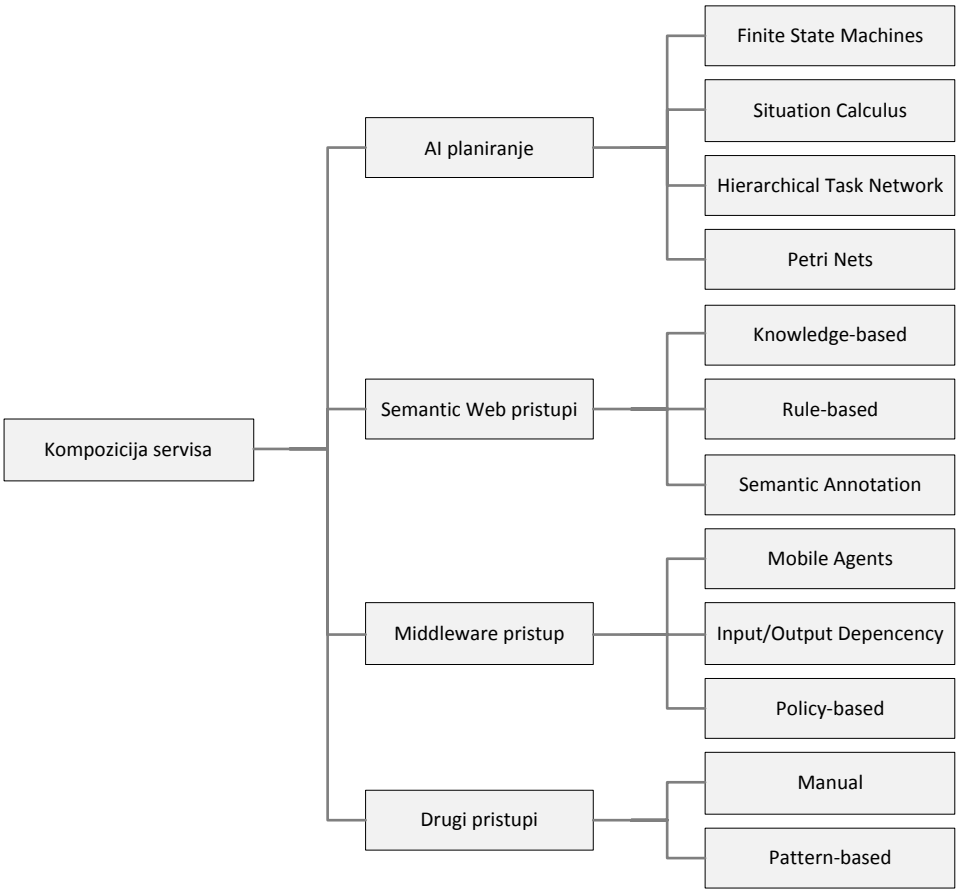


Slika 2.22: Grafički prikaz prve i druge generacije tehnologija veb servisa
(prilagođeno po [41])

2.3.3.3 Kompozicija veb servisa

Prethodno opisani standardi i specifikacije platforme veb servisa omogućavaju implementaciju jednostavnih interakcija između klijenta i veb servisa. U slučajevima kad implementacija poslovnog modela zahteva pozivanje više veb servisa, neophodno je da se izvrši agregacija njihovih funkcionalnosti. To se uobičajeno obavlja u okviru procesa, koji se naziva kompozicija servisa. Kompozicija servisa vrši se kombinovanjem i povezivanjem već postojećih, osnovnih ili prethodno komponovanih servisa, pri čemu klijent poziva kompozitni servis, kao bilo koji jednostavan servis, bez uvida u njegovu složenost.

Kompozicija servisa nije jednostavan zadatak i ona, pored otkrivanja veb servisa, uvodi novi važan problem, koji je potrebno rešavati u slučaju servisno orijentisane arhitekture. Postoji više pristupa za kompoziciju servisa, a jedna od mogućih kategorizacija pristupa data je na slici 2.23.



Slika 2.23: Kategorizacija kompozicije veb servisa
(prilagođeno po [57])

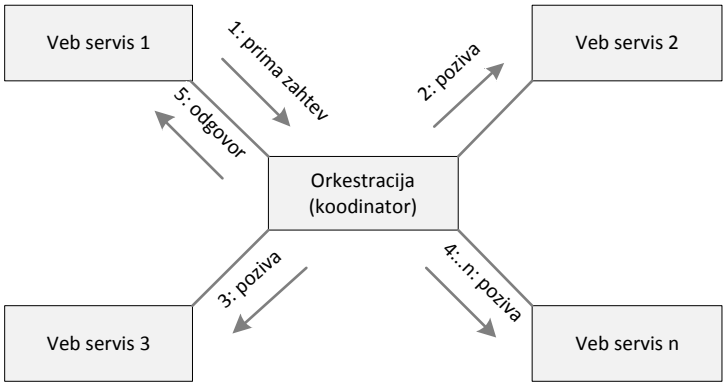
Takođe, često je citirana klasifikacija strategija za kompoziciju servisa, koja obuhvata pet kategorija [58]: statička i dinamička kompozicija; kompozicija zasnovana na modelu; kompozicija zasnovana na poslovnim pravilima; deklarativna kompozicija; automatska i manuelna kompozicija.

Za komponovanje servisa koriste se različite SOA tehnike, koje su najčešće statičke ili dinamičke. Statička kompozicija dešava se za vreme dizajniranja komponovanog servisa. Dinamička kompozicija obezbeđuje zamenu servisa dinamički, u toku izvršavanja. Dinamička kompozicija trenutno predstavlja predmet određenog broja istraživanja u oblasti servisne orijentacije. U radu [59] opisana su četiri različita modela za povezivanje servisa u kompoziciju:

- statičko povezivanje: URL adresa servisa je smeštena u kod;
- dinamičko povezivanje preko reference: URL adrese servisa smeštaju se u promenljive;
- dinamičko povezivanje pomoću pretraživanja: pre svakog pozivanja servisa kreira se upit, na osnovu koga se u registru pronalaze odgovarajući, potrebni servisi;
- dinamički izbor operacija: dinamički se određuje, ne samo servis koji će biti pozvan, već i njegova operacija.

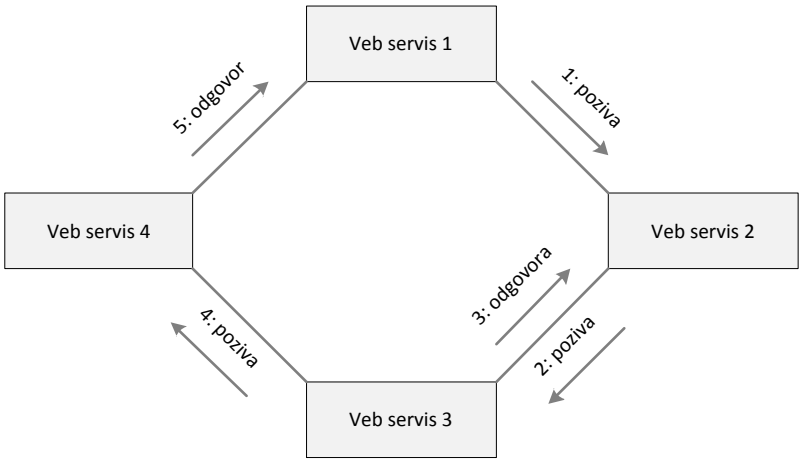
Aktuelni pristupi za kompoziciju veb servisa su orkestracija i koreografija.

Orkestracija opisuje kako veb servisi mogu komunicirati međusobno na nivou poruke. Orkestracija uključuje poslovnu logiku i redosled izvršavanja. Interakcija između veb servisa može povezivati aplikacije u okviru, ili van organizacije. Kao krajnji rezultat dobija se dugotrajni, transakcioni procesni model, sastavljen od više aktivnosti. U orkestraciji, centralni proces ima kontrolu nad uključenim veb servisima i koordinira izvršavanje različitih operacija veb servisa, pri čemu veb servisi ne znaju da su uključeni u kompoziciju i da su deo višeg poslovnog procesa. Jedino centralni koordinator orkestracije ima saznanja o tome, tako da je orkestracija centralizovana sa eksplicitnim definicijama operacija i redosledom pozivanja veb servisa. Orkestracija se uobičajeno opisuje pomoću BPEL jezika.



Slika 2.24: Grafički prikaz orkestracije

Koreografija prati niz poruka, koje mogu uključiti više učesnika i više izvora, poput: kupaca, snabdevača i poslovnih partnera. Umesto specifičnih poslovnih procesa, koje izvršava jedno preduzeće, koreografija je uobičajeno povezana sa razmenom javnih poruka, koja se obavlja između više veb servisa. Koreografija ne zavisi od centralnog koordinatora. Svaki veb servis, uključen u koreografiju, tačno zna kada treba da izvrši svoje operacije i sa kim treba da sarađuje. Koreografija se uobičajeno opisuje korišćenjem Web Services Choreography Description Language (WS-CDL).

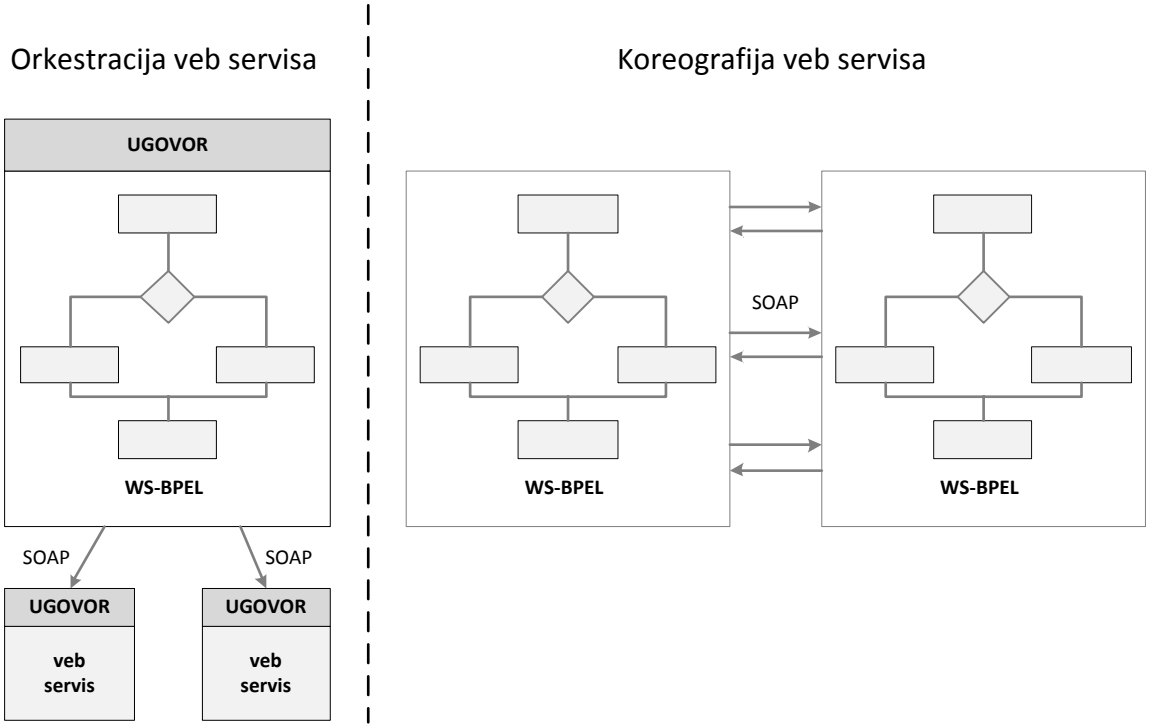


Slika 2.25: Grafički prikaz koreografije

Posmatrajući iz perspektive kompozicije veb servisa za izvršavanje poslovnog procesa, orkestracija ima prednosti u odnosu na koreografiju. Orkestracija je fleksibilnija, mada granica između orkestracije i koreografije nije uvek uočljiva. Orkestracija ima sledeće prednosti [32]:

- tačno se zna ko je odgovoran za izvršavanje celog poslovnog procesa;
- omogućava dodavanje veb servisa, bez njihovog znanja o činjenici da su deo poslovnog procesa;
- u slučaju pojave greške obezbeđuje alternativne scenarije.

Uporedni prikaz orkestracije i koreografije dat je na slici 2.26.



Slika 2.26: Uporedni prikaz orkestracije i koreografije

Iz poslovne perspektive, kompozicija veb servisa ima nekoliko prednosti [60]. Prvo, komponovani servisi omogućavaju preduzećima da smanje obim posla potreban za razvoj softverskih sistema, obezbeđujući na taj način, brzu adaptaciju na promene okruženja. Drugo, razvoj softverskih sistema, zasnovan na veb servisima, smanjuje rizike poslovanja, jer ponovno korišćenje postojećih servisa sprečava pravljenje novih grešaka. Treće, komponovanje veb servisa omogućava smanjenje potrebnih veština i napora za razvoj softverskih sistema. Na kraju, mogućnost da koriste najkvalitetnije servise, koji postoje na tržištu, omogućava preduzećima da povećaju svoj prihod.

3 Interoperabilnost kod servisno orijentisanog elektronskog poslovanja

Cilj poglavlja je da opiše područja istraživanja i metodologiju, koja je primenjena u ovoj disertaciji. Poglavlje uvodi koncept interoperabilnosti servisno orijentisanog razvoja softvera, kojim se bavi istraživanje, a zatim sledi diskusija o pristupima za rešavanje problema interoperabilnosti.

3.1 Koncept interoperabilnosti

Interoperabilnost se može definisati kao sposobnost softverskih sistema, koji se izvršavaju na različitim operativnim sistemima i različitom hardveru, da razmenjuju informacije u skladu sa tehničkim specifikacijama. Polazeći od navedenih karakteristika i mogućnosti interoperabilnosti, proističe da je upravo interoperabilnost jedan od kritičnih problema, koji je potrebno rešiti tokom budućeg razvoja i rasta obima elektronskog poslovanja.

I pored činjenice, da bavljenje pitanjima interoperabilnosti nije novo, diskusije o načinu primene i interpretacije su i dalje veoma aktuelne, a mišljenja su često oprečna. Za različite stručnjake, u zavisnosti od oblasti kojima se bave, pojam interoperabilnost ima druga značenja.

3.1.1 Definicije interoperabilnosti

U literaturi postoje brojne definicije interoperabilnosti. U ovom delu teze biće navedene neke od njih.

IEEE definiše interoperabilnost kao sposobnost dva, ili više sistema, ili komponenti, da razmenjuju informacije i da razmenjene informacije koriste [61].

Interoperabilnost predstavlja sposobnost sistema ili jedinice, da obezbedi ili prihvati usluge drugih sistema ili jedinica, i da tako razmenjenim uslugama, obezbedi efikasan zajednički rad, bez menjanja i degradiranja razmenjenih informacija [62].

U radu [63] interoperabilnost se definiše kao sposobnost međusobne komunikacije sistema i pristupa njihovim funkcionalnostima. Smatra se da interoperabilnost preduzeća obezbeđuje poslovnim entitetima (iz istih, ili različitih organizacija, bez obzira na njihovu lokaciju) mogućnost razmene, ili deljenja informacija, uzajamnim korišćenjem svih raspoloživih funkcionalnosti u distribuiranom i heterogenom okruženju.

TOGAF, Open Group Architecture Framework, definiše interoperabilnost kao [64]:

(1) sposobnost dva ili više sistema, ili komponenti, da razmenjuju i koriste zajedničke informacije, i
(2) sposobnost sistema da obezbede i prihvate usluge drugih sistema i da, tako razmenjene usluge, koriste za efikasan zajednički rad.

Standard ISO 14258 navodi da su dva sistema integrisana, ako postoji detaljni standardni format za sve sastavne komponente [65]. U odnosu na nivo integrisanosti sistema, interoperabilnost je:

- jedinstveni pristup povezivanja sistema, preko zajedničkih meta struktura konstitutivnih modela, kojima se obezbeđuje uspostavljanje semantičke jednakosti, ili
- ujedinjeni pristup, po kome se modeli moraju dinamički prilagoditi.

U saglasnosti sa prethodno navedenim definicijama, interoperabilnost:

- uključuje dva ili više entiteta (sistemi, komponente, jedinice, organizacije);
- omogućava komunikaciju (razmena informacija, usluga);
- ima za svrhu postizanje određenog cilja (da obezbedi efikasan zajednički rad i unapredi način rada).

Po ISO 14258 standardima, vezane za koncepte i pravila za modele preduzeća [65], postoje tri osnovna načina za uspostavljanje interoperabilnosti:

- **Integrisani pristup**: Postoji zajednički format za sve modele, koji mora biti na istom nivou detaljnosti kao i detaljnost modela. Zajednički format nije nužno standard, ali mora biti dogovoren od strane svih učesnika, koji grade sistem.
- **Ujedinjeni pristup**: Postoji zajednički format, ali samo na meta nivou. Meta model nije izvršni entitet, kao kod integrisanog pristupa, ali obezbeđuje sredstvo za semantičku jednakost pri mapiranju modela u sistem.
- **Federalni pristup**: Ne postoji zajednički format. Da bi se uspostavila interoperabilnost, partneri se prilagođavaju "u hodu". Prilikom primene ovog pristupa nijedan partner ne nameće svoje modele, jezike, ili metod rada.

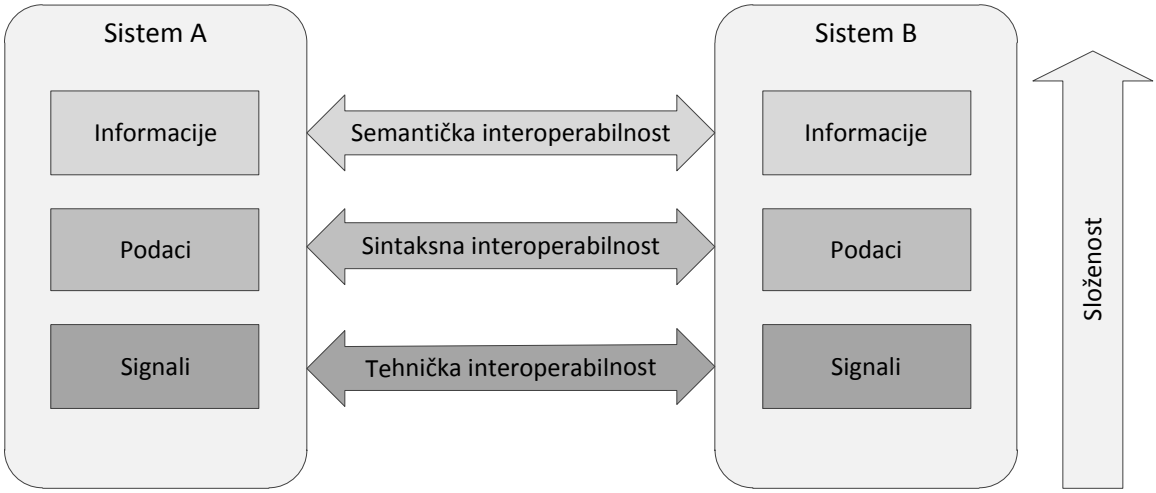
Često se pojam "interoperabilnost" pogrešno izjednačava sa pojmom "integracije sistema" [66]. Integracija u okviru preduzeća odnosi se na: fizičku integraciju (povezivanje uređaja i mašina, preko računarskih mreža); integraciju aplikacija (integracija softverskih aplikacija i sistema baza podataka) i poslovnu integraciju (koordinacija funkcija za upravljanje i nadgledanje poslovnih procesa). Sa stanovišta povezanosti, integrisani sistemi su čvrsto povezani, odnosno njihove komponente su međusobno zavisne i ne mogu biti izdvojene. Za razliku od integrisanih sistema, interoperabilni sistemi su slabo povezani. Dva integrisana sistema implicitno su interoperabilna, međutim, dva interoperabilna sistema ne moraju biti istovremeno i integrisana.

Na osnovu prethodne diskusije, može se uopšteno reći da se pod pojmom interoperabilnost podrazumeva sposobnost dva, ili više sistema, da sarađuju na organizacionom, semantičkom i sintaksnom nivou, uz upotrebu informaciono-komunikacionih tehnologija, sa ciljem da unaprede način rada i poslovanja. U slučaju SOA sistema, interoperabilnost podrazumeva sposobnost više sistema da međusobno efektivno koriste potrebne, raspoložive servise.

3.1.2 Nivoi interoperabilnosti

U ovom delu teze biće predstavljeno nekoliko tipova i aspekata interoperabilnosti. Sa ciljem da se uporede, različiti tipovi interoperabilnosti pozicioniraju se na različite nivoe skale interoperabilnosti. Viša pozicija na skali označava veći stepen postignute interoperabilnosti. Imajući u vidu navedenu mogućnost poređenja različitih tipova i aspekata interoperabilnosti, u literaturi se često govori o nivoima interoperabilnosti.

Evropski centar za interoperabilnost [67] navodi da se interoperabilnost može ostvariti na tehničkom, sintaksnom i semantičkom nivou. Navedeni nivou prikazani su na slici 3.1.



Slika 3.1: Nivoi interoperabilnosti
(prilagođeno po [67])

Tehnička interoperabilnost: Kod ovog nivoa interoperabilnosti postoje komunikacioni protokoli za razmenu podataka između sistema, preko kojih je uspostavljena komunikaciona infrastruktura za razmenu bitova i bajtova. Osnovne mreže i protokoli definisani su nedvosmisleno.

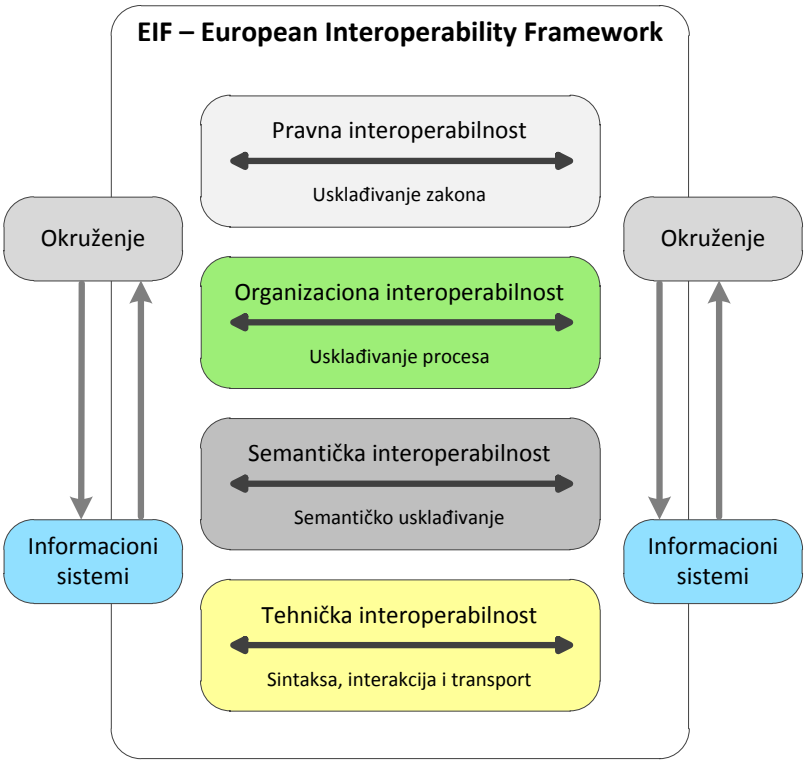
Sintaksna interoperabilnost: Ako su dva, ili više sistema, sposobna da razmenjuju podatke, oni pokazuju sintaksnu interoperabilnost. U osnovi se nalaze definisani formati podataka i komunikacioni protokoli. Primeri za sintaksnu interoperabilnost su XML i SQL standardi. Sintaksna interoperabilnost je preduslov za obezbeđenje višeg nivoa interoperabilnosti.

Semantička interoperabilnost: Obezbeđuje da su dva, ili više računarskih sistema, sposobni da razmenjuju informacije automatski, smisaono i precizno, i da tumače razmenjene informacije, kako bi se proizveli korisni rezultati, saglasno zahtevima krajnjih korisnika. Da bi se postigla semantička interoperabilnost, obe strane moraju da poštuju referentni model za razmenu informacija. Tokom razmene informacija, nedvosmisleno se definiše zahtev u pogledu sadržaja: skup poslatih informacija, u pogledu značenja, mora biti prepoznat i preslikan u skup primljenih informacija.

U poslednje vreme, u radovima se sve češće pojavljuje pojam "pragmatična interoperabilnost" [68] [69]. Ovaj nivo interoperabilnosti odnosi se na način korišćenja informacija, koje se razmenjuju porukama. Na nivou poruke, dogovor o značenju razmenjenih podataka (semantika) i strukturi podataka (sintaksa), nije dovoljan sa se postigne potpuna, efikasna i smislena saradnja. Značajan je i

način korišćenja podataka (pragmatika), odnosno, da li primalac poruke reaguje na primljenu poruku na način kako to pošiljalac poruke očekuje. U većini slučajeva poruke, poslate određenom sistemu, menjaju njegovo stanje, što proizilazi iz činjenice da se poruke uvek šalju sa namerom postizanja nekog efekta. Najčešće se željeni efekat ne postiže slanjem jedne poruke, već slanjem više poruka u određenom redosledu, koji je definisan protokolom interakcije. Zaključuje se da su sistemi pragmatično interoperabilni, kada su njihova očekivanja, o efektu razmenjenih poruka, ista. Problemi pragmatične interoperabilnosti nastaju, kada se željeni uticaj na sistem razlikuje od ostvarenog uticaja.

European Interoperability Framework [70] definiše interoperabilnost na tehničkom, semantičkom, organizacionom i pravnom nivou. U odnosu na [67], tehnička interoperabilnosti se postiže na isti način. U kontekstu EIF okvira, semantička interoperabilnost obuhvata sintaksni i semantički aspekt. Sintaksni aspekt opisuje tačan format informacija koje se razmenjuju, u smislu gramatike, formata i šeme. Semantički aspekt odnosi se na značenje i odnose između elemenata podataka, što uključuje razvoj rečnika za opis razmene podataka, kako bi se osiguralo razumevanje strana, koje međusobno komuniciraju. Organizaciona interoperabilnost, u suštini, podrazumeva integraciju poslovnih procesa i podataka, koji se razmenjuju. Pravna interoperabilnost nalazi se na najvišem nivou. Cilj pravne interoperabilnosti je da obezbedi usaglašenost sa pravnim sistemima država, koje razmenjuju informacije, na primer, u pogledu poštovanja zakona o zaštiti podataka u svim učestvujućim državama.



Slika 3.2: EIF nivoi interoperabilnosti
(prilagođeno po [63])

C4 Interoperability Framework [71] definiše četiri tipa interoperabilnosti, po čijim početnim slovima je okvir i dobio ime:

- *Connection*. Povezivanje je sposobnost informacionih sistema da razmenjuju signale, što se obezbeđuju uspostavljanjem fizičkog kontakta, odnosno konekcijom.
- *Communication*. Komunikacija je sposobnost informacionih sistema da razmenjuju podatke. Da bi se to postiglo, potrebno je da unapred postoje definisani formati i šeme za podatke, prihvaćeni od strane učesnika razmene.
- *Consolidation*. Konsolidacija je sposobnost informacionih sistema da razumeju podatke. Kako bi se u tome uspelo, potrebno je, između informacionih sistema uspostaviti opšte prihvaćeno značenje podataka, na primer, pomoću ontologije. Ovaj tip interoperabilnosti usredređen je na značenje podataka, njihovo tumačenje i semantiku.
- *Collaboration*. Saradnja je sposobnost sistema da deluju zajedno, što za rezultat ima promenu realnog sveta. Da bi uspeli u tome, potrebno je da postoji opšte prihvaćen način sporazumevanja za obavljanje funkcija/servisa/procesa/akcija. Ovaj tip interoperabilnosti usredređen je na proces, ponašanje i korišćenje podataka.

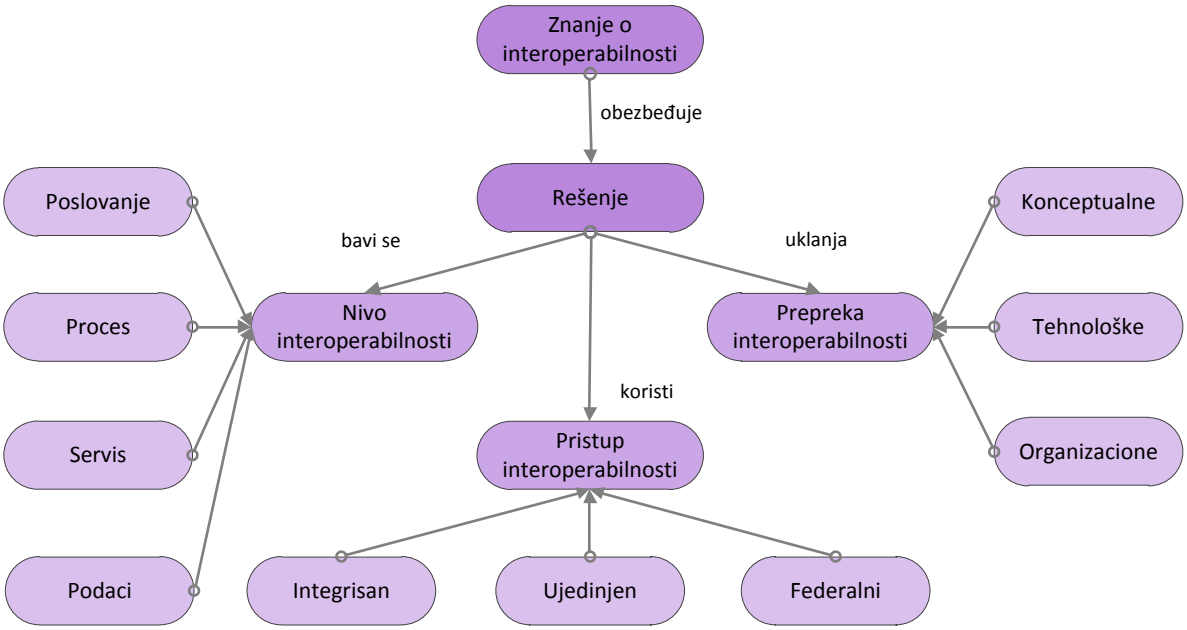
Skup tehnologija, preko kojih se može realizovati svaki od C4IF nivoa interoperabilnosti, prikazan je u tabeli 3.1.

Tabela 3.1: Tehnologije po C4IF nivoima interoperabilnosti
(prilagođeno po [71])

Connection	**Communication**	**Consolidation**	**Collaboration**
• kabl • infrared • bluetooth	• formati podataka • rečnici podataka • SQL • ER šema	• tezaurus • taksonomije • rečnici • RDF šeme • ontologije • tehnologije semantičkog veba	• jezici dijarama toka • BPML • servisne ontologije • SOA • veb servisi • tehnologije semantičkih veb servisa

Interoperabilnost može se posmatrati po nivoima na kojima se odvija [72]:

- **Interoperabilnost podataka**: Odnosi se na mogućnost zajedničkog rada različitih modela podataka (relacionih, hijerarhijskih) i različitih jezika za upite, da pronađu i dele informacije iz heterogenih izvora, koji se nalaze na različitim mašinama, sa različitim operativnim sistemima i sistemima za upravljanje bazama podataka.
- **Interoperabilnost servisa**: Bavi se identifikacijom, sastavljanjem i kreiranjem funkcija od različitih servisa/aplikacija (dizajniranih i implementiranih nezavisno), rešavanjem sintaksnih i semantičkih razlika, kao i pronalaženjem veza sa heterogenim, distribuiranim bazama podataka.
- **Interoperabilnost procesa**: Ima cilj da obezbedi zajednički rad različitih procesa. Često se u preduzeću nekoliko procesa izvršava u međusobnoj interakciji (serijski, ili paralelno). U slučaju umreženih preduzeća, potrebno je da se nađe način na koji se, od internih procesa dva preduzeća, može kreirati zajednički proces.
- **Interoperabilnost poslovanja**: Odnosi se na usaglašen način obavljanja posla između preduzeća, uprkos različitim: načinima odlučivanja; metodama rada; propisima i kulturi preduzeća; komercijalnim pristupima i slično, a na način da posao može biti razvijan i deljen između različitih preduzeća.



Slika 3.3: Osnovni koncepti interoperabilnosti preduzeća
(prilagođeno po [72])

Takođe, u radu [72] smatra se da preduzeća nisu interoperabilna usled interoperabilnih prepreka, koje mogu biti: konceptualne; tehnološke i organizacione. Prepreke konceptualne prirode odnose se na sintaksnu i semantičku različitost informacija, koje se razmenjuju, a tiču se modelovanja: i na poslovnom nivou i na nivou programiranja. Tehnološke prepreke odnose se na nekompatibilnost informacionih tehnologija (arhitektura i platforme, infrastruktura) i tiču se standarda za: predstavljanje; pamćenje; razmenu; obradu i komunikaciju podataka, korišćenjem softverskih sistema. Prepreke organizacione prirode odnose se na definiciju odgovornosti i nadležnosti, ili "ljudskog faktora", koji može biti prepreka za saradnju i obavljanje posla na potreban način.

3.2 Pristupi rešavanju problema interoperabilnosti

Problem interoperabilnosti javlja se kada se dva, ili više nekompatibilnih sistema, nađu u međusobnoj vezi. Postojeći softverski sistemi sastavljeni su od aplikacija, koje su razvijane u različitim periodima, od strane različitih razvojnih timova, koji su najčešće radili bez stvarne koordinacije. Pored toga, za razvoj aplikacija korišćene su različite tehnologije. U većini slučajeva, nove aplikacije kreiraju se korišćenjem, u tom momentu, najnovijih tehnologija, često bez kritičnog odnosa prema neophodnosti implementacije baš najnovijih tehnologija. Zaboravlja se da nova, trenutno aktuelna, tehnologija postaje zastarela kroz određeni period vremena. Takođe, većina ranijih aplikacija nije dizajnirana da komunicira sa drugim aplikacijama. Korišćenje komercijalnih softverskih paketa povećava problem, s obzirom da su dizajnirani da određeni problem rešavaju uopšteno. U toj situaciji, za povezivanje aplikacija potrebno je kreirati prilagođene linkove i interfejse.

Posmatrana na ovaj način, interoperabilnost je cilj koji treba dostići.

Iz praktične perspektive, bez obzira na brojnu literaturu, koja se bavi problemima interoperabilnosti, istraživanja su još uvek u ranoj fazi. Neki od problema, koje je potrebno i dalje rešavati, su:

- Korišćenje standarda ne garantuje uvek postizanje interoperabilnosti.
- Primena principa projektovanja ne garantuje uvek postizanje interoperabilnosti.
- Sveobuhvatno rešavanje zaštite informacija i bezbednosna pitanja u celoj organizaciji, koja su od ključnog značaja za obezbeđivanje pouzdanih elektronskih poslovnih odnosa i saradnje.
- Pitanja izvan domena tehnike, kao što su: komunikacija na kulturnom i ljudskom nivou i interakcije računara i domena specifičnih znanja, dokazano je da mogu postati prepreka u postizanju interoperabilnosti.
- Potreba da se analizira i oceni interoperabilnost sama po sebi.
- Brz razvoj tehnologije može da smanji stepen interoperabilnosti sistema, ukoliko se zahtevi za interoperabilnošću ne razmatraju na nivou dizajna.

Posmatrano iz različitih perspektiva, postoji nekoliko načina za postizanje interoperabilnosti. Opšte prihvaćeni načini za rešavanje problema interoperabilnosti su: standardi; okviri za interoperabilnost; okviri za arhitekturu informacionih sistema preduzeća; semantičke tehnologije, zasnovane na ontologijama; kao i servisna orijentacija.

3.2.1 Standardi i interoperabilnost

Termin "standardi" u disertaciji se koristi u širem smislu značenja i pokriva dokumente, koji obezbeđuju pravila, smernice i karakteristike za aktivnosti i njihove rezultate, sa ciljem postizanja optimalnog stepena reda u datom kontekstu [73].

Primena standarda olakšava postizanje interoperabilnosti sistema elektronskog poslovanja. Međutim, rad na razvoju i sprovođenje standarda ne znači da će interoperabilnost, i pored primene standarda, biti automatski postignuta.

Standardi u oblasti interoperabilnosti kreiraju se da bi osigurali konzistentnost između uređaja, platformi i sistema. Oni promovišu interoperabilnost sistema, tako što obezbeđuju da se proizvodi, kreirani korišćenjem različitih tehnologija, mogu zajedno koristiti [74].

Standard predstavlja sporazum, a u najboljem slučaju, sporazum predstavlja kompromis. Pokušaj da se sve kompanije saglase oko istih procesa, ili rečnika, za sve njihove aktivnosti je Herkulov, ako ne, i Sizifov posao [44]. Moguće je postići saglasnost oko opštih metoda i prihvatljivih razlika za kreiranje i implementaciju poslovnih procesa, mada kompanije često ne žele da smanje svoju konkurentsku prednost, koristeći iste standarde kao i njihovi konkurenti.

Do sada je kreirano više standarda, koji se bave različitim fazama razvoja sistema elektronskog poslovanja, ali je većina njih imala ograničen uspeh. Veliki broj standarda je složen, što je dovelo do nerazumevanja njihovog značenja. Posledice su nekompatibilnost implementacija i pored činjenice da su standardi korišćeni. Problem sa korišćenjem standarda je i što ih je previše. Neki standardi nisu dobro definisani ili se preklapaju sa drugim standardima, koje istovremeno predlažu konkurentske organizacije. Dešavanja oko standarda zbunjuju potencijalne korisnike, i oni odlažu njihovo usvajanje i primenu, bar dok neki standard ne postane šire prihvaćen.

Problem može nastati i što podržavajući jedan standard, kompanija nenamerno sprečava prihvatanje nekog drugog, alternativnog standarda. Standard nema operativni značaj, sve dok se partneri ne

usaglase kako da obavljaju posao na način definisan prihvaćenim standardom. Suština je da je posao, a ne standard, od kritične važnosti. Primena standarda je suštinski važna, jer osim obezbeđenja interoperabilnosti, uvodi pojedina rešenja, koja doprinose razvoju određene industrije. [44]

Povoljna okolnost je da distribuirano računarstvo trenutno prolazi kroz period tranzicije, od vlasničkog pristupa, ka pristupu zasnovanom na standardima. Ipak, veći deo današnje tehnologije još uvek je vlasnički, što uslovljava kupovinu različitih softverskih proizvoda od istog isporučioca, da bi se omogućila integrisanost i kompatibilnost.

Da bi se obezbedila neophodna interoperabilnost sistema elektronskog poslovanja, potrebno je koristiti novi koncept pri definisanja standarda. Postojeća standardizacija bila je prihvatljiva u vreme paketne obrade informacija. Aktuelni standardi su veoma kompleksni. Definišu samo razmenu podataka i način razmene podataka. Umesto toga, potrebno je da se kreiraju standardi, koji će na jednostavan način definisati dokumentovanje: šta, kada i zašto je potrebno da se informacije razmenjuju u realnom vremenu i u elektronski prihvatljivom formatu [75].

Ficdžerald i drugi, predložili su jednu od mogućih kategorizacija standarda [74]:

- *De facto* (ili neformalni) standardi su standardi, koje je priznala industrija, imajući u vidu široku upotrebu, a i pored činjenice da nisu zvanično usvojeni od strane organizacija za postavljanje standarda. U suštini, de facto standardi obezbedili su saglasnost na tržištu, iako nisu zvanično definisani, istraženi i propisani od strane referentne organizacije. De facto standardi uobičajeni su u IT sektoru.
- *De jure* (formalni) standarde definišu organizacije za donošenje standarda. Ovi standardi mogu biti ratifikovani od strane poznatih organizacija za standarde, kao što su International Organization for Standardization (ISO) ili International Electrotechnical Commission (IEC). Proces razvoja formalnih standarda podrazumeva konsenzus i neutralnost isporučioca.
- *Vlasnički* standardi su vlasništvo pojedinaca, ili organizacije. Vlasnici mogu da ograniče prava pristupa i korišćenja.

Veoma je značajno, sa stanovišta kreiranja i korišćenja standarda za interoperabilnost, objavljivanje ISO/IEC 14662 Open-edi referentnog modela [12]. Open-edi referentni model razvijen je prvenstveno sa ciljem da obezbedi okvir za koordinaciju postojećih i razvoj budućih međunarodnih standarda. Namenjen je za međusobnu saradnju organizacija, koje definišu standarde, sa ciljem da osiguraju koherentnost i integrisanje tehnika za modelovanje i opis: servisa, interfejsa servisa i protokola.

Open-edi opisuje značajne aspekte, relevantne za interoperabilnost informaciono-tehnoloških sistema. Referentni model, prikazan na slici 2.1, koji koriste Open-edi strane, ima dva aspekta:

- poslovni aspekt uključuje poslovne informacije, poslovne konvencije, sporazume i pravila između Open-edi strana;
- aspekt informacionih tehnologija, koji se u Open-edi sistemima koristi za podršku izvršavanju poslovnih transakcija.

Open-edi okvir često se u literaturi koristi za prikaz i razvrstavanje postojećih standarda i modela, koji se koriste tokom razvoja aplikacija elektronskog poslovanja [12].

Neki od uočenih problema sa standardima za modelovanje elektronskog poslovanja su [76]:

- Postoji više standarda koji se preklapaju.
- Postoji više organizacija za definisanje standarda.
- Postoje različiti stavovi o proširenju standarda.

U situaciji, kada se određeni standardi izaberu i koriste, dešava se da se ne koriste adekvatno. Razlog su neprecizna tumačenja. Posledica su problemi pri ostvarivanju interoperabilnosti sa drugim sistemima i ograničenja kod njihovog naknadnog korišćenja, prilikom proširenja delova sistema.

Korišćenje standarda može da ima veliki uticaj na interoperabilnost preduzeća. Međutim, mnogi standardi još uvek su na konceptualnom nivou, zbog čega nije moguća jednostavna implementacija na operativnom nivou [77]. Zato je neophodan dalji rad na standardima u oblasti jezika i podržanih platformi, posebno na kreiranju i izvršavanju modela poslovnog procesa.

3.2.2 Okviri za arhitekturu informacionih sistema preduzeća

Kreiranje sistema elektronskog poslovanja, zbog njegove složenosti i potrebe da komunicira sa drugim sistemima, uspešnije će se uraditi ukoliko je kreirana arhitektura informacionog sistema čitavog preduzeća.

Arhitektura, u svom osnovnom značenju, predstavlja strukturu i dizajn sistema, ili proizvoda. Arhitektura, kao disciplina, analizira problem i dekomponuje problem na njegove sastavne elemente. Izgradnja arhitekture informacionih sistema preduzeća podrazumeva razumevanje strukture preduzeća. Poznavanje arhitekture informacionih sistema preduzeća je preduslov za međusobnu saradnju sa okruženjem. Bez jasno definisane arhitekture, preduzeća ne mogu sarađivati efikasno, jer nemaju jasno definisane komponente i granice u okviru kojih, i preko kojih, treba da komuniciraju.

Postoji više definicija arhitekture informacionih sistema preduzeća, a neke od njih su:

- Gartner grupa [78] definiše arhitekuru informacionih sistema preduzeća: Arhitektura informacionih sistema preduzeća je proces prevođenja poslovne vizije i strategije u efektivnu promenu produzeća, preko kreiranja, komunikacije i poboljšanja ključnih principa i modela, koji opisuju buduće stanje preduzeća i omogućavaju njegov razvoj. Arhitektura informacionih sistema preduzeća uključuje ljude, procese, informacije i tehnologije u okviru preduzeća, kao i njihove međusobne relacije i relacije sa eksternim okruženjem.
- Po ANSI/IEEE 1471 standardu [79] arhitektura informacionih sistema preduzeća predstavlja osnovnu organizaciju sistema, koju čine: komponente, njihove međusobne veze i veze sa okruženjem. Sastavni deo arhitekture informacionih sistema preduzeća su i načela, koja regulišu dizajn i razvoj same arhitekture.
- Definicija arhitekture informacionih sistema preduzeća, koju je dao MIT Centar za istraživanje informacionih sistema [80], najbliža je pojmu arhitekture informacionih sistema preduzeća, koja se podrazumeva u ovoj tezi: Arhitektura informacionih sistema preduzeća predstavlja organizacionu logiku za ključne poslovne procese i IT mogućnosti, koji odražavaju integracione i standardizacione zahteve operativnog modela preduzeća.

Arhitektonski domeni, koji daju širi pogled na preduzeće i koje je potrebno modelovati kao deo arhitekture informacionih sistema preduzeća, su: poslovna arhitektura; aplikaciona/sistemska arhitektura; informaciona arhitektura i tehnološka arhitektura.

- *Poslovna arhitektura* opisuje strukturu i ponašanje sistema i pokriva: poslovne ciljeve; poslovne funkcije i mogućnosti; poslovne procese i uloge.
- *Informaciona arhitektura* opisuje strukturu, skladištenje i lokaciju podataka, koje sistem stvara i koristi.
- *Aplikaciona arhitektura* opisuje strukturu i ponašanje aplikacija, koje se koriste u poslovanju, i fokusira se na: njihovu međusobnu komunikaciju; komunikaciju sa korisnicima, kao i korišćenje i stvaranje podataka. Aplikaciona arhitektura ne bavi se internom strukturom podataka.
- *Tehnološka arhitektura* opisuje strukturu i ponašanje tehnološke infrastrukture, i pokriva: hardversku konfiguraciju klijentskih i serverskih čvorova; infrastrukturu aplikacija, koje rade na njima; infrastrukturne servise; protokole i mreže, koje povezuju aplikacije i čvorove.

Sastavni deo kreiranja arhitekture informacionih sistema preduzeća predstavlja izdvajanje i opisivanje poslovnih procesa i njihovo razlaganje na poslovne aktivnosti. Nivo dekompozicije poslovnih procesa zavisiće od faze modelovanja i korisnika, kojima su budući modeli namenjeni. Analizom ciljeva poslovanja i granica sistema, pojedini poslovni procesi uključuju se u sisteme elektronskog poslovanja. Aktivnosti u okviru poslovnog procesa izvršavaju se posredstvom servisa i aplikacija.

Kreiranje arhitekture informacionih sistema preduzeća efikasnije je uz korišćenje okvira, koji pomažu da se: definišu oblasti, koje će arhitektura obuhvatiti, i kategorizuju elementi za isporuku, čime se obezbeđuje organizovan i logičan pristup za arhitekte sistema preduzeća.

Generalno, termin "okvir" (engl. framework) odnosi se na mehanizam za strukturiranje koncepata i aktivnosti, odnosno zadataka, neophodnih za projektovanje i izgradnju sistema. Uobičajena praksa je da se, u mnogim disciplinama, okvir koristi da obezbedi podršku za struktuiranje sistema. Sadržaji okvira nisu ograničeni, već naprotiv, mogu da obuhvate potpuno različite artifakte: odgovarajuće pristupe; standarde; uloge; metode, tehnike i alate. U suštini, okvir može biti posmatran kao alat za organizovanje bilo kog oblika meta podataka preduzeća i da pomogne da se uspostavi interoperabilnost i kontrolišu troškovi razvoja sistema. Okvir za interoperabilnost obezbeđuje logičku strukturu za klasifikaciju i organizovanje kompleksnih informacija, neophodnih za postizanje što je moguće višeg nivoa interoperabilnosti.

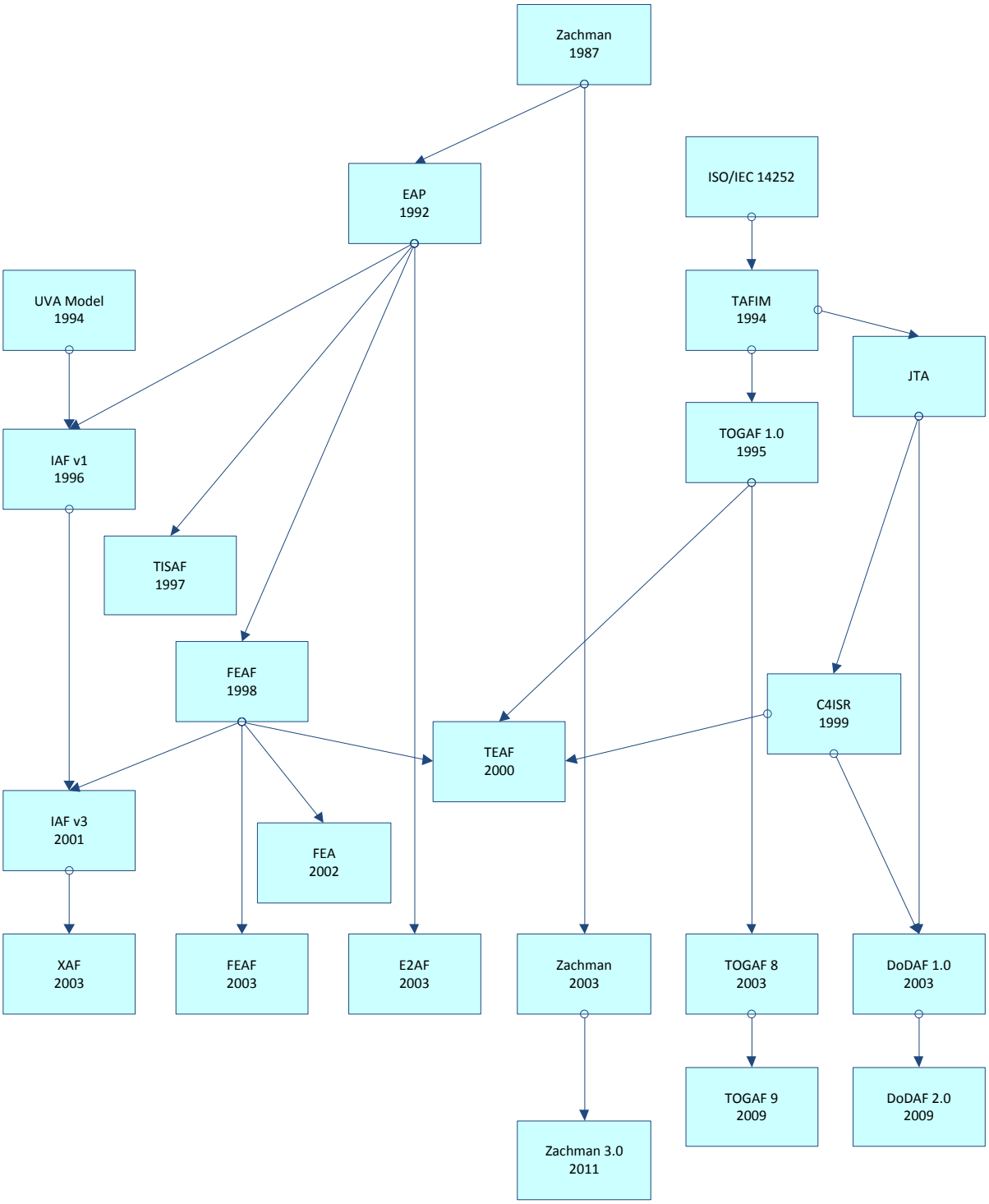
Okvir za arhitekturu informacionih sistema preduzeća mapira sve faze procesa razvoja softvera u okviru preduzeća, njihov međusobni odnos i komunikaciju, sa svrhom ostvarenja postavljenih ciljeva. Okviri pomažu stručnjacima da identifikuju oblasti koje je potrebno razmotriti, kao i odnos između njih. U tom smislu, okvir bi trebalo jasno da identifikuje sve poglede značajne za predstavljanje celokupne arhitekture. Uobičajeno, stručnjaci biraju i razvijaju skup pogleda, koji će omogućiti prikaz sistema iz više uglova.

Okvir za arhitekturu informacionih sistema preduzeća daje smernice za:

- definisanje oblasti, koje će arhitektura obuhvatiti;

- kategorizaciju elemenata za isporuku, koji će obezbediti organizovan i logičan put za pristup korisnika tim elementima.

Uočena uloga i značaj okvira za arhitekturu informacionih sistema preduzeća, doprinele su da bude kreirano više okvira. Dosadašnji razvoj okvira za arhitekturu informacionih sistema preduzeća prikazan je na slici 3.4.



Slika 3.4: Evolucija okvira za arhitekturu informacionih sistema preduzeća
(prilagođeno po [81])

Posmatranjem dijagrama sa slike 3.4 uočava se da većina današnjih okvira ima zajedničku osnovu. Autori novijih okvira nastojali su da unaprede, ili nadograde prethodno kreirane okvire. Pri tome, najčešći polazni okvir bio je Zahmanov okvir. Terminologija, korišćena kod okvira za arhitekturu informacionih sistema preduzeća, menjana je tokom vremena. Međutim, osnovne ideje u vezi funkcionalnosti i nivoa apstrakcije okvira, ostale su skoro iste.

Okvir, sam po sebi, nije metodologija i u tu svrhu se ne može koristiti. Okvir se može koristiti za prilagođavanje i struktuiranje procesa modelovanja, sa ciljem da izgrađeni modeli zadovolje ciljne aspekte i perspektive, uključene u sam okvir. Osim generisanja pojedinačnih modela, proces modelovanja uspostavlja vertikalnu i horizontalnu integraciju i konzistentnost između modela.

Buduća istraživanja i razvoj arhitekture informacionih sistema preduzeća trebalo bi da se zasnivaju na rigoroznim i precizno definisanim ontologijama za: skupove pojmova; relacije i osobine arhitekture informacionih sistema preduzeća [66]. Neophodno je da se, na osnovu opšte saglasnosti, razviju jezici za predstavljanje arhitekture, kao i potrebne metode i metrike, kako bi se predložena arhitektura mogla ispravno opisati, proceniti i porediti. Takođe, potrebno je razviti principe i šablone za dizajn arhitekture, a na osnovu primenjenih i proverenih arhitektonskih rešenja u konkretnoj industriji.

3.2.2.1 Zachman Framework

Većina današnjih okvira nastala je kao unapređenje ili nadogradnja Zahmanovog okvira. Zbog toga, kao i zbog činjenice da je poslužio kao osnova za razvoj okvira predloženog u ovoj tezi, Zahmanov okvir je detaljnije opisan.

Zahman (John Zachman) predstavio je 1987. godine prvi, i do sada najuticajniji, okvir za arhitekturu informacionih sistema preduzeća [82]. Ovaj okvir obezbeđuje pogled na subjekte i modele, potrebne za razvoj kompletne arhitekture informacionih sistema preduzeća. Zahmanov okvir predstavlja jednostavnu logičku strukturu za klasifikaciju i organizovanje opisnih prikaza preduzeća, koji su od značaja za upravljanje i razvoj preduzeća. Zahmanov okvir nezavistan je od bilo koje metodologije ili alata, pa se izbor metodologija i alata mora obaviti pre primene ovog okvira.

Zahmanov okvir imao je veliki uticaj na razvoj drugih okvira za arhitekturu informacionih sistema preduzeća (slika 3.5). Neki od najuticajnijih su: DODAF, TOGAF, FEAF. Zahmanov okvir detaljno je opisan na veb sajtu: http://www.Zachmaninternational.us/index.php/the-Zachman-framework.

Zahmanov okvir široko je korišćen pristup za razvoj i dokumentovanje arhitekture informacionih sistema u celom preduzeću. Kreiranje okvira Zahman je zasnovao na praksi razvoja tradicionalne arhitekture i inženjerstva. Dobijeni okvir po vertikalnoj osi prikazuje višestruke perspektive opšte arhitekture, a po horizontalnoj osi prikazuje klasifikaciju raznih artifakata arhitekture.

Svrha Zahmanovog okvira je da obezbedi osnovnu strukturu, koja podržava: organizaciju, pristup, integraciju, tumačenje, menadžment, razvoj i primenu skupa arhitektonskih predstava informacionih sistema organizacije. Takvi objekti, ili opisi arhitektonskih predstava, uobičajeno se nazivaju "artifakti".

Vertikalna osa, odnosno kolone Zahmanovog okvira, predstavljaju različite aspekte razumevanja preduzeća:

- kolona **Šta** - Opisuje entitete od značaja za poslovni sistem, viđene iz različitih uglova. Entiteti su, u ovom kontekstu, stvari za koje je potrebno obezbediti informacije. Primeri su: oprema, poslovni ciljevi, podaci.
- kolona **Kako** - Definiše funkcije, procese, operacije i aktivnosti, koje se odvijaju u preduzeću, ali u odnosu na različite uglove posmatranja i interesovanja.
- kolona **Gde** - Prikazuje geografske lokacije i međusobne veze aktivnosti u okviru preduzeća. Uključuje glavne geografske lokacije poslovanja, mreže i mesta odvijanja funkcija.
- kolona **Ko** - Predstavlja ljude u preduzeću i informacije za procenu sposobnosti i performansi. Ova kolona bavi se i interfejsom čovek-mašina i relacijama između ljudi i posla, koji obavljaju.
- kolona **Kada** - Prikazuje vreme, ili događaje.
- kolona **Zašto** - Opisuje motivaciju ljudi i preduzeća, odnosno, ciljeve preduzeća, poslovni plan, razloge koji stoje iza donošenja odluka i preduzimanja akcija.

Horizontalna osa, odnosno redovi Zahmanovog okvira obezbeđuju opštu taksonomiju preduzeća i predstavljaju različite uglove posmatranja:

- red **Perspektiva poslovnih izvršilaca** - opisuje viziju, misiju, kontekst, granice, arhitekturu i ograničenja preduzeća. Navodi čime bi preduzeće trebalo da se bavi. Predstavljen je modelom "crne kutije", s obzirom da su vidljivi samo ulazi i izlazi, dok je unutrašnje poslovanje netransparentno.
- red **Perspektiva poslovnog upravljanja** - definiše ciljeve, strategije, strukturu i procese, kojima se podržava misija preduzeća.
- red **Perspektiva arhitekata** - sadrži zahteve, ciljeve, aktivnosti i funkcije, koje čine poslovni model. Model sistema opisuje kako bi sistem trebalo da realizuje svoje funkcije. Unutrašnja struktura modela je transparentna, pa se model sistema svrstava u grupu modela "bela kutija".
- red **Perspektiva inženjera** - bavi se ograničenjima vezanim za ljude, alate, tehnologije i materijale.

	Šta	Kako	Gde	Ko	Kada	Zašto
Perspektiva poslovnih izvršilaca	Identifikacija podataka	Identifikacija procesa	Identifikacija lokacija	Identifikacija odgovornosti	Identifikacija vremenskih ciklusa	Identifikacija motivacije
Perspektiva poslovnog upravljanja	Definicija podataka	Definicija procesa	Definicija lokacija	Definicija odgovornosti	Definicija vremenskih ciklusa	Definicija motivacije
Perspektiva arhitekata	Predstavljanje podataka	Predstavljanje procesa	Predstavljanje lokacija	Predstavljanje odgovornosti	Predstavljanje vremenskih ciklusa	Predstavljanje motivacije
Perspektiva inženjera	Specifikacija podataka	Specifikacija procesa	Specifikacija lokacija	Specifikacija odgovornosti	Specifikacija vremenskih ciklusa	Specifikacija motivacije
Tehnička perspektiva	Konfiguracija podataka	Konfiguracija procesa	Konfiguracija lokacija	Konfiguracija odgovornosti	Konfiguracija vremenskih ciklusa	Konfiguracija motivacije
Perspektiva preduzeća	Instanciranje podataka	Instanciranje procesa	Instanciranje lokacija	Instanciranje odgovornosti	Instanciranje vremenskih ciklusa	Instanciranje motivacije

Slika 3.5: Zahmanov okvir 3.0 za arhitekturu informacionih sistema preduzeća

Opšte karakteristike Zahmanovog okvira su:

- Obezbeđuje dobru klasifikaciju pogleda svih zainteresovanih učesnika i pogleda na preduzeće.
- Okvir, u formi matrice, sadrži 36 ćelija, koje treba popuniti artifaktima (modelima).
- Prikazuje način povezivanja ćelija: horizontalno (između različitih perspektiva) i vertikalno (od koncepta do tehničke implementacije).
- Ne sadrži metodologiju za popunjavanje ćelija.

Uloga Zahmanovog okvira saglasna je nazivu: "okvir". Reč je samo o okviru, koji nije ni proces, ni metod, ni jezik, a nije ni alat. Zahmanov okvir je opšta 6x6 matrična šema, koja se može koristiti za organizovanje i proveru kompletnosti opisnih prikaza različitih, kompleksnih poslovnih sistema, kao što su: preduzeće; korisnici usluga; sistem.

Da bi se u potpunosti razumela arhitektura informacionih sistema preduzeća, neophodno je da se razviju modeli, koji predstavljaju potrebne perspektive i aspekte redova i kolona okvira. Za kreiranje modela trebalo bi koristiti notacije i alate, koji su, u svakom posebnom slučaju, najprikladniji za generisanje informacija, koje bi konkretan model trebalo da obezbedi. To mogu biti: spiskovi entiteta, model diferencijalnih jednačina, skup artifakata modelovan korišćenjem UML ili IDEF notacije, ili jednostavno: paragraf običnog teksta.

U vreme kada je napravljen, Zahmanov okvir je predstavljao novi način za otkrivanje informacija. I sa ove vremenske distance, izgleda toliko prirodno i logično, da je nemoguće ne postaviti pitanje "Zašto neko to nije uradio ranije?". Međutim, s obzirom da svaka od trideset šest ćelija, pomoću kojih je predstavljen okvir, kao rezultat trebalo bi da ima najmanje jedan dokument, razumljive su kritike

koje se odnose na količinu proizvedene dokumentacije, koju je potrebno obezbediti za popunu okvira. Neki istraživači tvrde da izgradnja arhitekture informacionih sistema preduzeća, korišćenjem Zahmanovog okvir nije jednostavan zadatak, jer je veliki broj ćelija prepreka za praktičnu primenljivost okvira. Takođe, smatraju da odnosi između različitih ćelija nisu definisani. Najčešće navođeni problemi kod primene Zahmanovog okvira su [83]:

- nedostatak metodologija, koje pokrivaju sve aspekte okvira;
- nepostojanje čvrstih pravila za povezivanje ćelija okvira;
- nedostatak popularnih notacija za modelovanje svih kolona okvira.

Razlog popularnosti i upotrebe Zahmanovog okvira leži u njegovoj fleksibilnosti i činjenici da može da definiše logičku strukturu bilo kog preduzeća, ili sistema. Za razliku od drugih okvira, razvijenih u oblasti arhitekture informacionih sistema preduzeća, Zahmanov okvir ne nameće metode i ograničenja stručnjacima u pogledu korišćenja unapred definisanih artifakata. Uprkos svim, citiranim nedostacima, Zahmanov okvir može se široko koristi. Trenutno aktuelna verzija 3.0 okvira, objavljena je 2011. godine.

Zahmanov okvir intenzivno se koristi u oblasti arhitekture informacionih sistema preduzeća. Pre svega, Zahmanov okvir je imao značajan uticaj, direktan ili indirektan, na kreiranje drugih okvira za arhitekturu. Zatim, Zahmanov okvir se koristi i kao okvir za opis standarda. Takođe, Zahmanov okvir služi kao referentni model za druge okvire za arhitekturu informacionih sistema preduzeća, odnosno za klasifikaciju artifakata arhitekture informacionih sistema preduzeća.

3.2.2.2 Odnos arhitekture informacionih sistema preduzeća i servisno orijentisane arhitekture

Servisno orijentisana arhitektura je arhitektonski stil. Koristi servise, u obliku softverskih komponenti, koje bi trebalo da obezbede ispunjavanje poslovnih zahteva. Cilj SOA je da, korišćenjem i ponovnom upotrebom servisa, omogući preduzeću brzo i efikasno prilagođavanje okolnostima u poslovnom okruženju.

Pri primeni SOA postoje određeni zadaci, koje je potrebno rešavati u svakom konkretnom slučaju posebno. Jedan od njih je sposobnost uklapanja SOA u arhitekturu čitavog preduzeća. Svrha arhitekture informacionih sistema preduzeća je mapiranje svih faza procesa razvoja softvera u preduzeću, usklađivanje njihovog međusobnog odnosa i komunikacije, i usklađivanje sadašnjih i budućih ciljeva preduzeća.

Postojanje velikog broja okvira za arhitekturu informacionih sistema preduzeća otežava izbor konkretnog okvira, koji će biti korišćen za razvoj, ili opis arhitekture. Većina okvira za arhitekturu informacionih sistema preduzeća kreirana je sa različitim namenama i za određene oblasti, kojima se preduzeća bave. Postojeći okviri zasnivaju se na različitim principima i strukturama. Bez obzira na razlike, većina savremenih okvira za arhitekturu informacionih sistema preduzeća ugrađuje principe servisne orijentacije, čije korišćenje treba da omogući kreiranje fleksibilnih poslovnih procesa, preko ponovnog korišćenja i komponovanja servisa.

Na drugoj strani, SOA stručnjaci koriste okvire za arhitekturu informacionih sistema preduzeća za implementaciju SOA. Arhitektura informacionih sistema preduzeća i servisno orijentisana arhitektura imaju slične poglede u odnosu na IT infrastrukturu. Kako bi se pojedinačni poslovni projekti, pre svega sistemi elektronskog poslovanja, uklopili u poslovanje preduzeća, arhitektura informacionih

sistema preduzeća i servisno orijentisana arhitektura posmatraju celo preduzeće i njegovu organizaciju. Pri tome, okviri za arhitekturu informacionih sistema preduzeća obezbeđuju SOA stručnjacima apstraktniji pogled na IT infrastrukturu i poslovne procese.

Princip servisne orijentacije može da se primenjuje u bilo kom delu arhitekture informacionih sistema preduzeća, ali se najčešće primenjuje za organizaciju softvera, koji podržava poslovne operacije preduzeća. Iz ugla SOA, softver preduzeća je organizovan kao skup softverskih servisa. Servisi su podržani od strane infrastrukture koja, zajedno sa njima, obezbeđuje protok informacija unutar preduzeća i sa okruženjem.

Sličnosti u kompoziciji arhitekture informacionih sistema preduzeća i SOA sistema, sugerišu da bi SOA trebalo da bude kreirana u skladu sa arhitekturom preduzeća (EA). Ovo ne znači da organizacija mora da ima definisanu arhitekturu, kako bi mogla da uspešno realizuje SOA. Međutim, ukoliko je arhitektura informacionih sistema preduzeća definisana, trebalo bi da bude integrisana sa SOA, što olakšava formulaciju SOA.

Tabela 3.2 prikazuje mapiranje između SOA i EA arhitektonskih domena [84]. Mapiranje jasno prikazuje da postoje značajna preklapanja između SOA i EA domena. Međutim, takođe je očigledno da je SOA podskup EA domena.

Tabela 3.2. Mapiranje SOA i EA arhitektonskih domena

Arhitektonski domeni	**EA**	**SOA**
Poslovni	poslovna arhitektura	poslovni procesi
Aplikacije	aplikaciona/sistemska arhitektura	servisi i komponente
Integracija	tehnološka arhitektura	integraciona arhitektura / ESB
Podaci	informaciona arhitektura	arhitektura podataka
Operacije	tehnološka arhitektura	kvalitet servisa, sigurnost, nadgledanje i infrastruktura

U radu [84] autori identifikuju sličnosti i razlike između SOA i EA arhitektonskih domena.

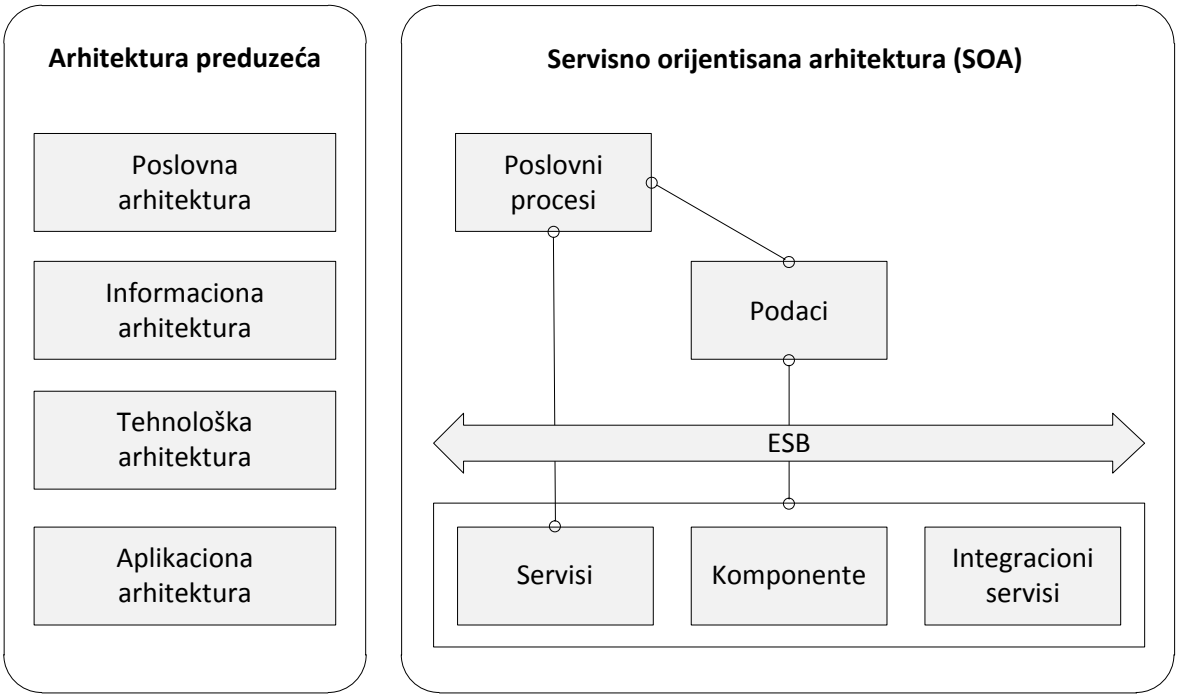
Sličnosti između SOA i EA arhitektonskih domena su:

- bave sličnim arhitektonskim domenima;
- cilj im je da usklade IT sa poslovanjem;
- koriste ulaze zasnovane na poslovnim ciljevima;
- zahtevaju slične strategije i planiranje aktivnosti.

Razlike između SOA i EA arhitektonskih domena su:

- EA se fokusira na definisanje poslovnih komponenti, dok se SOA fokusira na poslovne servise.
- EA se bavi okvirima za aplikacije i preduzeća, dok je predmet SOA modelovanje servisa.
- EA se bavi infrastrukturom na nivou preduzeća, uključujući servere, baze podataka, dok se SOA bavi infrastrukturom koja podržava servise, kao što je ESB.
- EA se bavi šablonima za integraciju na nivou preduzeća. SOA omogućava integraciju zasnovanu na korišćenju servisa.

Relacije između EA i SOA ilustrovane su slikom 3.6.



Slika 3.6: Odnos EA i SOA
(prilagođeno po [51])

Autori u radu [44] definišu odnos arhitekture informacionih sistema preduzeća i servisne orijentacije preko mapiranja SOA u Zahmanov okvir. Polazeći od činjenice da SOA predstavlja arhitektonski stil za izgradnju rešenja, logično polazište za primenu Zahmanovog okvira na SOA je presek kolone "Funkcija" (Kako?) i reda "Logički model sistema" (Perspektiva arhitekata). Međutim, SOA nije samo arhitektonski pristup, koji se bavi funkcionalnim aspektima sistema. SOA obuhvata poslovne procese, omogućene kompozicijom servisa i uslovljava način na koji preduzeća dele i predstavljaju informacije (preko semantičkih, logičkih i fizičkih modela podataka). SOA, istovremeno, definiše i način na koji mreža podržava aplikacije. Na osnovu navedenih karakteristika SOA, može se zaključiti da SOA, predstavljena preko Zahmanovog okvira, pored ćelije za arhitekturu aplikacije, pokriva i osam susednih ćelija, kao što je to predstavljeno na slici 3.7.

Pristup, po kome se popunjava samo deo ćelija Zahmanovog okvira je sasvim ispravan. Blumberg i drugi [44] ističu da samo mali broj preduzeća ima uslove da popuni svih 36 ćelija okvira. U suštini, potpuno popunjen Zahmanov okvir predstavlja granični slučaj, koji omogućava arhitektama preduzeća da se bave različitim aspektima svog preduzeća. Pri tome, redosled popunjavanja ćelija nije uvek obavezujući. Izvorno, ne postoji jedan jedini ispravan put kroz Zahmanov okvir, već arhitekte preduzeća trebalo bi da pronađu svoje sopstvene puteve. Proizilazi zaključak da se sistem, zasnovan na servisnoj orijentaciji, može razvijati korišćenjem Zahmanovog okvira.

	Šta	Kako	Gde	Ko	Kada	Zašto
Perspektiva poslovnih izvršilaca	Identifikacija podataka	Identifikacija procesa	Identifikacija lokacija	Identifikacija odgovornosti	Identifikacija vremenskih ciklusa	Identifikacija motivacije
Perspektiva poslovnog upravljanja	Definicija podataka	Definicija procesa	Definicija lokacija	Definicija odgovornosti	Definicija vremenskih ciklusa	Definicija motivacije
Perspektiva arhitekata	Predstavljanje podataka	SOA	Predstavljanje lokacija	Predstavljanje odgovornosti	Predstavljanje vremenskih ciklusa	Predstavljanje motivacije
Perspektiva inženjera	Specifikacija podataka	Specifikacija procesa	Specifikacija lokacija	Specifikacija odgovornosti	Specifikacija vremenskih ciklusa	Specifikacija motivacije
Tehnička perspektiva	Konfiguracija podataka	Konfiguracija procesa	Konfiguracija lokacija	Konfiguracija odgovornosti	Konfiguracija vremenskih ciklusa	Konfiguracija motivacije
Perspektiva preduzeća	Instanciranje podataka	Instanciranje procesa	Instanciranje lokacija	Instanciranje odgovornosti	Instanciranje vremenskih ciklusa	Instanciranje motivacije

Slika 3.7: Mapiranje SOA u Zahmanov okvir
(prilagođeno po [44])

Poređenjem EA, BPM i SOA uočavaju se određena preklapanja njihovih koncepata, aktivnosti, procesa i ishoda. Po članku [85], najveća vrednost se može dobiti konvergencijom njihovih arhitektura. Pojednostavljeno rečeno, u servisno orijentisanim okruženjima, rešenja zasnovana na SOA platformi obezbeđuju dizajn IT rešenja, BPM obezbeđuje optimizaciju procesa i okvir za razvoj poslovnih rešenja, a EA obezbeđuje i upravlja implementacijom arhitekture za čitavo preduzeće.

3.2.3 Okviri za interoperabilnost

Okvir za interoperabilnost obezbeđuje logičku strukturu za klasifikaciju i organizovanje informacija, neophodnih za postizanje određenog nivoa interoperabilnosti. Okvir za interoperabilnost deo je arhitekture informacionih sistema preduzeća, i bavi se integracijom i povezivanjem: elemenata unutar sistema i elemenata jednog sistema sa elementima drugih sistema.

Nekoliko istraživačkih inicijativa o interoperabilnosti, predložili su okvire razmatrajući interoperabilnost iz različitih uglova. Najpotpunije su predstavljeni sledeći okviri za modelovanje interoperabilnosti: ATHENA Interoperability Framework (AIF), European Interoperability Framework (EIF) i Interoperability Research for Networked Enterprises Applications and Software - Network of Excellence (INTEROP NoE). Navedeni okviri, koji su deo istraživanja o interoperabilnosti u okviru Evropske unije, ukratko su opisani u nastavku teze.

3.2.3.1 Projekat INTEROP NoE

Osnovni cilj projekta Interoperability Research for Networked Enterprises Applications and Software - Network of Excellence [86] je struktuiranje i oblikovanje evropske istraživačke aktivnosti, vezane za interoperabilnost aplikacija i softvera preduzeća, kao i formiranje stalne, evropske istraživačke zajednice, koja će uticati na formiranje i promenu postojećih standarda i pravila u oblasti interoperabilnosti. Najznačajniji rezultati trogodišnjeg rada na ovom projektu obuhvataju mape znanja i okvir za interoperabilnost.

Mapa znanja predstavlja okvir za klasifikaciju informacija. Obuhvata oblasti, povezane sa sledećim kategorijama: ontologije; modelovanje preduzeća i arhitekturu i platforme.

Tabela 3.3: Kategorije mape znanja
(prilagođeno po [86])

Ontologije	**Modelovanje preduzeća**	**Arhitektura i platforme**
Predstavljanje ontologija i zaključivanje	Umreženo, prošireno, virtualno i integrisano preduzeće	Razvoj tehnologija i arhitektura
Izrada ontologija i upravljanje		Standardi za integraciju preduzeća
Interoperabilnost ontologija	Jezici za modelovanje	Arhitektura informacionih sistema preduzeća Okviri za elektronsko poslovanje
Servisi za interoperabilnost aplikacija preduzeća zasnovani na ontologijama	Poslovni proces	Servisi
	Metodologije za izgradnju modela preduzeća	Upravljanje i izvršavanje poslova

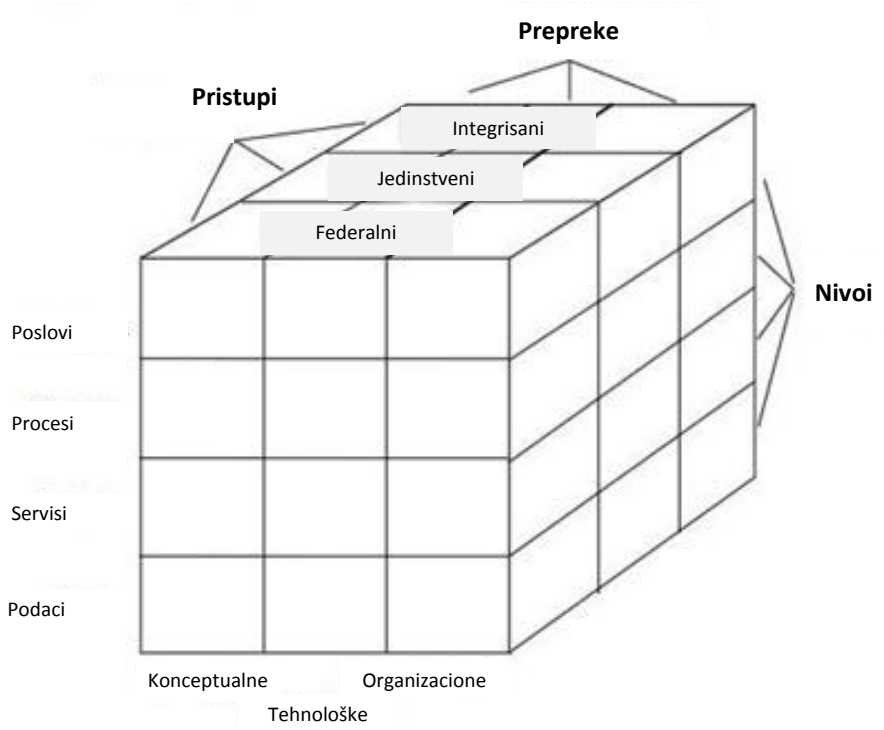
Po INTEROP NoE, okvir za interoperabilnost definiše tri osnovne dimenzije: dimenziju prepreka za interoperabilnost; dimenziju nivoa u preduzeću na kome se može uspostaviti interoperabilnost i dimenziju pristupa interoperabilnosti. Svaki presek navedenih dimenzija predstavlja novo područje za istraživanje interoperabilnosti.

Smatra se da organizacije nisu interoperabilne zbog: konceptualnih, tehnoloških i organizacionih prepreka.

Konceptualne prepreke vezane su za sintaksne i semantičke razlike između informacija, koje se razmenjuju. Konceptualne prepreke prisutne su pri modelovanju na visokom nivou apstrakcije. Tehnološke prepreke odnose se na nekompatibilnost informacionih tehnologija. Opisuju probleme, koji se tiču standarda za predstavljanje, pamćenje, razmenu, obradu i komunikaciju podataka, prilikom korišćenja računara. Organizacione prepreke odnose se na definisanje odgovornosti u organizaciji i nekompatibilnost organizacione strukture.

Definisani nivoi za uspostavljanje interoperabilnosti zasnivaju se na tehničkoj arhitekturi, koja je razmatrana u okviru projekta ATHENA. Odnose se na interoperabilnost: podataka; servisa; procesa i poslova.

Pristupi interoperabilnosti, po INTEROP NoE okviru, mogu biti: integrisani (postoji zajednički format za sve modele); jedinstveni (postoji zajednički format, ali samo na meta nivou) i federalni (ne postoji zajednički format, što implicira postojanje zajedničke ontologije).



Slika 3.8: INTEROP NoE - tri osnovne dimenzije

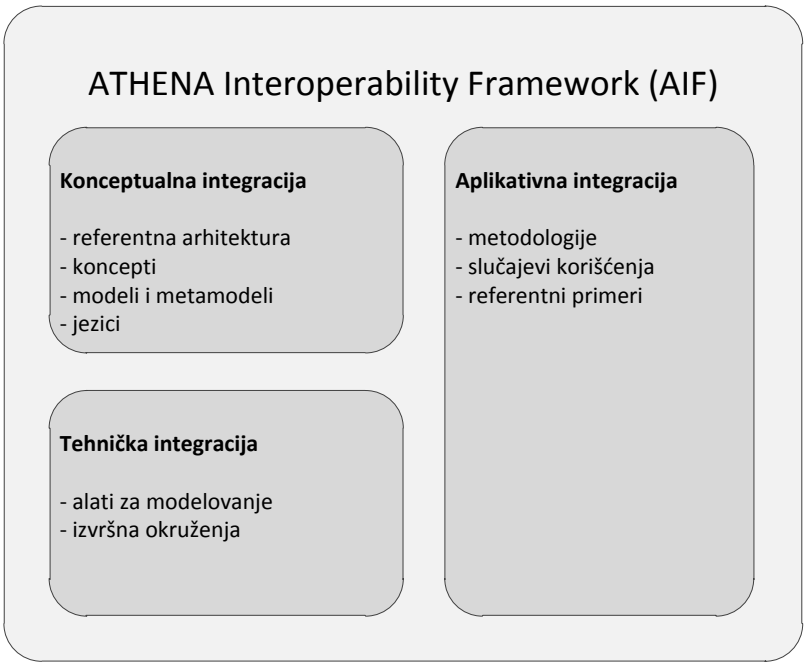
3.2.3.2 Projekat ATHENA

Učesnici projekta ATHENA [87] ističu originalnost projekta, koja je rezultat multidisciplinarnog pristupa. Tokom izrade projekta objedinjene su tri oblasti istraživanja, koje se bave razvojem interoperabilnih aplikacija i softvera organizacije:

- oblast arhitekture i platformi: za obezbeđenje okvira za implementaciju;
- oblast modelovanja organizacije: za definisanje interoperabilnih zahteva i podršku implementaciji rešenja;
- oblast ontologije: za identifikaciju semantičke interoperabilnosti u okviru organizacije.

ATHENA okvir za interoperabilnost, prikazan na slici 3.9, predlaže rešavanje pitanja interoperabilnosti na tri nivoa: konceptualnom; tehničkom i aplikativnom.

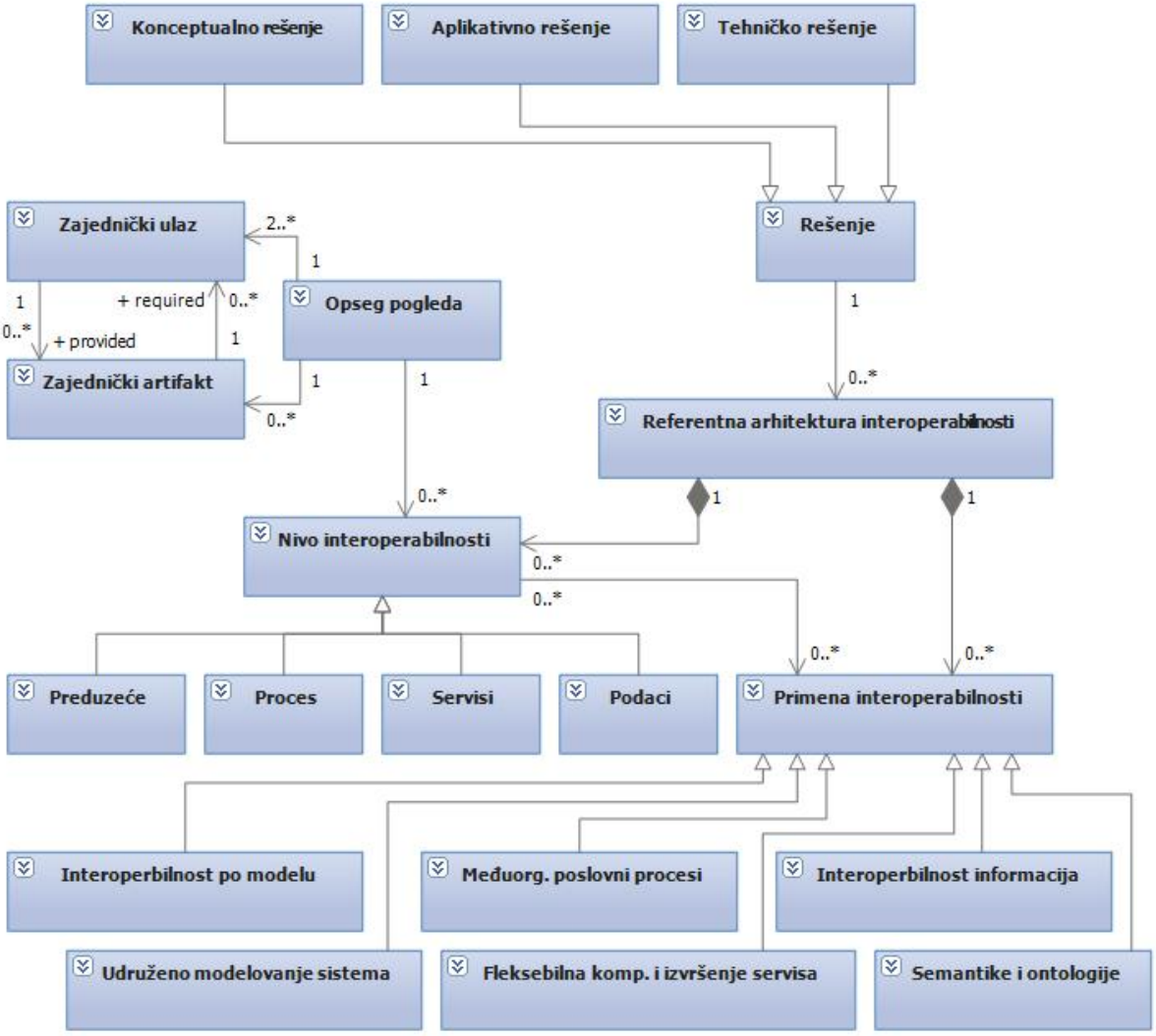
Konceptualni elementi su: referentne arhitekture; modeli i meta modeli; relacije između modela; jezici za modelovanje i koncepti. Konceptualni elementi koriste se za izgradnju konkretnog modela preduzeća. Tehnički nivo fokusira se na razvoj softvera i izvršna okruženja. ATHENA okvir definiše tehničku arhitekturu, koja obezbeđuje razvojne alate i izvršne platforme za integraciju procesa, servisa i informacija. Integracija aplikacija usmerena je na metodologije, standarde i modele određenih domena. Obezbeđuje smernice, principe i obrasce, koji se mogu koristiti za rešavanje problema interoperabilnosti.



Slika 3.9: Struktura ATHENA okvira za interoperabilnost
(prilagođeno po [87])

Referentna arhitektura za interoperabilnost odnosi se na skup nivoa za interoperabilnost i skup pristupa za interoperabilnost. Koncept "pristup interoperabilnosti" obuhvata šest tehnoloških pristupa, nastalih kao rezultat istraživačkih aktivnosti u ATHENA projektu, a koji podržavaju operacije na četiri nivoa interoperabilnosti:

- Interoperabilnost na nivou preduzeća/poslovnom nivou posmatra se kao organizaciona i operaciona sposobnost preduzeća da sarađuje sa drugim, eksternim organizacijama, uprkos različitih: načina za obavljanje posla, zakona, kulture i komercijalnih pristupa.
- Interoperabilnost na nivou procesa je sposobnost različitih procesa da rade zajedno. Proces definiše redosled izvršavanja funkcija u skladu sa specifičnim potrebama saradnje.
- Interoperabilnost na nivou servisa odnosi se na identifikovanje, komponovanje i izvršavanje različitih aplikacija, koje su dizajnirane i implementirane nezavisno.
- Interoperabilnost na nivou podataka odnosi se na upravljanje, razmenu i obradu različitih dokumenata, poruka i struktura.



Slika 3.10: AIF konceptualni model

ATHENA okvir bavi se struktuiranjem problema interoperabilnosti sa aspekata posla, znanja, tehnologije i semantike. Kao i INTEROP NoE, ATHENA okvir koristi pristup zasnovan na preprekama, sa ciljem da se otklone nekompatibilnosti na različitim nivoima interoperabilnosti: organizacionom, konceptualnom i tehnološkom nivou.

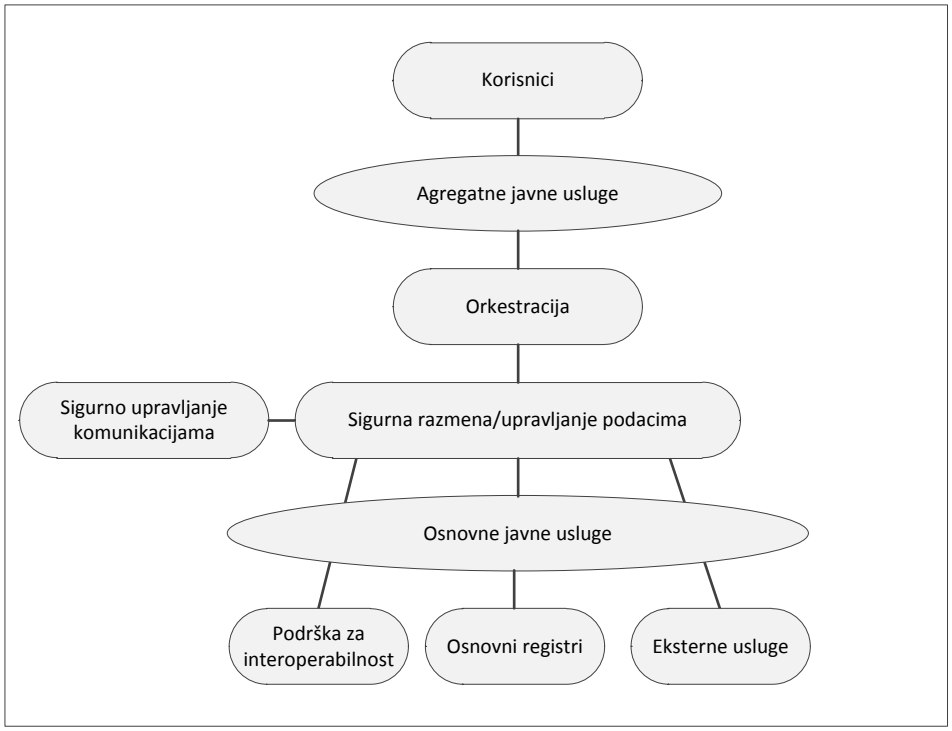
3.2.3.3 European Interoperability Framework

European Interoperability Framework [70] postavlja skup preporuka i smernica, koje bi trebalo da koriste države članice za razmenu podataka između javnih uprava, kao i razmenu podataka javnih uprava sa građanima i organizacijama. EIF verzija 1. publikovana je 2004. godine. Razvoj ovog okvira nastavljen je u okviru Interoperability Solutions for European Public Administrations (ISA) programa. EIF verzija 2. predstavljena je u decembru 2010. godine.

EIF okvir definiše interoperabilnost kao sposobnost ICT sistema i poslovnih procesa, koje oni podržavaju, da razmenjuju podatke i omogućavaju deljenje informacija i znanja.

Glavna karakteristika, verzije 1. EIF okvira, je čvrst stav prema otvorenim standardima i promovisanje otvorenog koda, sa ciljem da se ravnopravno tretira u odnosu na vlasničke kodove.

EIF verzija 2. definiše okvir za interoperabilnost kao dogovoreni pristup interoperabilnosti za organizacije, koje žele da sarađuju pri pružanju zajedničkih, javnih usluga, primenom u svom delokrugu rada. Okvir navodi niz zajedničkih elemenata, kao što su: rečnici; koncepti, principi i pravila; uputstva, preporuke, standardi, specifikacije i prakse. U odnosu na verziju 1, pored prethodno definisanih: semantičkog, tehničkog i organizacionog, dodat je novi nivo interoperabilnosti - pravni nivo (slika 3.2). Pojam "otvorenog standarda" više se ne koristi u EIF okviru, zbog posebnog značaja, koji ovaj pojam ima u EU. U EU, samo tehnička specifikacija, priznata od strane organa za standardizaciju, može se nazvati standardom. Međutim, mnogi ICT sistemi oslanjaju se na korišćenje specifikacija, razvijenih od strane drugih organizacija, kao što su konzorcijumi i forumi. Da bi se obuhvatila oba pojma, uveden je termin "formalizovana specifikacija". Takođe, u okviru konceptualnog modela za javne usluge, predlaže se da princip organizacije za dizajniranje evropskih, javnih usluga bude usredsređen na osnovne usluge, koje mogu biti povezane sa ciljem formiranja "agregatne usluge". Agregatne usluge trebalo bi da doprinesu da se u budućnosti uspostave i druge evropske, javne usluge.



Slika 3.11: Osnovni elementi EIF konceptualnog modela za javne usluge

3.2.4 Semantička interoperabilnost

Potpunu interoperabilnost između softverskih sistema veoma je teško postići. Trenutno, zahvaljujući razvijenosti standarda za veb servise, postignuta je sintaksna interoperabilnost. Sintaksna interoperabilnost dovoljna je za fizičku razmenu informacije, ali nedovoljna je za razumevanje informacija od strane različitih sistema. Zato se danas obavlja veliki broj istraživanja drugih tipova interoperabilnosti.

Semantička interoperabilnost, između servisa u SOA, ne razlikuje se značajno od razmene poruka, izraženih prirodnim jezikom između ljudi. U radu se podrazumeva da je semantička interoperabilnost

između dva sistema postignuta, ukoliko sistemi imaju sporazum o značenju poruka, koje razmenjuju među sobom. Semantička interoperabilnost suštinski je značajna pri primeni SOA, jer korisnici i kreatori servisa trebalo bi da razmenjuju informacije, koje će obe strane razumeti i koje se mogu koristiti za postizanje određenih mogućnosti. Nerazumljivi podaci nisu korisni, i bez obezbeđene semantičke usaglašenosti, podaci su samo niz simbola, koji nemaju smisla. Bez razumevanja značenja razmenjenih informacija, korisnici i kreatori servisa mogu ih pogrešno protumačiti, što može dovesti do neželjenih efekata u poslovanju.

Kreiranje SOA arhitekture, pomoću postojećih servisa, još uvek zahteva značajno učešće ljudi, za pronalaženje i korišćenje odgovarajućih servisa. Potreba za ljudskom intervencijom može se delimično pripisati činjenici da se standardi, koji se uobičajeno koriste kod opisa servisa, fokusiraju na sintaksni aspekt interfejsa servisa, izostavljajući podršku za pronalaženje i korišćenje servisa sa potrebnom funkcionalnošću. U klasičnom SOA scenariju diveloperi, prilikom kreiranja sistema korišćenjem SOA, uobičajeno traže dostupne servise, kako u okviru, tako i van svog preduzeća. Za identifikaciju servisa kandidata koriste se ključne reči i funkcija pretraživanja. Pre korišćenja servisa, neophodna je evaluacija sintaksnog opisa od strane ljudi. U mnogim situacijama, neophodno je uspostaviti dodatnu komunikaciju sa vlasnikom servisa, kako bi se razjasnile funkcionalnosti i značenja informacija, koje će se razmenjivati. Takođe, pre nego što se potrebni servisi izaberu i počnu da koriste, potrebno je obaviti testiranje poziva servisa. Navedene i potrebne aktivnosti, zahtevaju dosta rada, a realizovano rešenje je prilično statično, s obzirom da ograničava mogućnost brzog prilagođavanja promenama, što bi trebalo da bude ključna karakteristika i prednost SOA pristupa.

U slučaju klasične SOA, potreba da se funkcionalnost postojećih servisa poboljša, ili nedostupnost trenutno korišćenih servisa, zahteva dodatno angažovanje ljudi. Cilj semantičke SOA je dodavanje funkcija, koje će pomoći prevazilaženje navedenih ograničenja i koje će obezbediti mehanizam za automatizaciju poslova, koji trenutno zahtevaju ljudsku intervenciju. Ključno ograničenje klasične SOA predstavljaju standardi za opis veb servisa koji, osim jednostavnog opisa interfejsa, pružaju veoma malo informacija o servisu.

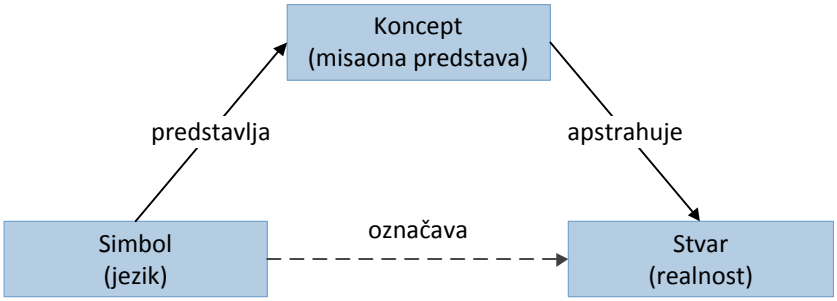
Semantička interoperabilnost trebalo bi da obezbedi da određeni simbol ima isto značenje za sve sisteme, koji koriste taj simbol. Problemi kod semantičke interoperabilnosti najčešće nastaju: kada različiti sistemi koriste različite koncepte, koje se odnose na iste stvari u realnom svetu; ili kada koriste iste koncepte za različite stvari u realnom svetu, ili odnose između njih. U osnovi, razlozi za postojanje problema pri uspostavljanju semantičke interoperabilnosti su sledeći:

- različiti konteksti (baze podataka, ontologije);
- različiti jezici za predstavljanje i logiku;
- isti koncepti, a različita imena, ili ista imena, a različiti koncepti;
- različiti pristupi konceptima (na primer, podklase nasuprot vrednosti svojstava);
- različiti nivoi detaljnosti;
- različiti domeni, koji se preklapaju.

To znači da su problemi kod semantičke interoperabilnosti posledica različitih apstrakcija, istog entiteta iz realnog sveta, ili različitog predstavljanja istog koncepta.

3.2.4.1 Pojam semantike

Semantika je nauka o značenju. U kontekstu informacionih sistema, semantika se definiše kao preslikavanje objekata iz stvarnog sveta u objekte informacionog sistema. Međusobni odnos može se predstaviti trouglom značenja, koji šematski prikazuje odnos između stvari, koncepata i simbola. Trougao značenja predstavljen je na slici 3.12.

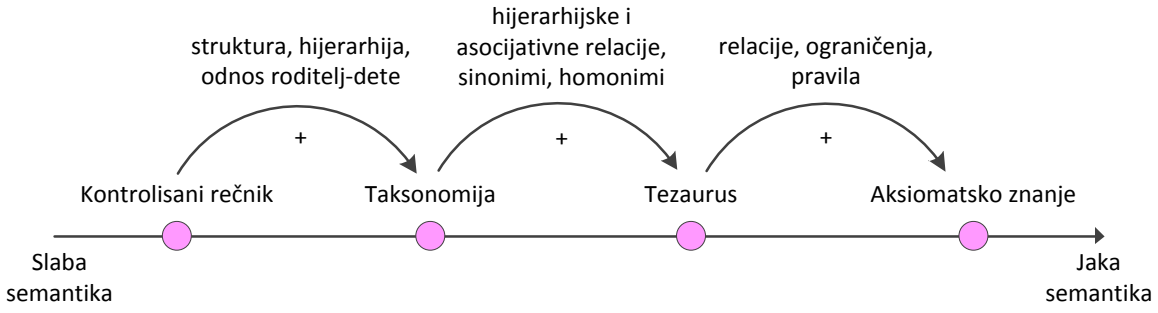


Slika 3.12: Trougao značenja

Metafore, smeštene u temenima trougla, predstavljaju:

- Stvar - bilo koji entitet (ili relacija između dva, ili više entiteta) u realnom svetu.
- Koncept - apstrakcija entiteta iz realnog sveta, sličnih karakteristika.
- Simbol - predstavlja koncept, opisan pomoću određenog jezika i odnosi se na stvari iz realnog sveta. Simbolom se može predstaviti ideja, fizički entitet ili proces.

Predstavljanje znanja oslanja se na ontologije - formalno iskazane prezentacije znanja iz domena interesovanja. Koncept ontologije preuzet je iz filozofije. Ontologija je u filozofiju uvedena sa ciljem da na najopštiji način identifikuju vrste stvari, koje postoje i da, kako se očekuje, predstavi suštinu znanja [88]. Ontologije mogu da obezbede različite nivoe formalizacije, kao što je prikazano na slici 3.13.



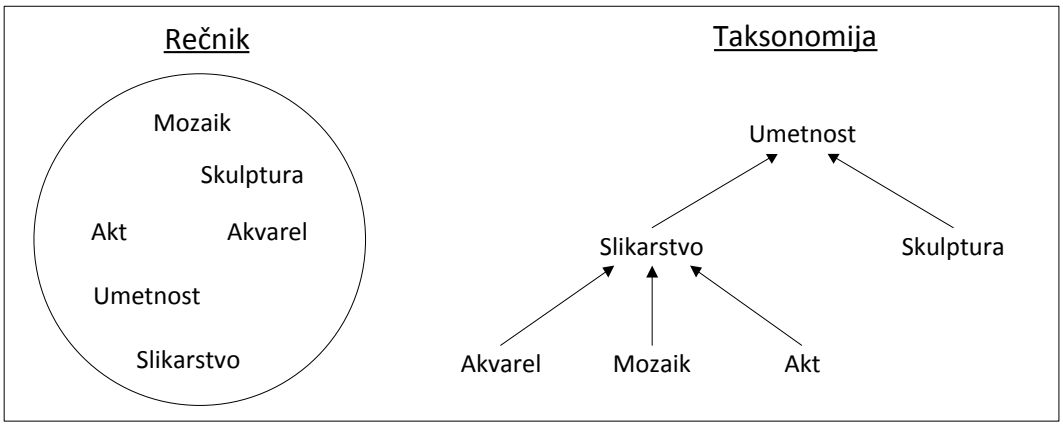
Slika 3.13: Nivoi semantike
(prilagođeno po [89])

Načini za modelovanje i organizovanje ontologija su: kontrolisani rečnici, taksonomije, tezaurusi i aksiomatska znanja.

Kontrolisani rečnici - Predstavljaju najslabiju formu ontologije. Sastavljeni su od spiska eksplicitno nabrojanih pojmova, kao što su: reči; fraze ili oznake. Svi pojmovi iz kontrolisanog rečnika trebalo bi da imaju jedinstveno i nedvosmisleno značenje. Kontrolisani rečnik je najjednostavniji način za opis

meta podataka i intenzivno se koristi za klasifikaciju pojmova. Kontrolisani rečnici ograničavaju izbor pojmova na dogovoreni, nedvosmisleni skup pojmova. Osnovni cilj kontrolisanog rečnika je da spreči korisnike da definišu sopstvene pojmove, koji mogu biti dvosmisleni, besmisleni ili pogrešno napisani.

Taksonomije - Predstavljaju predmetno zasnovane klasifikacije, koje hijerarhijski, uobičajeno u obliku drveta, organizuju pojmove u kontrolisanom rečniku. Hijerarhija u taksonomiji definiše odnos roditelj-dete.



Slika 3.14: Primer rečnika i taksonomije

Tezaurus - Predstavlja umreženu kolekciju pojmova iz kontrolisanog rečnika, sa konceptualnim relacijama između pojmova. Tezaurus je proširenje taksonomije, jer dozvoljava kreiranje i drugih relacija između hijerarhijski organizovanih pojmova.

Aksiomatsko znanje - Predstavlja najjaču formu ontologije. Slično kao i tezaurusi, aksiomatsko znanje uvodi specifikaciju semantičkih relacija između termina u kontrolisanim rečnicima, kao i formalna pravila o izgradnji složenih termina i odnosa. Uvedena formalna pravila obezbeđuju izvođenje novih znanja i formalno rezonovanje.

U radu [90] navode se tri oblasti, koje su pogodne za korišćenje ontologija:

- pri komunikaciji između ljudi;
- za postizanje interoperabilnosti između softverskih sistema;
- za poboljšanje dizajna i kvaliteta softverskih sistema.

Ontologija obezbeđuje način za podršku interoperabilnosti različitih sistema u distribuiranim i heterogenim okruženjima. Interoperabilnost različitih sistema zasniva se na istim ili različitim ontologijama. Interoperabilnost može se lakše realizovati u slučaju kada različiti sistemi koriste istu ontologiju. Pri tome jedan sistem koristi ontologiju za opis svojih zahteva i odgovora, a ostali sistemi istu ontologiju koriste za razumevanje dobijenih zahteva. Međutim, u većini slučajeva, sistemi koriste različite ontologije za opis svojih elemenata, pri čemu ne postoji standard za njihovu prezentaciju. Iz tog razloga, jedna od važnih aktivnosti je semantičko ujedinjenje koncepata, čiji je cilj unapređenje interoperabilnosti. Integracija ontologija može se izvršiti na nekoliko načina [63]:

- mapiranjem (uspostavljanje pravila preslikavanja između koncepata dve ontologije);
- poravnanjem (uspostavljanje dogovora između dve, ili više ontologija);
- transformacijom (promena strukture ontologije, zbog usklađivanja sa drugom ontologijom);
- fuzijom (izgradnja nove ontologije, objedinjavanjem dve, ili više postojećih).

Semantička interoperabilnost, zasnovana na ontologijama, veoma je aktuelna i svakako će biti predmet i budućih istraživanja.

3.2.4.2 Modeli za semantičku interoperabilnost

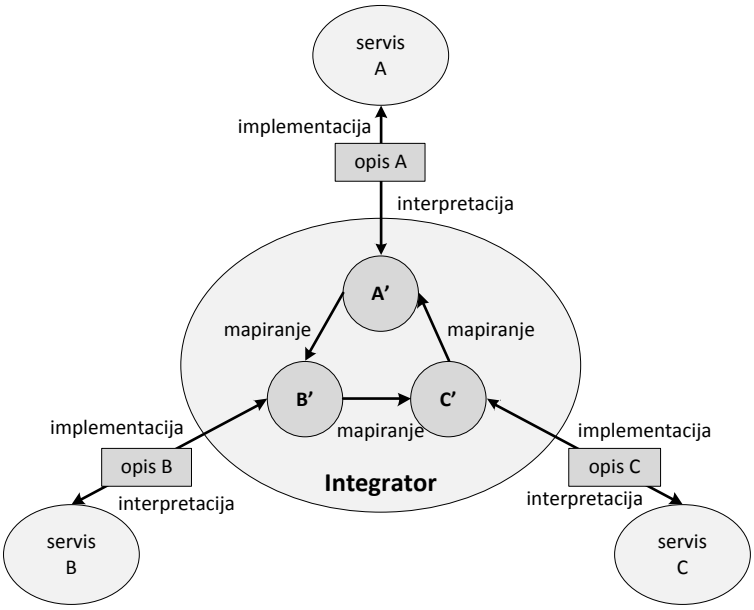
Klasifikacija modela za semantičku interoperabilnost može se izvršiti u okviru dve dimenzije [91]:

1. prema jednom od dva moguća načina za integraciono mapiranje:
   - prvi, po kome je šema za svaki servis mapirana u bilo koju drugu šemu ("svako-prema-svakom") i
   - drugi, po kome se svaki pojedinačni servis mapira u jednu šemu ("svi-prema-jednom")
2. prema mestu gde se izvršava integraciona logika:
   - centralizovano, u posebno izdvojenom čvoru, ili
   - decentralizovano, izvršenje se distribuira između više funkcionalno ekvivalentnih čvorova.

Kombinovanjem navedenih klasifikacionih kriterijuma, mogu se izvesti četiri interoperabilna modela.

Centralizovani "svako-prema-svakom" model

Kod centralizovanog, "svako-prema-svakom" modelu, servisi se interpretiraju i mapiraju u druge servise, bez korišćenja ontologija. Funkciju semantičke integracije obezbeđuje posebna komponenta, koja se naziva integrator. Integrator kreira model, za svaki od uključenih servisa, na osnovu sopstvene interpretacije opisa servisa.



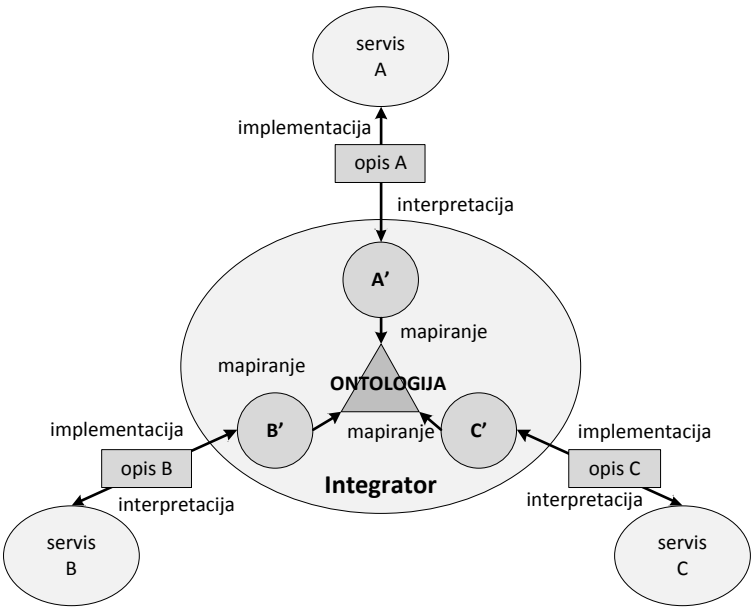
Slika 3.15: Centralizovani "svako-prema-svakom" model
(prilagođeno po [91])

Integrisani servisi, u ovom slučaju instance, najčešće su atomični, nezavisni i samostalni. Da bi se omogućila međusobna komunikacija između integrisanih servisa, preduslov je da postoji međusobna usaglašenost. U suprotnom, moguće semantičke greške otkrivaju se tek u fazi izvršavanja.

Model integrisanih servisa pogodan je za primenu u zatvorenom okruženju. Ontologije nisu uključene u model. U odnosu na druge modele za semantičku interoperabilnost, model "svako-prema-svakom" je skuplji i nepouzdaniji.

Centralizovani "svi-prema-jednom" model

Centralizovani model servisa "svi-prema-jednom", mapira ulazno/izlazne podatke u jednu jedinstvenu ontologiju, kojom upravlja specijalizovana aplikacija, ili servis (komponenta integrator na slici 3.16). Integrator kreira model za svaki servis posebno, na osnovu svoje interpretacije raspoloživih opisa servisa, na isti način, kao i u slučaju modela "svako-prema-svakom". Mapiranje obavlja integrator na osnovu jedinstvene ontologije. Dizajneri integratora zaduženi su za postavljanje semantičkog sloja, pa konzistentnost integratora zavisi od njihovog poznavanja sistema. Provajderi, kroz implementaciju servisa, obezbeđuju svoje sopstvene modele i nisu uključeni u razvoj integratora.

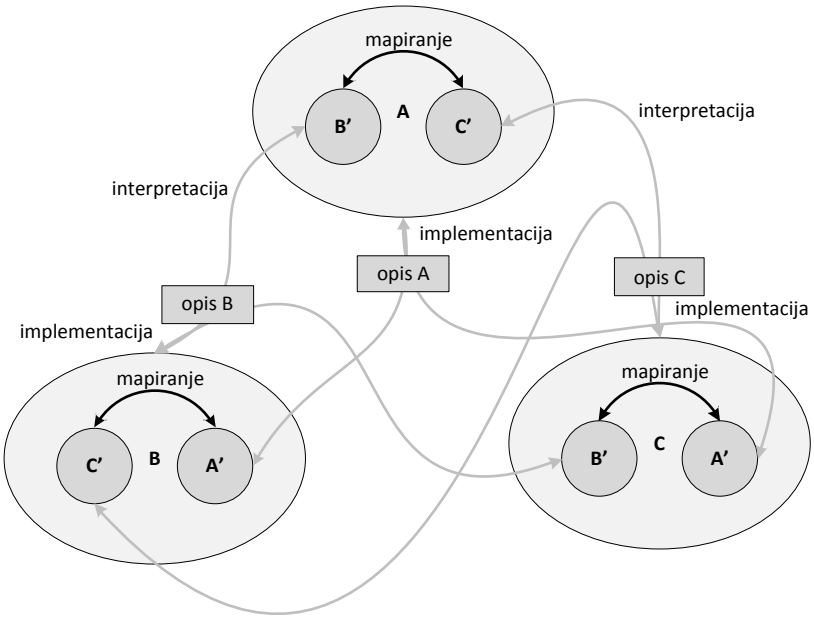


Slika 3.16: Centralizovani "svi-prema-jednom" model
(prilagođeno po [91])

Centralizovani model "svi-prema-jednom" pogodan je za primenu u dinamičkim okruženjima, na primer, u slučaju pozivanja veb servisa u B2B scenarijima. Primena modela nije jednostavna, jer zahteva pažljivo proučavanje odnosa između: izražajnosti jezika za ontologije; šema za mapiranje i integracionih algoritama. Značajna osobina modela "svi-prema-jednom" je mogućnost da se, uključenjem novih komponenti i izvršenjem njihovog mapiranja, proširi poslovni model.

Decentralizovani "svako-prema-svakom" model

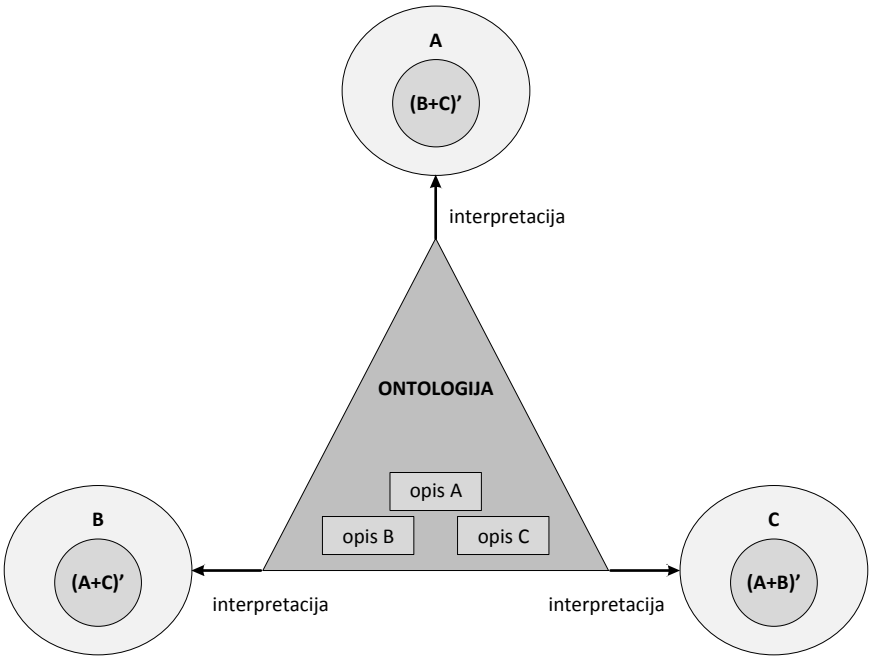
Integraciona logika decentralizovanog modela "svako-prema-svakom" je distribuirana i ne zasniva se na zajedničkim ontologijama. Model je karakterističan za slabo povezana okruženja. Svrstava se u "peer-to-peer" sistem. Semantike u sistemu su distribuirane i striktno međusobno izolovane. Odgovornost svakog od učesnika je da izvrši mapiranje prema ostalim učesnicima, koje želi da uključi. Usled nepostojanje centralnog modela, u cilju postizanja interoperabilnosti, potrebno je da se u svakom čvoru obezbedi mapiranje uključenih servisa.



Slika 3.17: Decentralizovani "svako-prema-svakom" model
(prilagođeno po [91])

Decentralizovani "svi-prema-jednom" model

Pravila mapiranja učesnika, u slučaju decentralizovanog "svako-prema-jednom" modela, obezbeđuju se na osnovu modela ontologije. Model ontologije uključuje semantiku i koncepte svih uključenih komponenti, a integraciona logika se izvršava distribuirano. U ovom modelu komponente su distribuirane, a razvoj i proširenje ontologija mora se, na neki način, obaviti centralizovano.



Slika 3.18: Decentralizovani "svi-prema-jednom" model
(prilagođeno po [91])

3.2.4.3 Semantičko označavanje veb servisa

U okviru SOA, servisi su definisani standardnim jezicima za opis servisa, kao što je WSDL, i jezicima za otkrivanje servisa, kao što je UDDI. Međutim, ovi jezici nisu dovoljno izražajni da omoguće proces automatskog otkrivanja i pronalaženja veb servisa. U suštini, podaci i operacije u okviru WSDL dokumenata ne mogu se mašinski, jasno interpretirati, ukoliko nisu semantički označeni.

Za označavanje veb servisa koriste se meta podaci iz ontologija. Označavanje može biti u obliku: komentara; beleške; pitanja ili nekog drugog tipa eksterne beleške, koja se dodaje dokumentu ili izabranom delu dokumenta.

Ručno označavanje servisa je dugotrajno i skupo. Ukoliko se servisi opišu semantički, mnogi od poslova, koji su prethodno zahtevali ljudsku intervenciju u izgradnji i održavanju SOA aplikacije, biće automatizovani. Na primer, servisi će moći da se otkrivaju na osnovu semantički opisanih funkcionalnosti, kao i da se biraju na osnovu definisanog kvaliteta servisa. Ovo će omogućiti dinamičko povezivanje sa potrebnim servisom u toku izvršavanja, odnosno uklanjanje hard-kodiranih informacija, tipičnih za klasičnu SOA. Nakon pojave novih servisa, koji nude potrebne funkcionalnosti, moći će da se razmatra zamena servisa koje sistem već koristi, novim, semantički unapređenim servisima. Osim toga, servis koji sistem koristi i koji više nije dostupan, moći će da se automatski zameni servisom, koji obezbeđuje potrebnu funkcionalnost.

Poslednjih godina razvijeno je nekoliko jezika za realizaciju semantičkog označavanja i obezbeđenje semantičke interoperabilnosti, koji omogućavaju predstavljanje: klasa; relacija; instanci i aksioma. Postojeće standarde za semantičko označavanje: RDF i RDF Schema, kao i OWL jezik za njihovo proširenje, predložio je konzorcijum W3C.

Resource Description Framework (RDF)

Resource Description Framework [92] obezbeđuje opšti metod za konceptualno opisivanje veb resursa i dizajniran je u obliku modela podataka za semantički veb. RDF se može koristiti za predstavljanje distribuiranih informacija i znanja, na način da ih mogu koristiti i obrađivati računarske aplikacije. Prvobitno je bio namenjen za predstavljanje meta podataka o veb resursima. Kasnije, RDF je generalizovan tako da se, pored korišćenja za predstavljanje meta podataka, može upotrebiti i za opis bilo kog resursa i relacija između resursa, u realnom svetu.

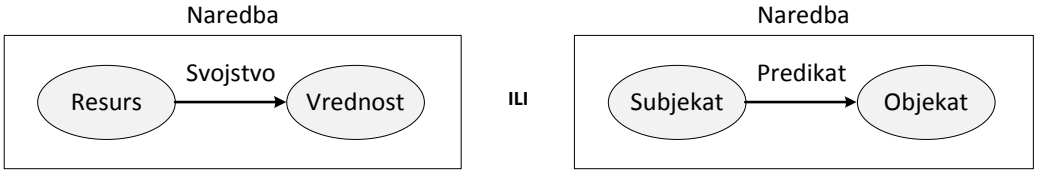
Poslednja verzija RDF sastoji se od šest dokumenata, koji zajedno opisuju različite aspekte RDF:

- RDF Primer - RDF je jezik za predstavljanje informacija o resursima na vebu;
- RDF Concept - RDF je okvir za predstavljanje informacija na vebu;
- RDF Schema i RDF Syntax - RDF je jezik opšte namene za predstavljanje informacija na vebu;
- RDF Semantics - RDF je jezik za izražavanje predloga preko preciznog, formalnog rečnika. Posebno se bavi predlozima, koje koristi RDF Schema, sa ciljem da pruži osnovu naprednijim jezicima sličnih namena.
- RDF Test Cases - sadrži skup testnih slučajeva, koji pokrivaju pravila definisana u prethodnim dokumentima.

Osnovni koncept RDF-a predstavljen je njegovim apstraktnim modelom, koji se koristi za predstavljanje znanja o realnom svetu. RDF koristi apstraktni model da dekomponuje informacije i znanje u delove, uključujući u svaki od tih delova jednostavna pravila o semantici. Cilj je da se obezbedi opšti metod, koji je dovoljno jednostavan i fleksibilan da izrazi bilo koju činjenicu, i istovremeno dovoljno struktuiran, da računarske aplikacije mogu da koriste predstavljeno znanje.

Apstraktni RDF model sastoji se od tri ključne komponente:

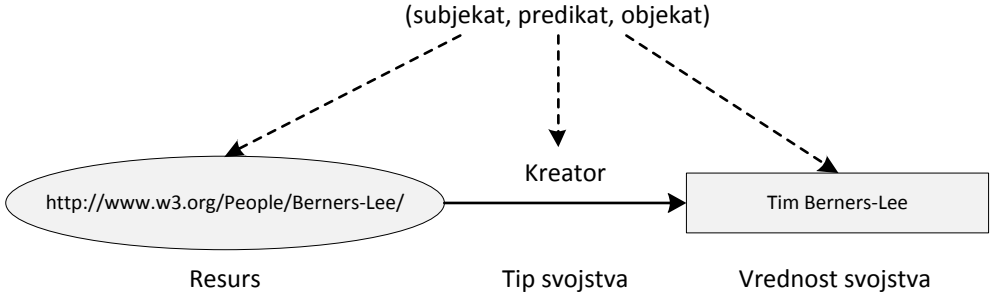
1. Subjekti (resursi): Stvari, koje RDF opisuje, nazivaju se subjekti, ili resursi. Resurs može biti veb strana, deo veb strane ili njihova kolekcija. Resurs može biti i objekat, kojem se ne pristupa direktno sa veb strane. Svi resursi identifikovani su pomoću *Uniform Resource Identifier* (URI).
2. Predikati (svojstva): Predstavljaju poseban aspekt, karakteristiku, atribut ili relaciju, koja se koristi da opiše resurs. Svako svojstvo ima određeno značenje. Definisane su dozvoljene vrednosti, tipovi resursa koje može da opiše, kao i odnosi sa drugim svojstvima. Identifikuje se preko URI.
3. Naredbe: Sastoji se od subjekta, predikata i objekta, odnosno, resursa sa imenovanim svojstvom i njegovom vrednošću. Objekat u naredbi može biti drugi resurs, ili literal (primitivni tip podatka definisan u okviru RDF šeme).



Slika 3.19: RDF naredba

Postoje dva komplementarna načina za predstavljanje informacija i znanja. Prvi način je preko skupa naredbi. Drugi način je preko RDF grafa. RDF graf je zasnovan na mreži, koja se sastoji od čvorova, povezanih usmerenim konektorima. Strelica na grafu uvek ukazuje na objekat u okviru naredbe. Graf predstavlja konjukciju (logički AND) svih naredbi, od kojih se sastoji.

RDF graf prikazuje potpuno iste informacije kao i napisana RDF naredba, jedino što grafički prikaz obezbeđuje da korisnici lakše sagledaju strukturu podataka.



Slika 3.20: Grafički prikaz RDF naredbe

```
<?xml version="1.0" ?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"  >
<rdf:Description rdf:about=" http://www.w3.org/People/Berners-Lee/">
<dc:creator>Tim Berners-Lee</dc:creator >
</rdf:Description>
</rdf:RDF>
```

Slika 3.21: RDF naredba

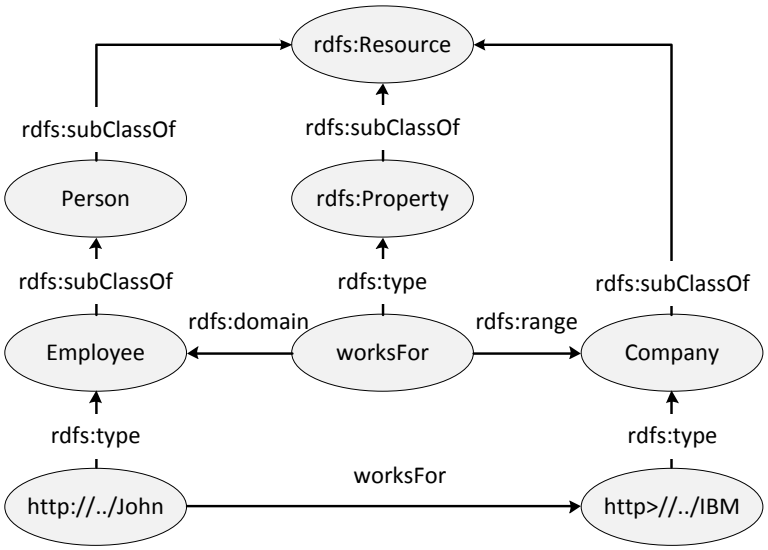
Sa ciljem da se omogući da se informacije i znanja, predstavljena pomoću RDF-a, zapamte, razmenjuju i pretražuju, potrebno je da se serializuju. Za RDF serializaciju koristiti se više formata: *XML; Notation-3* (N3); *Turtle; N-Triples*. Zajedničko za sve formate za RDF serializaciju je da podrazumevaju da će se RDF graf serializovati u samostalan dokument.

Za pronalaženje i manipulaciju RDF podataka koristi se *SPARQL Protocol And RDF Query Language* (SPARQL). SPARQL omogućava pretraživanje RDF trojki i grafova i podržava različite mogućnosti za kreiranje upita.

Poslednjih godina, W3C istražuje "prirodniji" način za zajedničko korišćenje RDF i XHTML. Rezultat istraživanja *je RDF in attributes* (RDFa), koji omogućava da se RDF ugradi direktno u XHTML dokument i izbegne postojanje odvojenog RDF dokumenta. Prednost ovakvog načina prikaza RDF je mogućnost da veb strane budu razumljive i ljudima i mašinama. U momentu pisanja teze, najpoznatije primene RDFa su *Google Rich Snippet* i *Yahoo! SearchMonkey*.

RDF Schema (RDFS)

RDF model podataka ne obezbeđuje mehanizme za opisivanje svojstava i relacija između svojstava i drugih resursa. Za tu svrhu može se koristiti RDF šema, koja predstavlja sistem za opis osnovnih karakteristika RDF jezika. RDF šema omogućava definisanje rečnika za RDF i njegovo hijerarhijsko organizovanje (Clas, subClassOf, type, Property, subPropertyOf). Pored toga, koristeći RDF šemu moguće je definisati taksonomiju klasa i svojstava i proširiti definicije za neke od elementa RDF.

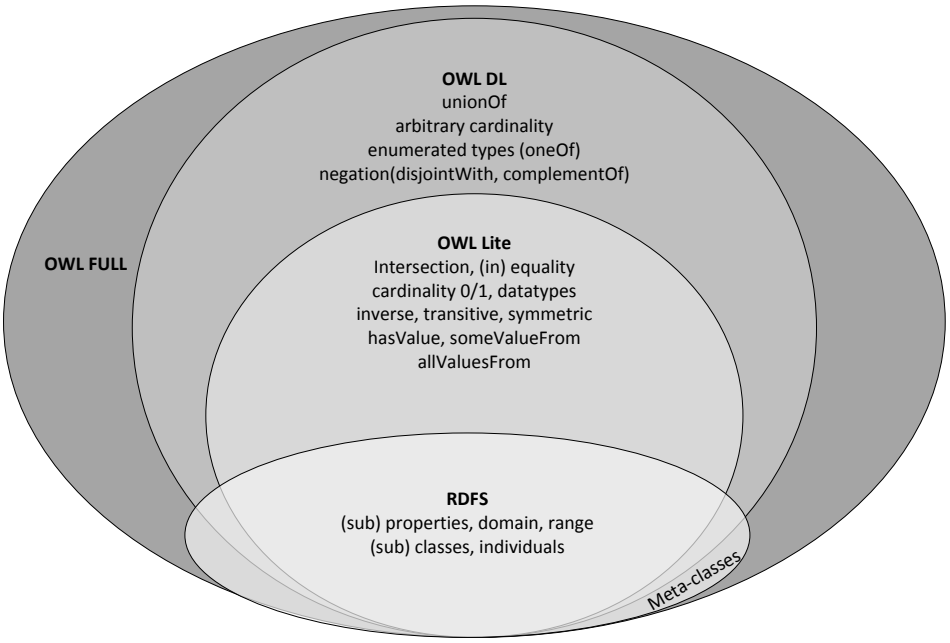


Slika 3.22: primer RDF šeme

Web Ontology Language (OWL)

Web Ontology Language je W3C standard za predstavljanje ontologija. Izgrađen je na osnovu RDF jezika. Pruža više mogućnosti u odnosu na RDF, zahvaljujući bogatijem rečniku i potpunijoj sintaksi. Napravljen je sa ciljem da se mašinski interpretira.

Postoje tri vrste OWL jezika: OWL Lite; OWL DL i OWL Full. Poseduju različitu izražajnost i jezičke konstrukcije. OWL Lite je najjednostavniji i napravljen je sa namerom da predstavi klasifikacionu hijerarhiju i jednostavna ograničenja. Na primer, OWL Lite podržava kardinalnost, ali samo tipa nula ili jedan. OWL DL obuhvata kompletan OWL rečnik, interpretiran isključivo preko jednostavnih ograničenja. OWL Full obuhvata kompletan OWL rečnik, tumačen šire nego u slučaju OWL DL.



Slika 3.23: Izražajnost OWL jezika i RDFS

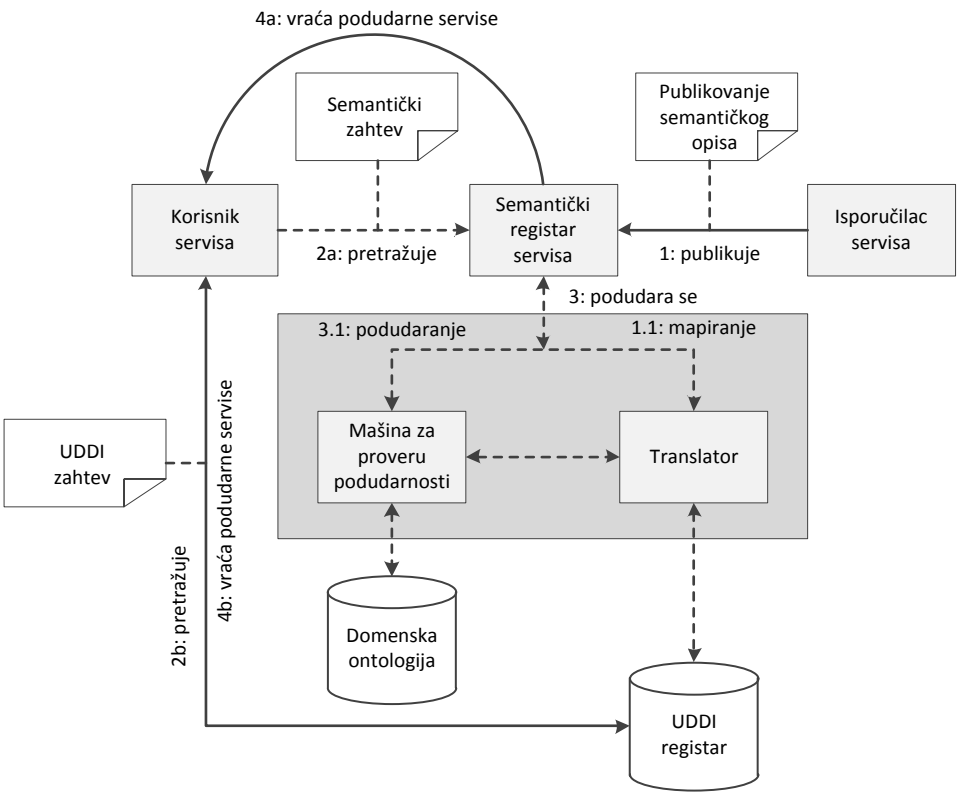
Pored prethodno navedenih i opisanih tehnologija i jezika za semantičko označavanje, pojavljuje se i određeni broj tehnologija i jezika iste, ili slične namene. Neke od njih su: DAML-S; OWL-S; WSDL-S; SWSO/SWSL; WSMO/WSML.

3.2.4.4 Otkrivanje semantički označenih veb servisa

UDDI obezbeđuje osnovnu podršku za publikovanje i otkrivanje veb servisa, u okviru ili van preduzeća. Otkrivanje veb servisa u UDDI registru obavlja se preko ključnih reči. Rezultati ovakvog načina otkrivanja često su netačni, a sam proces otkrivanja može dugo da traje. Povećanje broja servisa i njihova upotreba, uslovila je potrebu da se servisi semantički opisuju, kako bi mogli da se registruju i otkrivaju automatski.

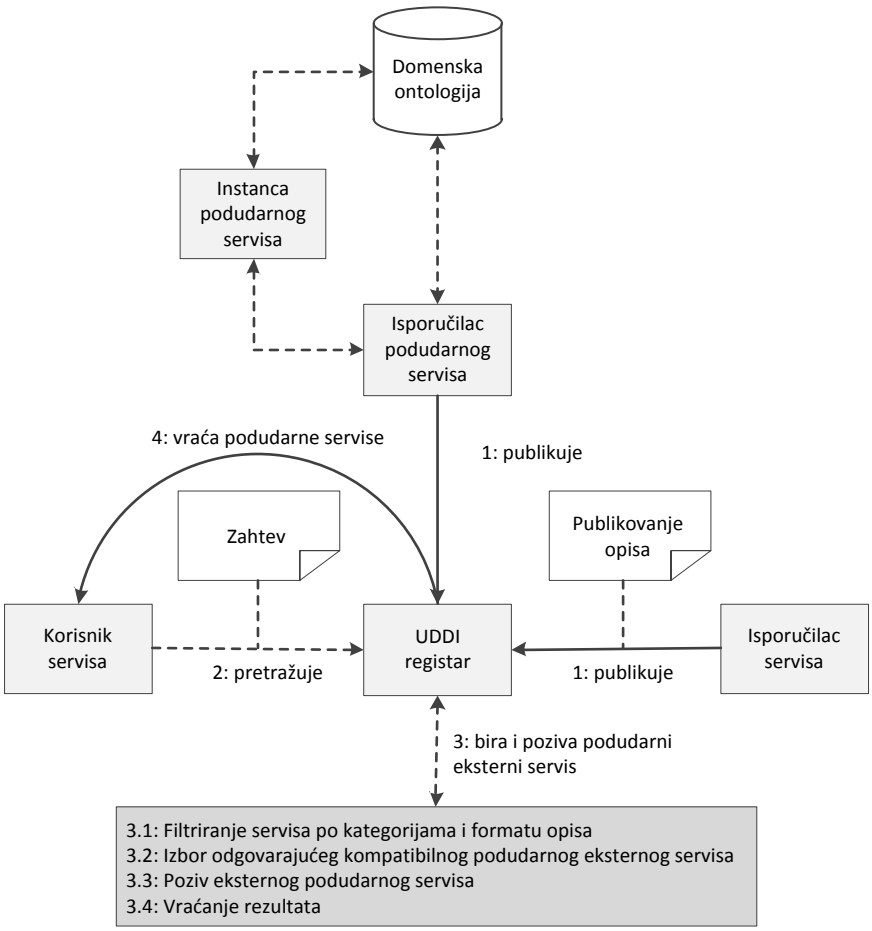
Tradicionalni UDDI registri mogu se koristiti i za otkrivanje semantički označenih veb servisa, jer, pored uobičajenih opisa veb servisa, UDDI registri mogu da sadrže, ili da imaju reference, na semantičke informacije o veb servisima. U tom slučaju, govori se o centralizovanom pristupu. Distribuirani pristup odnosi se na pronalaženje potrebnih servisa u peer-to-peer (P2P) sistemima.

Jedan od centralizovanih pristupa odnosi se na dodavanje semantike u UDDI registar. U radu [93] predlaže se proširenje UDDI registra, dodavanjem semantičkih oznaka na standardne UDDI elemente, kao što je tModel. Na taj način kombinuju se UDDI standard i semantički opisi veb servisa. Takođe, predloženo je korišćenje mašine za proveru podudarnosti, čija je uloga pronalaženje semantički označenih veb servisa. Po [94], nedostaci ovog pristupa ogledaju se u potrebi da se instalira dodatna infrastruktura u UDDI registar. Grafički prikaz procesa pronalaženja odgovarajućeg veb servisa, na osnovu semantičkih oznaka, dat je na slici 3.24.



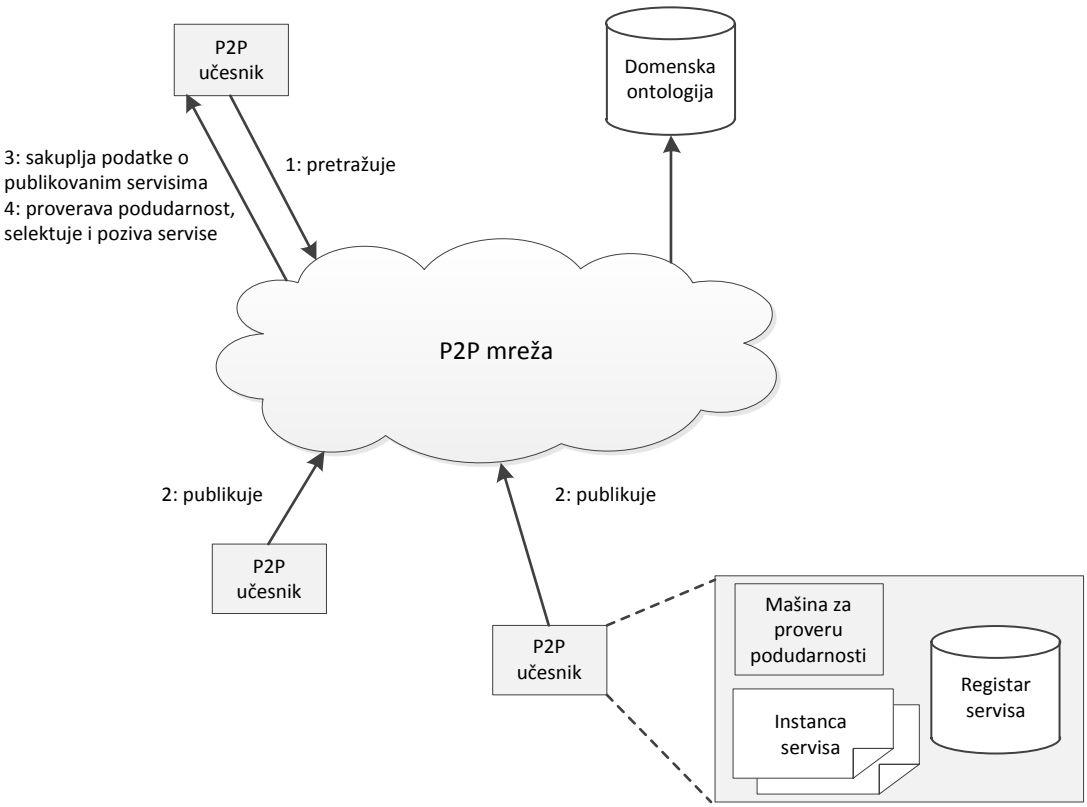
Slika 3.24: Otkrivanje servisa u semantički proširenom UDDI registru
(prilagođeno po [56])

Drugi centralizovani pristup po [94] ilustrovan je slikom 3.25, i odnosi se na eksternu proveru podudarnosti servisa. Algoritmi za proveru podudarnosti publikuju se kao veb servisi u UDDI registar. UDDI može da otkrije i izabere najpogodniji servis za svaki upućeni zahtev, i da ga pozove na osnovu semantičke podudarnosti. Kako bi podržao ovaj metod, UDDI registru treba dodati dva procesa. Prvi proces, na osnovu formata zahteva (da li sadrži ili ne semantičke oznake), određuje da li se može koristiti eksterna provera podudarnosti. Zatim se, na osnovu formata, bira i poziva odgovarajući servis za poređenje, i na kraju, informacije o otkrivenom servisu vraćaju se klijentu. Ovaj pristup eliminiše nedostatke prvog navedenog pristupa, a koji se odnose na potrebu da se instalira potrebna infrastruktura u UDDI registar. Međutim, pristup ne obezbeđuje koordinaciju potrebnu za kreiranje kompozicije od ad-hoc pozvanih servisa.



Slika 3.25: Eksterno provera podudarnosti servisa
(prilagođeno po [56])

Peer-to-peer (P2P) pristup svrstava se u distribuirane pristupe za pronalaženje servisa. Kod P2P pristupa, svaki P2P učesnik je provajder servisa, klijent servisa, ili istovremeno i provajder i klijent. Saglasno tome, svaki P2P učesnik treba da sadrži semantički opis svojih servisa, koje nudi, linkove prema domenskim ontologijama, kao i P2P komponentu za otkrivanje, koja je implementirana mašinom za proveru podudarnosti. Sa ciljem da pronađe traženi servis, servis klijenta pretražuje druge P2P učesnike u P2P mreži. Kada dobije semantički opis servisa, proces poređenja izvodi se lokalno. P2P pristup složeniji je u odnosu na prethodno opisane pristupe za pronalaženje odgovarajućih servisa. Sa druge strane, obezbeđuje veću pouzdanost i fleksibilnost. Uvek je publikovana poslednja verzija servisa, dok registar servisa, kod centralizovanog pristupa, može da sadrži i stare verzije. Takođe, klijent može da koristi svoj sopstveni algoritam za proveru podudarnosti.



Slika 3.26: P2P otkrivanje semantički označenih servisa
(prilagođeno po [56])

Nezavisno od pristupa koji se koristi, osnovni kriterijum za otkrivanje semantički označenih veb servisa je usklađivanje koncepata između klijenta i provajdera servisa, koje se postiže korišćenjem različitih mehanizama i algoritama za proveru podudarnosti.

4 Modelovanje interoperabilnosti sistema elektronskog poslovanja

Tokom planiranja sistema elektronskog poslovanja, neke od prvih odluka, koje se donose su izbor metodologije, metoda i alata, koji će biti korišćeni. S obzirom da je teško govoriti o "najboljoj metodologiji", "najboljem metodu" i "najboljem alatu", izbor može olakšati pregled postojećih iskustava.

Kada je reč o planiranju i razvoju sistema elektronskog poslovanja, do postavljenih ciljeva teško se dolazi bez efikasnog praćenja aktivnosti, pa je potrebno opredeliti se za primereni okvir. Imajući u vidu da su softverski sistemi delom "slika" realnog sveta, u praksi se često polazi od okvira opšte namene. Jedan od najpoznatijih i najopštijih je Zahmanov okvir. Striktna primena opštih okvira, s druge strane, ne mora da bude i garancija da će sistem, u svim delovima i iz svih aspekata, biti uspešno razvijen. Naime, metodologije, metode, alate i okvir trebalo bi birati imajući u vidu konkretni sistem, koji se modeluje.

Zajednička karakteristika okvira su različiti nivoi apstrakcije artifakata, kojima se opisuju različiti pogledi na sistem. Samim tim, za kreiranje konkretnih modela koriste se odgovarajuće tehnike modelovanja, namenjene za izradu konkretnih klasa modela. Dobijeni modeli trebalo bi da predstavljaju celovit pogled na sistem.

U prvom delu ovog poglavlja opisan je postupak za prilagođavanje Zahmanovog okvira potrebama modelovanja sistema elektronskog poslovanja. Polazeći od originalnog Zahmanovog okvira, a imajući u vidu zahteve i karakteristike servisno orijentisanog sistema, koji se modeluje i koji je potrebno razviti, prikazan je algoritam za punjenje ćelija Zahmanovog okvira. Na osnovu analize popunjenosti ćelija Zahmanovog okvira, izveden je i opisan novi okvir – SOFI okvir, koji bi trebalo da usmeri razvoj interoperabilnih, servisno orijentisanih sistema elektronskog poslovanja.

Nezavisno od svrhe modelovanja, jedan od mogućih načina za obezbeđenje konzistentnosti, na nivou celokupnog modela sistema, je primena generičkog meta modela u pojedinim fazama modelovanja sistema. Tokom rada na ovoj tezi, kreiran je generički meta model, koji bi trebalo da podrži izradu artifakata za popunjavanje ćelija Zahmanovog okvira. Razvoj generičkog meta modela obavljen je proširenjem i transformacijom trougla znanja i zasnovan je na ontologijama, poslovnim terminima i poslovnim konceptima. Kreirani generički meta model i mogućnosti primene za modelovanje kolekcija artifakata Zahmanovog okvira, opisani su u drugom delu ovog poglavlja.

4.1 Razvoj okvira za interoperabilnost

Jedan od ciljeva teze je da se obezbedi okvir sa infrastrukturnim komponentama, potrebnim za razvoj aplikacija elektronskog poslovanja, zasnovanim na SOA konceptima. Kreirani okvir trebalo bi da zadovolji neke od osnovnih principa:

- da bude kreiran na osnovu iskazanih potreba korisnika;
- da je zasnovan na standardima i šablonima, koji se primenjuju u oblasti servisno orijentisanog razvoja softvera;
- da omogući potrebnu interoperabilnost;
- da je primenljiv i praktičan za upotrebu;
- da ga karakteriše određeni stepen adaptivnosti, kako ne bi bio brzo prevaziđen;
- da pored sopstvenog, namenskog razvoja softverskog sistema, predvidi i mogućnost kupovine ili korišćenje postojećih, pojedinačnih softverskih komponenti.

Uloga okvira u prihvaćenoj metodologiji je da objedini skup: ideja, shvatanja i pravila, koji se koristi za planiranje ili odlučivanje, ili kao osnova za izgradnju artifakta, u ovom slučaju, za izgradnju sistema elektronskog poslovanja. Pored navedenih primena, okviri se mogu koristiti i za proveru validnosti modela.

Prezentovani okvir trebalo bi da isporuči dva ključna elementa [64]:

- opis metoda, pomoću kojih se popunjavaju ćelije za kreiranje interoperabilnih sistema elektronskog poslovanja;
- definicije pojedinačnih sadržaja ćelija okvira.

Primenljivost i uspešnost razvijenog okvira biće proverena na primeru razvoja konkretne aplikacije elektronskog poslovanja, zasnovane na SOA konceptu.

4.1.1 Metodologija i proces modelovanja okvira

Tokom procesa modelovanja okvira za razvoj sistema elektronskog poslovanja korišćena je jedinstvena metodologija. Izbor metodologije izvršen je na osnovu sledećih saznanja i kriterijuma:

- rezultati analize postavljenih zahteva;
- poznavanje prirode sistema, koji je potrebno razviti;
- raspoloživa opšta i specijalizovana literatura;
- uvid u dostupne opšte i specijalizovane standarde,
- kao i na osnovu dosadašnjeg iskustva autora.

Metodologija, predložena u ovoj tezi, obuhvata:

- identifikovanje važnih delova sistema elektronskog poslovanja i modelovanje arhitekture sistema, pri čemu se u prvom koraku koristi Zahmanov okvir,
- prepoznavanje elemenata, koje je potrebno povezati i integrisati, koristeći modele poslovnih procesa,
- kreiranje interoperabilnih, servisno orijentisanih rešenja, koristeći odgovarajuću, servisno orijentisanu arhitekturu.

Razvijena metodologija definiše, opisuje i preporučuje:

1. kreiranje i način korišćenja okvira;
2. izbor odgovarajućih jezika za kreiranje modela unutar ćelija okvira, a u zavisnosti od potrebne detaljnosti i namene;
3. razvoj modela unutar ćelija okvira i uspostavljanje veza između kreiranih modela;
4. grupisanje ćelija okvira za određivanje granica različitih pogleda;
5. korišćenje tehnike za proveru validnosti modela.

Metodologija za kreiranje okvira za modelovanje sistema elektronskog poslovanja, primenjena u ovoj tezi, obuhvatila je sledeće faze:

1. Određivanje artifakata
   Na osnovu analize životnog ciklusa razvoja softverskih sistema, identifikovani su artifakti, potrebni za razvoj interoperabilnih aplikacija elektronskog poslovanja.
2. Kreiranje inicijalne matrice
   Rezultat ove faze je uspostavljanje međusobnih veza i uslovljenosti identifikovanih artifakata u okviru faza životnog ciklusa razvoja softverskih sistema.
3. Popunjavanje Zahmanovog okvira
   Za kreiranje potrebnog SOA okvira, polazi se od Zahmanovog okvira. Zahmanov okvir popunjava se sadržajima ćelija inicijalne matrice iz tačke 2, a na osnovu ustanovljenog algoritma za popunjavanje.
4. Kreiranje SOFI okvira
   Delimično popunjen Zahmanov okvir, namenjen razvoju SOA sistema (iz tačke 3), metodom grupisanja popunjenih ćelija, mapira se u novi, SOFI okvir.

Opisana metodologija za razvoj sistema elektronskog poslovanja definiše načine za prikupljanje relevantnih informacija o procesima, poslovanju preduzeća i operativnim zahtevima, kao i načine za konzistentno i efikasno organizovanje, modelovanje i čuvanje dobijenih informacija. Osim razvoja sistema, metodologija obezbeđuje i naknadno ponavljanje koraka, preduzetih tokom kreiranja opisa arhitekture.

4.1.1.1 Artifakti dokumentacije i modela ćelija okvira

Artifakti, neophodni za popunjavanje okvira za interoperabilnost, određeni su korišćenjem Microsoft Solutions Framework (MSF) okvira [95]. MSF okvir upućuje na vrstu i redosled aktivnosti, koje treba obaviti tokom razvoja i isporuke softverskog rešenja, opisujući ceo životni ciklus softverskog rešenja. MSF nastao je na osnovu iskustava kompanije Microsoft, kupaca i prodavaca velikih softverskih projekata, Microsoft konsultanata, stečenim tokom realizacije razvoja sistema za poslovne korisnike, kao i postojeće prakse iz oblasti informacionih tehnologija. Opredeljenje za korišćenje MSF okvira procesa u ovoj tezi zasnovano je na fleksibilnosti i skalabilnosti primene MSF okvira tokom procesa kreiranja različitih vrsta softverskih rešenja, kao i na iskustvu autora teze, vezanim za njegovo korišćenje.

MSF okvir, na najvišem nivou aktivnosti, prati faznu realizaciju softverskog rešenja: faza predviđanja; faza planiranja; faza razvoja; faza stabilizacije i faza uvođenja. Pre početka realizacije navedenih faznih aktivnosti, MSF preporučuje prikupljanje i analizu zahteva. Inicijalno prikupljeni zahtevi mogu

se dodatno i detaljnije definisati u bilo kojoj fazi modelovanja. Pojedine aktivnosti, iz različitih faza, mogu se vremenski međusobno preklapati, odnosno, završetak prethodne faze nije uvek preduslov za početak naredne faze. Svaka faza MSF okvira završava se isporukom definisanih rezultata. Međutim, u zavisnosti od vrste i obima softverskog rešenja koje se realizuje, nije obavezno da svi rezultati, koje navodi MSF, u svim slučajevima budu i isporučeni.

U ovoj tezi se, za kreiranje faznih rezultata, najčešće predlaže korišćenje modela prikazanih UML dijagramima. Razlog proističe iz činjenice da se, u ovom momentu, većina softverskih specifikacija razvija pomoću notacija zasnovanih na UML jeziku. Isporučivi rezultati faza MSF okvira, prikazani u tabeli 4.1, prilagođeni su razvoju interoperabilnih aplikacija elektronskog poslovanja.

Tabela 4.1: Faze i isporučivi rezultati u MSF okviru

Faza MSF okvira	**Isporučivi rezultati**
faza predviđanja (izrada grubog opisa ciljeva i ograničenja projekta)	Spisak zahteva Vizija/oblast primene Koncept rešenja Spisak i opis lokacija
faza planiranja (određuje šta treba da se razvija i planira kako bi se kreiralo rešenje) - konceptualni dizajn - logički dizajn - fizički dizajn	Slučajevi korišćenja Scenariji upotrebe Dijagram aktivnosti / BPMN Prototip UI Konceptualni (domenski) model klasa Konceptualni model podataka Dijagram sekvenci Logički model klasa Model servisa Dijagram isporuke Dizajn UI Arhitektura sistema Logički model podataka Model klasa Specifikacija servisa Komponente Implementacione tehnologije Opis UI ekrana Fizički dizajn podataka
faza razvoja (kreiranje rešenja)	Izvorni kod Izvršne datoteke Instalacioni skriptovi Konfiguraciona podešavanja Konačna funkcionalna specifikacija Elementi za podršku performansi Specifikaciji za testiranje i testni slučajevi
faza stabilizacije (testiranje rešenja)	Konačno izdanje Napomene o izdanju Podrška performansi Rezultati testiranja Izvorni kod i alatke za testiranje Projektni dokumenti Pregled markera
faza uvođenja (uvođenje rešenja)	Informacioni sistemi Skladište dokumentacije Plan obuke Izveštaj o završetku projekta

4.1.1.2 Kreiranje inicijalne matrice

Polazeći od faza MSF okvira i isporučivih rezultata po fazama, kreirana je inicijalna matrica, u kojoj redovi predstavljaju isporučive rezultate, odnosno artifakte, a kolone faze MSF okvira. Sadržaj ćelija ukazuje na međusobne veze i uslovljenosti između kolona i redova u matrici, označavajući koja konkretna aktivnost stvara (**S**) određeni artifakt i koje, već postojeće artifakte, koristi (**K**) pri kreiranju sopstvenog artifakta.

Faze razvoja, stabilizacije i uvođenja nisu prikazane u matrici, jer se izvršavaju na način definisan u fazi planiranja, zbog čega nisu obuhvaćene predloženom metodologijom.

Tabela 4.2: Matrica artifakata i njihovih zavisnosti po MSF fazama

Artifakti	**Faza predviđanja**		**Faza planiranja**					
			Konceptualni dizajn		**Logički dizajn**		**Fizički dizajn**	
	Definisanje i analiza zahteva	Opis ciljeva i ograničenja	funkcija sistema	podataka	funkcija sistema	podataka	funkcija sistema	podataka
Spisak zahteva	S	K	K	K				
Vizija/oblast primene		S	K	K				
Koncept rešenja		S	K	K				
Spisak i opis lokacija		S	K					
Slučajevi korišćenja			S		K			
Scenariji upotrebe			S		K			
Dijagram aktivnosti/BPMN			S		K			
Prototip UI			S		K			
Konceptualni (domenski) model klasa			S		K			
Dijagram isporuke			S		K			
Konceptualni model podataka				S		K		
Dijagram sekvenci					S		K	
Logički model klasa					S		K	
Model servisa					S		K	
Dijagram isporuke					S		K	
Dizajn UI					S		K	
Arhitektura sistema					S		K	
Logički model podataka						S		K

Artifakti	Faza predviđanja		Faza planiranja					
			Konceptualni dizajn		Logički dizajn		Fizički dizajn	
	Definisanje i analiza zahteva	Opis ciljeva i ograničenja	funkcija sistema	podataka	funkcija sistema	podataka	funkcija sistema	podataka
Dijagram klasa							S	
Specifikacija servisa							S	
Dijagram komponenti							S	
Implementacione tehnologije							S	
Dijagram isporuke							S	
UI ekrani							S	
Fizički model podataka								S

Grupisanje ćelija matrice izvršeno je prema fazama MSF. Dobijene grupe ćelija (submatrice) označene su različitim bojama, čineći inicijalnu matricu preglednijom.

4.1.1.3 Popunjavanje Zahmanovog okvira

Pri izboru okvira za modelovanje sistema elektronskog poslovanja postojale su dve mogućnosti: da se prihvati neki od već postojećih i često korišćenih okvira, ili da se razvije, ili izvede iz postojećih - nov okvir. Imajući u vidu da je sistem elektronskog poslovanja zasnovan na servisno orijentisanom razvoju softvera, striktna primena postojećih okvira, verovatno, ne bi omogućila kreiranje validnog modela sistem.

Jedan od najčešće korišćenih okvira je Zahmanov okvir. Međutim, bez obzira na popularnost, korišćenje Zahmanovog okvira u praksi nije jednostavno, jer ne postoje opšte i jedinstvene metode i alati, koji omogućavaju kreiranje arhitekture informacionih sistema preduzeća iz svih potrebnih perspektiva. Predloženo je nekoliko načina za rešavanje ovog problema [96] [83], sa istom idejom: za popunjavanje ćelija okvira koriste UML dijagrame.

Polazeći od činjenice da je većina poznatih okvira, kao što su FEAF, TEAF, EAP, izvedena iz Zahmanovog okvira, a uvažavajući konstataciju njihovih autora da su uspešno kreirali novi, sopstveni okvir, postavljen je sličan zadatak:

"Polazeći od Zahmanovog okvira, potrebno je kreirati nov okvir, koji će, svojom aktivnom ulogom tokom procesa modelovanja, dovesti do kreiranja validne arhitekture budućeg sistema elektronskog poslovanja, zasnovanog na SOA arhitekturi".

Do novog, SOFI okvira, došlo se nakon određenog broja iteracija. Proces modelovanja prolazio je kroz sledeće, glavne faze:

1. Pražnjenje (inicijalizacija) ćelija Zahmanovog okvira.
2. Popunjavanje ćelija Zahmanovog okvira na osnovu baze definisanih artifakata.

3. Provera sadržaja ćelija u odnosu na kriterijume za konvergenciju iterativnog postupka.

Na osnovu filozofije Zahmanovog okvira, ustanovljeni su kriterijumi i algoritmi za procenu konvergencije procesa za izvođenje novog okvira:

1.	kriterijum	prazna ćelija
	algoritam	proveriti da li je zahtev suviše opšti, ili ne postoji
2.	kriterijum	suviše uopšten zahtev ili model, pa se mapiranje vrši u više od jedne ćelije
	algoritam	proveriti da li je moguća dekompozicija zahteva ili modela
3.	kriterijum	centri gravitacije (grupisanja) popunjenih ćelija
	algoritam	proveriti da li su popunjene ćelije grupisane u odnosu na susedne kolone i redove, ili raspoređene po celom prostoru okvira

Kriterijumi za konvergenciju procesa kreiranja okvira nisu bili formalno izraženi, već je konvergencija proveravana na osnovu ekspertske procene.

1. faza: Pražnjenje (inicijalizacija) ćelija Zahmanovog okvira

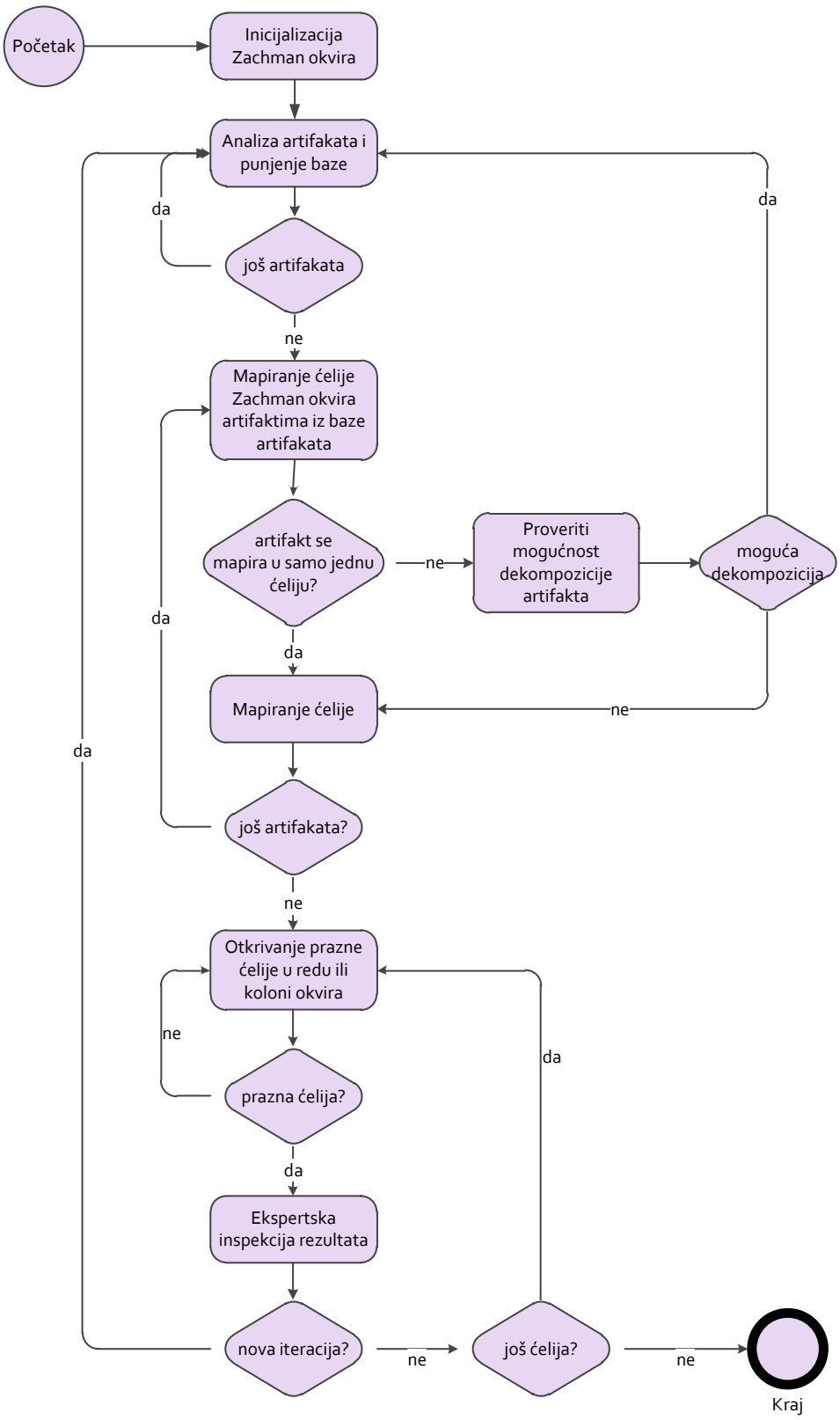
Inicijalizovan Zahmanov okvir prikazan je na slici 4.1. Postojeći sadržaji ćelija upućuju na očekivani, budući sadržaj, kojim će biti zamenjeni.

	Šta	Kako	Gde	Ko	Kada	Zašto
Perspektiva poslovnih izvršilaca	Identifikacija podataka	Identifikacija procesa	Identifikacija lokacija	Identifikacija odgovornosti	Identifikacija vremenskih ciklusa	Identifikacija motivacije
Perspektiva poslovnog upravljanja	Definicija podataka	Definicija procesa	Definicija lokacija	Definicija odgovornosti	Definicija vremenskih ciklusa	Definicija motivacije
Perspektiva arhitekata	Predstavljanje podataka	Predstavljanje procesa	Predstavljanje lokacija	Predstavljanje odgovornosti	Predstavljanje vremenskih ciklusa	Predstavljanje motivacije
Perspektiva inženjera	Specifikacija podataka	Specifikacija procesa	Specifikacija lokacija	Specifikacija odgovornosti	Specifikacija vremenskih ciklusa	Specifikacija motivacije
Tehnička perspektiva	Konfiguracija podataka	Konfiguracija procesa	Konfiguracija lokacija	Konfiguracija odgovornosti	Konfiguracija vremenskih ciklusa	Konfiguracija motivacije
Perspektiva preduzeća	Instanciranje podataka	Instanciranje procesa	Instanciranje lokacija	Instanciranje odgovornosti	Instanciranje vremenskih ciklusa	Instanciranje motivacije

Slika 4.1: Inicijalizovani Zahmanov okvir

2. faza: Popunjavanje ćelija Zahmanovog okvira

Proces popunjavanja ćelija inicijalizovanog Zahmanovog okvira (slika 4.1) prikazan je sledećim algoritmom:



Slika 4.2. Algoritam popunjavanja ćelija Zahmanovog okvira

3. faza: Provera sadržaja ćelija u odnosu na kriterijume za konvergenciju iterativnog postupka

Rezultati iteracija opisanog procesa podložne su ekspertskom odlučivanju. Nije moguća jednostavna automatizacija procesa: kako u pogledu primene i provere navedenih kriterijuma, tako i u slučaju kada se donosi odluka da se proces zaustavi. Najčešći izlazni rezultat je šuplja matrica.

Osnovna namena Zahmanovog okvira je da ponudi šemu za organizovanje artifakata arhitekture informacionih sistema preduzeća, ili bilo kog drugog sistema, uvažavajući poglede različitih učesnika. Upravo zato, ćelije Zahmanovog okvira popunjene su prethodno definisanim artifaktima, nastalim modelovanjem perspektiva različitih učesnika i zainteresovanih strana, u odnosu i na poslovni sistem i na budući softverski sistem.

Rezultat procesa mapiranja je delimično popunjen Zahmanov okvir, prikazan na slici 4.3.

	Šta	Kako	Gde	Ko	Kada	Zašto
Perspektiva poslovnih izvršilaca	- Spisak zahteva - Vizija/ oblast primene	- Koncept rešenja	- Spisak i opis lokacija			
Perspektiva poslovnog upravljanja	ER dijagram ili dijagram klasa - Konceptualni model podataka	- Slučajevi korišćenja - Scenariji upotrebe - Dijagram aktivnosti / BPMN - Prototip UI - Konceptualni (domenski) model klasa	- Dijagram isporuke			
Perspektiva arhitekata	ER dijagram ili dijagram klasa - Logički model podataka	- Dijagram sekvenci - Logički model klasa - Model servisa - Dizajn UI - Arhitektura sistema	- Dijagram isporuke			
Perspektiva inženjera	ER dijagram ili dijagram klasa - Fizički model podataka	- Dijagram klasa - Specifikacija servisa - Dijagram komponenti - Implementacione tehnologije - UI ekrani	- Dijagram isporuke			
Tehnička perspektiva						
Perspektiva preduzeća						

Slika 4.3: Delimično popunjen Zahmanov okvir

Zadatak daljih analiza i iteracija je da se dobijena šuplja matrica, prikazana slikom 4.3, dimenzija 6x6, transformacijama svede na popunjenu matricu, koja ne mora biti kvadratna.

4.1.1.4 Kreiranje SOFI okvira

Na slici 4.3 prikazan je delimično popunjen Zahmanov okvir. Prateći predloženu metodologiju, u ovoj fazi izvršeno je mapiranje sadržaja delimično popunjenog Zahmanovog okvira u novi okvir: *Service Oriented Framework for Interoperability* (SOFI okvir), koji je prikazan na slici 4.4.

	Entiteti	Funkcije	Lokacija
Opis sistema i zahtevi	- Spisak zahteva - Vizija/oblast primene	- Koncept rešenja	- Spisak i opis lokacija
Poslovni model	ER dijagram ili dijagram klasa - Konceptualni model podataka	- Slučajevi korišćenja - Scenariji upotrebe - Dijagram aktivnosti / BPMN - Prototip UI - Konceptualni (domenski) model klasa	- Dijagram isporuke
Model sistema	ER dijagram ili dijagram klasa - Logički model podataka	- Dijagram sekvenci - Logički model klasa - Model servisa - Dizajn UI - Arhitektura sistema	- Dijagram isporuke
Tehnološki model	ER dijagram ili dijagram klasa - Fizički model podataka	- Dijagram klasa - Specifikacija servisa - Dijagram komponenti - Implementacione tehnologije - UI ekrani	- Dijagram isporuke

Slika 4.4: SOFI okvir

Ćelije okvira sadrže opise sistema, posmatrane iz različitih uglova. Uobičajeno, opisi su prikazani u obliku jednog ili više modela.

Razvijeni SOFI okvir podržava modelovanje od "vrha prema dnu", počev od kreiranja modela višeg stepena detaljnosti, do modela nižeg stepena detaljnosti. Modeli istog stepena detaljnosti smešteni su u isti red okvira. Redovi okvira odgovaraju perspektivama učesnika uključenim u kreiranje sistema. Učesnici, u skladu sa svojim ulogama, kreiraju potrebne modele. Kolone okvira ukazuju: na podatke koje sistem koristi; na funkcije koje sistem treba da obavlja, kao i na lokacije na kojima će se nalaziti podaci, hardverske i softverske komponente, koje će izvršavati funkcije sistema.

Prvi red okvira sagledava sistem iz perspektive poslovnih izvršilaca, koji utiču na određivanje veličine sistema, obim i njegovu osnovnu namenu. Popunjen je artifaktima kreiranim tokom faze predviđanja, a u okviru MSF modela procesa.

Drugi red okvira sadrži konceptualne modele podataka, funkcija i lokacija sistema. Modeli u ovom redu sagledavaju preduzeće iz perspektivne poslovnih analitičara. Odnose se na poslovne modele preduzeća, koji prikazuju poslovne entitete i procese, kao i njihov međusobni odnos. Modeli, smešteni u ovaj red okvira, ne prikazuju sve detalje strukture sistema i nezavisni su od platforme, na kojoj će se sistem izvršavati.

Ćelije drugog reda okvira popunjene su artifaktima konceptualnog dizajna, obezbeđenim u fazi planiranja, a u okviru MSF modela procesa:

- U kolonu, koja bi trebalo da odgovori na pitanje: "Koje će podatke koristiti ili stvarati sistem?", mapiran je ER dijagram, ili dijagram klasa, kojim su predstavljeni podaci na konceptualnom nivou.
- U kolonu, koja bi trebalo da odgovori na pitanje: "Koje funkcije sistema će se obavljati?", mapirano je više modela, koji na konceptualnom nivou definišu funkcije, procese, operacije i aktivnosti. Obuhvaćeni su: slučajevi korišćenja, scenariji upotrebe, BPMN dijagrami, prototip korisničkog interfejsa i konceptualni (domenski) model klasa.
- U kolonu, koja treba da odgovori na pitanje: "Gde će se odvijati funkcije, gde su locirane mreže i glavne lokacije poslovanja?", mapiran je UML dijagram isporuke, koji prikazuje poslovne logističke mreže, u ovom slučaju, na konceptualnom nivou.

Treći red okvira obuhvata model sistema. Pripadajući modeli nezavisni su od platforme na kojoj će se sistem izvršavati. Modeli, smešteni u ćelije u ovog reda, sagledavaju preduzeće iz perspektive analitičara, odgovornih za logički dizajn sistema. To znači, da će u ćelije ovog reda biti mapirani artifakti logičkog dizajna, nastali u fazi planiranja, a u okviru MSF modela procesa:

- U kolonu podataka mapiran je ER dijagram, ili dijagram klasa, koji prikazuje model podataka na logičkom nivou.
- U kolonu funkcija mapirano je više modela, proisteklih iz postojećih modela konceptualnog nivoa. Mapirani modeli prikazuju dalju dekompoziciju sistema.
- U kolonu lokacija, koja bi trebalo da prikaže sistemske lokacije i veze, mapiran je UML dijagram isporuke.

Osnovna namena SOFI okvira je modelovanje interoperabilnih, servisno orijentisanih sistema elektronskog poslovanja, koji su po strukturi distribuirani i koji komuniciraju preko sopstvenih granica s drugim sistemima i aplikacijama. Iz navedenih razloga, potrebno je da se u ovom redu, modelovanje funkcija i arhitekture sistema zasniva na SOA principima. Oslanjanjem na SOA principe (detaljno su objašnjeni u 2. poglavlju ove teze) tokom modelovanja, omogućeno je korišćenje autonomnih, slabo povezanih servisa, koji komuniciraju među sobom posredstvom poruka. Osim toga, značajna osobina primene servisa je mogućnost naknadnih korišćenja i kompozicije, a za potrebe izvršavanja određenih poslovnih aktivnosti.

Četvrti red okvira sadrži detaljne, tehnološke modele, kreirane u odnosu na izvršnu platformu sistema. Kreirani modeli definišu perspektivu inženjera, odgovornih za fizički dizajn sistema. S tom namenom, u ćelije ovog reda mapirani su artifakti fizičkog dizajna, nastali u fazi planiranja MSF modela procesa.

Kreirani SOFI okvir predstavlja prilagođen, objedinjen i dokumentovan pristup razvoju sistema elektronskog poslovanja:

- Prilagođen je kreiranju interoperabilnih, servisno orijentisanih sistema elektronskog poslovanja.
- Objedinjuje različite poglede učesnika tokom razvoja sistema.
- Detaljno su opisani način kreiranja i način korišćenja.

Predložen i opisan SOFI okvir, sam po sebi, ne može se efikasno koristiti, ukoliko izostane određeni broj konkretnih preporuka za primenu. U narednom poglavlju detaljnije je opisano korišćenje SOFI okvira.

4.1.2 Korišćenje SOFI okvira

Stručnjacima IT namenjen je veliki broj alata: od alata za projektovanje, programiranje, testiranje softverskih rešenja, do alata za izradu veb stranica i slično. "Pravi alat" stručnjaci IT najčešće procenjuju i biraju u zavisnosti od konkretnih potreba, na osnovu iskustva i afiniteta. Međutim, često se sreće još jedan kriterijum - verovatno i presudan: "Da li je korišćenje alata dovoljno dobro dokumentovano?". Slična konstatacija odnosi se i na korišćenja okvira za modelovanje softverskih sistema. Sa ciljem da se olakša korišćenje i izbor "pravog alata", u narednim poglavljima detaljnije je objašnjena primena SOFI okvira.

4.1.2.1 Metod za korišćenje SOFI okvira

Korišćenje SOFI okvira obezbeđuje struktuiran i kontrolisan način razvoja interoperabilnih sistema elektronskog poslovanja, zasnovanih na servisnoj orijentaciji. Metod uspostavlja redosled popunjavanja ćelija okvira, ukazuje na međuzavisnosti ćelija i definiše kriterijume za proveru sadržaja pojedinačnih ćelija i okvira u celini.

Ćelije su označene slovima, a artifaktima, pored oznake ćelije kojoj pripadaju, pridružen je redni broj. Inicijalni SOFI okvir prikazan je na slici 4.5:

	Entiteti	Funkcije	Lokacija
Opis sistema i zahtevi	A1. Spisak zahteva A2. Vizija/oblast primene	B1. Koncept rešenja	C1. Spisak i opis lokacija
Poslovni model	D1. ER dijagram ili dijagram klasa - Konceptualni model podataka	E1. Slučajevi korišćenja E2. Scenariji upotrebe E3. Dijagram aktivnosti / BPMN E4. Prototip UI E5. Konceptualni (domenski) model klasa	F1. Dijagram isporuke
Model sistema	G1. ER dijagram ili dijagram klasa - Logički model podataka	H1. Dijagram sekvenci H2. Logički model klasa H3. Model servisa H4. Dizajn UI H5. Arhitektura sistema	I1. Dijagram isporuke
Tehnološki model	J1. ER dijagram ili dijagram klasa - Fizički model podataka	K1. Dijagram klasa K2. Specifikacija servisa K3. Dijagram komponenti K4. Implementacione tehnologije K5. UI ekrani	L1. Dijagram isporuke

Slika 4.5: Inicijalni SOFI okvir sa označenim sadržajima ćelija

Popunjavanje ćelija SOFI okvira vrši se sukcesivno, po koracima koji obuhvataju kreiranje predviđenih dokumenata i modela. Opisani koraci i redosled punjenja ćelija odgovaraju razvoju novog sistema i prate smer "odozgo na dole".

1. korak

   A1 - Spisak zahteva prethodi svim drugim aktivnostima izrade rešenja. Kreira se na osnovu analize poslovnih potreba i ograničenja i opisuje šta bi sistem, koji se modeluje, trebalo da ispuni. Ne zavisi od drugih artifakata. Artifakti u ćelijama B, D i E direktno zavise od definisanih zahteva. Zahtevi se mogu redefinisati, ukoliko se tokom dizajniranja rešenja ukaže potreba za izmenama.

   A2 - Vizija/oblast primene kreira se na osnovu spiska zahteva (A1). Vizija predstavlja kratku, dovoljno jasnu i motivacionu rečenicu, koja uspostavlja zajednički pogled i obezbeđuje konsenzus tima u odnosu na buduću realizaciju sistema.

2. korak

   B1 - Koncept rešenja kreira se nakon određivanja spiska zahteva i vizije/oblasti primene (A1, A2). Koncept rešenja predstavlja grubi nacrt sistema, opisuje pristup tima pri ispunjavanju ciljeva projekta i obezbeđuje ulaze za fazu planiranja.

3. korak

   C1 - Spisak i opis lokacija obuhvata inicijalnu listu lokacija postojećih i budućih hardverskih i softverskih komponenti sistema. Predstavlja osnovu za kreiranje dijagrama isporuke na nivou poslovnog modela.

4. korak

   E1 - Slučajevi korišćenja kreiraju se na osnovu zahteva, vizije i koncepta rešenja (A1, A2, B1). Slučajevi korišćenja su UML tipovi dijagrama ponašanja i koriste se za prikaz funkcionisanja sistema iz ugla korisnika. Slučajevi korišćenja predstavljaju procese sistema i uključuju sve moguće događaje i situacije. Koriste se za identifikaciju sistema, učesnika, definisanje interakcija između učesnika i sistema i određivanje granice sistema.

   E2 - Scenariji upotrebe kreiraju se na osnovu slučajeva korišćenja (E1). Scenariji upotrebe predstavljaju detaljan i potpuni semantički opis slučajeva korišćenja. Uobičajeno sadrže preduslove i završne uslove, dele aktivnosti u diskretne korake i identifikuju izuzetke. Veoma su korisni za kasnije faze dizajna sistema, mada se u praksi retko pišu, jer njihova izrada zahteva dosta vremena. Tokom izrade, potrebno je proveriti svaki slučaj korišćenja i scenarija upotrebe, a u odnosu na postojeće zahteve, ostalu raspoloživu dokumentaciju i uz saradnju svih zainteresovanih strana za sistem.

   E3 - BPMN dijagrami koriste se za prikaz modela poslovnih procesa i kreiraju se na osnovu slučajeva korišćenja i scenarija upotrebe (E1, E2). BPMN dijagrami namenski su napravljeni da omoguće predstavljanje modela poslovnih procesa, kako složenih, koji se sastoje od više aktivnosti, tako i jednostavnih, koji se sastoje od jedne aktivnosti. U kasnijim fazama modelovanja, prikazane BPMN aktivnosti predstavljaju kandidate za buduće servise. Za prikaz modela poslovnih procesa mogu se koristiti i UML dijagrami aktivnosti, međutim,

preporuka je da se koriste BPMN dijagrami, zbog sposobnosti da predstave sve aspekte modelovanih poslovnih procesa.

E4 - Prototip korisničkog interfejsa kreira se u slučajevima kada postoji front-end aplikacija, na primer portal, koja će komunicirati sa servisima, ili klasama. Kreira se na osnovu slučajeva korišćenja, scenarija upotrebe i BPMN dijagrama (E1, E2, E3). U ovoj fazi prototip se može predstaviti skicom, bez ulaženja u detaljan izgled korisničkog interfejsa. Obezbeđuje prikaz glavne strukture i osobine korisničkog interfejsa i ilustruje moguće puteve kretanja korisnika kroz aplikaciju. Takođe, omogućava da se u komunikaciji sa korisnicima brzo identifikuju potencijalni problemi i istraži alternativni dizajn, sa ciljem izrade interfejsa, koji će korisnici prihvatiti.

E5 - Konceptualni (domenski) model klasa kreira se na osnovu slučajeva korišćenja, scenarija upotrebe i BPMN dijagrama (E1, E2, E3). Predstavlja poslovne objekte na visokom nivou, uspostavlja termine, kojima se opisuje sistem i prikazuje relacije između objekata sistema. Može se koristiti tokom verifikacije i provere razumevanja domena problema.

5. korak

D1 - Konceptualni model podataka definiše informacije važne za sistem (entitete) i asocijacije između parova entiteta (relacije). Predstavlja semantiku sistema. Grafički se predstavlja pomoću Entity Relationship (ER) dijagrama, ili dijagrama klasa. Konceptualni model podataka radi se na osnovu zahteva, slučajeva korišćenja i scenarija upotrebe (A1, E1, E2).

6. korak

F1 - Dijagram isporuke obezbeđuje statički pogled na konfiguraciju čvorova sistema i komponenti, koje će se izvršavati u tim čvorovima. Grubi model isporuke definiše se tokom modelovanja poslovanja. Kasnijom razradom, tokom modelovanja sistema i kreiranja tehnološkog modela, rade se detaljniji dijagrami (I1, L1), koji u potpunosti definišu preporučenu hardversku i softversku konfiguraciju sistema koji se realizuje.

7. korak

H1 - Dijagram sekvenci kreira se na osnovu BPMN dijagrama, slučajeva korišćenja i scenarija upotrebe (E1, E2, E3). Prikazuje interakcije između objekata sistema, u sekvencijalnom redosledu izvršavanja. Važna karakteristika dijagrama sekvenci je praćenje vremena po vertikalnoj osi: interakcija počinje pri vrhu dijagrama i završava se pri dnu. Paralelne, vertikalne linije prikazuju procese ili objekte, koji istovremeno postoje. Horizontalne linije prikazuju poruke, koje razmenjuju procesi ili objekti, po redosledu u kome su nastale. Dijagrami sekvenci pogodni su za prikaz dinamičkog ponašanja sistema, koji sadrži mali broj procesa i objekata.

H2 - Logički model klasa kreira se na osnovu slučajeva korišćenja, BPMN dijagrama, domenskog modela i dijagrama sekvenci i (E1, E3, E5, H1). Ukoliko postoje, najbolje je koristiti scenarija upotrebe (E2), jer su najdetaljniji. Klase su ljudi, ili stvari, preuzeti iz modela, na osnovu kojih se kreira logički model klasa. Funkcija predstavlja određenu akciju, koju poslovni objekat treba da izvrši. Akcija može biti operacija, ili transformacija, koja se može primeniti na objekat, ili je objekat može izvršiti. Funkcije se koriste da implementiraju poslovna pravila, manipulišu podacima, ili pristupaju informacijama. Funkcija može izvršiti bilo koju aktivnost, koja se može opisati skupom pravila. Za identifikaciju funkcije, potrebno

je odrediti: šta objekat treba da radi, vrstu podataka koju treba da održava i koje akcije treba da izvrši; atribute objekata i relacije, koje postoje između njih. Identifikovane funkcije dodeljuju se uočenim klasama.

H3 - *Model servisa* kreira se nakon logičkog modela klasa, izdvajanjem klasa i njihovih odgovarajućih funkcija, koje se mogu implementirati servisima. Preporučen način, za identifikovanje servisa, zasniva se na povezivanju aktivnosti iz BPMN dijagrama sa funkcijama iz modela klasa. Neke aktivnosti poslovnog procesa mogu se lako svrstati u aktivnosti, koje se ne izvršavaju korišćenjem servisa. Na primer, neke manuelne aktivnosti, ne mogu se automatizovati, ili ne postoji potreba za automatizacijom. Istovremeno, neke druge aktivnosti su dobri kandidati za izvršavanje pomoću servisa. Kao dodatni kriterijum, može se posmatrati da li odvijanje određene aktivnosti poslovnog procesa izlazi iz granica jednog sistema i prelazi u druge, susedne sisteme, i da li je potrebno uspostavljanje interoperabilnosti. Dobro je proveriti da li identifikovani servisi zadovoljavaju poslovne zahteve i ciljeve. Izdvojeni servisi ne čine funkcionalni sistem, postoje samostalno, i uobičajeno su deo eksterne kompozicije, koja određuje vreme pozivanja i redosled izvršavanja servisa. Nakon identifikacije servisa, potrebno je opredeliti se za samostalno kreiranje, ili korišćenje već raspoloživih servisa.

H4 - Dizajn korisničkog interfejsa na logičkom nivou kreira se na osnovu logičkog modela klasa (H2) i prototipa korisničkog interfejsa (E4), koji je nacrtan tokom konceptualnog dizajna. Dizajn UI, na ovom nivou, pomaže da se opiše tok komunikacije sistema sa korisnicima, elementima korisničkog interfejsa, kao što su: meniji; dugmad; tekstualna polja; liste i slično.

H5 - Arhitektura sistema opisuje strukturu sistema i način sprovođenja funkcionalnih i integracionih zahteva sistema. U slučaju SOA sistema, arhitektura prikazuje način na koji servisi, nezavisni od platforme, u ulozi provajdera i klijenta, komuniciraju među sobom, a sa ciljem da podrže poslovne procese. Tokom međusobne komunikacije, servisi koriste XML standarde i protokole poput: WSDL jezika za opis servisa i SOAP protokola za razmenu poruka.

8. korak

G1 - Logički model podataka kreira se na osnovu konceptualnog modela podataka, ili dijagrama klasa (D1). Prethodno uočenim entitetima, ili klasama, pridružuju se atributi, primarni i strani ključevi. Logički model podataka koristi poslovna imena za entitete i atribute i nezavistan je od tehnologije. Potrebno je da bude normalizovan.

9. korak

I1 - Dijagram isporuke definiše na visokom nivou isporuku komponenti sistema. Ukoliko je definisan dijagram isporuke pri kreiranju poslovnog modela (F1), on predstavlja osnovu za kreiranje dijagrama isporuke na ovom nivou. Za većinu sistema u praksi, dijagram isporuke kreira se tokom izrade tehnološkog modela sistema.

10. korak

K1 - Dijagram klasa kreira se na osnovu logičkog dijagrama klasa (H2). Obezbeđuje transformaciju logičkih klasa u definicije klasa, uključujući: njihove interfejse; identifikaciju novih klasa, koje prethodno nisu uočene i ukoliko je potrebno, konsolidaciju klasa.

K2 - Specifikacija servisa kreira se na osnovu modela servisa (H3). U ovom koraku se određuje način implementacije i detaljni dizajn servisa. Servisi se mogu implementirati kao: komponente; veb servisi i REST servisi. Trenutno, dominantan način realizacije servisne orijentacije je pomoću veb servisa. Veb servisi podržavaju interoperabilnost preko skupa standarda, zasnovanih na XML jeziku, kao što su WSDL, SOAP, UDDI. Potrebno je definisati ugovor veb servisa, odnosno identifikovati elemente i odgovarajuće tipove podataka, kao i način interakcije sa klijentima.

K3 - Dijagram komponenti kreira se na osnovu dijagrama klasa (K1) i omogućava logičko pakovanje klasa u komponente. Svrha kreiranja dijagrama komponenti je određivanje komponenti i definisanje zavisnosti između njih.

K4 - Implementacione tehnologije koriste se za razvoj sistema i sastoje se od: programskih jezika i interfejsa za programiranje aplikacija (API); servera i serverskih tehnologija, kao i drugih tehnologija, potrebnih za implementaciju sistema. Važno je identifikovati ih, jer utiču na način razvoja sistema i plan nabavke potrebnog softvera.

K5 - Ekrani korisničkog interfejsa prikazuju detaljan izgled ekrana, preko kojih sistem komunicira sa korisnicima. Ukoliko je definisan, dizajn korisničkog interfejsa na nivou modela sistema (H4), čini osnovu za kreiranje detaljnog prikaza ekrana. Međutim, u slučaju velikog broja sistema, dizajn korisničkog interfejsa kreira se tek u ovom koraku modelovanja sistema.

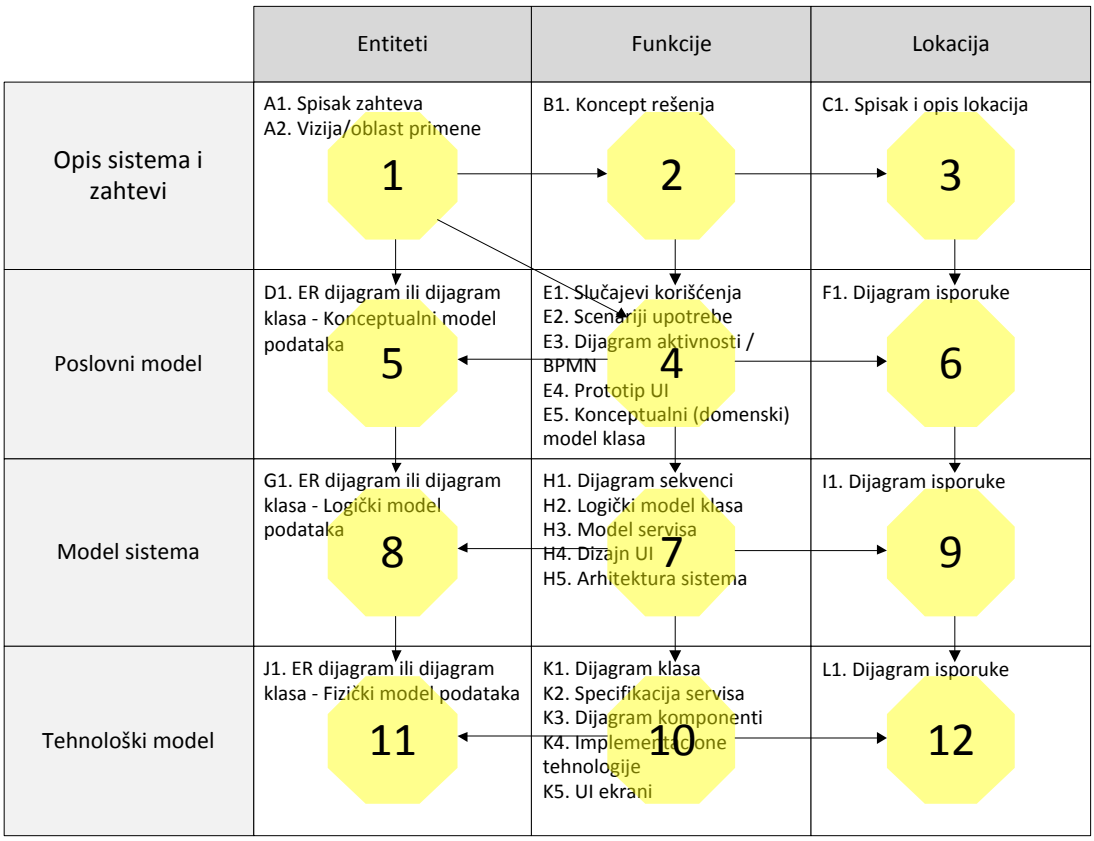
11. korak

J1 - Fizički model podataka kreira se za na osnovu logičkog modela (G1) saglasno sintaksi izabrane platforme, na kojoj će podaci biti skladišteni. Definiše imena tabela i kolona; tipove podataka i sve artifakte baze podataka, potrebne za kreiranje relacija između tabela i postizanje željenih performansi, kao što su: indeksi; definicije ograničenja; povezane tabele; particionisane tabele, ili klasteri. Može se koristiti i za procenu potrebnog prostora za odabrani sistem baze podataka.

12. korak

L1 - Dijagram isporuke definiše: mrežnu topologiju, koja ukazuje na hardverske lokacije i njihovo povezivanje; topologiju podataka i komponenti, koja ukazuje na lokacije komponenti sistema, servisa i podataka, saglasno mrežnoj topologiji. Ukoliko je prethodno definisan model isporuke na logičkom nivou (I1), koristiće se za kreiranje ovog, detaljnog dijagrama isporuke.

SOFI okvir, prikazan na slici 4.5, odredio je sadržaj ćelija, a opisani koraci upućuju na redosled punjenja ćelija i međuzavisnost sadržaja ćelija. Primena prethodno navedenih pravila za punjenje SOFI okvira, šematski je prikazana na slici 4.6.



Slika 4.6: Redosled popunjavanja i međuzavisnosti sadržaja ćelija SOFI okvira

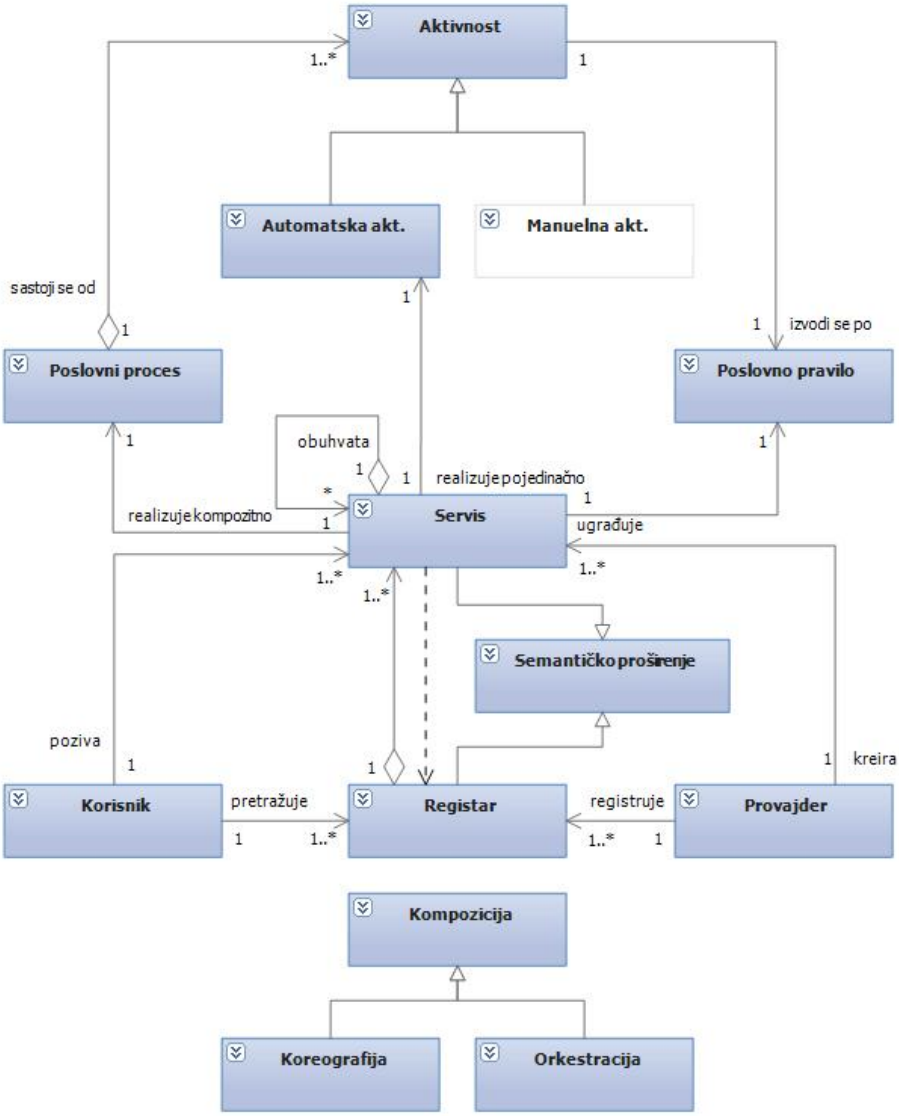
Popunjeni SOFI okvir namenjen je da obezbedi potrebna dokumenta i modele za razvoj servisno orijentisanog sistema elektronskog poslovanja. Uključene napomene ukazuju na uloge dokumenata i modela, mogućnost obaveznog ili opcionog uključenja u okvir, ili mogućnost da se navedeni dokument ili model zameni nekim drugim, odgovarajućim. Opisani metod popunjavanja SOFI okvira primenjen je tokom razvoja Portala za pretraživanje i prezentaciju kulturnih dobara u Srbiji, koji je predstavljen u poglavlju 5. ove teze.

4.1.2.2 Arhitektura sistema kreiranih pomoću SOFI okvira

Modelovanje arhitekture predstavlja preduslov za efikasan razvoj i upravljanje informacionim sistemima preduzeća. Uočena je mogućnost da modeli arhitekture predviđaju ponašanje informacionih sistema, delujući u svom okruženju i pre implementacije i puštanja sistema u rad. Modeli arhitekture omogućavaju sagledavanje posledica različitih mogućih scenarija, podržavajući donošenje poslovnih odluka. Zbog uticaja na tip i organizaciju sistema, model arhitekture jedan je od artifakata SOFI okvira.

SOFI okvir trebalo bi da omogući struktuirano kreiranje servisno orijentisanih sistema elektronskog poslovanja. Fokus servisno orijentisanog modela i podržanih tehnologija usmeren je ka interoperabilnosti i rešavanju praktičnih problema, koji se javljaju zbog razlika u platformama i programskim jezicima delova sistema, koje je potrebno povezati. S obzirom da prirodno podržavaju slabo povezivanje, ugovore, apstrakciju i mogućnost kompozicije, dobar izbor za realizaciju servisno orijentisanih sistema su veb servisi. Osnovna svrha kreiranja veb servisa je izvršavanje pojedinačnih poslovnih aktivnosti ili, ukoliko se uključe u kompoziciju, izvršavanje celovitih poslovnih procesa.

Analizirajući moguće načine za kreiranje arhitekture sistema, autor ove teze razvio je meta model arhitekture, koji preporučuje za modelovanje i organizovanje elemenata, neophodnih za konstituisanje servisno orijentisanih sistema elektronskog poslovanja. Izvedeni meta model arhitekture sistema sastavni je deo SOFI okvira i predstavljen je na slici 4.7.



Slika 4.7: Meta model arhitekture sistema

Predloženi meta model arhitekture sistema zasnovan je na povezivanju: servisno orijentisane arhitekture i modela poslovnih procesa; na konzistentnom, semantičkom proširenju registra i servisa, i prikazuje sastavne delove arhitekture sistema, kao i relacije između uključenih delova sistema. Elementi arhitekture obuhvataju poslovne procese i aktivnosti, koje se izvode na osnovu poslovnih pravila, kao i realizaciju poslovnih aktivnosti pomoću servisa.

U prikazanom meta modelu arhitekture, važne uloge preuzimaju provajder, korisnik i registar servisa. Provajder servisa kreira i publikuje servise. Korisnik servisa pronalazi publikovane servise, povezuje se sa njima i koristi ih. Treću ulogu preuzima registar servisa, koji sadrži informacije o dostupnim servisima. Ne mora da postoji, ili da se koristi, u situacijama kada su lokacije potrebnih

servisa unapred poznate, ili kada provajderi servisa žele da kontrolišu korišćenje svojih servisa. Registar servisa može da bude javno dostupan preko Interneta, ili da postoji interno, u okviru preduzeća.

Sistemi elektronskog poslovanja najčešće su po prirodi složeni, pa implementacija obično zahteva pozivanje više veb servisa. Istovremeno pozivanje više veb servisa zahteva agregaciju njihovih funkcionalnosti, koja se može obaviti kompozicijom potrebnih veb servisa, preko orkestracije ili koreografije.

Zbog potrebe da se omogući čuvanje semantičkih podataka o servisima i omogući njihovo automatsko registrovanje i otkrivanje, u predloženi meta model arhitekture sistema uključen je element "Semantičko proširenje". Semantičko proširenje može se izvesti:

- pridruživanjem semantičkog opisa u postojeći opis veb servisa, korišćenjem jezika za semantičko označavanje (RDF, OWL);
- dodavanjem semantičkih oznaka na standardne elemente registra servisa (tModel);
- kreiranjem jednog ili više nestandardnih, dodatnih elemenata, koji podržavaju semantičko opisivanje servisa, slično centralizovanom "svi-prema-jednom" modelu, koji je prikazan na slici 3.16.

Semantičko proširenje nudi mogućnost da se servisi konzistentno opišu istom semantikom, zasnovanom na domenskim ontologijama. Opseg primenljivosti uslovljen je isključivo vrstom domenske ontologije, kojom se opisuju razmatrani, realni sistemi. Rezultat je da semantička struktura predloženog meta modela arhitekture sistema omogućava semantičko povezivanje servisno orijentisanih sistema.

4.1.2.3 Grupisanje ćelija okvira i podržani pogledi na sistem

Kombinovanjem i grupisanjem sadržaja ćelija kreiranog SOFI okvira, stiče se opšti ili celovit pogled na sistem; definisani su SOA domeni i naznačen je odnos BPM i SOA. Na ovaj način razgraničene grupe ćelija okvira zajedno opisuju sistem elektronskog poslovanja, zasnovan na servisnoj orijentaciji.

1. Opšti pogled na sistem

Opšti pogled na sistem definisan je sadržajima svih ćelija SOFI okvira. Ćelije SOFI okvira grupisane su po fazama životnog ciklusa procesa razvoja softverskih rešenja, primenom sledeće šeme:

- analiza i specificiranje zahteva;
- konceptualni dizajn;
- logički dizajn i
- fizički dizajn.

Potrebni artifakti, saglasno MSF modelu procesa, kreiraju se u okviru navedenih faza životnog ciklusa procesa razvoja softvera.

Analiza i specificiranje zahteva obuhvata određivanje granica, obima i namene sistema, koji se kreira. U ovoj fazi analiziraju se i definišu poslovni zahtevi, ograničenja i cilj kreiranja sistema i navodi se spisak i opis lokacija delova sistema. Dobijeni rezultati obezbeđuju sadržaje ćelijama: A; B i C.

Konceptualni dizajn sagledava sistem iz ugla poslovnih korisnika. Tokom ove faze prezentuju se rešenja na visokom nivou. Zahtevi se modeluju kreiranjem: slučajeva korišćenja i scenarija upotrebe; modela konteksta (skice ekrana) i konceptualnog modela podataka. Dobijeni rezultati obezbeđuju sadržaje sledećim ćelijama: D; E i F.

Logički dizajn sagledava sistem iz ugla arhitekata sistema. Rezultati ove faze su: klase i servisi; korisnički interfejs; arhitektura sistema i logički model podataka. Dobijeni rezultati obezbeđuju sadržaje sledećim ćelijama: G; H i I.

Fizički dizajn sagledava sistem iz ugla inženjera. Cilj ove faze je da se odrede osnove za implementaciju sistema: dijagram klasa i specifikacija servisa; dijagram komponenti; izabrane tehnologije; arhitektura i dizajn infrastrukture; opis UI ekrana i fizički model podataka. Dobijeni rezultati obezbeđuju sadržaje sledećim ćelijama: J; K i L.

Grupisane ćelije SOFI okvira, prema fazama životnog ciklusa razvoja softvera, prikazane su na slici 4.8.

	Entiteti	Funkcije	Lokacija	
Opis sistema i zahtevi	A1. Spisak zahteva A2. Vizija/oblast primene	B1. Koncept rešenja	C1. Spisak i opis lokacija	*Analiza i specificiranje zahteva*
Poslovni model	D1. ER dijagram ili dijagram klasa - Konceptualni model podataka	E1. Slučajevi korišćenja E2. Scenariji upotrebe E3. Dijagram aktivnosti / BPMN E4. Prototip UI E5. Konceptualni (domenski) model klasa	F1. Dijagram isporuke	*Konceptualni dizajn*
Model sistema	G1. ER dijagram ili dijagram klasa - Logički model podataka	H1. Dijagram sekvenci H2. Logički model klasa H3. Model servisa H4. Dizajn UI H5. Arhitektura sistema	I1. Dijagram isporuke	*Logički dizajn*
Tehnološki model	J1. ER dijagram ili dijagram klasa - Fizički model podataka	K1. Dijagram klasa K2. Specifikacija servisa K3. Dijagram komponenti K4. Implementacione tehnologije K5. UI ekrani	L1. Dijagram isporuke	*Fizički dizajn*

Slika 4.8: Grupisane ćelije SOFI okvira prema fazama životnog ciklusa razvoja softvera

Izvedeni SOFI okvir uključuje prikaz različitih pogleda, koji zajedno opisuju sistem elektronskog poslovanja, zasnovan na servisnom razvoju softvera. Polazeći od prikazanog sadržaja ćelija SOFI okvira, a promenom ugla posmatranja, moguće je definisati SOA domene i odnos između SOA i BPM.

2. SOA domeni

Definisanje SOA domena vrši se posmatranjem ćelija SOFI okvira, prikazanog na slici 4.5, iz drugog ugla, a koristeći sledeću šemu:

- SOA poslovni domen;
- SOA informacioni domen;
- SOA servisni domen i
- SOA infrastrukturni domen.

	Entiteti	Funkcije	Lokacija
Opis sistema i zahtevi	A1. Spisak zahteva A2. Vizija/oblast primene	B1. Koncept rešenja	C1. Spisak i opis lokacija
Poslovni model	D1. ER dijagram ili dijagram klasa - Konceptualni model podataka	E1. Slučajevi korišćenja E2. Scenariji upotrebe E3. Dijagram aktivnosti / BPMN E4. Prototip UI E5. Konceptualni (domenski) model klasa	F1. Dijagram isporuke
Model sistema	G1. ER dijagram ili dijagram klasa - Logički model podataka	H1. Dijagram sekvenci H2. Logički model klasa H3. Model servisa H4. Dizajn UI H5. Arhitektura sistema	I1. Dijagram isporuke
Tehnološki model	J1. ER dijagram ili dijagram klasa - Fizički model podataka	K1. Dijagram klasa K2. Specifikacija servisa K3. Dijagram komponenti K4. Implementacione tehnologije K5. UI ekrani	L1. Dijagram isporuke
	Informacioni domen	*Servisni domen*	*Infra-strukturni domen*

Poslovni domen

Slika 4.9: SOA domeni u SOFI okviru

SOA poslovni domen uključuje model poslovnih procesa, opisuje komponovanje poslovnih procesa od servisa, navodi semantičke podatke, koje model koristi i opisuje odnos servisa definisanih modelom. U poređenju sa arhitekturom preduzeća, SOA poslovni domen delimično odgovara poslovnoj arhitekturi preduzeća, s obzirom da ne definiše uloge ljudi i njihove odgovornosti, u odnosu na kreiranje strukture i opis ponašanja sistema. SOA poslovni domen odnosi se na sadržaje sledećih ćelija: D; E i F.

SOA informacioni domen uključuje zajedničku SOA semantiku i opisuje koji su i kako su definisani zajednički semantički podaci, uključujući njihov odnos sa ostalim operacionim i analitičkim podacima. U poređenju sa arhitekturom preduzeća, SOA informacioni domen odgovara informacionoj arhitekturi preduzeća, jer opisuje strukturu, skladištenje i lokaciju podataka. SOA informacioni domen odnosi se na sadržaje sledećih ćelija: D; G i J.

SOA servisni domen obuhvata sva rešenja SOA i opisuje: načine kreiranja servisa; kako su postojeće aplikacije izložene kao servisi; komponovanje poslovnih procesa od servisa; načine na koji servisi podržavaju druge aplikacije i u kakvom su odnosu sa drugim aplikacijama, kao što su portali, B2B aplikacije i slično. U poređenju sa arhitekturom preduzeća, SOA servisni domen odgovara aplikacionoj arhitekturi preduzeća. SOA servisni domen odnosi se na sadržaje sledećih ćelija: E; H i K.

SOA infrastrukturni domen opisuje način na koji infrastruktura servisa podržava servise, distribuciju servisa i način na koji se SOA infrastrukturni domen integriše u ostali deo infrastukture preduzeća. U odnosu na arhitekturu informacionih sistema preduzeća, SOA infrastrukturni domen delimično odgovara tehnološkoj arhitekturi preduzeća, s obzirom da ne opisuje u potpunosti tehnološku arhitekturu, protokole i mreže. SOA infrastrukturni domen odnosi se na sadržaje sledećih ćelija: F, I, L.

3. Odnos između BPM i SOA

BPM bavi se kreiranjem poslovnih procesa sistema, nadgledanjem izvršavanja, njihovom optimizacijom i podešavanjem performansi. U tom kontekstu, svaki poslovni proces modelovan je kao skup individualnih aktivnosti, koji se u sistem implementiraju preko servisa.

Imajući u vidu definiciju SOA sa tehničkog aspekta, SOA je opisana u SOFI okviru presekom kolone "Funkcije" i reda "Model sistema". Lokacija rezultujuće ćelije pozicionira SOA u odnosu na celokupnu arhitekturu sistema. Međutim, po definiciji SOA sa poslovnog aspekta, SOA obuhvata poslovne procese realizovane kompozicijom servisa, prikazuje kako sistemi dele i predstavljaju informacije (preko semantičkih, logičkih i fizičkih modela podataka) i opisuje lokacije izvršavanja servisa u mreži.

Imajući u vidu, da bi servisno orijentisani sistemi elektronskog poslovanja trebalo da budu fokusirani na poslovne procese, kao i da bi upravljanje poslovnim procesima kod sistema elektronskog poslovanja trebalo da bude servisno orijentisano, značajno je da se model SOA proveri i usaglasi sa rezultatima BPM.

Pogled na odnos BPM - SOA definiše se sadržajem ćelija SOFI okvira, primenom sledeće šeme:

- BPM - modelovanje poslovanja i procesa obuhvaćeno je sadržajem ćelija: A; B; C; D; E i F;
- SOA - servisno orijentisana arhitektura obuhvaćena je sadržajem sledećih ćelija: D; E; F; G; H i I.

	Entiteti	Funkcije	Lokacija	
Opis sistema i zahtevi	A1. Spisak zahteva A2. Vizija/oblast primene	B1. Koncept rešenja	C1. Spisak i opis lokacija	*BPM*
Poslovni model	D1. ER dijagram ili dijagram klasa - Konceptualni model podataka	E1. Slučajevi korišćenja E2. Scenariji upotrebe E3. Dijagram aktivnosti / BPMN E4. Prototip UI E5. Konceptualni (domenski) model klasa	F1. Dijagram isporuke	
Model sistema	G1. ER dijagram ili dijagram klasa - Logički model podataka	H1. Dijagram sekvenci H2. Logički model klasa H3. Model servisa H4. Dizajn UI H5. Arhitektura sistema	I1. Dijagram isporuke	*SOA*
Tehnološki model	J1. ER dijagram ili dijagram klasa - Fizički model podataka	K1. Dijagram klasa K2. Specifikacija servisa K3. Dijagram komponenti K4. Implementacione tehnologije K5. UI ekrani	L1. Dijagram isporuke	

Slika 4.10: Odnos BPM i SOA

Posmatrajući SOA arhitekturu u odnosu na BPM, uočava se da postoji preklapanje u drugom redu SOFI okvira, što ukazuje da su BPM i SOA međusobno zavisni i da se moraju sinhrono primenjivati pri dizajniranju sistema.

4.1.2.4 Opšte preporuke za korišćenje SOFI okvira

Korišćenje SOFI okvira trebalo bi da podrži razvoj interoperabilnih, servisno orijentisanih sistema elektronskog poslovanja, koji uključuju specifične zahteve. Prilikom korišćenja okvira, osim primene prethodno opisanog metoda za popunjavanje ćelija, potrebno je uzeti u obzir i sledeće, opšte preporuke:

- Koristiti prihvaćene, na XML-u zasnovane, standarde i specifikacije. Oni su kreirani da bi uspostavili konzistentnost između uređaja, platformi i sistema, i već samom njihovom upotrebom omogućava se određena interoperabilnost.
- Poslovne zahteve definisati preko poslovnih procesa, a aktivnosti u okviru poslovnog procesa povezati sa servisima, koji će preuzeti na sebe izvršenje konkretnog zadatka.
- Servise kreirati kao veb servise, zbog njihove zasnovanosti na otvorenim standardima i specifikacijama, i zato što se mogu publikovati, pronaći i pokrenuti preko veba.
- Kreirati veb servise tako da budu slabo povezani, autonomni, da ne čuvaju stanje i da se mogu otkriti.
- Razmotriti kreiranje, ili korišćenje već postojećih ontologija, kako bi se uspostavila semantička interoperabilnost sistema, koji međusobno komuniciraju.
- Razmotriti korišćenje generičkog meta modela, koji će poslužiti za kreiranje i proveru semantičke konzistentnosti kolekcija artifakata, kojima se popunjavaju ćelije SOFI okvira.
- Koristiti generički meta model za uspostavljanje interoperabilnosti sistema, koji se dizajnira.

4.1.3 Poređenje SOFI okvira sa drugim izvedenim okvirima

Na prvi pogled, izgleda da je tema kreiranja novog okvira na osnovu Zahmanovog okvira, istražena detaljno od strane drugih istraživača. Većina predloženih okvira ima za cilj da Zahmanov okvir dopuni metodama i alatima, koji pokrivaju sve aspekte sistema, koji se modeluje.

Najpoznatiji okviri, izvedeni iz Zahmanovog okvira, su Federal Enterprise Architecture Framework (FEAF), Treasury Enterprise Architecture Framework (TEAF), Enterprise Architecture Planning (EAP). Izvedeni su sa ciljem da opišu arhitekturu celokupnog preduzeća.

Pored navedenih, veoma poznatih okvira, u stručnim radovima mogu se pronaći različite, ne toliko poznate i korišćene, interpretacije Zahmanovog okvira. Među predloženim okvirima postoje određene razlike.

U radu [83] autori predlažu korišćenje UML poslovnih profila za opis ćelija u drugom i trećem redu Zahmanovog okvira, sa ciljem da se prevaziđe nepostojanje dobro definisane metodologije i konkretne notacije za definisanje arhitekture informacionih sistema preduzeća. U radu su prikazani rezultati upitnika, koji su popunjavali domenski eksperti različitih profila. Analiza rezultata pokazala je da su UML dijagrami za kreiranje različitih modela sistema dobar izbor, jer su poznati većini ispitanika.

U radu [97], sa ciljem da se dobiju kompatibilni metodi za sve ćelije Zahmanovog okvira, autori predlažu primenu integrisanog jezika, koji je zasnovan na Model-Driven Architecture (MDA). Na osnovu provere u praksi, autori smatraju da predloženi metod ima prednosti u odnosu na druge metode, koje su analizirali u svom radu. Za opis opšte arhitekture informacionih sistema preduzeća koriste Zahmanov okvir, ne objašnjavajući kako da se kreiraju servisno orijentisana rešenja.

U radu [98], predlaže se metod, koji pruža smernice za razvoj arhitekture informacionih sistema preduzeća, popunjavanjem Zahmanovog okvira artifaktima, u smeru od vrha prema dnu. Za tu namenu koriste sopstveni alat. Takođe, uvode novi koncept - "veznu ćeliju", koju koriste da iskažu semantički odnos između ćelija u redovima Zahmanovog okvira.

U radu [96], Zahmanovom okviru dodaju novu, treću dimenziju i ilustruju korišćenje proširenog Zahmanovog okvira, preko različitih aspekata i njihovih projekcija. Za ilustraciju potreba proširenja Zahmanovog okvira, navode četiri dostupna pristupa za korišćenje UML jezika: Rational Unified Process (RUP), Popkin Process for Enterprise Architecture, Select Perspective i Model Driven Architecture (MDA). U radu se ne daje metod za popunjavanje okvira.

Predložena rešenja uglavnom pokušavaju da odrede sopstvene perspektive posmatranja sistema i da, na različite načine, pokriju ćelije izvedenog okvira stereotipima dijagrama klasa. Nijedan od prikazanih okvira ne predlaže način za obezbeđenje servisno orijentisanih rešenja. Izuzev autora rada [98], nijedan od prikazanih radova ne predlaže metod za razvoj arhitekture informacionih sistema preduzeća, niti prikazuje zavisnosti između ćelija okvira. Takođe, evidentan je nedostatak opisa artifakata, kojima bi ćelije okvira trebalo da budu popunjene, kao i način određivanja artifakata.

SOFI okvir, opisan u ovoj tezi, kreiran je sa ciljem da se podrži izgradnja sistema elektronskog poslovanja, zasnovanog na servisnoj orijentaciji. Kreirani okvir izveden je iz Zahmanovog okvira. Ćelije Zahmanovog okvira popunjavane su artifaktima, koji odgovaraju fazama životnog ciklusa razvoja softverskih sistema. Dobijeni, delimično popunjeni Zahmanov okvir, transformisan je u novi - SOFI okvir. Sa ciljem da se olakša primena SOFI okvira, predložen je metod za njegovo korišćenje, koji ukazuje na redosled popunjavanje ćelija i uspostavljanje njihove međuzavisnosti. Okvir ne propisuje broj artifakata, koje svaka ćelija treba da sadrži, već preporučuje njihove moguće tipove. Implementacija tipova artifakata, predloženog okvira, usaglašava se sa potrebama sistema elektronskog poslovanja, koji se razvija. Metod za primenu SOFI okvira sugeriše upotrebu procesa za predstavljanje logički povezanih poslovnih aktivnosti. Za modelovanje poslovnih procesa predlaže se izrada BPMN dijagrama. BPMN dijagrami omogućavaju formalizovano uočavanje zadataka i aktivnosti poslovnog procesa, koji se realizuju, delimično ili u potpunosti, posredstvom servisa. Preporučeno je korišćenje veb servisa, čime se zadovoljavaju kritični preduslovi za uspostavljanje interoperabilnosti.

SOFI okvir je celovit metodološki okvir, koji: specificira kolekciju artifakata, kojima se popunjavaju ćelije okvira; obezbeđuje poglede na sistem iz ugla različitih korisnika; daje metod za popunjavanje ćelija okvira; daje opšte preporuke za primenu; objašnjava kako kreirati servisno orijentisana rešenja; preporučuje način za povezivanje modela poslovnih procesa sa modelom servisa; uključuje meta model arhitekture sistema.

Na osnovu navedenih osobina može se zaključiti da SOFI okvir prevazilazi određene nedostatke prethodno opisanih okvira, pa se može preporučiti za modelovanje servisno orijentisanih sistema elektronskog poslovanja.

4.2 Razvoj generičkog meta modela za interoperabilnost

Izbor i primena odgovarajućeg okvira predstavlja početak aktivnosti, čiji je cilj izgradnja arhitekture sistema. Aktivnosti projektnog tima nastavljaju se popunjavanjem ćelija okvira. Svaka ćelija okvira, iz posebne perspektive, opisuje pogled na sistem. Opisani pogled obuhvata jedan, ili više modela. Tako obezbeđeni sadržaji (modeli, spiskovi, ili slobodni tekstualni opisi), predstavljaju meta podatke za kreiranje arhitekture. Tokom popunjavanja ćelija okvira, menja se apstraktnost: polazi se od gornjih redova i opštih modela, da bi se donji redovi okvira popunjavali detaljnijim modelima. Ukoliko postoje i odgovaraju, koriste se već urađeni modeli. U suprotnom, do potrebnih modela dolazi se meta modelovanjem.

Cilj meta modelovanja je izgradnja kolekcije koncepata, u okviru određenog domena [99]. Pri tome, na različitim nivoima apstrakcije, nastaju dve vrste artifakata: modeli i meta modeli. Model je apstrakcija pojava u realnom svetu, dok je meta model apstrakcija samog modela, koja naglašava njegovu generičku strukturu.

Meta model obuhvata konstrukcije i pravila, potrebna za kreiranje specifičnih modela u oblasti posmatranog domena i može se posmatrati iz tri različite perspektive [100]:

- kao skup gradivnih blokova i pravila za kreiranje modela;
- kao model posmatranog domena i
- kao instanca nekog drugog modela.

U slučaju planiranja sistema elektronskog poslovanja, značajno je obezbediti efektivnu razmenu poslovnih transakcija i informacija sa drugim sistemima. Dobra osnova za uspostavljanje interoperabilnosti, posebno u slučaju labavo povezanih sistema, su poslovni koncepti i poslovni termini, opisani meta modelima. U tom slučaju, meta modeli mogu se koristiti [101]:

- kao šema za semantičke podatke, koji se razmenjuju ili pamte;
- kao jezik, koji podržava određeni metod ili proces i
- kao jezik za izražavanje dodatne semantike postojećih informacija.

Osnovna namena SOFI okvira je da podrži razvoj servisno orijentisanih sistema elektronskog poslovanja, uspostavljanjem semantičke interoperabilnosti. Imajući u vidu mogućnosti primene meta modela [101], u narednom poglavlju biće predstavljen generički meta model, namenjen za kreiranje i proveru semantičke konzistentnosti kolekcija artifakata, kojima se popunjavaju ćelije SOFI okvira.

4.2.1 Meta modelovanje

Uspostavljanje interoperabilnosti u sistemima elektronskog poslovanja, zasnovano na korišćenju prikazanog SOFI okvira, moguće je obezbediti semantičkim povezivanjem uključenih sistema. Semantičko povezivanje omogućavaju modeli različitih nivoa apstrakcija, koji se zasnivaju na: semantici; poslovnim terminima i konceptima; ontologijama i poslovnim pravilima.

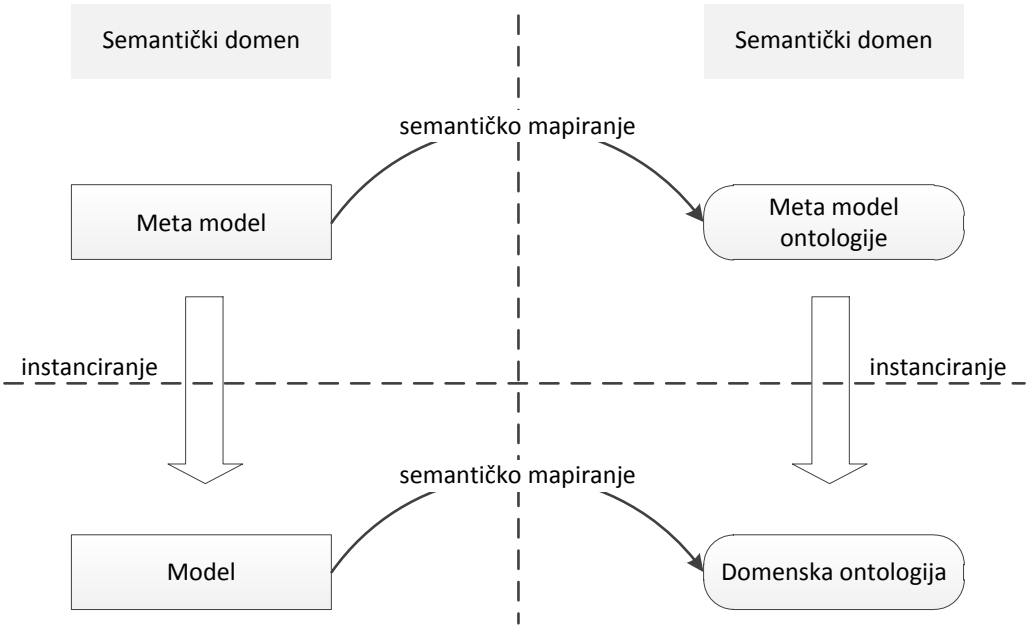
Uloga semantike u modelima je da objasni poslovanje sistema koji se modeluje. Za semantički opis sistema koriste se poslovni termini i fraze, pri čemu frazu čini jedna ili više reči. U kontekstu kreiranja meta modela, poslovni koncept označava realnu stvar, događaj ili proces i može se predstaviti jednim poslovnim terminom [88]. Poslovni koncept se u modelu pojavljuje kao simbol (na primer: osoba), kao atribut simbola (na primer: pol), ili kao instanca simbola (na primer: student "A. B.").

Koristeći govorni jezik, poslovni koncept određenog sistema može se opisati sa više različitih reči, ili fraza. Različite reči i fraze, koje opisuju jedan isti poslovni koncept, su sinonimi. U suprotnom slučaju, kada neke od reči ili fraza, u različitim industrijama, imaju drugačije značenje, govori se o homonimima. Sa ciljem da se obezbedi jedinstveno značenje, pri kreiranju meta modela, umesto reči i fraza govornog jezika, koriste se poslovni termini. Poslovni termini obuhvataju opšte prihvaćene reči i fraze, kojima se uobičajeno opisuju sistem i procesi u sistemu, uključujući i specifične reči, vezane za industriju (ili sistem), da bi se opisali tehnički aspekti procesa, ili drugi aspekti sistema.

U meta modelima, svaki poslovni termin isključivo je predstavljen jednim simbolom, koji definiše opseg poslovnog termina. Logički i funkcionalno povezan skup simbola predstavlja rečnik. Ovako kreiran rečnik čini ontologiju.

Zadatak analitičara sistema je da ontologijama predstave semantiku sistema, opisujući stvari i procese, koji postoje u sistemu i njegovom okruženju. Ontologija je tesno povezana sa semantikama, pri čemu se ontologija bavi znanjem sistema, a semantika direktno ukazuje šta nešto znači [88]. U meta modelima svaki simbol može da bude predmet klasifikacija jedne ili više ontologija, pri čemu svaki simbol mora biti uključen u neku od ontologija. Formalno gledano, svaki validan meta model je ontologija. Međutim, ne moraju sve ontologije eksplicitno biti meta modeli. Iz navedenog razloga, može se reći da meta model ontologije, u slučaju konkretnog sistema, predstavlja model sistema [100].

Slična ideja razrađena je u radu [102], u kome se predlaže razvijanje dva tipa ontologija: ontologije meta modela i ontologije domena problema, koje se zovu domenske ontologije. Domenska ontologija je instanciranje ontološkog meta modela. Model je instanca meta modela i semantički se interpretira domenskom ontologijom.

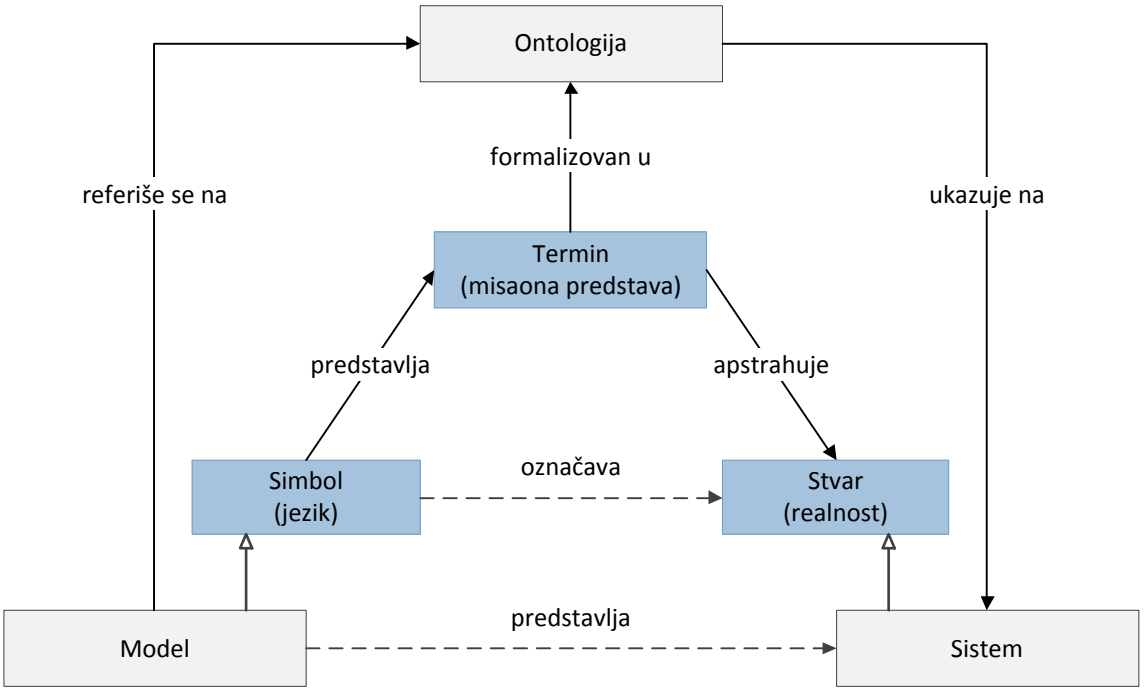


Slika 4.11: Odnosi između ontologija, modela i meta modela
(prilagođeno po [102])

Osnovna ideja semantičkog modelovanja sistema zasniva se na proširenju semantičkog trougla značenja, koji je prikazan slici 4.12. Potreba za proširenjem proističe iz činjenice, da se simbol, koji predstavlja: ideju; fizički entitet ili proces, razlikuje od realne pojave, ili manifestacije toga što predstavlja.

Unutrašnji trougao, na modelu sa slike, ukazuje da simboli nemaju direktan odnos prema stvarima, koje predstavljaju. Relacija između simbola i stvari može se uspostaviti samo posredno, preko termina kojima se opisuju stvari, predstavljene odgovarajućim simbolima. Odnosno, interpretacija ili percepcija stvari moguća je jedino kada subjekt (stvar) poseduje termin, koncept ili mentalni model, koji je povezan sa simbolom.

Analogni odnosi postoje i između elemenata: model; ontologija i sistem (spoljni trougao). Modeli su predstavljeni posebnim simbolom. Veza između modela i sistema nije direktna i neposredno moguća, jer je model rezultat konstrukcionog procesa, koji karakteriše kreativnost ideje i shvatanja autora.

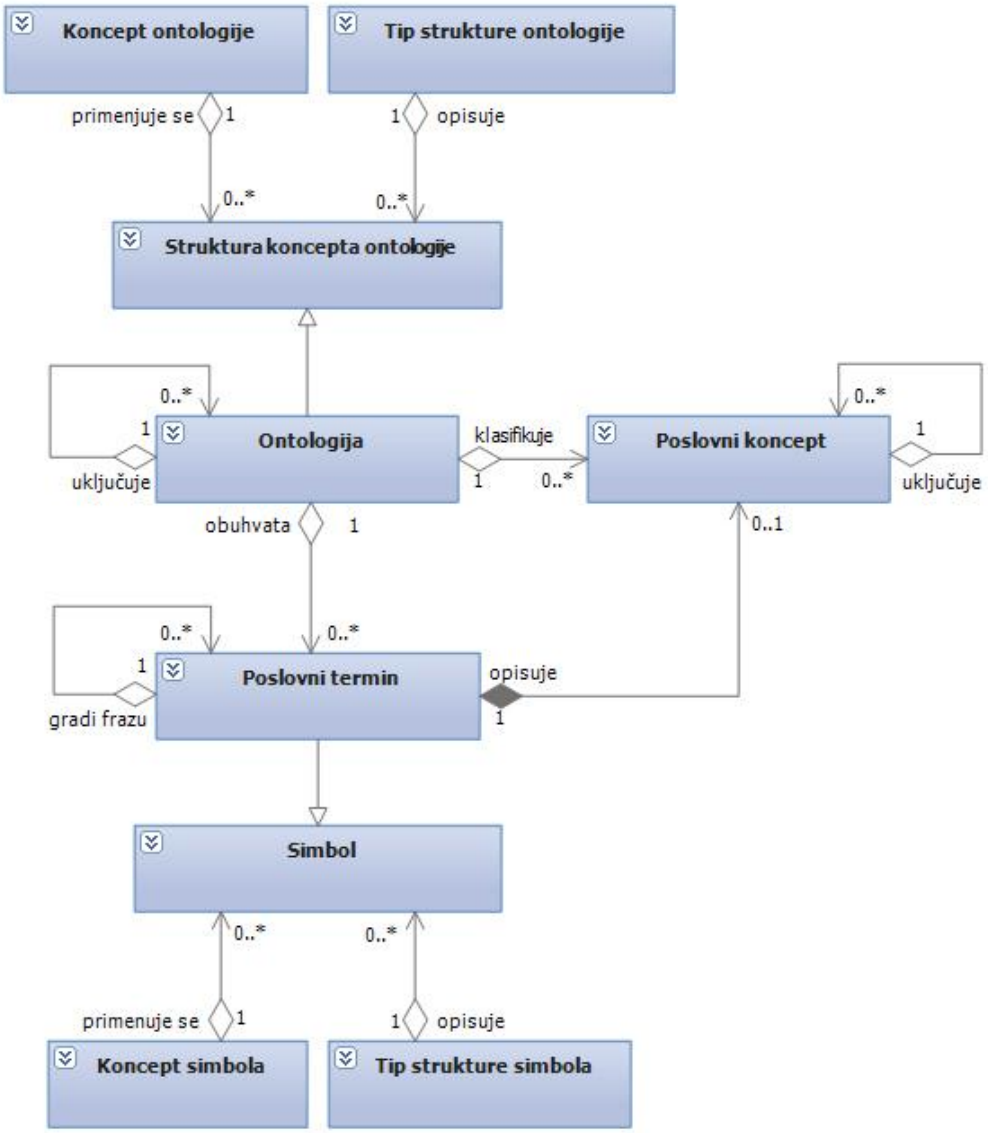


Slika 4.12:. Prošireni semantički trougao znanja

4.2.2 Generički meta model

Polazeći od opisanog koncepta i strukture ontologija, predstavljenog proširenim semantičkim trouglom znanja na slici 4.12, izveden je generički meta model. Generički meta model predstavljen je na slici 4.13. Namenjen je:

- da obezbedi osnovu za semantičku interoperabilnost, a na osnovu ontologija;
- da omogući kreiranje kolekcija modela, koje obezbeđuju interoperabilnost, a kojima se popunjavaju određene ćelije SOFI okvira;
- da obezbedi osnovu za kreiranje i proveru drugih modela iz SOFI okvira.



Slika 4.13. Generički meta model

Prikazani generički meta model ima višestruku ulogu tokom kreiranja kolekcija modela za popunjavanje ćelija SOFI okvira. Može da podrži: kreiranje modela na različitim nivoima apstrakcije i u odnosu na različite poglede na sistem; kreiranje šema za semantičke podatke; da obezbedi dodatne, semantičke opise za pojedine modele; da korišćenjem simbola obezbedi jezik za opisivanje određenih metoda i procesa. Sledi kratak opis navedenih mogućnosti, ilustrovan primerima primene modela za potrebe kreiranja pojedinih artifakata ćelija SOFI okvira.

- Šema za semantičke podatke
  Podržana je sledećim klasama generičkog meta modela: koncept ontologije, tip strukture ontologije, struktura koncepta ontologije, ontologija, poslovni koncept i poslovni termin. Šema za semantičke podatke, koji se razmenjuju ili pamte, mapirana je prirodnim jezikom sistema. Omogućava primenu jedinstvenog, konzistentnog rečnika i zaključivanja tokom kreiranja pojedinačnih artifakata SOFI okvira. Primeri primene su: spisak zahteva; slučajevi korišćenja.

- Nivoi apstrakcije i podržani pogledi
  Logička struktura generičkog meta modela podržava izradu pojedinačnih meta modela, na različitim nivoima apstrakcije i iz različitih uglova posmatranja sistema. Različite nivoe apstrakcije direktno podržavaju klase iz generičkog meta modela: ontologija, poslovni koncept i poslovni termin. Na drugoj strani, različiti uglovi posmatranja uključeni su korišćenjem predefinisanog skupa poslovnih termina za opisivanje poslovnih koncepata. Primeri primene različitih nivoa apstrakcije su: spisak zahteva; konceptualni model podataka; logički model podataka. Primeri primene različitih pogleda na sistem su: koncept rešenja; BPMN dijagrami.
- Jezik za dodatno semantičko opisivanje postojećih modela
  U slučaju SOA arhitekture, interoperabilnost obezbeđuje korišćenje servisa. Opisani SOFI okvir namenjen je za modelovanje sistema elektronskog poslovanja, koji se zasnivaju na semantičkoj interoperabilnosti. Semantička interoperabilnost uspostavlja se korišćenjem veb servisa. Pri tome, mogu se koristiti već dostupni veb servisi, ili se mogu posebno kreirati, za potrebe realizacije konkretnog sistema. Bilo da se koriste javni, ili privatni veb servisi i registri veb servisa, efektivnost otkrivanja potrebnih veb servisa povećava se ukoliko su dodatno semantički opisani. Dodatno opisivanje veb servisa podržavaju sledeće klase iz generičkog meta modela: ontologija, poslovni koncept, poslovni termin. Primeri primene su: model servisa; specifikacija servisa.
- Jezik za podršku metodama i procesima
  Artifakti, koji pripadaju tehnološkom modelu SOFI okvira sadrže kolekcije modela, namenjene IT inženjerima. Obuhvataju informacije, potrebne za kreiranje fizičkih struktura baza podataka, kreiranje procedura, klasa i metoda i slično. Ukoliko se primeni, generički meta model obezbediće konzistentnost potrebnih informacija i simbola preko sledećih klasa iz generičkog meta modela: koncept simbola, tip strukture simbola i simbol. Primeri primene su: fizički model podataka; dijagram klasa; specifikacija servisa; dijagram komponenti.

Opis generičkog meta modela i primeri mogućeg korišćenja ukazali su na mogućnosti primene, za kreiranje kolekcija modela za popunjavanje ćelija SOFI okvira. Očekuje se:

- da šema za semantičke podatke obezbedi semantičku konzistentnost modela;
- da izvedeni model omogući kreiranje artifakata na različitim nivoima apstrakcije, koji se mogu opisati konzistentnom kolekcijom simbola i koji će omogućiti različitim učesnicima jedinstven pogled na sistem;
- da podrži dodatno semantičko opisivanje potrebnih veb servisa, ili već postojećih modela;
- da omogući semantičku interoperabilnost sistema elektronskog poslovanja, uključenjem semantički opisanih veb servisa.

Generički meta model razvijen je proširenjem i tranformacijom trougla znanja. Zasnovan je na ontologijama, preko kojih se uspostavlja veza između objekata realnog sistema i simbola iz modela, kojima se realni objekti sistema predstavljaju. Predloženi generički meta model obezbeđuje osnovu za semantičku interoperabilnost i kreiranje kolekcije modela za popunjavanje ćelija SOFI okvira. Primena generičkog meta modela nije ograničena nivoom apstrakcije i pogledom na sistem.

Verifikacija razvijenog SOFI metodološkog okvira i generičkog meta modela opisana je u narednom poglavlju.

5 Verifikacija razvijenog metodološkog okvira

Verifikacija SOFI metodološkog okvira za razvoj interoperabilnih, servisno orijentisanih aplikacija elektronskog poslovanja sprovedena je tokom modelovanja Sistema za pretraživanje i prezentaciju različitih vrsta kulturnih dobara (KD) u Srbiji (u daljem tekstu: Portal).

Tokom izrade Portala pokazalo se da su odabrani sadržaji, kulturna dobra koja se čuvaju u različitim ustanovama kulture, relevantan skup artifakata, dovoljan za verifikaciju svih aspekata primene razvijenog metodološkog okvira. Obuhvaćene vrste KD karakteriše izuzetna semantička raznovrsnost, s obzirom da su nastajale u okviru različitih aktivnosti svakodnevnog života i u dugom vremenskom periodu. Osim toga, ne postoji centralizovana i jedinstvena baza podataka o njima. Ad hoc tražene podatke, sadržane u upitu korisnika, potrebno je pronaći u pojedinačnim bazama podataka ustanova kulture, koristeći isključivo njihove, semantički označene veb servise.

U ovom poglavlju biće prikazana primena SOFI metodološkog okvira, tokom svih faza razvoja Sistema za pretraživanje i prezentaciju KD, počev od definisanja i analize zahteva, zaključno s analizom dobijenih rezultata i mogućnosti.

S obzirom da je reč o razvoju novog sistema, proces njegovog modelovanja odvijao se u smeru od "vrha prema dnu", a saglasno usvojenoj metodologiji. Prvo su kreirani modeli višeg stepena apstrakcije, nakon čega su na nižim nivoima, sukcesivnim dekompozicijama, izvedeni modeli neophodne detaljnosti. Osnovu za kreiranje modela predstavljao je generički meta model, prikazan u poglavlju 4.2.2 ove teze.

5.1 Primena SOFI metodološkog okvira

Mogućnosti i uspešnost metodološkog okvira, za uspostavljanje interoperabilnosti i primenu SOA arhitekture, moguće je proveriti na primeru distribuiranog poslovnog sistema. Odabrani poslovni sistem za verifikaciju okvira, trebalo je da zadovolji određeni broj neophodnih kriterijuma:

- da je definisan i standardizovan u pogledu odvijanja poslovnih procesa i granica sistema;
- da je organizovan oko distribuiranih baza podataka o artifaktima od značaja za poslovni sistem;
- da su podaci o artifaktima poslovnog sistema semantički bogati.

Dodatni kriterijumi, koji su uticali na izbor poslovnog sistema za analizu i realizaciju su:

- aktuelnost;
- originalno rešenje;
- nije u potpunost realizovan u praksi – ni u svetu, a ni kod nas i
- mogućnost logičkog modelovanja i fizička izvodljivost.

Imajući u vidu navedene zahteve i kriterijume, verifikacija razvijenog metodološkog okvira za primenu SOA arhitekture i uspostavljanje interoperabilnosti sprovedena je tokom izrade Portala, namenjenog za pretraživanje i prezentaciju različitih vrsta kulturnih dobara u Srbiji.

Za planiranje Portala korišćen je SOFI metodološki okvir, koji obuhvata potrebna dokumenta i modele, neophodne za definisanje zahteva, izradu modela i implementaciju projektovanih rešenja. Pored specifikacije potrebne dokumentacije i modela, okvir je preuzeo i kontrolne funkcije planiranja Portala, zasnovane na pravilima popunjavanja ćelija, a u smislu provere međusobnih veza dokumenata i modela, njihove potpunosti, granuliteta i uslovljenosti. SOFI metodološki okvir, detaljno opisan u poglavlju 4. i grafički predstavljen na slici 4.6, korišćen je tokom svih faza procesa planiranja Portala. Ćelije na slici 4.6 označene su slovima, kojima je pridružen redni broj za svaki artifakt, koji ćelija sadrži. Označavanje ćelija okvira olakšalo je referisanje na konkretna dokumenta i modele, sadržane u njima, i korišćeno je tokom verifikacije sistema.

U narednom poglavljima dati su detaljan opis Sistema za pretraživanje i prezentaciju KD u Srbiji, koji je predmet analize, navedena je vizija sistema i prikazana su i objašnjena kreirana dokumenta i izvedeni modeli, neophodni za razvoj Portala.

5.2 Opis Portala i ciljne grupe

Dobro razumevanje sistema, koji se analizira, osnov je za uspešnu izradu potrebne dokumentacije i modela za razvoj. Iz navedenog razloga, objašnjeni su osnovni principi zaštite KD u Srbiji, data je klasifikacija i opisane su vrste artifakata, koja su predmet analize.

5.2.1 Opis sistema – priroda i prezentacija kulturnih dobara

Na početku procesa verifikacije urađen je opis poslovnog sistema koji je, polazeći od ciljeva projekta, obuhvatio definicije, prirodu i klasifikacije kulturnih dobara, kao i mogućnosti njihove prezentacije.

5.2.1.1 Priroda, klasifikacije i zaštita kulturnih dobara

Najveći broj kulturnih dobara (KD) nastao je kao rezultat svakodnevnih životnih aktivnosti i potreba i imao je upotrebnu vrednost.

U Srbiji, a na osnovu naučno-stručne prakse specijalizovanih ustanova kulture i zakonskih rešenja, sva KD svrstana su u dve osnovne grupe, a u saglasnosti sa UNESCO klasifikacijom [103]: materijalna KD i nematerijalna KD.

Materijalna KD po prirodi su predmeti, objekti ili prostor. Materijalna KD klasifikuju se u okviru pet vrsta KD:

1. Muzejski predmeti (arheološki nalazi, etnografski predmeti, nakit, oružje, oruđe, predmeti za svakodnevnu upotrebu i slično; primerci faune i flore)
   Posebna grupa muzejskih predmeta su likovna i vajarska dela (slike, grafike, skulpture, mozaici, vitraži) i instalacije. Ovu grupu predmeta karakteriše isključivo estetska vrednost. Proglašenjem predmeta za KD, prestaje njegovo svakodnevno korišćenje.

2. Arhivska građa (arhivski fondovi, zbirke, pojedinačna dokumenta i rukopisi)
   Stvaralac arhivske građe najčešće je organizacija, koja se po zakonu naziva registratura ili fond. Odabrani materijal, koji nastaje u registraturi, u određenim vremenskim intervalima preuzima nadležni (istorijski) arhiv, u kome se sređuje, obrađuje i trajno štiti. Preuzeta arhivska građa se, saglasno propisanim procedurama, može i dalje koristiti, ali se ne sme menjati.
3. Filmski materijal i predmeti vezani za film (filmska dokumentacija, fotografije, plakati; delovi scenografije i kostimi; uređaji za snimanje i reprodukciju filmskog materijala)
   Filmski materijal sa statusom KD, uz preduzimanje mera zaštite i poštovanje zakonskih procedura, može se koristiti za javno prikazivanje i stvaranje novih filmskih i srodnih umetničkih proizvoda (filmska produkcija; TV produkcija; pozorište, edukacija i slično).
4. Građevine i ambijentalne celine (sakralni objekti; arheološka nalazišta; utvrđenja i fortifikacioni objekti; kuće i staništa; spomenici; objekti infrastrukture, industrijski i privredni objekti; ambijentalne celine i slično)
   Navedeni objekti nastali su kao rezultat životnih potreba, a za nepokretno kulturno dobro (NKD) proglašavaju se na osnovu prepoznatih i dokumentovanih kulturno-umetničkih i istorijskih vrednosti. Proglašenjem za KD, objekat se može i nadalje koristiti i menjati, ali uz poštovanje naloženih mera zaštite.
5. Bibliotečki materijal (izdanja beletristike i stručne literature; časopisi i novinska izdanja; mape, karte, razglednice i fotografije).
   Korišćenje bibliotečkog materijala sa statusom KD najčešće je dozvoljeno, ali u skladu s propisanim merama zaštite.

U nematerijalna KD svrstavaju se duhovna i socijalna dobra:

1. Pojave vezane za društveni život:
   a. godišnji običaji (Božić, Uskrs, porodične slave i zavetine, litije)
   b. običajno pravo;
   c. običaji iz životnog ciklusa (rođenje, venčanje, smrt).
2. Duhovna kultura:
   a. verovanja (mitska bića; kultna mesta; vradžbine; bajanje);
   b. narodna umetnost: muzika (sviranje i gradnja muzičkih instrumenata), igra, pesma;
   c. jezik, znanja, umotvorine i usmena tradicija;
   d. veštine i rukotvorine (ćilimarstvo, tkačka radinost, ručni radovi, grnčarstvo, drevni zanati).

Ustanove kulture u Srbiji, pored stalnih aktivnosti u okviru brojnih procesa zaštite KD, po zakonu su u obavezi da štićena KD javno prezentuju. Pojedinačna, ali nepotpuna prezentacija KD, obavlja se u okviru stalnih i povremenih izložbenih postavki, u različitim publikacijama, u štampi, na radiju i televiziji.

Pri procenjivanju mogućnosti celokupne prezentacija KD, potrebno je uzeti u obzir određeni broj ograničenja. Neka od bitnih ograničenja su:

- broj pojedinačnih KD je izuzetno veliki;
- proces stručno-naučne obrade pojedinačnog KD u praksi najčešće predugo traje - i do 20 godina;

- u najvećem broju ustanovama za zaštitu KD ne koriste se jedinstveni modeli podataka za kreiranje baza podataka, iako su logički modeli urađeni, ekspertski provereni i verifikovani [104].

5.2.1.2 Veb portal u funkciji prezentacije i zaštite KD

Jedan od opštih svetskih trendova je sistemska zaštita kulturne materijalne i nematerijalne baštine. Pri tome, pojam "zaštita" ima šire značenje i podrazumeva više obaveza za ustanove za zaštitu KD. Obaveze ustanova zaštite KD ne prestaju završetkom faze naučno-stručne obrade KD, dokumentovanjem podataka o KD i fizičko-tehničkim obezbeđenjem artifakata, već obuhvataju i primerenu prezentaciju KD.

Suočene sa novim zahtevima, ustanove za zaštitu KD prihvataju novi način prezentacije obrađenih KD - izrađuju multimedijalne veb sajtove. U vreme pisanja ove, u Srbiji je postojalo oko sto veb sajtova ustanova za zaštitu KD. Reč je o veb sajtovima biblioteka, istorijskih arhiva, muzeja, zavoda za zaštitu spomenika kulture i o veb sajtu Jugoslovenske kinoteke. Uočeni su pokušaji podizanja veb portala, namenjenih za prezentaciju jedne vrste KD, na primer: www.eternitas.rs, međutim projekat je u zastoju, ili je obustavljen.

Multimedijalni veb sajt ustanove kulture, ili celokupne mreže ustanova za zaštitu određene vrste KD, u funkciji prezentacije KD, a u odnosu na druge medije, ima i prednosti i nedostatke.

Prednosti veb sajta u odnosu na druge medije:

- brza izrada i niska cena po jedinici informacije;
- aktuelnost, proističe iz mogućnosti relativno brze i lake izmene i dopune sadržaja;
- dostupan u svakoj tački u kojoj postoji veza s Internetom;
- indeksirani sadržaji s direktnim pristupom traženim sadržajima itd.

Neki od nedostataka veb sajta su:

- virtuelnost - veb sajt ne postoji u obliku artifakta, ukoliko se ne uzmu u obzir povremene kopije na CD ili DVD nosioce zapisa;
- najčešće je vremenski diskretan sistem, bez memorije - sadržaj veb sajta, posmatran iz ugla posetioca, ima samo jedno stanje - trenutno, bez mogućnosti uvida u prethodne, izmenjene ili obrisane sadržaje (prethodna diskretna stanja veb sajta).

Većina, trenutno dostupnih veb sajtova ustanova za zaštitu KD, ima statične sadržaje. Najčešće se objavljuju podaci o ustanovi i istorijatu ustanove, trenutnim dešavanjima, izdatim publikacijama, a veb sajt je ilustrovan često nepotpunim prikazom određenog broja izabranih artifakata.

Prezentovani sadržaji, i ponekad neažurne informacije, ukazuju da su ciljevi, postavljeni pre izrade veb sajta, bili nejasni i nepotpuni, ili nisu dostignuti. Osim toga, teško je prepoznati ciljne grupe, kojima je veb sajt namenjen, a moguća pretpostavka je da je reč o "opštoj ciljnoj grupi".

5.2.2 Ciljne grupe i primeri korišćenja sistema

Da bi se odredio profil korisnika, potrebno je da se, u okviru opšte Internet populacije, uoče grupe značajnije motivisane za pretraživanje i korišćenje podataka o KD.

Nakon urađenih analiza korišćenja veb sajtova ustanova kulture za zaštitu KD, zaključeno je da budući sistem mora da zadovolji široku ciljnu grupu korisnika. Očekuje se da će Portalu pristupati i "nekvalifikovani" korisnici, ali i poznavaoci prezentovanih sadržaja – "kvalifikovani posetioci". I pored te činjenice, nije se moglo zadovoljiti konstatacijom da je Portal namenjen "opštoj ciljnoj grupi". Zato, umesto fraze: "opšta ciljna grupa", svi budući posetioci svrstani su u "hibridnu ciljnu grupu".

U sledećem koraku, a na osnovu urađenih analiza, unutar "hibridne ciljne grupe" precizno su identifikovane podgrupe korisnika.

Struktura hibridne ciljne grupe
U slučaju prezentacije i zaštite KD, izdvojene su zainteresovane grupe posetilaca. Vezuju se za oblasti profesionalnih aktivnosti, ili za oblasti ličnog interesovanja. Uočene su sledeće ciljne grupe budućih korisnika sistema:

1. državne institucije koje su posredno ili neposredno zadužene za zaštitu KD;
2. eksperti u oblasti proučavanja i zaštite KD;
3. institucije i pojedinci koji se bave publikovanjem i mediji;
4. turistička industrija;
5. kvalifikovani pojedinci (lično interesovanje i korišćenje informacija za potrebe edukacije) i
6. nekvalifikovani pojedinci (posetioci koji bez posebnog motiva i namere pristupaju sadržajima Portala).

Prepoznavanje i analiza članova hibridne ciljne grupe omogućavaju kreiranje scenarija i definisanje preciznijih zahteva, kojima Portal mora da odgovori.

1. Državne institucije zadužene za zaštitu KD

Posetioci Portala, zaposleni u državnim institucijama za zaštitu KD, pripadaju grupi profesionalnih korisnika informacija o KD. Imaju upravljačku i kontrolnu funkciju. Pored opštih informacija o KD, posebnu pažnju poklanjaju klasifikaciji KD, preduzetim merama i ostvarenim rezultatima tehničko-fizičke zaštite KD. Takođe, interesuje ih i kvalitet prezentacije KD. Uz navedene korisnike, ovoj grupi mogu se priključiti carinska služba i služba Ministarstva unutrašnjih poslova, koje se bave istragama vezanim za otuđenje i neovlašćeni promet KD. Predmet njihovog interesovanja pretežno su konkretna KD i detaljni podaci o njima, odnosno izvod iz dosijea određenog KD. Pri pretraživanju sistema najčešće zadaju konkretne pojmove i stručne fraze, očekujući mogućnost dubinskog pretraživanja raspoloživih izvora podataka.

Mogući scenario broj 1.1
Službenik Ministarstva kulture dobio je zadatak da uradi prigodnu prezentaciju Republike Srbije, povodom međunarodnog obeležavanja godišnjice završetka I svetskog rata. Centralna proslava obaviće se u jednoj od zemalja pobednica i svaka od članica saveza dobiće određeni, uređeni

prostor. Prezentacija bi trebalo da obuhvati pisana dokumenta, prigodne predmete, fotografije i filmski materijal iz tog perioda. Veličina raspoloživog prostora je ograničena, a odabrani materijali trebalo bi da predstavljaju tematsku celinu. S obzirom da je reč o završetku rata, odlučeno je da tema bude: "Pripreme za proboj i proboj Solunskog fronta". Očekuje se da odabrani predmeti i materijali budu originalni, karakteristični za Srbiju, privlačni, da predstavljaju tematsku celinu, kao i da simbolizuju hrabrost i odvažnost srpskog vojnika. Potrebni predmeti su najvećim delom zaštićena KD i nalaze se u muzejima i arhivima, a filmski materijal u Jugoslovenskoj kinoteci. Širi izbor materijala obuhvatio bi: slike i filmski materijal sa Solunskog polja i Kajmakčalana; zapovesti i ratni dnevnici; delovi vojničke opreme; fotografije srpskog groblja na Kajmakčalanu; doček srpskih vojnika u oslobođenim gradovima; pesme i kola srpskih vojnika iz tog perioda.

Korisnik pristupa Portalu, kreira više upita koristeći unapred pripremljene ključne reči i fraze, koje zavise od sinopsisa buduće izložbe. Spisak raspoloživih predmeta i materijala, koji zadovoljavaju tematiku prezentacije, dobiće kreiranjem više sukcesivnih upita.

Mogući scenario 1.2
Na graničnom prelazu carinik je, rutinskim pregledom automobila koji je napuštao Srbiju, otkrio kolekciju antičkih, srebrnih novčića. Koristeći raspoložive, numizmatičke kataloge, služba Carine identifikuje pronađene novčiće i o tome izveštava nadležno odeljenje MUP. U okviru provere porekla kolekcije, stručnjak MUP, ekspert za KD, pristupa Portalu. Na osnovu podataka, dobijenih od službe Carine, sukcesivno zadaje ključne reči za pretraživanje, s namerom da ustanovi da li je otkrivena kolekcija novčića deo zbirke nekog od muzeja u Srbiji.

2. Eksperti u oblasti proučavanja i zaštite KD
Ekspert u oblasti proučavanja i zaštite KD pripada grupi profesionalnih korisnika informacija o KD. Reč je o specijalistima i izuzetno dobrim poznavaocima oblasti vezanih za proučavanje i zaštitu KD. Predmet njihovog interesovanja su pojedinačna KD, ali i komparativni podaci, vezani za različita KD.

Ekspert kreira upit koristeći stručne pojmove i fraze. Pre pristupa podacima o konkretnom KD, očekuje širi spisak relevantno rangiranih rezultata pretraživanja.

Mogući scenario broj 2.
Doktor M. M. radi u Etnološkom institutu, pri Srpskoj akademiji nauka. Prateći trenutne svetske trendove, odlučio je da detaljnije prouči pojedine običaje i verovanja iz istočne Srbije. Cilj mu je da ustanovi ispunjenost kriterijuma za nominovanje, nekog od njih, za upis u spisak svetske nematerijalne kulturne baštine. Pripremajući se za buduće istraživanje, namerava da se upozna s postojećim, već zaštićenim KD (predmeti; fotografije i filmski materijal; audio i video zapisi; dokumenta i bibliografske jedinice). Pretpostavlja da su mogući izvori podataka: muzeji, istorijski arhivi, Jugoslovenska kinoteka, zavičajna odeljenja lokalnih biblioteka. Dodatni izvori informacija mogli bi biti: Filmske novosti, lokalne radio i TV stanice, lokalna štampa.

3. Institucije i pojedinci koji se bave publikovanjem i mediji
Korisnike informacija o KD ili novinare, za potrebe publikovanja najčešće interesuju opšti podaci o KD. Ne pripadaju grupi profesionalnih korisnika. Zadate reči i fraze za pretraživanje su opšte i jednostavnije. Značajno je da se ovoj grupi korisnika, uz spisak rezultata pretraživanja, ponudi i procenjena relevantnost sadržaja u odnosu na zadatu reč ili frazu iz upita.

Mogući scenario broj 3.
Novinar lokalne televizije dobio je zadatak da uradi reportažu o starim zanatima, karakterističnim za teritoriju opštine. Reportaža bi trebalo da bude zabavno-edukativnog karaktera i namenjena je gledaocima različitog uzrasta i obrazovanja. U okviru priprema za snimanje reportaže, novinar mora pronaći odgovarajuće izvore i evidentirati moguće priloge: fotografije, arhivska dokumenta, stara godišta lokalnih novina. Pretpostavlja da su mogući izvori podataka gradski muzej, lokalni istorijski arhiv i gradska biblioteka.

4. Turistička industrija
Interesovanje turističke industrije za KD je promenljivo i zavisi od opštih kretanja u ovoj industriji. Više je izraženo u slučaju nacionalnih i lokalnih turističkih organizacija, a mnogo manje i ređe u slučaju turističkih agencija.

Predmet interesovanja su opšti podaci o pojedinačnim KD, uključujući ilustrativne fotografije. Pri pretraživanju sadržaja, posvećenim KD, zadaju opšte ključne reči i jednostavne fraze. Pristup određenom, traženom KD može im olakšati spisak rezultata, dopunjen stepenom relevantnosti.

Mogući scenario broj 4.
Nacionalna turistička organizacija priprema prezentaciju Srbije na međunarodnom sajmu turizma u Parizu. Početna ideja je da se promoviše putovanje Dunavom, s obilaskom usputnih praistorijskih, antičkih i srednjevekovnih spomenika kulture: Vinča i Lepenski Vir; Singidunum, Smederevska i Golubačka tvrđava; Tabula Traiana, Viminacium i Felix Romuliana. Plan je da se navedeni lokaliteti i pokretni nalazi sa njih ilustruju mapama tog područja, fotografijama i kratkim, pratećim tekstovima. Pretpostavka je da su mogući izvori informacija veb sajtovi nadležnih ustanova za zaštitu KD: muzeji i zavodi za zaštitu spomenika kulture. Prednost je ukoliko se do potrebnih informacija može doći sa Portala, koji bi ponudio objedinjene i potpune spiskove, uz navođenje relevantnosti raspoloživih materijala.

5. Kvalifikovani pojedinci
Kvalifikovani pojedinci pripadaju neprofesionalnoj ciljnoj grupi, s obzirom da se ne mogu svrstati u neku od navedenih strukovnih organizacija. Međutim, njihovo interesovanje, u pogledu određenog KD ili cele grupe KD, konkretno je i relativno precizno definisano. Kreirajući upit za pretraživanje zadaju stručne pojmove i složenije fraze. Ispunjenje potreba ovih korisnika podrazumeva prezentovanje detaljnijih podataka o pojedinačnom KD, komparativne preglede i navođenje relevantnosti rezultata pretraživanja.

Mogući scenario broj 5.
Akademski vajar K. V. godinama se bavi proučavanjem i izradom replika antičkog i srednjevekovnog nakita. Ovom prilikom želi da uradi kopiju srednjevekovne minđuše, iskopane na lokalitetu Dvor.

S namerom da uradi što je moguće verniju kopiju, pre izrade novog komada nakita nastoji da prikupi podatke o materijalima i tehnikama originalnog primerka. Važni su mu: doba nastanka, mesto nalaza i podaci o starim objektima u blizini. Posao bi mu olakšale kvalitetne fotografije nalaza, visoke rezolucije. Pri kreiranju upita za pretraživanje mogućih izvora podataka navodi stručne reči i složenije fraze. Očekuje da rezultat pretraživanja, pored spiska, sadrži i procenu relevantnosti ponuđenih sadržaja. Mogući izvori podataka su gradski muzej, Arheološki institut SANU i nadležni zavod za zaštitu spomenika kulture.

6. Nekvalifikovani pojedinci
Posetioci, koji na portal dolaze zadavanjem opšte poznatih i uobičajenih ključnih reči i jednostavnih fraza, deo su neprofesionalne ciljne grupe. Njihovi zahtevi i potrebe zadovoljavaju se pristupom opštim sadržajima, koji su, najčešće, potpuno relevantni u odnosu na postavljene upite.

Mogući scenario broj 6.
Učenica gimnazije A. N. radi maturski rad na temu slikarstva Leonida Šejke. Namera joj je da Šejkino slikarstvo prikaže hronološki, vezujući ga za lični život umetnika. Biografske podatke prikupila je relativno lako, ali ostao joj je teži deo posla: da prouči i prikaže faze razvoja umetnika i pripadnost likovnim pravcima tokom umetnikovog života i stvaralaštva. Značajni podaci su: vreme nastanka dela; primenjene tehnike i materijali; zastupljeni motivi; fotografije za ilustrovanje rada. Mogući izvori informacija su jedino muzeji, u kojima se čuvaju Šejkina dela, a ne može računati na podatke o delima, koje se nalaze u kolekcijama privatnih vlasnika.

Prepoznavanje kvalifikovanih i nekvalifikovanih korisnika može se iskoristiti da se pri planiranju Portala u većem broju slučajeva zadovolje stvarne potrebe korisnika.

5.3 Implementacija metodološkog okvira

Navedeni opis Portala, uključujući ograničenja i preduslove za realizaciju, bili su osnov za detaljnu analizu, identifikaciju i analizu potreba ciljnih grupa – budućih korisnika sistema i izbor raspoloživih informacionih tehnologija za njegovu realizaciju. Predloženo rešenje zasnovano je na veb portalu, čija je uloga da uspostavi vezu između korisnika sistema i distribuiranih baza podataka nadležnih ustanova kulture.

Primenom raspoloživih informacionih tehnologija, a sa ciljem da Sistem za prezentovanje KD u Srbiji na pravi način promoviše kulturnu baštinu i delimično zadovolji zakonske obaveze specijalizovanih ustanova kulture i potrebe posetilaca u pogledu prezentacije KD, neophodno je uzeti u obzir različite potrebe i ponašanje ciljnih korisnika informacija.

Zato bi Portal trebalo razvijati prema procenjenim potrebama i ponašanju "hibridnog korisnika", čiji je profil mnogo složeniji od nedefinisanog profila nedefinisane "opšte ciljne grupe", a korišćenjem SOFI metodološkog okvira.

5.3.1 Opis Portala i zahtevi

Zahtevi, vizija i ciljevi Portala urađeni su na osnovu analize poslovnih potreba i ograničenja i opisuju šta bi veb portal, koji se modeluje, trebalo da ispuni.

5.3.1.1 Zahtevi Portala, ograničenja i preduslovi za realizaciju

Polazeći od interesa korisnika Portala i mogućnosti ustanova za zaštitu KD, osnovni zahtevi Portala bili su:

- Da se korisnicima Portala omogući postavljanje proizvoljno definisanog upita za pretraživanje podataka o KD. Pri tome, korisnik Portala ne mora da poseduje posebna znanja o klasifikacijama KD i subjektima, koji obezbeđuju tražene podatke.

- Da Portal, na osnovu analize upita korisnika, prosledi zahtev za podacima samo užem skupu ustanova kulture, za koje se očekuje da poseduju KD sa karakteristikama iz upita korisnika.
- Da ustanova kulture za zaštitu KD, kojoj je Portal prosledio zahtev za podacima, nema neposrednu komunikaciju s korisnikom, pa samim tim ni potrebu za dodatnom administracijom i servisiranjem korisnika. Da ustanova kulture svoje aktivnosti, vezane sa upit korisnika, završi slanjem odgovora sistemu.
- Da Portal, nakon prihvatanja odgovora ustanova kulture, kojima je prosledio upit, objedini spisak svih KD, koja zadovolje postavljeni upit, nezavisno od vrste KD i ustanove, koja se o njima stara.
- Da Portal prosledi odgovor korisniku na postavljeni upit u obliku objedinjenog spiska KD, koja su zadovoljila kriterijume iz upita.
- Da korisnik, na osnovu dobijenog odgovora, može da zatraži od Portala podatke o konkretnom KD sa dobijenog spiska, ili da postavi novi upit.

Ograničenja Portala proistekla su iz postavljenih granica sistema. U saglasnosti sa definisanim zahtevima, Portal je samo posrednik između korisnika i baza podataka ustanova kulture, koje se bave zaštitom KD. Postojeće baze podataka ustanova kulture nalaze se na različitim platformama, a podaci o logičkim i fizičkim strukturama nisu dostupni, pa će se komunikacija sa bazama podataka obavljati isključivo preko posredničkog softvera.

Osnovni preduslovi za realizaciju Portala obuhvataju:

- Da se planiranje Portala zasniva na jedinstvenom meta modelu, koji dovoljno dobro opisuje različite vrste KD.
- Da se koristi jedinstveni rečnik - tezaurus, koji opisuje različite vrste KD.
- Da se ustanove kulture, koje sprovode zaštitu KD, priključe Portalu za pretraživanje i prezentovanje KD.
- Da uključene ustanove kulture imaju baze podataka o štićenim KD i da su saglasne da određeni skup podataka o KD pruže na uvid korisnicima Portala.

5.3.1.2 Vizija i ciljevi sistema

Nakon analize opisa i ciljeva zaštite KD, navedenih u poglavlju 5.2.1 ove teze, usaglašen je sledeći iskaz vizije sistema:

"Očekujemo da se svim posetiocima veb portala "Kulturna dobra Srbije" omogući jedinstveno pretraživanje i prezentacija podataka o svim vrstama kulturnih dobara u Srbiji, nezavisno od ustanova za zaštitu KD, koje se o njima staraju.".

Ovako formulisana vizija ukazala je na osnovne ciljeve i mogućnosti sistema:

- Ugao korisnika sistema
  Nezavisno od poznavanja principa zaštite i klasifikacija KD, korisnik će moći da kreira proizvoljan upit, zasnovan jedino na ličnom interesovanju i ličnim potrebama. Kao odgovor na postavljeni upit, dobiće spisak svih KD, koja zadovoljavaju postavljene kriterijume.

- Ugao specijalizovanih ustanova za zaštitu KD
  Ustanove kulture mogu iskoristiti mogućnost da prezentuju podatke o KD, o kojima se staraju, široj grupi zainteresovanih korisnika sistema. Pored ispunjenja zakonskih obaveza, unapredićе sopstveni nivo konkurentnosti u okviru grupe srodnih ustanova.

Zahtevi sistema, ograničenja i preduslovi za realizaciju, i definisana vizija i ciljevi sistema, osnov su za određivanje koncepta rešenja.

5.3.1.3 Koncept rešenja

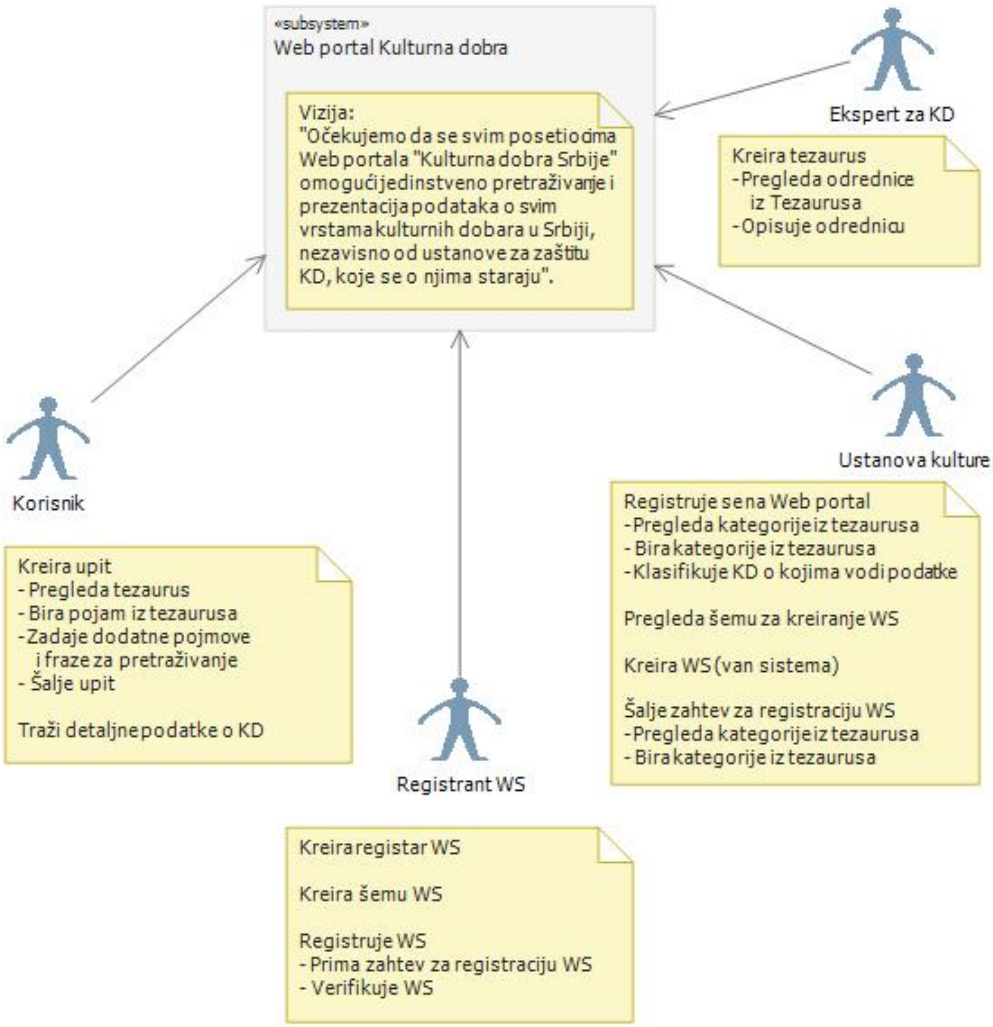
Analizom trenutnog stanja, uočeno je da postojeći veb sajtovi, koji se bave prezentovanjem i zaštitom KD, pružaju određene informacije. Međutim, korisniku se nudi suviše mnogo pojedinačnih izvora podataka, koje mora da pregleda, a zatim da sam proceni relevantnost ponuđenih informacija u odnosu na svoje, upitom iskazane potrebe.

Polazeći od ciljeva i zahteva sistema, odlučeno je da osnova budućeg rešenja bude novi veb portal. Bez namere da se menjaju i dopunjuju postojeće veb prezentacije ustanova kulture, opredeljenje je da se iskoriste bogatiji izvori informacija o KD: raspoložive baze podataka ustanova kulture, koje se priključe Portalu.

Sa ciljem, da se celokupnoj hibridnoj grupi obezbede kvalitetne usluge i za njih značajne informacije, veb portal bi trebalo da zadovolji određene funkcionalne zahteve, kao što su:

- Plan ili projekat izrade i korišćenja veb portala zasnovan je na meta modelu baza podataka za evidentiranje podataka o KD.
- Interfejs za kreiranje korisničkih upita, zasnovan je na postojećim tezaurusima. Izbor, verifikaciju i dopunu tezaurusa obavljaju eksperti za zaštitu kulturnih dobara.
- Podaci, potrebni za prikaz informacija o pojedinačnim KD, ili grupi KD, primarno se nalaze u postojećim bazama podataka ustanova za zaštitu KD, kojima nije moguć direktan pristup. Ustanove kulture priključuju se veb portalu, uz inicijalnu registraciju i navođenje vrsta KD o kojima se staraju. Prijavljene ustanove kulture upisuju se u jedinstveni registar i veb portal sa njima komunicira posredstvom veb servisa.
- Veb servise kreiraju same ustanove kulture, pridržavajući se unapred definisane šeme za opis veb servisa. Kreirani veb servis, ustanova kulture semantički opisuje i registruje preko veb portala.
- U zavisnosti od semantike kreiranog upita, odgovarajućim bazama podataka pristupa se ad hoc izborom odgovarajućih veb servisa.
- Rezultati pretraživanja prikazuju se u obliku spiska KD, koji zadovoljava postavljeni korisnički upit.

Šematski prikaz koncepta rešenja prikazan je na slici 5.1.



Slika 5.1: Koncept rešenja

5.3.2 Poslovni model

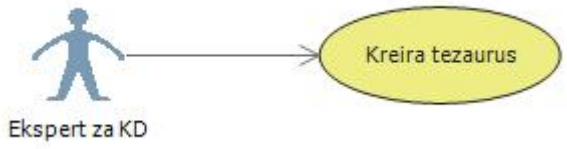
SOFI metodološki okvir za dizajniranje poslovnog modela, predviđa izradu sledećih artifakata:

- slučajevi korišćenja;
- scenariji upotrebe;
- BPMN dijagram;
- prototip UI (skica korisničkog interfejsa);
- konceptualni (domenski) model klasa;
- konceptualni model podataka;
- dijagram isporuke.

Kolekcije navedenih artifakata, kojima se popunjavaju ćelije SOFI okvira, kreirane su konsultovanjem generičkog meta modela, koji je prikazan u odeljku 4.2.2 ove teze. U ovom poglavlju dati su prikazi i komentari izvedenih modela.

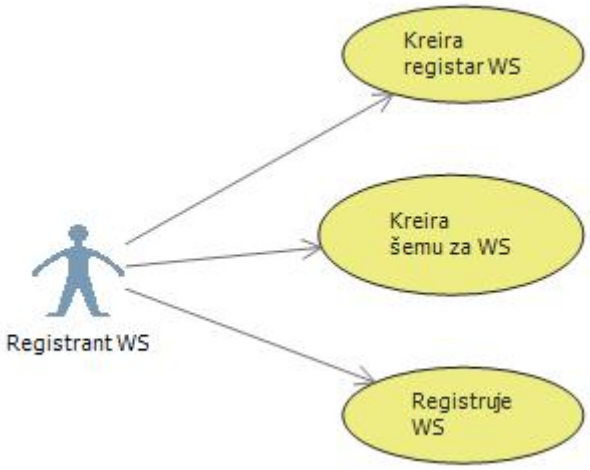
5.3.2.1 *Slučajevi korišćenja*

UML dijagrami slučajeva korišćenja urađeni su na osnovu zahteva, vizije i koncepta rešenja, koji su postavljeni sistemu za pretraživanje i prezentaciju podataka o KD. Sistem opslužuje korisnike, koji imaju sledeće uloge: ekspert za KD, registrant WS, ustanova kulture, vlasnik podataka o KD i krajnji korisnik sistema. Odgovarajući slučajevi korišćenja sukcesivno su prikazani na slikama 5.2, 5.3, 5.4 i 5.5.



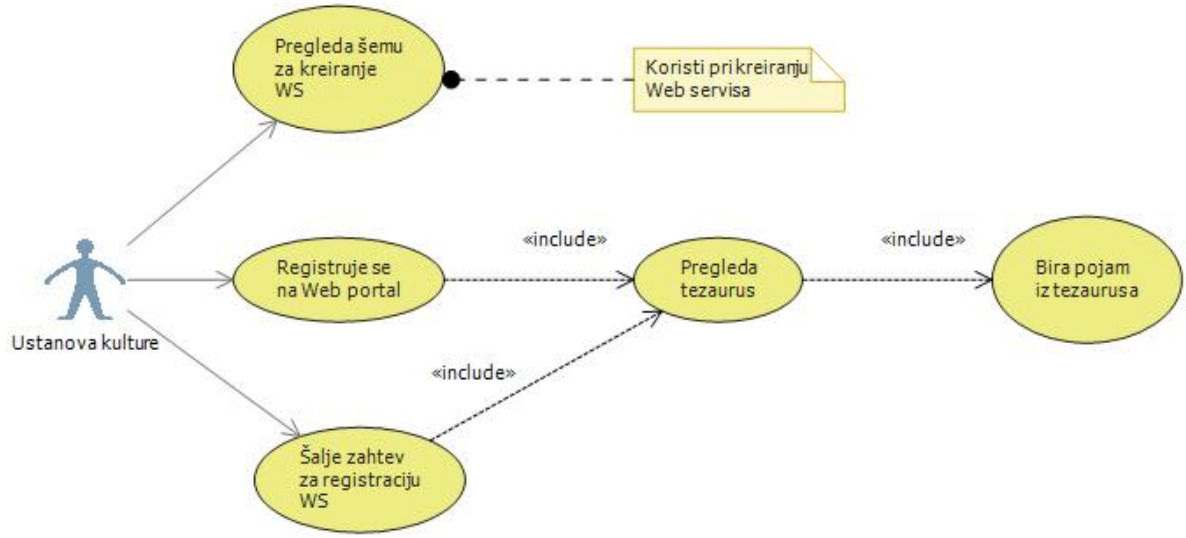
Slika 5.2: Slučajevi korišćenja za učesnika: Ekspert za KD

Funkcije eksperta za KD obuhvataju kreiranje i vođenje tezaurusa. Jednom kreirani tezaurus vrlo se retko menja i dopunjuje novim pojmovima, s obzirom da se novi pojmovi retko pojavljuju. Samim tim, aktivnost se retko izvodi, najčešće jednokratno.



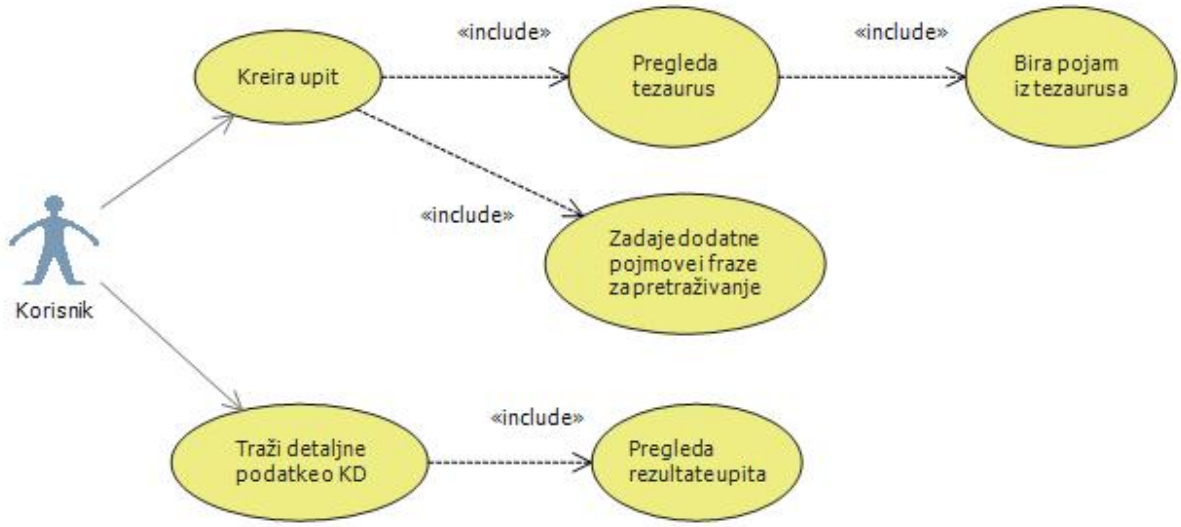
Slika 5.3: Slučajevi korišćenja za učesnika: Registrant WS

Aktivnosti registranta WS opisane su sledećim funkcijama: kreira registar veb servisa (WS); kreira šemu za WS i registruje WS. Kreiranje registra WS i šeme za WS su inicijalne aktivnosti i izvršavaju se jednokratno. Aktivnost "registrovanje WS" izvršava se u momentu, kada ustanova kulture prosledi zahtev za registraciju svog veb servisa.



Slika 5.4: Slučajevi korišćenja za učesnika: Ustanova kulture

Funkcije ustanove kulture su: pregled šeme koju će koristiti pri kreiranju veb servisa; pristupanje (upisivanje ustanove kulture u registar) i navođenje vrsta KD o kojima se stara; registracija kreiranog veb servisa i njegov opis u vidu semantičke klasifikacije na osnovu tezaurusa. Preduslovi za realizaciju aktivnosti ustanove kulture su već kreirani: tezaurus; registar WS i šema za kreiranje WS.



Slika 5.5: Slučajevi korišćenja za učesnika: Korisnik

Aktivnosti krajnjih korisnika opisane su funkcijama koje obuhvataju: biranje ključnih reči za kreiranje upita i pregled rezultata upita. Preduslov za izvršenje aktivnosti je postojanje tezaurusa.

Portal, pored obezbeđenja interfejsa ekspertu za KD, ustanovama kulture i krajnjem korisniku, preuzima ulogu posrednika između korisnika: krajnjeg korisnika sistema i ustanova kulture. Osnovne funkcije sistema su: da kreira tezaurus; da vodi registar ustanova kulture, koje su pristupile portalu i kreirale svoje veb servise; da analizira upite korisnika i šalje zahteve relevantnim ustanovama kulture, a na osnovu opisa vrsta KD, koje su ustanove kulture prijavile pri registrovanju i da pripremi i

prikaže rezultate upita krajnjem korisniku. Navedene funkcije na sebe preuzimaju uloge: Ekspert za KD i Registrant WS.

5.3.2.2 Scenariji upotrebe

Scenariji upotrebe rađeni su na osnovu UML dijagrama slučajeva korišćenja. U tabeli 5.1 prikazan je jedan od urađenih slučajeva upotrebe.

Tabela 5.1: Scenario upotrebe za slučaj korišćenja "Korisnik"

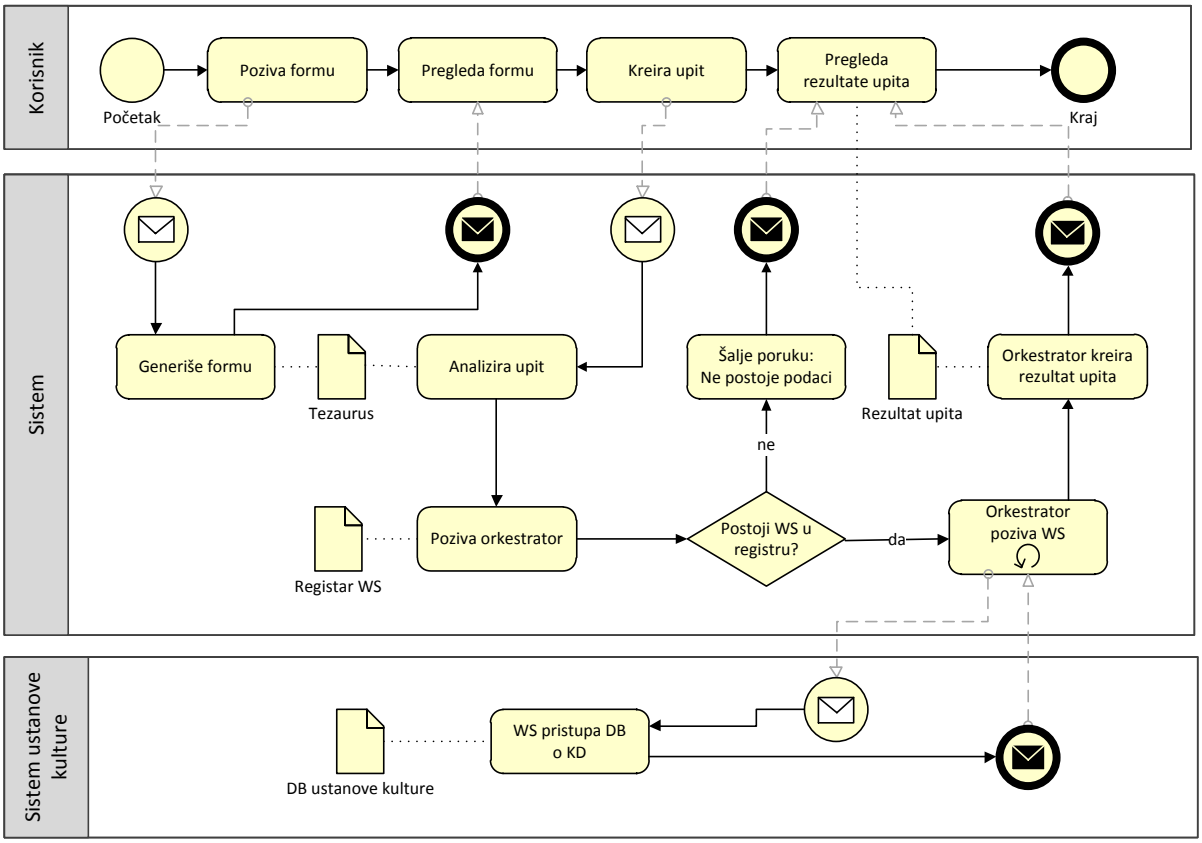
Slučaj korišćenja:	Korisnik (prikazan na slici 5.5)
Naslov:	Pretraživanje podataka o kulturnim dobrima (KD)
Namena:	Vraća podatke o jednom, ili više KD
Opis scenarija:	Korisnik veb portala želi da pretraži i pregleda podatke o jednom, ili više KD, koja odgovaraju pojmovima iz upita, koji je postavio.
Pretpostavke i preduslovi:	Veb portal je dostupan korisniku.
Učesnici:	Korisnik
Glavni tok procesa:	1. Slučaj korišćenja počinje kada korisnik pozove formu za pretraživanje podataka o KD. 2. Sistem kreira formu, u koju uključuje kategorije pojmova iz tezaurusa. 3. Korisnik kreira upit, birajući jedan, ili više ponuđenih pojmova i ako želi, pridodaje svoju frazu za pretraživanje KD. 4. Sistem prikazuje spisak KD, koja su zadovoljila korisnikov upit. 5. Slučaj korišćenja se završava kada se odabrane informacije isporuče.
Alternativni tok procesa:	1. Ukoliko ne postoje KD, koja zadovoljavaju upit korisnika, sistem šalje korisniku odgovarajuću poruku. 2. Slučaj korišćenja se restartuje da bi se korisniku obezbedila ispravka, ili postavljanje novog upita.
Koristi/dopunjuje (uses/extends)	Uključuje: 1. Pregleda tezaurus 2. Bira pojmove iz tezaurusa 3. Pregleda rezultate upita Dopunjuje: ne
Učestanost procesa:	Vrlo često.
Ovlašćenje	Slobodan pristup, nije potrebno logovanje

Na priloženom scenariju upotrebe uočeno je nekoliko bitnih činjenica, od značaja za razvoj sistema:

- Pored opisa glavnog slučaja korišćenja, opisan je i alternativni tok, čime je ceo slučaj korišćenja potpuno podržan.
- Korisnik ima slobodan pristup Portalu, odnosno, ne zahteva se registracija korisnika za pretraživanje sadržaja.
- Predviđa se visoka učestanost zahteva.

5.3.2.3 BPMN dijagram

BPMN dijagram urađen je na osnovu koncepta rešenja, slučajeva korišćenja i scenarija upotrebe.



Slika 5.6: BPMN dijagram

BPMN dijagram obuhvatio je tri celine (bazena): Korisnik, Sistem i Sistem ustanove kulture. U okviru bazena prikazani su: inicijatori aktivnosti; redosled, dinamika i uslovljenost izvršavanja aktivnosti; međusobna razmena poruka, kao i komunikacija između aktivnosti, koje se izvršavaju u različitim bazenima.

Centralni deo kreiranog BPMN dijagrama objašnjava aktivnosti za pristup bazama podataka ustanova kulture, koje se nalaze van sistema i kojima nije moguće neposredno pristupiti, da bi se pretražili podaci o KD. Zato, saglasno konceptu rešenja, za pristupanje bazama podataka o KD, koristiće se odgovarajući veb servisi ustanova kulture. Da bi se omogućilo efikasno otkrivanje veb servisa iz registra, analizator upita obaviće semantičko mapiranje originalnog upita korisnika u formalizovani upit, uključenjem pojmova iz tezaurusa. Kreirani, formalizovani upit omogućava da aktivnost "Orkestrator pôziva WS" otkrije i pozove samo veb servise ustanova kulture, koji će najpreciznije odgovoriti zahtevu korisnika. Naknadnom analizom ustanoviće se način izvršavanja aktivnosti "Orkestrator pôziva WS".

5.3.2.4 Prototip UI

Korisnički interfejs Portala realizovaće se korišćenjem veb tehnologija. Osnovne forme za komunikaciju krajnjeg korisnika sa sistemom i ustanove kulture sa sistemom su:

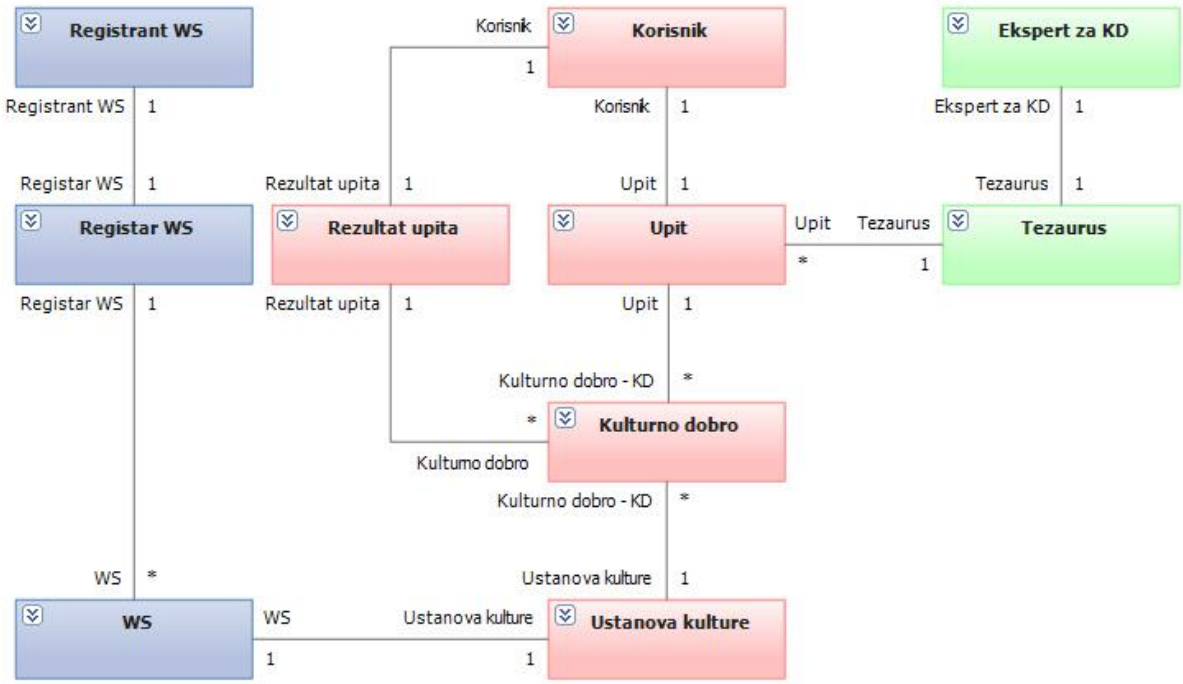
- Forma na kojoj ekspert za KD kreira tezaurus.
- Forma za prijavljivanje ustanova kulture koju, sa elementima tezaurusa obezbeđuje sistem. Ustanova kulture, prilikom prijavljivanja, upisuje svoje osnovne podatke, a nakon toga, može da registruje svoje veb servise.

- Forma za registrovanje veb servisa namenjena je ustanovama kulture, koje su se već priključile veb portalu. Ustanova kulture registruje i opisuje svoj veb servis, koristeći uputstva i pravila, koje obezbeđuje sistem.
- Forma na kojoj korisnik kreira upit, a na osnovu tezaurusa, koji mu obezbeđuje sistem.
- Forma sa spiskom rezultata upita, koje mu je vratio sistem, i na kojoj može da odabere detaljan prikaz podataka o nekom KD sa dobijenog spiska.

Konačni izgledi navedenih formi prikazani su u 5.3.3.4 ove teze.

5.3.2.5 *Konceptualni (domenski) model klasa*

UML dijagram konceptualnog (domenskog) modela klasa urađen je na osnovu koncepta sistema, slučajeva korišćenja, scenarija upotrebe i BPMN dijagrama. Domenski model klasa prikazan je na slici 5.7.



Slika 5.7: Domenski model klasa

Izvedeni model obuhvatio je, osim domenskih klasa, i međusobne veze, odnose i uslovljenost klasa. Model prikazuje najvažnije vrste klasa u domenu veb portala. Na modelu se uočavaju tri vrste klasa: klase za registrovanje veb servisa, klase za korišćenje veb portala i klase za kreiranje tezaurusa.

5.3.2.6 Konceptualni model podataka

Konceptualni model podataka urađen je na osnovu zahteva, slučajeva korišćenja i scenarija upotrebe. Imajući u vidu opis i analizu zahteva sistema, uključujući granice i postojeća ograničenja sistema, konceptualni model podataka rađen je bez uvida u logičke i fizičke strukture baza podataka o KD, koje su u vlasništvu ustanova kulture.

Polazeći dalje od zahteva sistema, model podataka morao je da podrži korišćenje kontrolisanog rečnika u obliku tezaurusa. Iz navedenih razloga, urađena su dva konceptualna modela podataka,

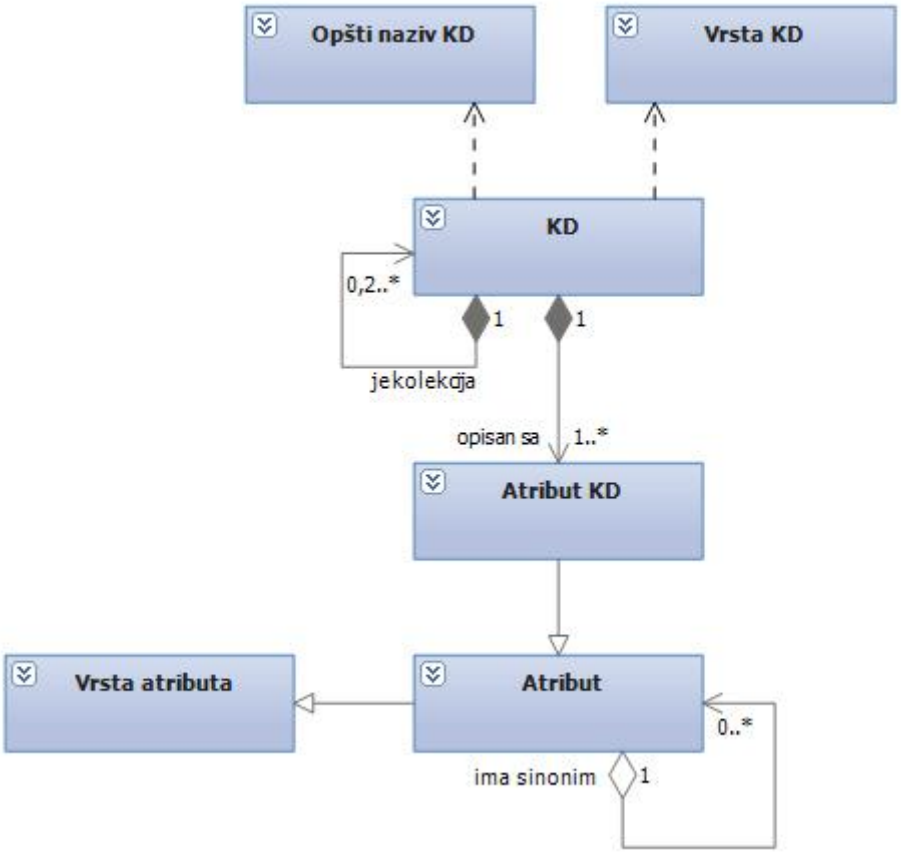
koji se međusobno dopunjuju. Prvo je urađen meta model baza podataka o KD, nakon čega je urađen meta model tezaurusa. Navedeni modeli kreirani su na osnovu generičkog meta modela, koji je predstavljen u odeljku 4.2.2 ove teze.

Meta model baza podataka o KD

Rezultati analize profila korisnika veb portala i razmatranje mogućih scenarija u potpunosti su definisali potrebe korisnika. Sa namerom da veb portal za pretraživanje i prezentaciju podataka o KD u Srbiji zadovolji sve identifikovane grupe korisnika i njihove potrebe, mora se omogućiti intuitivno kreiranje upita za pretraživanje i prikaz informacija o KD. Pri tome, treba imati u vidu činjenicu da se različite vrste KD, artifakti, značajno razlikuju. Razlikuju se: po prvobitnoj nameni; poreklu i starosti; dimenzijama; materijalima i tehnikama izrade. Osim toga, veb portal trebalo je u potpunosti da obuhvatiti dve, po pojavi suštinski različite grupe KD: materijalna i nematerijalna KD. Materijalna KD su prostorno i pojavno potpuno definisana. Nematerijalna KD manifestuju se pre svega u okviru misaonih procesa, koji se, ponekad, dopunjuju dodatnim, fizičkim predstavama.

Polazeći od činjenice da se raspoloživi podaci o KD nalaze u različitim bazama podataka ustanova za zaštitu KD, čiji logički i fizički modeli nisu poznati, urađen je meta model baza podataka o KD. Cilj izrade meta modela je da preuzme ulogu nedostupnih logičkih i fizičkih modela baza podataka o KD i da, na osnovu njega, sistem korisnicima ponudi opštu, intuitivnu formu za kreiranje upita.

Kreirani meta model zasnovan je na analizi životnog ciklusa KD: od nastanka i prvobitne namene; akvizicije (nabavke), pa do naučno-stručne obrade i proglašenja artifakta za KD. Dobra osnova za izradu meta modela baza podataka o različitim vrstama KD, bili su rezultati analize životnog ciklusa KD [104]. Meta model baza podataka o KD prikazan je na slici 5.8.



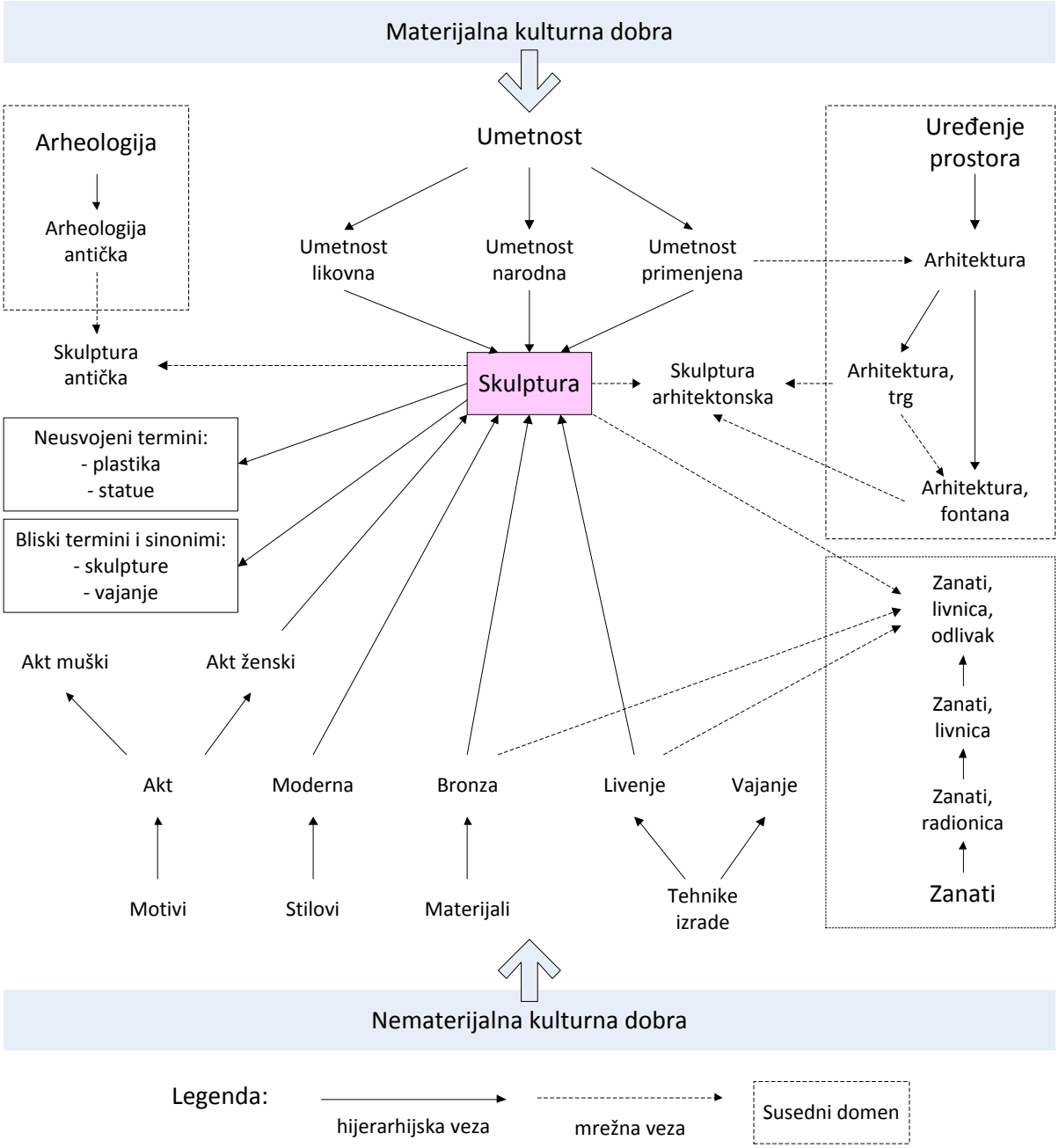
Slika 5.8: Meta model baza podataka o KD

Konceptualni model tezaurusa u oblasti KD

Tezaurus je tematski ograničen skup pojmova, zasnovan na kontrolisanom rečniku. Namena i opšte karakteristike tezaurusa opisane su u poglavlju 4. ove teze. Osnova tezaurusa su odrednice i pododrednice. Glavni deo tezaurusa čine opšte odrednice. Primeri opštih odrednica u oblasti KD su [105] [106] [107]:

- Vrste likovnih umetnosti: Apstraktna umetnost, Narodna umetnost, Umetnost naivaca, Narodno graditeljstvo;
- Oblasti likovnih umetnosti: Arhitektura, Slikarstvo, Skulptura, Grafika, Primenjena umetnost;
- Pravci i stilovi umetnosti: Barok, Romantizam, Moderna;
- Vrste umetničkih dela: Arhitektonski spomenik, Skulptura, Slika;
- Nazivi artefakata: Zgrada, Trg, Spomenik, Oružje, Novac, Nakit, Odeća;
- Tehnike izrade: Bakropis, Akvarel, Livenje, Kovanje, Mašinska obrada;
- Materijali izrade: Ulje, Bronza, Glina, Zlato;
- Etnička pripadnost: Srpska, Makedonska, Vlaška;
- Geografske odrednice: Istočna Srbija, Banat, Raška, Kosovo i Metohija, Bugarska;
- Opšte poznate i značajne ličnosti: Milan Konjović, Nikola Tesla, Ivan Meštrović.

Uloga pododrednica u tezaurusu je da dopune odrednicu i ukažu na aspekt posmatranja artifakta: teritorijalnu povezanost, vremensku ograničenost, etničku pripadnost i slično. Prema ulozi koju imaju, pododrednice se svrstavaju u: tematske, geografske, vremenske i pododrednice forme.

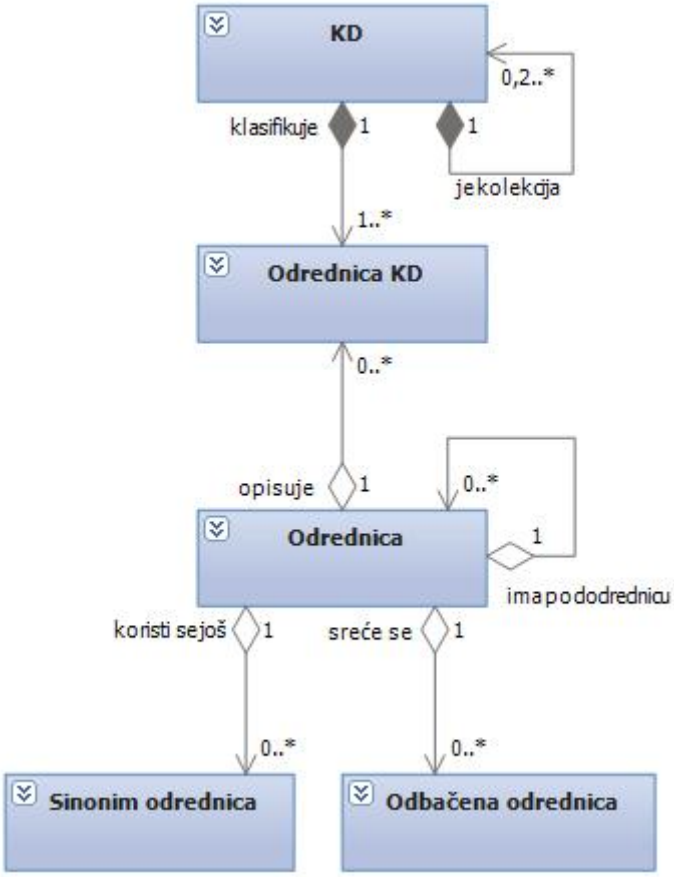


Slika 5.9: Primer odrednice "Skulptura"

Prikazani tezaurus u oblasti KD objedinio je dve vrste KD: materijalna KD i nematerijalna KD. Na primeru samo jedne odrednice (Skulptura), jasno se uočava složenost strukture tezaurusa. Određeni broj termina ima dvostruku ulogu: koriste se i kao odrednice i kao pododrednice. Primeri su: Bronza (kao materijal i vrsta tehnike); Moderna (kao umetnički pravac, ali i kao pododrednica KD) i slično. Sastavni deo tezaurusa su neusvojeni termini. Uočava se da je prikazana struktura hijerarhijsko-mrežna. Primeri hijerarhije su: Umetnost – Umetnost, likovna – Skulptura. Način uspostavljanja mrežne strukture zavisi od vrste tezaurusa. Ukoliko je reč o domenskom tezaurusu u oblasti KD, odrednice, koje učestvuju u mrežnoj strukturi, uključiće se u domenski tezaurus po osnovu tematske pripadnosti. Primeri su: Skulptura, antička i Skulptura, arhitektonska. Preko navedenih odrednica uspostavlja se veza sa susednim oblastima, u ovom slučaju sa: Arheologijom i Uređenjem prostora.

U slučaju da je reč o domenskom tezaurusu u oblasti Uređenja prostora, odrednica Skulptura, arhitektonska bi se, po osnovu tematske pripadnosti, uključila u domenski tezaurus u oblasti Uređenja prostora.

Polazeći od namene, sadržaja i logičke strukture tezaurusa, kao domenske ontologije [102] i generičkog meta modela, koji je opisan u odeljku 4.2.2 ove teze, kreiran je konceptualni model tezaurusa u oblasti KD. Dobijeni model prikazan je na slici 5.10.



Slika 5.10: Konceptualni model tezaurusa u oblasti KD

Meta model baza podataka o KD i konceptualni model tezaurusa u oblasti KD poslužili su:

- za uvođenje u sistem domenskog tezaurusa, koji dovoljno dobro reprezentuje podatke o KD;
- za kreiranje forme za prijavljivanje ustanova kulture, registraciju i semantičko opisivanje njihovih veb servisa;
- za kreiranje forme za upit na veb portalu, na kojoj se korisniku nude tematske odrednice, sadržane u predmetnom, kontrolisanom rečniku - u ovom slučaju iz kreiranog tezaurusa o KD;
- za ad hoc pozivanje WS, potrebnih za realizaciju korisničkog upita;
- za međusobnu razmenu podataka između veb portala i odabranih WS, a na osnovu njihove semantike.

5.3.3 Model sistema

SOFI metodološki okvir za dizajniranje modela sistema, predviđa izradu sledećih kolekcija modela:

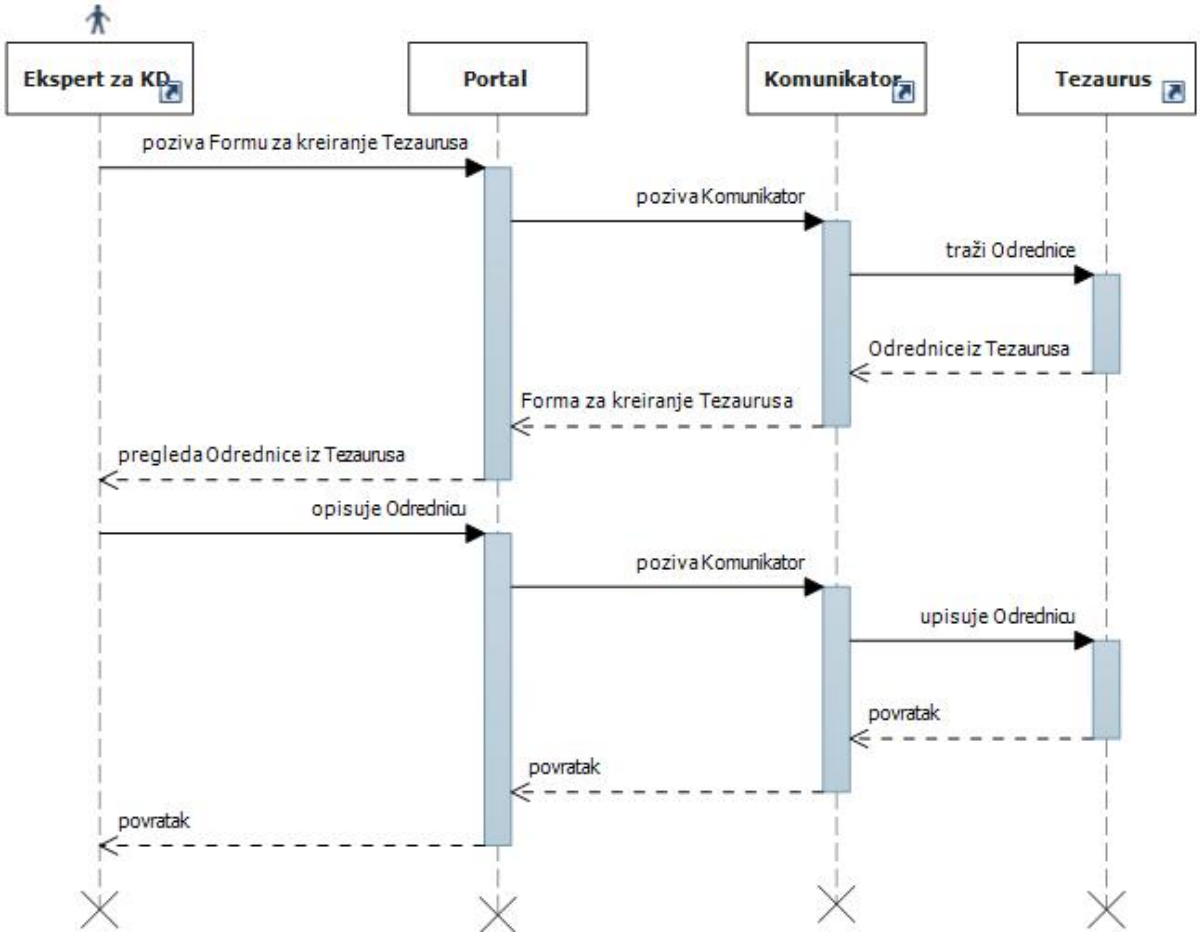
- dijagram sekvenci;
- logički model klasa;
- model servisa;
- dizajn UI;
- arhitektura sistema;
- logički model podataka;
- logički dijagram isporuke.

U ovom poglavlju dati su prikaz i komentari izvedenih modela.

5.3.3.1 Dijagrami sekvenci

Modelovanje sistema počinje kreiranjem UML dijagrama sekvenci: UML dijagram sekvenci za kreiranje tezaurusa; UML dijagram sekvenci za kreiranje šeme WS; UML dijagram sekvenci za registraciju WS ustanova kulture i UML dijagram sekvenci za pretraživanje podataka o KD.

UML dijagram sekvenci za kreiranje tezaurusa



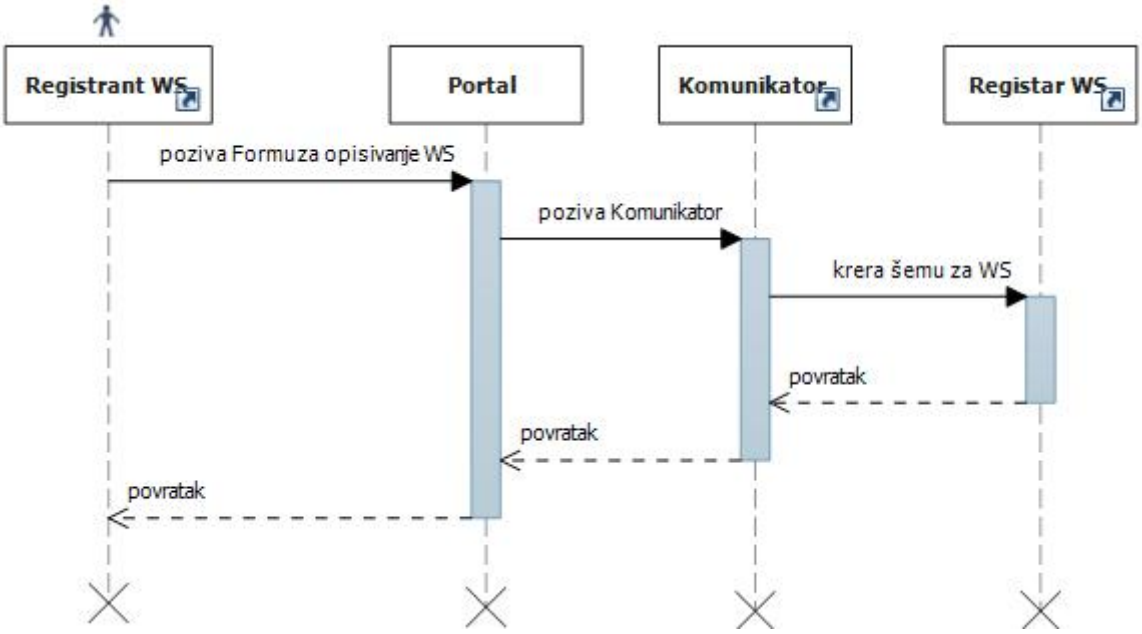
Slika 5.11: Dijagram sekvenci za kreiranje tezaurusa

Kreiranje tezaurusa obavlja ekspert za KD. Formu za kreiranje tezaurusa sistem prosleđuje preko komunikatora. Dijagram sekvenci za kreiranje tezaurusa prikazan je na slici 5.11.

Uloga tezaurusa u sistemu je da obezbedi kriterijume i uslove za uspostavljanje interoperabilnosti između veb portala i sistema uključenih ustanova kulture. Uspostavljanje interoperabilnosti izvodi se u okviru dve faze. U pripremnoj fazi, tezaurus se koristi da ustanove kulture opišu sebe i svoje WS. Ovakav, "kontrolisani" opis WS ustanova kulture omogućava uspostavljanje dinamičke interoperabilnosti, u cilju pronalaženja rezultata na upit korisnika portala, ad hoc pozivanjem samo WS, čiji opisi odgovaraju pojmovima iz formalizovanog upita korisnika.

UML dijagram sekvenci za kreiranje šeme WS

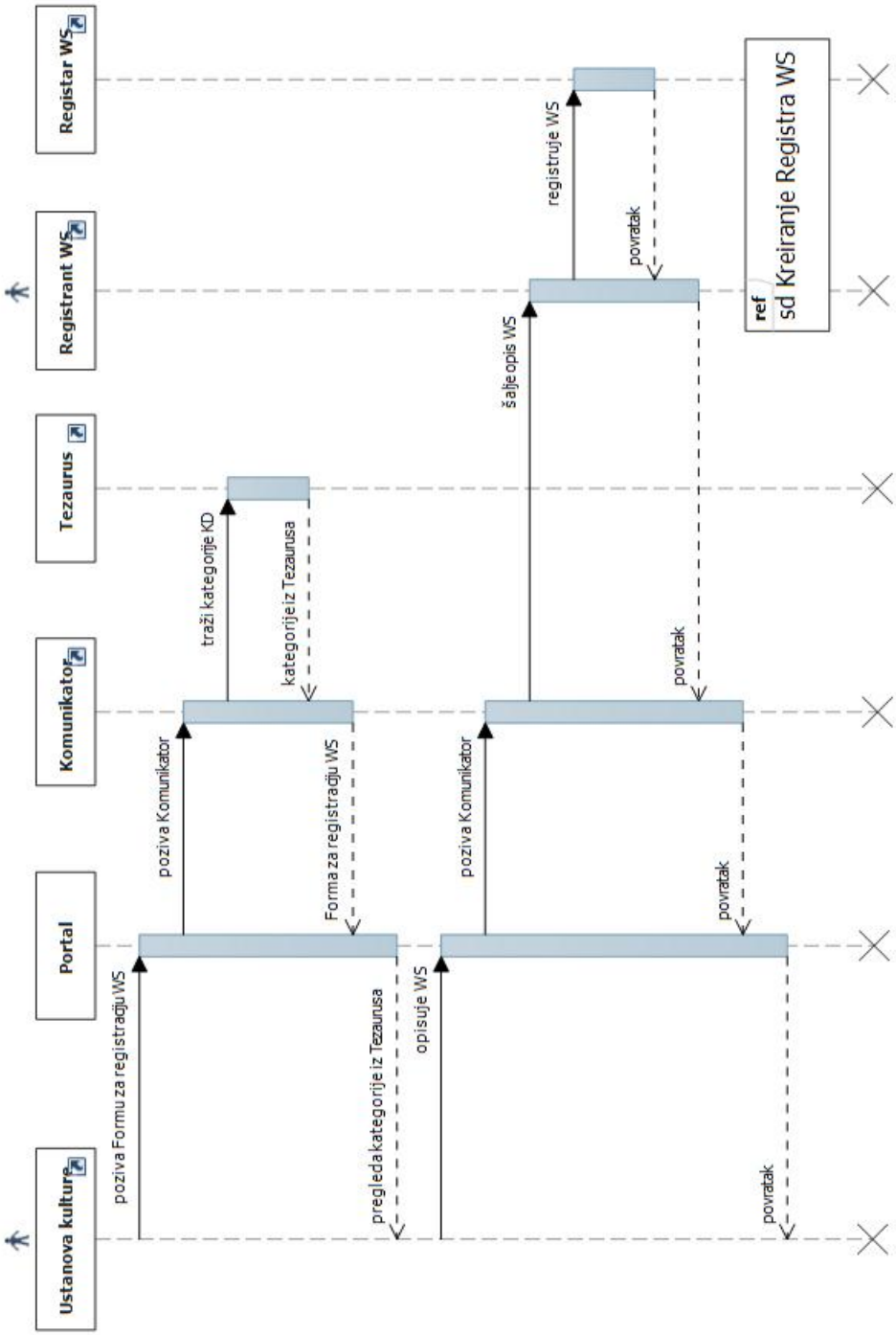
Kreiranje šeme WS obavlja registrant WS, posredstvom komunikatora. Šema WS upisuje se u registar WS. Dijagram sekvenci za kreiranje šeme WS prikazan je na slici 5.12.



Slika 5.12: Dijagram sekvenci za kreiranje šeme WS

Dijagram sekvenci za registraciju WS ustanova kulture

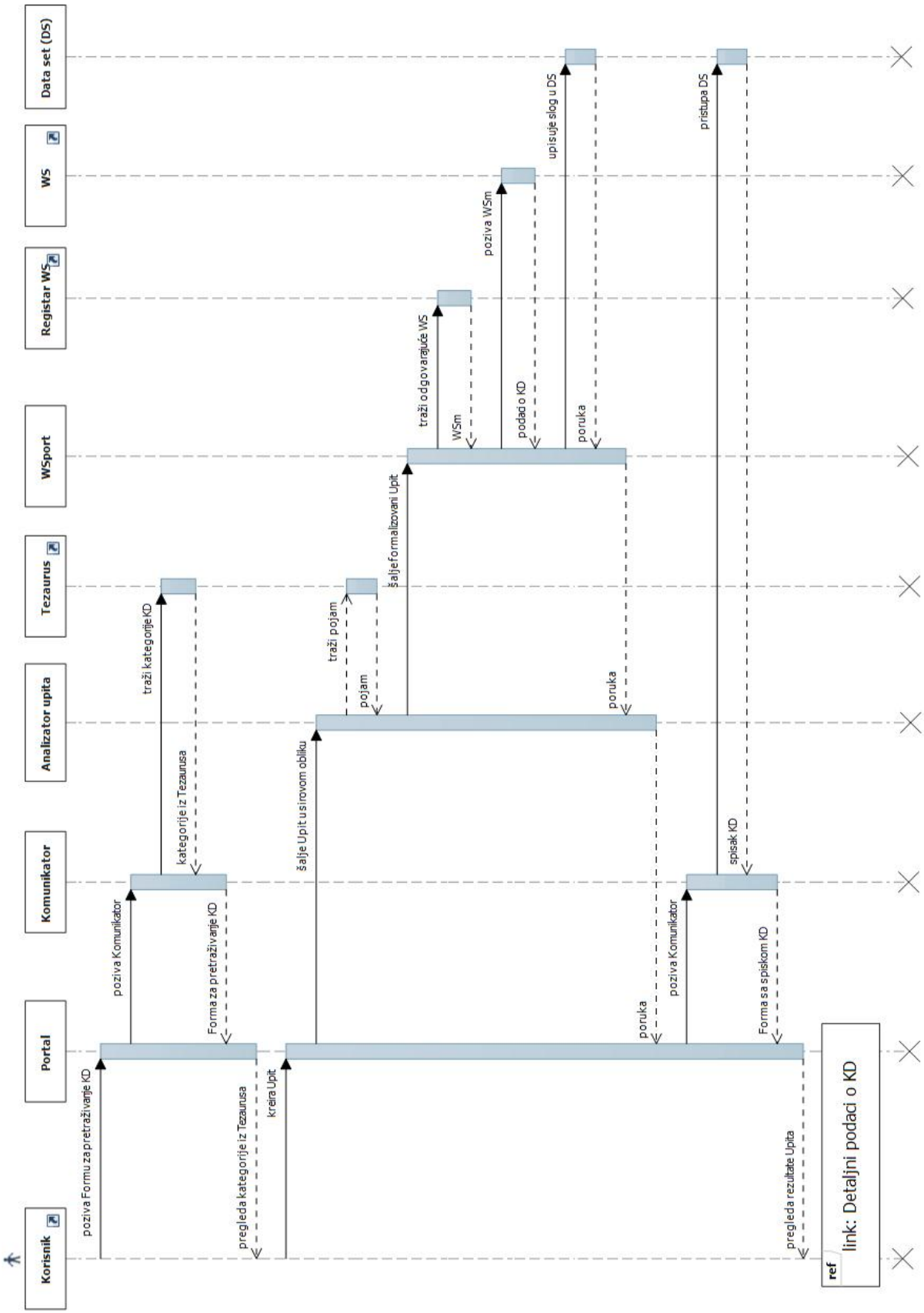
Komunikacija veb portala, sa podacima o KD, obavlja se posredstvom WSm, koje kreiraju ustanove za zaštitu KD. Kreiranje WSm vrši se na osnovu unapred zadatog inicijalnog opisa, uspostavljenih pravila i šeme za kreiranje veb servisa ustanova kulture. Elementi sistema, koji učestvuju u procesu registracije WSm i kreiranju registra WSm su: komunikator, tezaurus i registrant. Uloga tezaurusa je da, pri registraciji veb servisa, obezbedi kontrolisani skup reči i fraza za opis WSm, kojima će komunikator napuniti formu za registraciju WSm. Opisan WSm preuzima registrant i upisuje ga u registar veb servisa. Dijagram sekvenci za registraciju veb servisa ustanova kulture prikazan je na slici 5.13.



Slika 5.13: Dijagram sekvenci za registrovanje WS ustanova kulture

Dijagram sekvenci za pretraživanje podataka o KD

Glavni elementi sistema, koji uspostavljaju interoperabilnost pri pretraživanju baza podataka o KD su: tezaurus i WS ustanova kulture. Dijagram sekvenci za pretraživanje podataka o KD prikazan je na slici 5.14.



Slika 5.14: Dijagram sekvenci za pretraživanje podataka o KD

Prikazani elementi sistema imaju dvostruku ulogu: da obezbede semantiku, potrebnu za uspostavljanje interoperabilnosti i da, kao funkcije sistema, izvrše zadate upite korisnika.

Semantički preduslovi obezbeđeni su uvođenjem tezaurusa. Opis i konceptualni model tezaurusa prikazani su u odeljku 5.3.2.6.

- Komunikator
  - Komunikator, na poziv Portala, a konsultujući tezaurus, priprema korisniku portala formu za kreiranje upita.
  - Komunikator, na poziv Portala, pristupa tabeli sa rezultatima upita (Data set - DS) i priprema spisak sa rezultatima, koji vraća Portalu.
- Analizator upita
  - Analizator upita, na poziv Portala, prihvata i formalizuje upit korisnika.
  - Formalizovani upit, u obliku parametara, analizator upita prosleđuje portalskom veb servisu (WSport).

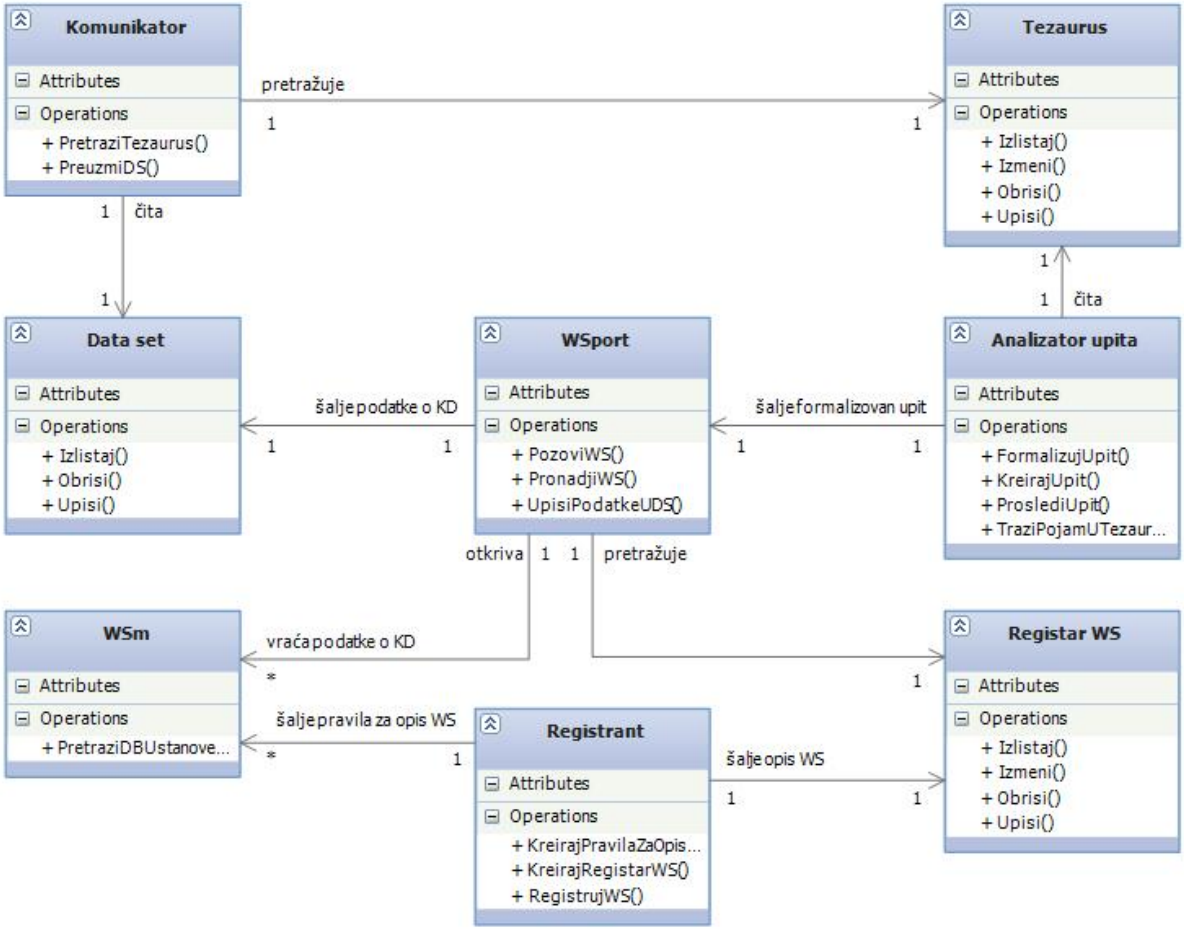
U sistem su uključene dve vrste veb servisa: portalski veb servis (WSport) i veb servisi ustanova kulture (WSm):

- portalski veb servis - WSport
  WSport je najaktivniji element sistema. Preuzima na sebe izvršavanje više funkcija:
  - Pristupa registru WS.
  - Otkriva WS ustanova kulture, koji semantički zadovoljavaju formalizovane parametre upita.
  - Orkestrira veb servise ustanova kulture.
  - Prihvata podatke, koje mu vraća pozvani WSm i kreira zapis o KD u tabeli DS.
- veb servisi ustanova kulture - WSm
  Zadatak WSm ustanova kulture je, da na poziv WSport, pristupe podacima ustanove kulture, čije su vlasništvo i da, veb servisu WSport, vrate podatke o KD, koja zadovoljavaju upit korisnika.

Na UML dijagramima sekvenci prikazani su elementi sistema i njihove međusobne veze, na osnovu čega su identifikovane klase i servisi sistema.

5.3.3.2 Logički model klasa

Na osnovu analize dijagrama sekvenci, identifikovane su sledeće klase: komunikator; tezaurus; analizator upita; registrant; registar WS; Data set i veb servisi (WSport i WSm). Dijagram logičkog modela klasa prikazan je na slici 5.15.



Slika 5.15: Dijagram klasa

Uloga komunikatora u sistemu je da obezbeđuje podatke za kreiranje elemenata za korisničke forme sistema. Potrebne podatke komunikator preuzima iz tezaurusa i Data seta, izvršavajući operacije: PretraziTezaurus() i PreuzmiDS().

Registrant je u sistemu preuzeo funkcije vezane za registraciju veb servisa ustanova kulture: WSm. Zadaci registranta su da: kreira registar WS; kreira pravila za opis WSm i registruje prijavljeni WSm ustanove kulture.

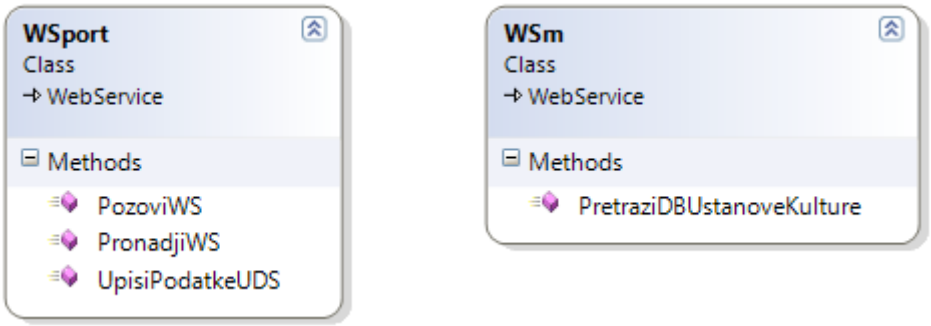
Analizator upita je, u pogledu funkcija, jedan od sofisticiranih objekata sistema, čiji je zadatak da obezbedi deo preduslova za uspostavljanje interoperabilnosti. Originalni, delimično neformalizovani upit korisnika, koji dobija od sistema, analizator upita preslikava u potpuno formalizovani upit. Formalizacija upita obavlja se cikličnim izvršavanjem operacije "TraziPojamUTezaurusu", pri čemu se značenja nestruktuiranih pojmova, iz originalnog upita korisnika, mapiraju u suštinski bliske pojmove iz tezaurusa. Novokreirani, formalizovani upit analizator upita prosleđuje veb servisu WSport.

Objekti: tezaurus, registar WS i Data set sistem koristi za pristup i manipulaciju podacima.

U narednom poglavlju opisani su veb servisi WSport i WSm.

5.3.3.3 Model servisa

Logički model servisa obuhvata dve vrste veb servisa: WSport i servisi ustanova kulture - WSm. Njihov zadatak je da u međusobnoj saradnji obezbede odgovor na upit korisnika. Modeli WSport i WSm prikazani su na slici 5.16.



Slika 5.16: Servisi WSport i WSm

WSport je najaktivniji deo sistema, s obzirom da ima ključnu ulogu u pronalaženju odgovora na upite korisnika. WSport, nakon prihvatanja formalizovanog upita, koji mu prosleđuje analizator upita, pristupa registru WS, sa ciljem da u registru WS otkrije veb servise WSm, čiji semantički opis odgovara parametrima formalizovanog upita. Otkrivene veb servise WSport orkestrira asinhrono i nakon primljenog odgovora, dobijene podatke o KD, upisuje u Data set.

Veb servisi ustanova kulture kreirani su po šemi registra WS, a semantički su opisani preko pojmova iz domenskog tezaurusa. Imajući u vidu da je domenski tezaurus, između ostalog, kreiran i na osnovu meta modela struktura baza podataka ustanova kulture, semantika opisa WSm predstavlja najbolju moguću interpretaciju stvarnih sadržaja baza podataka o KD. Zato je, kontrolisani, semantički opis WSm višestruko značajan za sistem. Omogućava da ih WSport direktno otkrije u registru WS, a zatim, da se u bazi podataka ustanove kulture, sa velikom verovatnoćom, pronađu odgovori na upit korisnika. Rezultat pretraživanja baze podataka, sa osnovnim podacima o KD i linkom na detaljnije podatke, WSm vraća veb servisu WSport. Kvalitet dobijenih rezultata zavisi od mogućnosti WSm da mapiraju upit na svoju bazu, od kompletnosti i kvaliteta podataka o KD. Bolji rezultati će se postići u slučajevima kada su strukture baza podataka ustanova kulture normalizovane.

Saradnja veb servisa WSport i WSm uspostavlja se ad hoc otkrivanjem i pozivanjem otkrivenog WSm. Na taj način, ugrađeni princip opisivanja WSm omogućava uspostavljanje interoperabilnosti na semantičkom nivou. Polazeći od navedene činjenice, pouzdanost uspostavljanja interoperabilnosti zavisi od sledećih elemenata sistema: kvaliteta i obima domenskog tezaurusa; opisa WSm i kvaliteta modela baza podataka ustanova za zaštitu KD.

5.3.3.4 Korisnički interfejs

Osnovni korisnički interfejs urađen je na osnovu koncepta i zahteva sistema. Prateći koncept sistema, a sa ciljem da se korisnicima Portala olakša kreiranje upita, logičke celine forme grupisane su u jasno izdvojene celine. Ključnu ulogu u okviru korisničkog interfejsa, bilo da je namenjen krajnjim korisnicima Portala, ili ustanovama kulture, preuzima domenski tezaurus. Time se postiglo

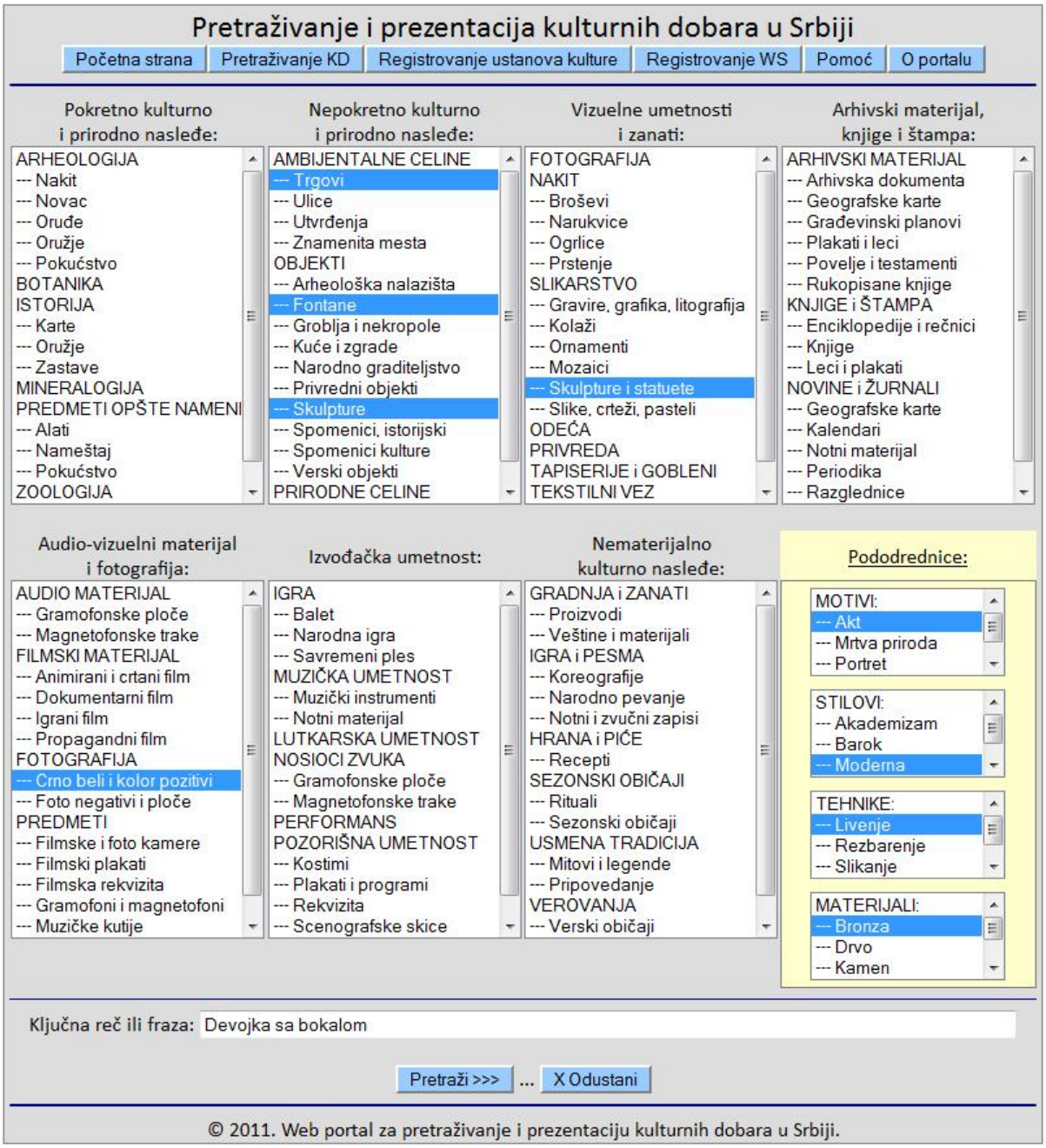
da svaki korisnik, nezavisno od cilja i stepena poznavanja oblasti KD, može intuitivno da kreira i najsloženije upite.

Ustanovama za zaštitu KD namenjena je forma za pristupanje Portalu i prijavljivanje veb servisa. Korišćenjem domenskog tezaurusa za inicijalno punjenje sadržaja forme, olakšano im je opisivanje i registrovanje WS, uz obezbeđenje semantičke konzistentnosti, zasnovane na domenskom tezaurusu.

Osnovne forme za komunikaciju krajnjeg korisnika sa Portalom i ustanove kulture sa Portalom su:

1. Forma na kojoj ekspert za KD kreira tezaurus.
2. Forma na kojoj korisnik kreira upit, sa pojmovima domenskog tezaurusa, koji obezbeđuje Portal.
3. Forma sa spiskom rezultata upita, koje je Portal vratio korisniku i na kojoj korisnik može da odabere detaljan prikaz podataka o nekom KD sa dobijenog spiska.
4. Forma za prijavljivanje ustanova kulture sa pojmovima domenskog tezaurusa, koje obezbeđuje Portal. Ustanova kulture, prilikom prijavljivanja, upisuje osnovne, matične podatke, a nakon toga, može da registruje svoje veb servise.
5. Forma za registrovanje veb servisa. Namenjena je ustanovama kulture, koje su se već priključile veb portalu. Ustanove kulture registruje svoj veb servis koristeći potrebna uputstva i pravila, koje obezbeđuje sistem.

Korisnička forma za kreiranje upita prikazana je na slici 5.17.



Slika 5.17. Korisnička forma za kreiranje upita

Urađeni interfejs zadovoljio je više važnih kriterijuma. Već na prvi pogled deluje samoobjašnjavajuće, što nije beznačajno u slučaju da pristupaju korisnici bez posebnog poznavanja oblasti KD. Drugi izuzetno bitan kriterijum, je mogućnost jednostavnog, intuitivnog kreiranja i najsloženijih upita, a zahvaljujući podršci, koju pruža aktivno korišćenje domenskog tezaurusa.

Primenjeni dizajn interfejsa za registrovanje i opisivanje WS, namenjen ustanovama kulture, pored lakoće korišćenja, obezbedio je podršku domenskog tezaurusa, čime se obezbeđuje konzistentan opis registrovanih WS, nezavisno od vrste KD koja se opisuju.

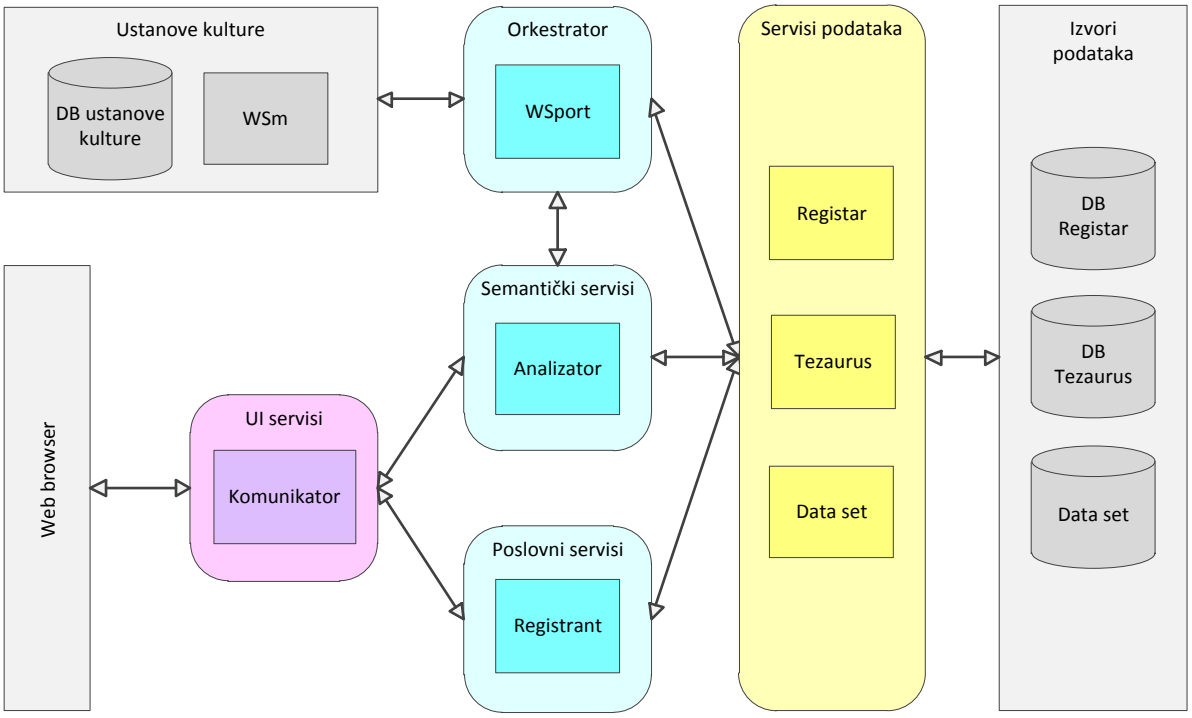
5.3.3.5 *Arhitektura sistema*

Arhitektura "Sistema za pretraživanje i prezentaciju KD" kreirana je na osnovu meta modela za arhitekturu, koji je definisan u poglavlju 4.1.2.2 ove teze. Arhitektura sistema zasnovana je na servisnoj orijentaciji i njene najvažnije elemente predstavljaju: veb servisi WSport i WSm i registar veb servisa.

Ulogu provajdera preuzimaju ustanove kulture. Njihov zadatak je da kreiraju sopstvene veb servise - WSm i da informacije o njima upišu u registar servisa. Pri publikovanju, WSm se semantički opisuje.

Za izvršavanje poslovnog procesa, koji omogućava posetiocu Portala da pronađe tražena kulturna dobra, uloga veb servisa WSport je da otkrije odgovarajuće WSm u registru veb servisa, da ih orkestrira i pozove na izvršavanje. Povezivanje WSm u kompoziciju je dinamičko i obavlja se pretraživanjem registra veb servisa na osnovu semantičkih kriterijuma. U predloženoj arhitekturi WSport ima dvostruku ulogu: on je i korisnik servisa i obavlja posao orkestratora servisa.

Registar WS obezbeđuje čuvanje osnovnih informacija o WSm, kao i njihov semantičku opis, kreiran u skladu sa domenskim tezaurusom. Njegova osnovna svrha je da omogući pretraživanje i pronalaženje informacija o WSm.

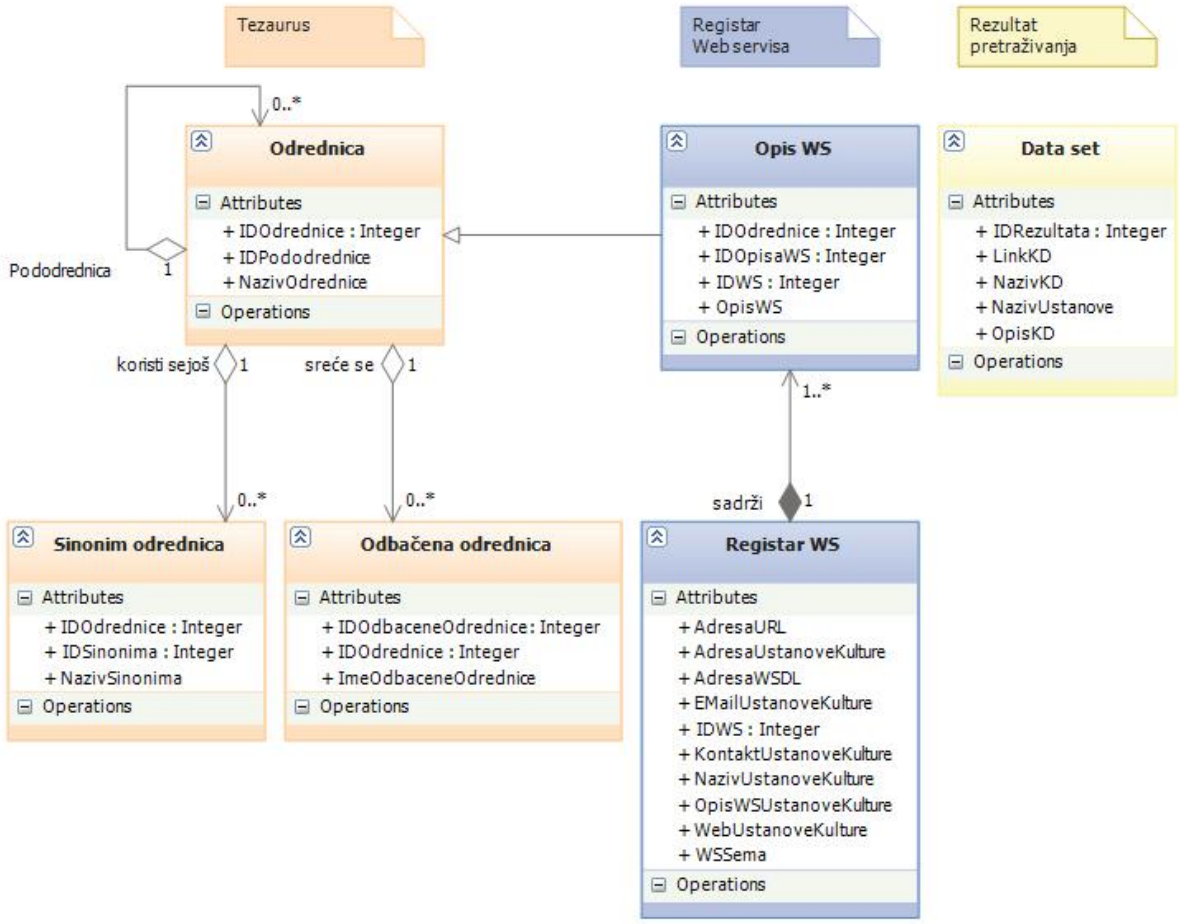


Slika 5.18: Arhitektura Sistema za pretraživanje i predstavljanje KD

Planirana arhitektura sistema je višeslojna. Karakteristične komponente sistema svrstane su u servisne slojeve, koji obezbeđuju: kreiranje korisničkog interfejsa, izvršavanje poslovnih pravila i pristup podacima. Na prikazanom modelu arhitekture sistema prikazani su samo servisi i komponente, specifični za sistem koji se modeluje.

5.3.3.6 Logički model podataka

Logički model podataka urađen je na osnovu konceptualnog modela podataka, dodavanjem identifikovanih ključeva, indeksa i atributa. Model je prikazan na slici 5.19.



Slika 5.19: Logički model podataka

Na modelu se uočavaju tri grupe objekata: Tezaurus, Registar veb servisa i Rezultat pretraživanja KD. Za razliku od ostalih, objekat Data set je privremeni objekat, čija je uloga da prihvati rezultate pretraživanja. Sadržaj Data seta kreira WSport, na osnovu odgovora pozvanih WS. Komunikator koristi Data set, s ciljem da korisniku vrati formatizovane rezultate upita.

Tezaurus je modelovan samoreferentnom strukturom, koja je obuhvatila: odrednice, uže odrednice, odbačene odrednice i sinonime. Registru veb servisa, u odnosu na javne registre, dodata je unakrsna tabela: Opis WS. Dodavanjem ovog objekta izvršena je klasifikacija veb servisa ustanova kulture, saglasno vrstama KD o kojima vode podatke. Informacije o klasifikaciji veb servisa predstavljaju osnovni kriterijum za izbor WS, koji su relevantni u odnosu na sadržaj upita korisnika. To znači da će WSport pozvati samo WS ustanove kulture, čija se klasifikacija poklapa sa sadržajem formalizovanog upita.

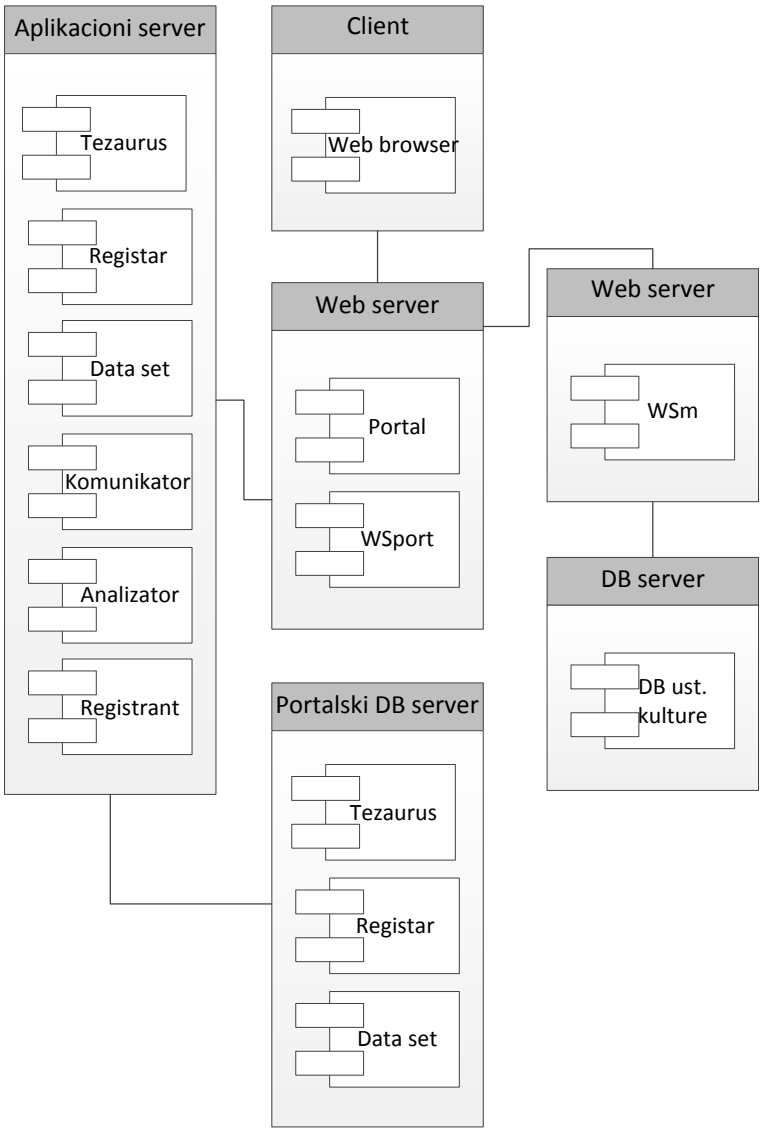
5.3.3.7 Logički dijagram isporuke

Logički model isporuke ukazuje na lokacije komponenti sistema, servisa i podataka. Struktura lokacija obuhvata: veb servere, aplikacione servere i DB servere.

Portalski veb server hostuje veb portal. Aplikacioni server hostuje ostale komponente sistema. Portalski DB server je deo sistema, na kome se nalaze Registar WS, Tezaurus i Data set sa rezultatima upita korisnika.

Na drugim veb serverima hostuju se WSm ustanova kulture. Lokacijski mogu biti na samom portalu, ali i na bilo kojoj drugoj lokaciji na Internetu.

Posebna grupa DB servera pripada ustanovama kulture i na njima se nalaze baze podataka o njihovim KD. DB serveri ustanova kulture nalaze se van granica sistema.



Slika 5.20: Logički dijagram isporuke

Logički model isporuke, prikazan na slici 5.20, u potpunosti zadovoljava zahteve i funkcionalnost sistema i lokacijski nije uslovljen. Prvenstveno iz praktičnih razloga, zbog efikasnijeg administriranja sistema u celini, preporuka je da DB server s Tezaurusom, Registrom WS i Data setom, kao i veb server s registrom WSm budu na lokaciji portalskog veb servera.

5.3.4 Tehnološki model

SOFI metodološki okvir za dizajniranje tehnološkog modela, predviđa izradu sledećih artifakata:

- dijagram klasa;
- specifikacija servisa;
- dijagram komponenti;
- implementacione tehnologije;
- UI ekrani;
- ER dijagram ili dijagram klasa - fizički model podataka;
- dijagram isporuke.

Deo WSDL dokumenta, kreiran za WSport servis, prikazan je na slici 5.21.

```
▼<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tm=
 xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:mime="http://s
 xmlns:s="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://schemas.xmlsoa
 xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" targetNamespace="http://tempuri.
 ▼<wsdl:types>
  ▼<s:schema elementFormDefault="qualified" targetNamespace="http://tempuri.or
   ▶<s:element name="PronadjiWS">...</s:element>
   ▶<s:element name="PronadjiWSResponse">...</s:element>
   ▶<s:element name="OrkestrirajWS">...</s:element>
   ▶<s:element name="OrkestrirajWSResponse">...</s:element>
   ▶<s:element name="UpisiPodatkeUKD">...</s:element>
   ▶<s:element name="UpisiPodatkeUKDResponse">...</s:element>
   </s:schema>
  </wsdl:types>
 ▶<wsdl:message name="PronadjiWSSoapIn">...</wsdl:message>
 ▶<wsdl:message name="PronadjiWSSoapOut">...</wsdl:message>
 ▶<wsdl:message name="OrkestrirajWSSoapIn">...</wsdl:message>
 ▶<wsdl:message name="OrkestrirajWSSoapOut">...</wsdl:message>
 ▶<wsdl:message name="UpisiPodatkeUKDSoapIn">...</wsdl:message>
 ▶<wsdl:message name="UpisiPodatkeUKDSoapOut">...</wsdl:message>
 ▼<wsdl:portType name="WSportSoap">
  ▶<wsdl:operation name="PronadjiWS">...</wsdl:operation>
  ▶<wsdl:operation name="OrkestrirajWS">...</wsdl:operation>
  ▶<wsdl:operation name="UpisiPodatkeUKD">...</wsdl:operation>
  </wsdl:portType>
 ▶<wsdl:binding name="WSportSoap" type="tns:WSportSoap">...</wsdl:binding>
 ▶<wsdl:binding name="WSportSoap12" type="tns:WSportSoap">...</wsdl:binding>
 ▼<wsdl:service name="WSport">
  ▶<wsdl:port name="WSportSoap" binding="tns:WSportSoap">...</wsdl:port>
  ▶<wsdl:port name="WSportSoap12" binding="tns:WSportSoap12">...</wsdl:port>
  </wsdl:service>
 </wsdl:definitions>
```

Slika 5.21. Deo WSDL dokumenta za WSport servis

Tehnološki modeli uslovljeni su implementacionim tehnologijama, koje su odabrane za realizaciju sistema. U ovom poglavlju, zbog specifične uloge u sistemu, prikazana je specifikacija servisa. Ostali modeli nisu prikazani, s obzirom da namenjeni inženjerima i nisu od značaja za prikaz sistema.

S obzirom da specifikacija servisa treba da definiše ugovor veb servisa, odnosno opiše elemente i odgovarajuće tipove podataka, kao i način interakcije sa klijentima, kreirani su WSDL dokumenti za identifikovane servise u prethodnim fazama dizajniranja sistema.

5.3.5 Diskusija implementacije metodološkog okvira

U prethodnim poglavljima ove teze obrađeni su interoperabilni, servisno orijentisani sistemi elektronskog poslovanja, a zatim je predložen metodološki okvir, koji bi trebalo da podrži razvoj konkretnog softverskog sistema.

U ovom poglavlju opisana je verifikacija razvijenog SOFI metodološkog okvira za uspostavljanje interoperabilnosti servisno orijentisanih sistema. Verifikacija je sprovedena tokom modelovanja veb portala, namenjenog za prezentaciju i pretraživanje različitih vrsta KD u Srbiji. Karakteristike izabranog sistema, za koji je rađena verifikacija SOFI okvira, zadovoljile su osnovne preduslove, potrebne za verifikaciju interoperabilnosti i primenu SOA arhitekture:

- Sistem ustanova kulture za zaštitu KD je distribuiran i ne može se direktno pristupiti bazama podataka o KD, koje su vlasništvo različitih, nadležnih ustanova kulture.
- Tokom pretraživanja podataka o KD, bazama podataka ustanova kulture pristupa se isključivo posredstvom veb servisa, koje kreiraju vlasnici baza podataka.
- Nakon analize i formalizovanja upita korisnika veb portala, ne pretražuju se sve raspoložive baze podataka o KD, već samo baze podataka čiji, unapred zadati, semantički sadržaji zadovoljavaju postavljeni upit.
- Kriterijumi za semantička označavanja definisani su jednoznačno i nalaze se u domenskom tezaurusu, koji je deo veb portala.

Softverski sistem za pretraživanje i prezentaciju različitih vrsta KD u Srbiji do sada nije rađen, pa je reč o dizajnu potpuno novog sistema. Samim tim, proces modelovanja odvijao se u smeru od "vrha prema dnu", a saglasno usvojenoj metodologiji. Tokom izrade sistema striktno je praćen redosled popunjavanja ćelija SOFI okvira, a prema šemi prikazanoj na slici 4.6.

Verifikacija SOFI okvira, tokom izrade sistema, trebalo je, između ostalog, da odgovori na sledeća pitanja:

- Da li predloženi redosled popunjavanja ćelija SOFI okvira omogućava kreiranje artifakata na način, koji obezbeđuje njihovo logičko i hijerarhijsko povezivanje?
- Da li se za potrebe kreiranja kolekcija artifakata, kojima se popunjavaju ćelije SOFI okvira, može koristiti generički meta model?
- Da li su za artifakt, koji se kreira, obezbeđene sve potrebne informacije, koje bi trebalo da preuzme od prethodno kreiranih artifakata?
- Da li artifakti, kojima je popunjena određena ćelija SOFI okvira, zadovoljavaju kriterijum granularnosti?

- Da li artifakti jedne ćelije SOFI okvira, ispunjavaju uslov apstrakcije i prostorne ograničenosti na samo tu, konkretnu ćeliju?

Tokom planiranja sistema, primenjena su pravila za popunjavanja SOFI okvira. Koristeći generički meta model, kreirane su dve osnovne vrste dokumenata i modela, kojima su definisani sistem i zahtevi sistema i opisani su: poslovni model; model sistema i tehnološki model.

- Prva grupa dokumenta i modela opisuje prirodu poslovnog sistema i zahteve postavljene softverskom sistemu. Urađeni artifakti smešteni su u prva dva reda SOFI okvira i najvećim delom obuhvataju rezultate analize i specificirane zahteve budućeg softverskog rešenja.
- Druga grupa dokumenata i modela obuhvata detaljne modele sistema, namenjene pre svega IT diveloperima.

Tokom verifikacije SOFI okvira pokazalo se da su artifakti, sadržani u ćelijama prvog reda (opis sistema i zahtevi), neophodni i izuzetno korisni za realizaciju svih narednih aktivnosti planiranja razvoja sistema. Dobar opis sistema, uz jasne i usaglašene zahteve, omogućio je pravolinijski razvoj, bez ponavljanja prethodnih aktivnosti i potrebe za naknadnim usaglašavanjima poslovnih analitičara i sistem analitičara.

Verifikacija razvijenog metodološkog okvira urađena je na primeru semantički izuzetno bogatog sistema za pretraživanje različitih vrsta KD u Srbiji. Priroda sistema omogućila je da se provere mogućnost korišćenja generičkog meta modela za razvoj kolekcija modela, kojima su popunjavane ćelije SOFI okvira. Ispostavilo se da je izuzetno koristan u svim fazama planiranja sistema: počev od definisanja zahteva, pa do izrade tehnološkog modela. Primenjeni generički meta model: obezbedio je konzistentan pogled na sistem, preko šeme za semantičke podatke i podržao je izradu modela različitih nivoa apstrakcije i različite poglede na sistem. Posebno značajan rezultat primene generičkog meta modela je model domenskog tezaurusa, koji učestvuje u uspostavljanju interoperabilnosti preko semantički opisanih veb servisa. Ovako opisani veb servisi omogućavaju ad hoc otkrivanje i selektivno pozivanje veb servisa isključivo na osnovu sadržaja upita korisnika.

Na osnovu analize primene SOFI okvira, može se konstatovati da su tehničke specifikacije SOFI okvira proverene i da su mogućnosti primene verifikovane na primeru. Primenjeni SOFI okvir ispunio je svoj zadatak u domenu definisanja i provere kompletnosti potrebne dokumentacije i modela, potrebnih za planiranje i razvoj konkretnog sistema zasnovanog na interoperabilnosti i servisnoj orijentaciji.

6 Naučni i stručni doprinosi

Tokom istraživanja, predstavljenih u ovoj tezi, razmatrane su oblasti: servisno orijentisana arhitektura i interoperabilnost sistema elektronskog poslovanja. Pregledom dostupne literature, potvrđena je aktuelnost i interesovanje za rešavanje uočenih problema. Za postizanje zajedničkog cilja: uspostavljanje interoperabilnosti servisno orijentisanih sistema elektronskog poslovanja, istraživači predlažu kreiranje okvira, koji se mogu svrstati u nekoliko grupa. Pri tome, prezentuju svoja viđenja modelovanja poslovnih procesa i primenu SOA. Imajući u vidu trenutno stanje u navedenim, glavnim oblastima istraživanja, autor ove teze predložio je novi okvir, prilagođen i namenjen SOA, kao i generički meta model, koji predstavlja osnovu za kreiranje i praćenje artifakata, potrebnih za celovito i konzistentno planiranje sistema elektronskog poslovanja.

Na osnovu rezultata verifikacije predloženog okvira i generičkog meta modela, može se zaključiti da istraživanja, predstavljena u ovoj tezi, doprinose rešavanju pojedinih problema u oblasti interoperabilnosti servisno orijentisanog sistema elektronskog poslovanja. Osnovni doprinosi istraživanja, u okviru teze, obuhvataju:

- Razvoj metodološkog okvira za povezivanje na ad-hoc principu u sistemima elektronskog poslovanja.
- Razvoj generičkog meta modela za obezbeđenje semantičke konzistentnosti sistema i opisivanje i otkrivanje komponenti u sistemima elektronskog poslovanja, zasnovanih na servisno orijentisanoj arhitekturi.
- Predlog arhitekture za semantičku interoperabilnost poslovnih procesa na ad-hoc principu.
- Verifikaciju razvijenog okvira, na primeru poslovnog sistema ustanova kulture, a u cilju dinamičkog izvršavanja procesa za pretraživanje i prezentovanje informacija, dokumenata i ilustrativnih materijala o različitim vrstama kulturnih dobara.

Slede opis i obrazloženje navedenih doprinosa.

- **Metodološki okvir za povezivanje na ad-hoc principu u sistemima elektronskog poslovanja**

  U tezi je prikazan razvoj metodološkog okvira za dizajniranje sistema elektronskog poslovanja, koji omogućava primenu servisno orijentisane arhitekture i obezbeđuje povezivanje sa drugim sistemima na ad-hoc principu. Kreirani okvir izveden je iz Zahman okvira, popunjavanjem ćelija artifaktima, u zavisnosti od faza životnog ciklusa razvoja softverskih sistema. Popunjavanje ćelija Zahman okvira vršeno je prema posebno razvijenom algoritmu. Dobijeni, delimično popunjeni Zahman okvir, transformisan je u novi, SOFI okvir. Preporučena metodologija opisala je i definisala:

1. Način korišćenja izvedenog okvira;
2. Izbor odgovarajućih jezika za kreiranje kolekcija artifakata unutar ćelija okvira, a u zavisnosti od pogleda na sistem i potrebne detaljnosti modela;
3. Uspostavljanje i proveru veza između kreiranih modela;
4. Grupisanje ćelija matrice za određivanje granica sistema, u odnosu na različite poglede zainteresovanih pri planiranju sistema;
5. Korišćenje tehnika za proveru validnosti kreiranih modela.

Izvedeni okvir ne propisuje broj artifakata, koje svaka ćelija treba da sadrži, već preporučuje njihove moguće tipove, a implementacija tipova artifakata usaglašava se sa potrebama sistema elektronskog poslovanja, koji se planira. Sa ciljem da se olakša i omogući dosledna primena izvedenog okvira, predložen je metod za njegovo korišćenje, koji podržava sve korake za izgradnju rešenja: od poslovnih zahteva, do realizacije softvera. Za modelovanje poslovnih procesa i predstavljanje logički povezanih aktivnosti preporučuje se izrada BPMN dijagrama, koji omogućavaju formalizovano izdvajanje aktivnosti poslovnog procesa, koji će biti realizovani posredstvom servisa. Da bi se zadovoljili kritični preduslovi za uspostavljanje interoperabilnosti, preporučeno je korišćenje veb servisa.

Planiranje sistema elektronskog poslovanja uključuje izradu modela različitih nivoa detaljnosti i analizu potreba i pogleda na sistem iz različitih uglova. Istraživanja, izvršena tokom izrade ove teze, ukazala su na probleme povezivanja razvijenih modela i obezbeđenje njihove konzistentnosti. Nakon razmatranja mogućih rešenja, razvijen je metodološki okvir za povezivanje na ad-hoc principu u sistemima elektronskog poslovanja. Preporučena metodologija obezbeđuje konzistentno planiranje sistema i uključuje: namenski metodološki okvir; metod za primenu izvedenog okvira; primenu BPMN dijagrama za predstavljanje poslovnih procesa; korišćenje BPMN dijagrama za formalizovano izdvajanje poslovnih aktivnosti, koje će se realizovati korišćenjem veb servisa, čime će preduslovi za uspostavljanje interoperabilnosti biti zadovoljeni.

- **Generički meta model za obezbeđenje semantičke konzistentnosti sistema i opisivanje i otkrivanje komponenti u sistemima elektronskog poslovanja, zasnovanih na servisno orijentisanoj arhitekturi**

Razvijeni metodološki okvir preporučuje tipove artifakata za popunjavanje ćelija okvira. Uključeni modeli formalno obezbeđuju kontinuitet i kontrolu procesa planiranja sistema, u odnosu na različite nivoe detaljnosti modela i različite poglede zainteresovanih za planiranje sistema. Sama činjenica da su, saglasno metodološkom okviru, svi potrebni modeli urađeni, nije dovoljna. Imajući to u vidu, tokom izrade ove teze, deo istraživanja posvećen je rešavanju problema međusobne, sadržajne usklađenosti uključenih modela.

Proces planiranja sistema započinje izradom modela, koji se zasnivaju na govornom jeziku, nastavlja se izradom modela, koji koriste grafičke prikaze i prepoznatljive govorne termine. Najdetaljniji modeli namenjeni su specijalistima IT i prikazani su dijagramima, simbolima i najčešće stručnim terminima. Istražujući moguća rešenja za obezbeđenje sadržajne usklađenosti različitih modela, autor ove teze predložio je generički meta model, koji jedinstvenom, domenskom semantikom podržava izradu svih artifakata za popunjavanje kreiranog okvira. Razvoj generičkog meta modela obavljen je proširenjem i transformacijom trougla znanja i

zasnovan je na domenskim ontologijama, poslovnim terminima i poslovnim konceptima. Uključenjem semantike, predloženi generički meta model može se koristiti:

- kao šema za semantičke podatke da obezbedi semantičku konzistentnost modela;
- kao model za kreiranje artifakata, koji se mogu opisati konzistentnom kolekcijom simbola, čime se obezbeđuje jedinstvena osnova za međusobnu komunikaciju učesnika;
- da podrži dodatno semantičko opisivanje potrebnih veb servisa, ili već postojećih modela;
- da omogući semantičku interoperabilnost sistema elektronskog poslovanja, uključenjem semantički opisanih veb servisa.

Zaključak je da se istovremenom primenom razvijenog metodološkog okvira i izvedenog generičkog meta modela, tokom procesa planiranja sistema elektronskog poslovanja, zasnovanih na servisno orijentisanoj arhitekturi, može obezbediti formalna i sadržajna konzistentnost modela, potrebnih za celovit razvoj sistema i stvoriti preduslov za semantičku interoperabilnost sistema.

- **Arhitektura za semantičku interoperabilnost poslovnih procesa na ad-hoc principu**

Planiranje sistema započinje definisanjem zahteva i kreiranjem određenog broja scenarija. Pri kreiranju modela sistema uključuju se opisane, postojeće potrebe preduzeća ali, u okviru mogućnosti predviđanja, i budući zahtevi sistema i moguće promene okruženja. Osnovni zadatak modela arhitekture sistema je da objedini i organizuje kreirane kolekcije artifakata. Osim realizacije sistema, dobro urađen model arhitekture sistema trebalo bi da omogući prilagodljivost budućeg softverskog sistema promenama u okruženju i samom sistemu.

U ovoj tezi predložen je meta model arhitekture za semantičku interoperabilnost, koji organizuje delove arhitekture sistema, potrebne za obavljanje poslovnih funkcija i uspostavlja relacije između uključenih delova sistema. Predložena arhitektura povezuje poslovne procese sa servisima, preko kojih se realizuju pojedinačne poslovne aktivnosti. Poslovna pravila, ugrađena u servise, omogućavaju realizaciju poslovnih procesa kompozitno, preko koreografije, ili preko orkestracije servisa. Tokom daljeg istraživanja, razmatrana je dopuna standardnih registara semantičkim opisom registrovanih servisa, kao i njihovo otkrivanje na osnovu tog opisa. Razmatrajući moguća rešenja, autor ove teze opredelio se za semantičko proširenje osnovne SOA, koje je uključeno u meta model arhitekture sistema. Semantičko proširenje nudi mogućnost da se registar servisa i servisi konzistentno opišu istom semantikom. Rezultat je da semantička struktura predloženog meta modela arhitekture sistema omogućava semantičko povezivanje servisno orijentisanih sistema. S obzirom da prirodno podržavaju slabo povezivanje, ugovore, apstrakciju i mogućnost kompozicije, za realizaciju servisno orijentisano orijentisane arhitekture sistema odabrani su veb servisi.

Daljom analizom predloženog meta modela arhitekture sistema, ustanovljeno je da su obezbeđeni preduslovi u pogledu adaptivnosti i naknadnih proširenja sistema. Posmatrano iz ugla ove teze, pojam "adaptivnost" ne podrazumeva mogućnost budućeg softverskog sistema da samostalno prepozna i da se prilagodi svim mogućim promenama okruženja. Cilj je ograničen na promene u samom sistemu i pojedine promene u okruženju. Konkretno, zahvaljujući semantičkom proširenju, budući softverski sistem adaptivan je na moguće česte semantičke

promene, inicirane u okruženju, čime je očuvana osobina semantičke interoperabilnosti. U pogledu samog sistema, adaptivnost se ispoljava u slučajevima naknadnih proširenja sistema. Naime, u slučaju proširenja registra, nakon upisa novih servisa, sistem će sačuvati osobinu semantičke interoperabilnosti poslovnih procesa, na ad-hoc principu, ravnopravno otkrivajući i "stare" i "nove" veb servise.

Baveći se problemima za uspostavljanje semantičke interoperabilnosti poslovnih procesa, na ad-hoc principu, autor ove teze predložio je meta model arhitekture sistema, zasnovan na povezivanju SOA i poslovnih procesa i na konzistentnom, semantičkom proširenju registra i servisa. Verifikacija meta modela arhitekture izvršena je na primeru konkretnog sistema elektronskog poslovanja. Pokazalo se da semantičko proširenje, zasnovano na domenskom tezaurusu, omogućava: konzistentno semantičko proširenje registra i servisa; ad-hoc otkrivanje i kompoziciju potrebnih veb servisa i konačno, uspostavljanje interoperabilnosti isključivo sa sistemima, koji zadovoljavaju semantiku upita korisnika sistema.

Zaključak autora ove teze je da se predloženi meta model arhitekture sistema, zasnovan na povezivanju poslovnih procesa i SOA i semantičkom proširenju registra i servisa, može uspešno primeniti za uspostavljanje ad-hoc semantičke interoperabilnosti, određenih klasa sistema elektronskog poslovanja. Opseg primenljivosti uslovljen je isključivo vrstom domenske ontologije, kojom se opisuju razmatrani, realni sistemi.

- **Verifikacija razvijenog okvira, na primeru poslovnog sistema ustanova kulture, a u cilju dinamičkog izvršavanja procesa za pretraživanje i prezentovanje informacija, dokumenata i ilustrativnih materijala o različitim vrstama kulturnih dobara**

Verifikacija razvijenog metodološkog okvira i izvedenog generičkog meta modela urađena je tokom modelovanja interoperabilnog, servisno orijentisanog sistema elektronskog poslovanja za pretraživanje i prezentovanje kulturnih dobara u Srbiji. Predloženo rešenje zasnovano je na veb portalu, čija je uloga da uspostavi vezu između korisnika sistema i distribuiranih baza podataka ustanova kulture.

Aktivnosti planiranja sistema inicirane su definisanjem zahteva, vizije i ciljeva, a nastavljene analizom profila budućih korisnika i funkcionalnosti, koje bi sistem trebalo da zadovolji. Na osnovu navedenih dokumenata kreiran je koncept rešenja, zasnovan na korišćenju domenskog tezaurusa i semantički opisanih veb servisa i registra.

Prateći metodološki okvir, planiranje sistema nastavljeno je izradom modela za popunjavanje ćelija okvira, sa ciljem da se opišu različiti pogledi na sistem i da se obezbede potrebne specifikacije za razvoj sistema. Popunjeni okvir obuhvatio je: slučajeve korišćenja; scenarija upotrebe; BPMN dijagram; skice korisničkog interfejsa; konceptualni model klasa i podataka i dijagram isporuke. Pri kreiranju pojedinačnih artifakata, konsultovan je generički meta model, koji je preuzeo kontrolnu funkciju za uspostavljanje sadržajne konzistentnosti uključenih modela. Razvijeni modeli definisali su elemente SOA i uspostavili veze SOA sa aktivnostima poslovnih procesa sistema. Uočene su sledeće vrste servisa: semantički, poslovni, UI i servisi podataka (registar, domenski tezaurus). Konačna arhitektura sistema obuhvatila je: navedene servise; izvore podataka (registar, domenski tezaurus) i orkestrator servisa. Organizovani elementi arhitekture omogućili su:

- integrisanje različitih izvora podataka, ad-hoc otkrivanjem, komponovanjem i pozivanjem veb servisa, čiji opisi odgovaraju semantički formalizovanom upitu korisnika (komunikator - analizator - domenski tezaurus - orkestrator);
- adaptivnost sistema u odnosu na semantičko proširenje, inicirano u okruženju sistema (domenski tezaurus);
- adaptivnost sistema u odnosu na proširenje domenskog tezaurusa;
- otvorenost sistema za uključenje proizvoljnog broja ustanova iz poslovnog domena sistema (registrant - domenski tezaurus - registar).

Verifikacija metodološkog okvira izvršena je tokom procesa planiranja sistema za pretraživanje i prezentovanje kulturnih dobara u Srbiji i opisana je u poglavlju 5. ove teze. Predloženi metodološki okvir, u saradnji sa generičkim meta modelom, omogućio je izradu, formalno i sadržajno konzistentnih modela, za kreiranje arhitekture sistema. Izvedena servisno orijentisana arhitektura sistema, zasnovana na domenskom tezaurusu, omogućava uspostavljanje semantičke interoperabilnosti sa uključenim sistemima ustanova kulture. Istovremeno, sistem podržava: kreiranje opšteg mehanizma za registraciju veb servisa; ad-hoc otkrivanje, komponovanje i pozivanje na izvršenje servisa isključivo na osnovu upita korisnika i integrisanje podataka iz različitih baza podataka u jedinstveni skup podataka.

Zaključak autora ove teze je da se, pored primene u oblasti kulturnih dobara, razvijeni metodološki okvir i generički meta modeli, zasnovani na domenskim ontologijama, mogu uspešno primeniti za planiranje sistema elektronskog poslovanja i u drugim poslovnim domenima. Uspešnost primene, pored kvaliteta definisanih zahteva sistema i urađenih modela elemenata arhitekture sistema, značajno zavisi od strukture i sadržaja domenske ontologije, s obzirom da se na nju oslanjaju sve faze životnog ciklusa razvoja i korišćenja softverskog sistema.

Na osnovu istraživanja, predstavljenih u ovoj tezi, zaključeno je da je moguće unaprediti modelovanje interoperabilnosti sistema elektronskog poslovanja koristeći razvijeni metodološki okvir i razvijeni semantički orijentisan, generički meta model, koji obezbeđuju semantičku konzistentnost modela sistema i koji se, kombinovano, mogu koristiti za kreiranje opisa servisnih komponenti i njihovo dinamičko pozivanje preko aktivnosti poslovnog procesa.

7 Dalji pravci rada i istraživanja

Rezultati istraživanja, prikazani u ovoj tezi, pored već navedenih doprinosa, ukazuju da uspešna realizacija sistema elektronskog poslovanja zahteva obezbeđenje određenih preduslova i rešavanje uočenih problema. Na osnovu stečenog iskustva i izvedenih zaključaka, uočeni su mogući pravci daljeg rada i istraživanja.

Trenutno, semantičke tehnologije nisu dovoljno zastupljene u okvirima za elektronsko poslovanje. Prihvatanje semantičkih tehnologija, sa ciljem da se unapredi interoperabilnost, uslovljeno je dopunom važećih standarda i specifikacija. U tom smislu, trebalo bi i nadalje istraživati mogućnosti ugradnje semantike u specifikacije, koje se bave servisima i poslovnim procesima.

Uspešnost primene semantičke interoperabilnosti značajno zavisi od strukture i sadržaja ontologija. Zbog toga je potrebno uložiti napor za proširenje postojećih i kreiranje novih ontologija, pre svega u granicama pojedinačnih oblasti poslovanja, sa motivom da se u narednim fazama međusobno povežu.

Sada se čini da su opšti UDDI registri stvar prošlosti, jer se u praksi retko koriste: zbog svoje kompleksnosti; nepostojanja mogućnosti za semantički opis i pretraživanje servisa; nedostatka mehanizma za obezbeđenje sigurnosti i zbog ograničenja primene isključivo na SOAP veb servise. Autor ove teze smatra da postoji opravdanje da se oživi ideja registara servisa. U tom smislu, potrebno je istražiti mogućnosti razvoja specijalizovanih UDDI registara, za pojedinačne oblasti poslovanja, pri čemu će se koristiti odgovarajuće domenske ontologije. Razvoj specijalizovanih UDDI registara dobra je prilika da se pojednostave kompleksne strukture postojećih UDDI registara i da se uključe potrebne funkcije sigurnosti.

Jedan od pravaca istraživanja je proširenje metodološkog okvira, koji bi trebalo dopuniti metodama i tehnikama za modelovanje poslovanja (primeri su: e^3-value, REA, BMO i IDEF0). U tom slučaju, potrebno je i istraživati mogućnosti povezivanja modela poslovanja sa modelima poslovnih procesa, čime bi se otklonio diskontinuitet, koji postoji između njih. Uspešno rešenje problema, omogućilo bi vertikalno povezivanje svih nivoa pri kreiranju sistema elektronskog poslovanja.

8 Zaključak

U ovoj tezi prikazan je jedan pristup za modelovanje interoperabilnosti servisno orijentisanih sistema elektronskog poslovanja. Obavljeni istraživački rad usmeren je ka nekoliko oblasti, u okviru dva glavna koncepta: servisno orijentisana arhitektura i interoperabilnost sistema elektronskog poslovanja. Istraživanje je motivisano potrebom da se unapredi modelovanje interoperabilnosti servisno orijentisanih sistema elektronskog poslovanja i pretpostavkom da se unapređenje može postići razvojem metodološkog okvira i generičkog meta modela. Konačno rešenje, autor ove teze predložio je analiziranjem i kombinovanjem više poznatih metoda i tehnika, dodajući sopstveni, novi metodološki okvir i semantički generički meta model.

Pregledom tehnika i metoda za modelovanje sistema elektronskog poslovanja, koje se odnose na poslovni operacioni pogled i koje obezbeđuju definisanje modela poslovanja i modela poslovnog procesa, a koristeći BPMN dijagram za praćenje pojedinačnih poslovnih aktivnosti, identifikovani su kandidati za servise. Model arhitekture sistema izveden je objedinjenim modelovanjem i uspostavljanjem veze između standardne SOA i poslovnih procesa, a za realizaciju servisno orijentisanih sistema preporučeni su semantički opisani veb servisi. Predložena arhitektura sistema, uz primenu jedinstvene semantičke osnove, zasnovane na domenskim ontologijama, omogućava jednoznačno praćenje i proveru konzistentnosti procesa modelovanja i uspostavljanje semantičke interoperabilnosti slabo povezanih sistema.

Završna analiza primene predloženog modela arhitekture sistema, okvira za interoperabilnost i generičkog meta modela pokazala je da je moguće planski i sistematski usaglasiti poglede poslovnih analitičara i analitičara sistema. Imajući u vidu da je, tokom planiranja konkretnog sistema, omogućeno modelovanje različitih pogleda na sistem i kreiranje dokumenata i modela različite detaljnosti i namene, zaključeno je da su mogućnosti i opravdanost primene predložene arhitekture sistema, okvira i generičkog meta modela verifikovane.

Stav autora ove teze je da izbor metodologija za planiranje sistema elektronskog poslovanja, pa i ostalih softverskih sistema, ne treba zasnivati isključivo na trendovima i atraktivnosti nekih od njih. Naprotiv, "prava metodologija" je metodologija prilagođena prirodi klase sistema koji se modeluju, metodologija koja će obezbediti potrebne modele za tačnu i efikasnu realizaciju softverskog sistema, primenom aktuelnih i primerenih, a dostupnih tehnologija. Navedeni stav proistekao je iz obavljenih istraživanja obrazloženih u ovoj tezi.

Literatura

[1] E. Kasanen, L. Kari, and A. Siitonen, "The Constructive Approach in Management Accounting Research," *Journal of Management Accounting Research*, vol. 5, pp. 243-264, 1993.

[2] S.T. March and G.F. Smith, "Design and Natural Science Research on Information Technology," *Decision Support Systems*, vol. 15, no. 4, pp. 251-266, Dec 1995.

[3] L. Gerstner, *Who Says Elephants Can't Dance? How I Turned Around IBM*. New York: Harper Business, 2002.

[4] E. Turban, D. King, D. Viehland, and J. Lee, *Electronic Commerce 2008: A Managerial Perspective*. Upper Saddle River, New Jersey: Prentice-Hall, 2008.

[5] N. Melao, "e-Business Processes and e-Business Process Modelling: A State-of-Art Overview," *International Journal of Services Technology and Management*, vol. 11, no. 3, pp. 293-322, 2009.

[6] O. El Sawy, *Redesigning Enterprise Processes for e-Business*. Boston: McGraw-Hill, 2001.

[7] M. Earl, "Evolving the E-Business," *Business Strategy Review*, vol. 11, no. 2, pp. 33-38., 2000.

[8] M. Henning, "The rise and fall of CORBA," *ACM Queue*, vol. 4, no. 5, pp. 28-34, 2006.

[9] E. Turban and J. E. Aronson, *Decision Support Systems and Intelligent Systems*. Upper Saddle River, New Jersey: Prentice Hall, 2001.

[10] Wikipedia. [Online]. http://en.wikipedia.org/wiki/Enterprise_architecture

[11] J. Gordijn and H. Akkermans, "Designing and Evaluating E-Business Models," *IEEE Intelligent Systems*, vol. 16, no. 4, pp. 11-17, 2001.

[12] J. Dorn, C. Grun, H. Werthner, and M. Zapletal, "A Survey of B2B Methodologies and Technologies: From Business Models towards Deployment Artifacts," *Information Systems and E-Business Management*, vol. 7, no. 2, pp. 123–142, 2009.

[13] H. Weigand et al., "Value Object Analysis and the Transformation from Value Model to Process Model," in *2nd International Conference on Interoperability of Enterprise Software and Applications*, Bordeaux, France, 2006, pp. 55-56.

[14] W.E. McCarthy, "The REA Accounting Model: A Generalized Framework for Accounting Systems in a Shared Data Environment," *The Accounting Review*, pp. 554-578, 1982.

[15] A. Osterwalder and Y. Pigneur, "An e-Business Model Ontology for Modeling e-Business," in *15th Bled Electronic Commerce Conference e-Reality: Constructing the e-Economy*, Bled, Slovenia, 2002.

[16] "INTEGRATION DEFINITION FOR FUNCTION MODELING (IDEF0)," NIST, Federal Information Processing Standards Publications (FIPS PUBS) 1993.

[17] J. Gordijn, A. Osterwalder, and Y. Pigneur, "Comparing two Business Model Ontologies for Designing e-Business Models and Value Constellations," in *18th Bled eConference eIntegration in Action*, Bled, Slovenia, 2005.

[18] B. Andersson et al., "Towards a Common Ontology for Business Models," Interop project in EU 6th FP 2006.

[19] C. Huemer, A. Schmidt, H. Werthner, and M. Zapletal, "A UML Profile for the e3-Value e-Business Model Ontology," in *Proceedings of the Third International Workshop on Business/IT Alignment and Interoperability (BUSITAL'08) held in conjunction with CAiSE'08 Conference*, Montpellier, France, 2008.

[20] J. Gordijn, H. Akkermans, and H. van Vliet, "Business Modelling is not Process Modelling," in *Workshops on Conceptual Modeling Approaches for E-Business and The World Wide Web and Conceptual Modeling: Conceptual Modeling for E-Business and the Web*, 2000, pp. 40-51.

[21] T.H. Davenport and J.E. Short, "The New Industrial Engineering: Information Technology and Business Process Redesign," *Sloan Management Review*, vol. 31, no. 4, Jul 1990.

[22] P. Dalmaris, E. Tsui, B. Hall, and B. Smith, "A framework for the improvement of knowledge-intensive business processes," *Business Process Management Journal*, vol. 13, no. 2, pp. 279 - 305, 2007.

[23] H. M Kim and R. Ramkaran, "Best practices in e-business process management Extending a re-engineering framework," *Business Process Management Journal*, vol. 10, no. 1, pp. 27-43., 2004.

[24] S. Haberl. (2003) University of South Australia, School of Computer and Information Science. [Online]. http://www.cis.unisa.edu.au/~cissh/research/webflow/bpdl.html

[25] P. Kueng, P. Kawalek, and P. Bichler, "How to compose an object-oriented business process model?," in *IFIP WG8.1/WG8.2 Working Conference*, Atlanta, GA, 1996.

[26] M. Havey, *Essential Business Process Modeling*. Sebastopol, CA: O'Reilly Media, 2005.

[27] J. Mendling, H. Reijers, and W. M. van der Aalst. (2008) Queensland University of Technology - Digital Repository. [Online]. http://eprints.qut.edu.au/12340/1/12340.pdf

[28] S. A. White. (2004) OMG. [Online]. http://www.omg.org/bpmn/Documents/Introduction_to_BPMN.pdf

[29] (2006, Mar) OMG. [Online]. http://www.bpmn.org/Documents/OMG_Final_Adopted_BPMN_1-0_Spec_06-02-01.pdf

[30] M. zur Muehlen. (2007) Queensland University of Technology - BPM 2007 Conference. [Online]. http://bpm07.fit.qut.edu.au/program/slides/Thursday/Thursday-Tutorials/Muehlen.pdf

[31] (2003) IBM developer Works. [Online]. http://www.ibm.com/developerworks/library/specification/ws-bpel/

[32] M. Juric, B. Mathew, and P. Sarang, *Business Process Execution Language for Web Services*, 2nd ed. Birmingham, UK: Packt Publishing, 2006.

[33] D. Chappell. (2005) Opinari from David Chappell. [Online]. http://www.davidchappell.com/html_email/opinari_no14_10_05.html

[34] B. Silver. (2005) BPMInstitute. [Online]. http://www.bpminstitute.org/articles/article/article/bpms-watch-bpm-and-soa-one-technology-two-communities.html

[35] P. Wohed, W.M.P. van der Aalst, M. Dumas, A.H.M. Hofstede, and N. Russell. (2005) Queensland University of Technology - Digital Repository. [Online]. http://eprints.qut.edu.au/2977/1/2977.pdf

[36] A. Tsai, "EPC Workflow Model to WIFA Model Conversion," in *IEEE International Conference on Systems, Man, and Cybernetics*, Taipei, Taiwan, 2006, pp. 2758-2763.

[37] (2006) UMM Development Site. [Online]. http://umm-dev.org/umm-specification/

[38] N. Russell, W. M van der Aalst, A.H.M. Hofstede, and P. Wohed, "On the suitability of UML 2.0 activity diagrams for business process modelling," in *Proceedings of the 3rd Asia-Pacific Conference on Conceptual Modelling*, Hobart, Australia, 2006.

[39] C. Emig, J. Weisser, and S. Abeck, "Development of SOA-Based Software Systems – an Evolutionary Programming Approach," in *AICT/ICIW*, Guadeloupe, French Caribbean, 2006.

[40] J.C. Recker, M. zur Muehlen, K. Siau, J. Erickson, and M. Indulska, "Measuring method complexity : UML versus BPMN," in *15th Americas Conference on Information Systems*, San Francisco, California, 2009.

[41] T. Erl, *Service-Oriented Architecture: Concepts, Technology, and Design*. Upper Saddle River, NJ: Pearson Education, 2005.

[42] B. Lublinsky, "SOA Design: Meeting in the Middle," *JavaPro*, no. Special Issue, 2004.

[43] D. Krafzig, K. Banke, and D. Slama, *Enterprise SOA: Service-Oriented Architecture Best Practices*. Upper Saddle River, NJ.: Prentice Hall, 2004.

[44] J. Bloomberg and R. Schmelzer, *Service Orient or Be Doomed! How Service Orientation Will Change Your Business*. Hoboken, New Jersey: John Wiley & Sons, Inc., 2006.

[45] N. Bieberstein, S. Bose, L. Walker, and A. Lynch, "Impact of service-oriented architecture on enterprise systems, organizational structures, and individuals," *IBM Systems Journal*, vol. 44,

no. 4, 2005.

[46] D. Box, "A Guide to Developing and Running Connected Systems with Indigo," *MSDN Magazine*, 2004.

[47] Y. V. Natis. (2003) Gartner Research. [Online]. http://www.gartner.com/DisplayDocument?doc_cd=114358

[48] T. Kohlborn, E. Fielt, A. Korthaus, and M. Rosemann, "Towards a service portfolio management framework, Evolving Boundaries and New Frontiers: Defining the IS Discipline," in *20th Australasian Conference on Information Systems*, Melbourne, 2009.

[49] Gartner Research. [Online]. http://www.gartner.com/technology/it-glossary/soa-service-oriented-architecture.jsp

[50] Wikipedia. [Online]. http://en.wikipedia.org/wiki/Service-oriented_architecture

[51] M. Rosen, *Applied SOA: Service-Oriented Architecture and Design Strategies*. Hoboken, New Jersey: Wiley Publishing, 2008.

[52] T. Kohlborn, A. Korthaus, T. Chan, and M. Rosemann, "Identification and Analysis of Business and Software Services - A Consolidated Approach," *IEEE Transactions on Services Computing*, vol. 2, no. 1, pp. 50 - 64, 2009.

[53] D. Fensel and C. Bussler, "The web service modeling framework WSMF," *Electronic Commerce Research and Applications*, vol. 1, pp. 113-137, 2002.

[54] (2004) World Wide Web Consortium. [Online]. http://www.w3.org/TR/ws-arch/

[55] *Patterns: Service-Oriented Architecture and Web Services*, 1st ed.: IBM Redbooks, 2004.

[56] V. Tsetsos, C. Anagnostopoulos, and S. Hadjiefthymiades, "Semantic Web Service Discovery: Methods, Algorithms, and Tools," in *Semantic Web Services: Theory, Tools and Applications*, J. Cardoso, Ed.: IDEA Group Publishing, 2007, ch. 11, pp. 240-280.

[57] G. Kapitsaki et al., "Service composition: State of the art and future challenges," in *IST Mobile and Wireless Communications Summit*, Budapest, 2007, pp. 1-5.

[58] S. Dustdar and W. Schreiner, "A Survey On Web Services Composition," *International Journal of Web and Grid Services*, vol. 1, no. 1, 2005.

[59] G. Alonso, F. Casati, H. Kuno, and V. Machiraju, *Web Services. Concepts, Architectures and Application*. Berlin Heidelberg: Springer-Verlag, 2004.

[60] M.P. Papazoglou, P. Traverso, S. Dustdar, and F. Leymann, "Service-Oriented Computing: a Research Roadmap," *International Journal of Cooperative Information Systems*, vol. 17, no. 2, pp. 223–255, 2008.

[61] "IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossarie," Institute of Electrical and Electronics Engineers, New York, 1990.

[62] L.S. Winters, M.M. Gorman, and A. Tolk, "Next Generation Data Interoperability: It's all About the Metadata," in *Fall Simulation Interoperability Workshop*, Orlando, 2006.

[63] F.B. Vernadat, "Technical, semantic and organizational issues of enterprise interoperability and networking," *Annual Reviews in Control*, vol. 34, no. 1, pp. 139-144, 2010.

[64] (2009) Open Group. [Online]. http://pubs.opengroup.org/architecture/togaf9-doc/arch/

[65] "Industrial Automation Systems – Concepts and Rules for Enterprise Models," International Organization for Standardization, ISO 14258, 1999.

[66] D. Chen, G. Doumeingts, and F. Vernadat, "Architectures for enterprise integration and interoperability: Past, present and future," *Computers in Industry*, vol. 59, no. 7, pp. 647-659, 2008.

[67] (2009) Semantic Interoperability Centre Europe. [Online]. https://joinup.ec.europa.eu/system/files/doc/guidelines_studies/semic-eu-study-on-methodology-v1.2.pdf

[68] S. Pokraev, D. Quartel, M. Steen, and M. Reichert, "Semantic service modeling: Enabling system interoperability," in *International Conference on Interoperability for Enterprise Software and Applications (I-ESA'06)*, Bordeaux, France, 2006, pp. 221–231.

[69] C.H. Asuncion and M.J. Sinderen van, "Pragmatic Interoperability: A Systematic Review of Published Definitions," in *IFIP TC5 International Conference on Enterprise Architecture, Integration and Interoperability, EAI2N 2010*, Brisbane, Australia, 2010.

[70] "European Interoperability Framework for Pan-European eGovernment Services," European Commission, Luxembourg, ISBN 92-894-8389-X, 2004.

[71] V. Peristeras and K. Tarabanis, "The Connection, Communication, Consolidation, Collaboration Interoperability Framework (C4IF) For Information Systems Interoperability," *IBIS – Interoperability in Business Information Systems*, vol. 1, no. 1, pp. 61-72, 2006.

[72] D. Chen, B. Vallespir, and N. Daclin, "An Approach for Enterprise Interoperability Measurement," in *International Workshop on Model Driven Information Systems Engineering: Enterprise, User and System Models (MoDISE-EUS'08)*, Montpellier, France, 2008.

[73] ETSI. [Online]. http://www.etsi.org/WebSite/Standards/WhatIsAStandard.aspx

[74] A.M. Fitzgerald and K.M. Pappalardo, "Moving towards open standards," *SCRIPTed - A Journal of Law, Technology & Society*, vol. 6, no. 2, pp. 467-483, 2009.

[75] K.D. Naujok. (2009) illumonus. [Online]. http://www.illumonus.com/business-modeling-methodology

[76] S. Kemsley, "The Role of Standards in BPM," in *5th International Conference on Business Process Management*, Brisbane, Australia, 2007.

[77] M. Zelm and K. Kosanke, "Standardisation in Interoperability," in *IMS Workshop*, Zurich, 2007.

[78] Gartner Research. [Online]. http://www.gartner.com/technology/it-glossary/enterprise-architecture.jsp

[79] "Recommended Practice for Architecture Description of Software-Intensive Systems," IEEE Standards Association, ANSI/IEEE 1471, 2000.

[80] MIT SLOAN Center for Information Systems Research. [Online]. http://cisr.mit.edu/research/research-overview/classic-topics/enterprise-architecture/

[81] J. Schekkerman, *How to Survive in the Jungle of Enterprise Architecture Frameworks*, 3rd ed. Bloomington: Trafford Publishing, 2006.

[82] J. Zachman. (1997) Concepts of Framework for Enterprise Architecture: Background, Description and Utility. [Online]. http://www.ies.aust.com/PDF-papers/zachman3.pdf

[83] A. Fatolahi and F. Shams, "An investigation into applying UML to the Zachman Framework," *nformation Systems Frontiers*, vol. 8, no. 2, 2006.

[84] I. Mamdouh and G. Long. (2007) developerWorks. [Online]. http://www.ibm.com/developerworks/webservices/library/ws-soa-enterprise2/

[85] C.T. Jensen et al. (2008) developerWorks. [Online]. http://www.ibm.com/developerworks/websphere/bpmjournal/0812_jensen/0812_jensen.html

[86] "Interoperability Research for Networked Enterprises Applications and Software, Network of Excellence," 2003.

[87] "ATHENA IS 507849, Advanced Technologies for Interoperability of Heterogeneous Enterprise Networks and their Applications," FP6-2002-IST-1, 2003.

[88] D. Hay, *DATA MODEL PATTERNS: A Metadata Map*. San Francisco, CA: Morgan Kaufmann, 2006.

[89] J. Cardoso, C. Anagnostopoulos, and S. Hadjiefthymiades, "The Syntactic and the Semantic Web," in *Semantic Web Services: Theory, Tools and Applications*, J. Cardoso, Ed.: IDEA Group Publishing, 2007, ch. 1, pp. 1-23.

[90] R. Jasper and M. Uschold, "A framework for understanding and classifying ontology applications," in *IJCAI99 Workshop on Ontologies and Problem-Solving Methods*, Stockholm, 1999.

[91] G. Vetere and M. Lenzerini, "Models for Semantic Interoperability in Service Oriented Architectures," *IBM Systems Journal*, vol. 44, pp. 887-904, 2005.

[92] [Online]. http://www.w3.org/RDF/

[93] M. Paolucci, T. Kawamura, T. Payne, and K. Sycara, "Importing the Semantic Web in UDDI," in *CAiSE '02/ WES '02 Revised Papers from the International Workshop on Web Services, E-Business, and the Semantic Web*, London, 2002.

[94] J. Colgrave, R. Akkiraju, and R. Goodwin, "External matching in UDDI," in *International Conference of Web Services (ICWS)*, San Diego, 2004, pp. 226-233.

[95] K. Harding, Ed., *MCSD Self-Paced Training Kit: Analyzing Requirements and Defining Microsoft.NET Solution Architectures*. Redmond, Washington: Microsoft Press, 2003.

[96] V. Jovanovic, S. Mrdalj, and A. Gardiner, "A Zachman Cube," *Issues in Information Systems*, vol. VII, no. 2, pp. 257-262, 2006.

[97] S.S. Ostadzadeh, F. Shams Aliee, and S. Ostadzadeh, "A method for consistent modeling of Zachman Framework cells," *Advances and Innovations in systems, Computing science and software engineering*, pp. 375-380, 2007.

[98] C.M. Pereira and P. Souza, "A Method to Define an Enterprise Architecture using the Zachman Framework," in *Symposium on Applied Computing*, 2004, pp. 1366-1371.

[99] Wikipedia. [Online]. http://en.wikipedia.org/wiki/Meta-Modeling

[100] W. Pidcock. (2003) InfoGrid. [Online]. http://infogrid.org/wiki/Reference/PidcockArticle

[101] E. Johannes. (2002) InfoGrid. [Online]. http://infogrid.org/wiki/Reference/WhatIsMetaModeling

[102] M. Saeki and H. Kaiya, "On Relationships among Models, Meta Models and Ontologies," in *6th OOPSLA Workshop on Domain-Specific Modeling*, Portland, 2006.

[103] (2009) UNESCO. [Online]. http://www.culturalpolicies.net/web/files/134/en/FCS_2009.doc

[104] N. Jevremović and Z. Cvetković, "MISS - Muzejski informacioni sistem Srbije," *Muzeološke sveske*, vol. 8, 1996.

[105] M. Cvrčanin, *Tezaurus za likovne umetnosti - Metodologija izrade*. Beograd, Srbija: Narodna biblioteka Srbije, 1983.

[106] S. Jovanović, *Tezaurus za oblast arheologije*. Beograd, Srbija: Narodna biblioteka Srbije, 1990.

[107] Lj. Kovačević, *Tezaurus za uređenje prostora: arhitektura, građevinarstvo, prostorno planiranje, urbanizam*. Beograd, Srbija: Narodna biblioteka Srbije, 1996.

Spisak slika

Slika 1.1: Grafički prikaz strukture teze ... 6
Slika 2.1: Open-EDI referentni model ... 11
Slika 2.2: Pregled tehnika i metoda za poslovni operacioni pogled ... 12
Slika 2.3: e^3-value koncepti ... 13
Slika 2.4: REA šablon ... 14
Slika 2.5: BMO elementi ... 15
Slika 2.6: IDEF0 model aktivnosti ... 15
Slika 2.7: Definicija poslovnog procesa ... 18
Slika 2.8: Relacija između procesa i servisa ... 18
Slika 2.9: BPM arhitektura ... 20
Slika 2.10: Osnovni skup BPMN elemenata ... 23
Slika 2.11: Funkcionalni obim i odnos BPMN 1.0 i BPMN 2.0 ... 23
Slika 2.12: Relacija između modelovanja i izvršavanja poslovnog procesa ... 26
Slika 2.13: Osnovni skup elemenata UML dijagrama aktivnosti ... 27
Slika 2.14: ebXML arhitektura ... 28
Slika 2.15: Odnos servisa i poslovne logike ... 33
Slika 2.16: Pristupi dizajniranju SOA sistema ... 38
Slika 2.17: BPM i SOA ... 40
Slika 2.18: Standardni scenario veb servisa ... 42
Slika 2.19: Osnovni slojevi platforme veb servisa ... 43
Slika 2.20: WSDL 1.1 i WSDL 2.0 dokumenti i relacije između elemenata ... 44
Slika 2.21: UDDI model podataka ... 45
Slika 2.22: Grafički prikaz prve i druge generacije tehnologija veb servisa ... 47
Slika 2.23: Kategorizacija kompozicije veb servisa ... 48
Slika 2.24: Grafički prikaz orkestracije ... 49
Slika 2.25: Grafički prikaz koreografije ... 49
Slika 2.26: Uporedni prikaz orkestracije i koreografije ... 50
Slika 3.1: Nivoi interoperabilnosti ... 53
Slika 3.2: EIF nivoi interoperabilnosti ... 54
Slika 3.3: Osnovni koncepti interoperabilnosti preduzeća ... 56
Slika 3.4: Evolucija okvira za arhitekturu informacionih sistema preduzeća ... 61
Slika 3.5: Zahmanov okvir 3.0 za arhitekturu informacionih sistema preduzeća ... 64
Slika 3.6: Odnos EA i SOA ... 67
Slika 3.7: Mapiranje SOA u Zahmanov okvir ... 68
Slika 3.8: INTEROP NoE - tri osnovne dimenzije ... 70
Slika 3.9: Struktura ATHENA okvira za interoperabilnost ... 71

Slika 3.10: AIF konceptualni model ... 72
Slika 3.11: Osnovni elementi EIF konceptualnog modela za javne usluge ... 73
Slika 3.12: Trougao značenja ... 75
Slika 3.13: Nivoi semantike ... 75
Slika 3.14: Primer rečnika i taksonomije ... 76
Slika 3.15: Centralizovani "svako-prema-svakom" model ... 77
Slika 3.16: Centralizovani "svi-prema-jednom" model ... 78
Slika 3.17: Decentralizovani "svako-prema-svakom" model ... 79
Slika 3.18: Decentralizovani "svi-prema-jednom" model ... 80
Slika 3.19: RDF naredba ... 81
Slika 3.20: Grafički prikaz RDF naredbe ... 82
Slika 3.21: RDF naredba ... 82
Slika 3.22: primer RDF šeme ... 83
Slika 3.23: Izražajnost OWL jezika i RDFS ... 84
Slika 3.24: Otkrivanje servisa u semantički proširenom UDDI registru ... 85
Slika 3.25: Eksterno provera podudarnosti servisa ... 86
Slika 3.26: P2P otkrivanje semantički označenih servisa ... 87
Slika 4.1: Inicijalizovani Zahmanov okvir ... 94
Slika 4.2. Algoritam popunjavanja ćelija Zahmanovog okvira ... 95
Slika 4.3: Delimično popunjen Zahmanov okvir ... 96
Slika 4.4: SOFI okvir ... 97
Slika 4.5: Inicijalni SOFI okvir sa označenim sadržajima ćelija ... 99
Slika 4.6: Redosled popunjavanja i međuzavisnosti sadržaja ćelija SOFI okvira ... 104
Slika 4.7: Meta model arhitekture sistema ... 105
Slika 4.8: Grupisane ćelije SOFI okvira prema fazama životnog ciklusa razvoja softvera ... 107
Slika 4.9: SOA domeni u SOFI okviru ... 108
Slika 4.10: Odnos BPM i SOA ... 110
Slika 4.11: Odnosi između ontologija, modela i meta modela ... 114
Slika 4.12:. Prošireni semantički trougao znanja ... 115
Slika 4.13. Generički meta model ... 116
Slika 5.1: Koncept rešenja ... 128
Slika 5.2: Slučajevi korišćenja za učesnika: Ekspert za KD ... 129
Slika 5.3: Slučajevi korišćenja za učesnika: Registrant WS ... 129
Slika 5.4: Slučajevi korišćenja za učesnika: Ustanova kulture ... 130
Slika 5.5: Slučajevi korišćenja za učesnika: Korisnik ... 130
Slika 5.6: BPMN dijagram ... 132
Slika 5.7: Domenski model klasa ... 133
Slika 5.8: Meta model baza podataka o KD ... 135
Slika 5.9: Primer odrednice "Skulptura" ... 136
Slika 5.10: Konceptualni model tezaurusa u oblasti KD ... 137
Slika 5.11: Dijagram sekvenci za kreiranje tezaurusa ... 138
Slika 5.12: Dijagram sekvenci za kreiranje šeme WS ... 139
Slika 5.13: Dijagram sekvenci za registrovanje WS ustanova kulture ... 140
Slika 5.14: Dijagram sekvenci za pretraživanje podataka o KD ... 141
Slika 5.15: Dijagram klasa ... 143

Slika 5.16: Servisi WSport i WSm..... 144
Slika 5.17. Korisnička forma za kreiranje upita 146
Slika 5.18: Arhitektura Sistema za pretraživanje i predstavljanje KD..... 147
Slika 5.19: Logički model podataka 148
Slika 5.20: Logički dijagram isporuke..... 149
Slika 5.21. Deo WSDL dokumenta za WSport servis..... 150

Spisak tabela

Tabela 2.1: Razlike između modela poslovanja i modela poslovnog procesa 17
Tabela 3.1: Tehnologije po C4IF nivoima interoperabilnosti 55
Tabela 3.2. Mapiranje SOA i EA arhitektonskih domena 66
Tabela 3.3: Kategorije mape znanja 69
Tabela 4.1: Faze i isporučivi rezultati u MSF okviru 91
Tabela 4.2: Matrica artifakata i njihovih zavisnosti po MSF fazama 92
Tabela 5.1: Scenario upotrebe za slučaj korišćenja "Korisnik" 131