

Modelovanje procesa tehničke realizacije usluga jezicima ontologija

Aleksandar Aladžić
Elektrotehnički fakultet
Univerzitet u Beogradu
Aleksandar.Aladzic@mtel.ba

Boško Nikolić
Elektrotehnički fakultet
Univerzitet u Beogradu
nbosko@etf.rs

Sažetak - U ovom članku je predstavljen prijedlog načina modelovanja procesa tehničke realizacije usluga u telekomunikacionom operateru ontološkim jezicima. Procesi su modelovani modelovanjem njihovih sastavnih elemenata odgovarajućim konstruktima ontoloških jezika, pri čemu je izvršeno preslikavanje jezika, interno razvijenog za definisanje procesa tehničke realizacije u odgovarajuće konstrukte ontoloških jezika. Elementi internog jezika su opisani odgovarajućim elementima ontoloških jezika, kao i njihovim vezama. Na ovaj način, poboljšava se jasnoća modela zbog inherentne osobine ontoloških jezika za formalizmom, a takođe, dobija se i mogućnost provjere logičke konzistentnosti samog modela.

Ključne riječi-ontologije; modelovanje; procesi; jezici ontologija;

I UVOD

Za interne potrebe definisanja procesa realizacije usluga, u M:telu je razvijen interni grafički jezik za njihovo modelovanje. Razlog za upotrebu ovakvog jezika za definisanje procesa, leži u potrebi da se procesi permanentno mijenjaju, kako zbog poslovnih promjena u Kompaniji, tako i zbog sve češćeg uvođenja novih tehnologija i servisa, čija realizacija implicira drugačiji način implementacije, što kao posljedicu ima različit način realizacije. Na ovaj način, softverski sistem za tehničku realizaciju servisa i usluga postaje otporniji na promjene, zadovoljava

trenutne potrebe poslovnih procesa i produžava rok korisnog trajanja u upotrebi fleksibilnim prilagođenjima novim potrebama. Ovako opšti pristup rješavanju problema svakako ima i neke svoje nedostatke. Najvažniji od njih je nedostatak jasnih formalizama u implementiranim mehanizmima koji bi mogli dovesti do različitih interpretacija realizovanih procesa, što bi se onda preslikalo i na poslovni model. Ovaj problem bi se mogao manifestovati i kroz rukovanje sistemom od strane različitih administratora [1]. Takođe, ako uzmemo u obzir da su danas softverski sistemi uglavnom povezani sa drugim sistemima, sa kojima više ili manje saraduju, nameće se potreba za formalizmima, kako u samim internim procesima, tako i u procesima nastalim integracijom sa drugim sistemima, a takođe, potrebno je i da rješenje bude moguće obraditi računom. Jedan od načina obezbjeđenja potrebnog formalizma bi bio korištenje nekog već postojećeg sistema baziranog na formalizmu koji ove probleme već ima riješene i adekvatnim modelovanjem našeg sistema pomoću tog sistema. U ovu svrhu nam može pomoći koncept ontologije, kao i jezici na kojima su ontologije predstavljene, jer su bazirane na formalnom logičkom modelu. Ovdje ćemo prikazati jedno rješenje tog problema, modelovanjem internog grafičkog jezika interno razvijenom ontologijom.

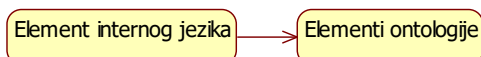
II KONCEPT REALIZACIJE

Postoji više načina realizacije ovog problema. Jedan od načina je da se iz modela generišu metapodaci, koji se zatim preslikaju u ciljanu ontologiju, i sa kojim ontologija ne gubi vezu, nego se međusobno dopunjuju[2]. Ovakav pristup je dobar ako se povezuju

III PREGLED INTERNOG GRAFIČKOG JEZIKA ZA REALIZACIJU PROCESA

već postojeći entiteti: postojeći jezik procesa sa postojećom ontologijom iz domena ontologija procesa. Pošto za naš interni grafički jezik razvijamo sopstvenu ontologiju, nemamo potrebu za tim dodatnim slojem apstrakcije, jer ćemo je potpuno prilagoditi našim internim potrebama. Takođe, već su razvijene i ontologije za domen poslovnih procesa, kao što su npr.: Business Management Ontology - BMO [3], Toronto Virtual Enterprise –TOVE [4], Enterprise Ontology [5], kao i metodi njihove upotrebe [6]. Motivacija za realizaciju interne ontologije je proizašla iz potrebe da se procesi modeluju ontologijom koja je potpuno prilagođena internim potrebama i koja, u zavisnosti od potreba, može da se dalje modifikuje i prilagođava po potrebi. Moguć je i pristup modelovanja već nabrojanim postojećim ontologijama, ali bi to nametalo probleme u njihovom prilagođenju internim potrebama, jer, semantičko značenje entiteta iz postojeće ontologije, ne mora nužno da, u potpunosti, odgovara internim potrebama, pa se, u tom slučaju, nameće potreba za dodatnim prilagođenjima i njihovom ažuriranjem, u slučaju promjena. Pristup korišten u ovom radu je da se svi elementi internog grafičkog jezika predstave odgovarajućim ontološkim konceptima ili njihovom kombinacijom objedinjenom u jednu ontologiju, a da se konkretni procesi dobiju instanciranjem modelovanih elemenata iz ontologije. U slučaju integracije sa nekim drugim sistemom, dovoljno bi bilo samo povezati ontologije dva sistema, tako da bi drugi sistem imao jednoznačno određene koncepte. Korišteni koncept realizacije je prikazan na slici 1.

Ontologiju ćemo realizovati u jeziku OWL, koji je preporučen za tu svrhu od strane W3C Konzorcijuma, što, pored reprezentacije ontologije, omogućava i da se vrše dodatna rezonovanja nad podacima, što može da dovede do otkrivanja novih činjenica, kao i otkrivanje eventualne nekonzistentnosti u procesima i njihovim podacima. Takođe, ovaj jezik je dobro podržan od strane proizvođača softvera, pa neće biti problema prilikom realizacije ovog rješenja, jer se, u tom slučaju, mogu implementirati svi željeni OWL konstrukti.



Slika 1. Prikaz načina povezivanja koncepata iz korepondentnih domena

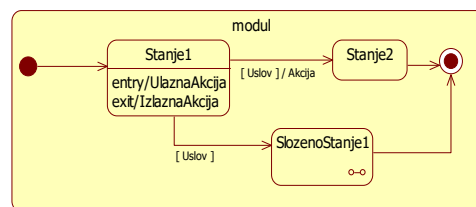
Motivacija za razvoj internog grafičkog jezika je proizašla iz potrebe za sistemom koji će imati fleksibilan dizajn, i koji će moći da se mijenja u skladu sa potrebama i u kratkom vremenskom roku. Iz ličnog iskustva, u radu sa gotovim rješenjima, pokazalo se da takva rješenja, niti u potpunosti mogu da se prilagode potrebama kompanije, niti se promjene mogu uraditi brzo, a često ni na jednostavan način. Interni grafički jezik je dizajniran tako da omogući maksimalnu fleksibilnost u definisanju procesa realizacije neke usluge ili servisa. Ovo je postignuto definisanjem elemenata jezika na nivou metamodela i njihovim instanciranjem u određenom modelu. Trudili smo se da definišemo dovoljan broj različitih generičkih metaelemenata čijim kombinacijama možemo da dobijemo bilo koji proces koji nam je potreban za tehničku realizaciju usluga i servisa. Na slici 2. je prikazan primjer jednog procesa realizovanog internim grafičkim jezikom, koji obuhvata sve mehanizme koji se nalaze u jeziku, kako bi se prikazali svi elementi grafičkog jezika. Prikaz je predstavljen pomoću elemenata UML - a. Interni grafički jezik ima sledeće elemete:

Modul – metaelement koji označava neki skup globalnih funkcionalnosti. On određuje koje su generalne mogućnosti procesa, kao i najopštiji skup podataka koji postoji u procesu,

Start – čvor koji označava početak procesa,

Stop – čvor koji označava kraj procesa,

Stanje – čvor koji označava da se proces nalazi u nekom stanju/fazi u toku svog izvršavanja, kada se nad njim vrše neke aktivnosti. Takođe, ono ima i **SLA** (Service Level Agreement) – koje znači koliko vremena je dozvoljeno da proces može provesti u tom stanju, i da li je dozvoljena promjena korisnika u tom stanju.



Slika 2. Proces implementiran internim grafičkim jezikom

Prilikom ulaska ili izlaska iz stanja moguće je generisati neku akciju. Akcija predstavlja neku aktivnost koja se može pokrenuti prilikom ulaska ili izlaska iz stanja. Takođe, ove akcije mogu biti i uslovljene, tj. nad njima se mogu definisati i uslovi pod kojim se generišu. Može se definisati i više akcija, ali se tada redoslijed njihovog izvršavanja definiše prioritizacijom. Takođe, način prelaska iz stanja u stanje može biti manuelni ili automatski,

Složeno stanje – predstavlja stanje koje se sastoji od više stanja povezanih tranzicijama. Ovako su, u principu, riješeni potproces koji se mogu generisati iz nekog procesa. Potproces koji mogu biti serijski ili paralelni, što je riješeno mogućnostima pokretanja jednog ili više procesa u nekom stanju, dok se, prilikom pokretanja procesa, naglašava da li se ostali procesi mogu nastaviti, ili ostaju blokirani do završetka novopokrenutog procesa. Sve što se može uraditi u elementarnom stanju, može se uraditi i u složenom stanju.

Resolution (prelaz) – tranzicija između stanja. Takođe, i na prelazu se može definisati neka akcija koja se generiše prilikom ovog prelaza, a ta akcija takođe, može biti i uslovljena nekim uslovom koji može biti prosti ili složeni. Svakako, i u ovom slučaju može biti više akcija, a njihov redoslijed izvršavanja se takođe realizuje prioritizacijom.

Uslov – nad tranzicijama se mogu definisati i uslovi koji moraju da budu ispunjeni da bi sistem iz jednog

stanja prešao u drugo preko određene tranzicije. U ovom slučaju, na tranziciju ne mora da bude postavljena akcija, nego se uslov odnosi na uslov prelaska iz stanja u stanje.

Koncept paralelizacije procesa je realizovan aktiviranjem akcije, u određenoj fazi, koja kreira novi potproces, koji se odvija paralelno sa glavnim procesom. Ovakvih akcija može biti i više, što implicira da se može kreirati više paralelnih procesa.

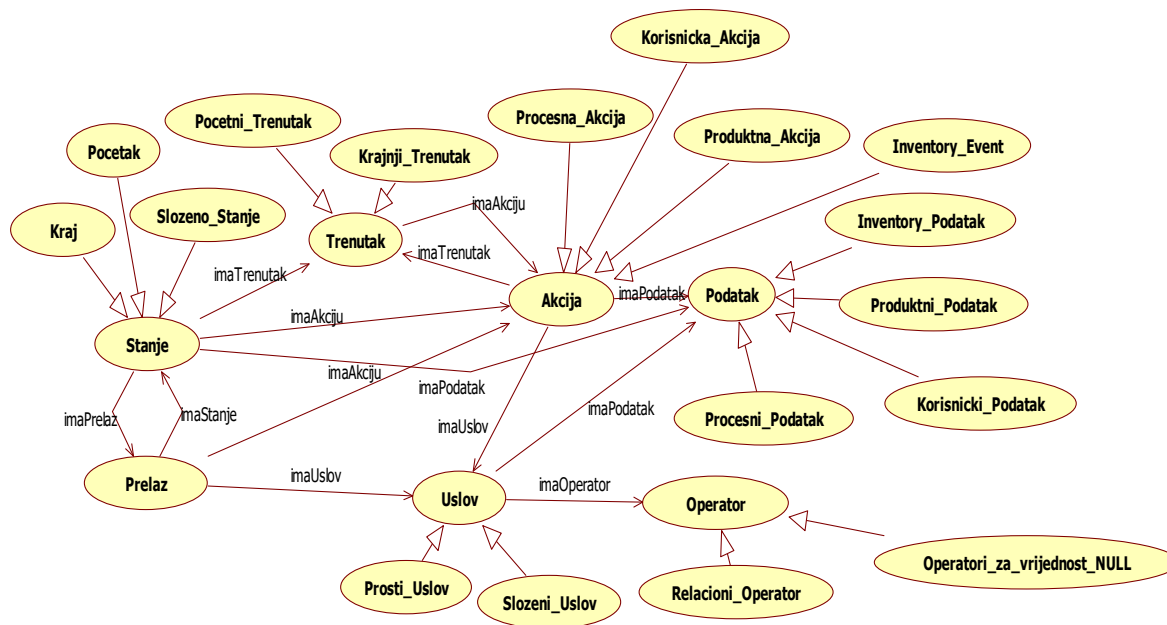
Sinhronizacija procesa je riješena akcijama koje se izvršavaju u nekom od stanja, mogu biti uslovljene, a utiču na status željenog procesa.

Gorepredstavljani elementi predstavljaju elemente modela, čijim se kombinovanjem i adekvatnim podešavanjem, dobije model procesa realizacije neke usluge ili servisa. Instanciranjem modela, dobije se konkretan proces za konkretnu realizaciju neke konkretne usluge ili servisa po koracima propisanim metamodelom procesa.

IV ONTOLOGIJA

Na slici 3. je prikazana realizovana ontologija.

U realizaciji ontologije su korišteni ontološki koncepti koji se mogu predstaviti jezikom OWL. Instanciranjem odgovarajućih elemenata ontologije, možemo da realizujemo bilo koji proces na ontološkom nivou.



Slika 3. Realizovana ontologija

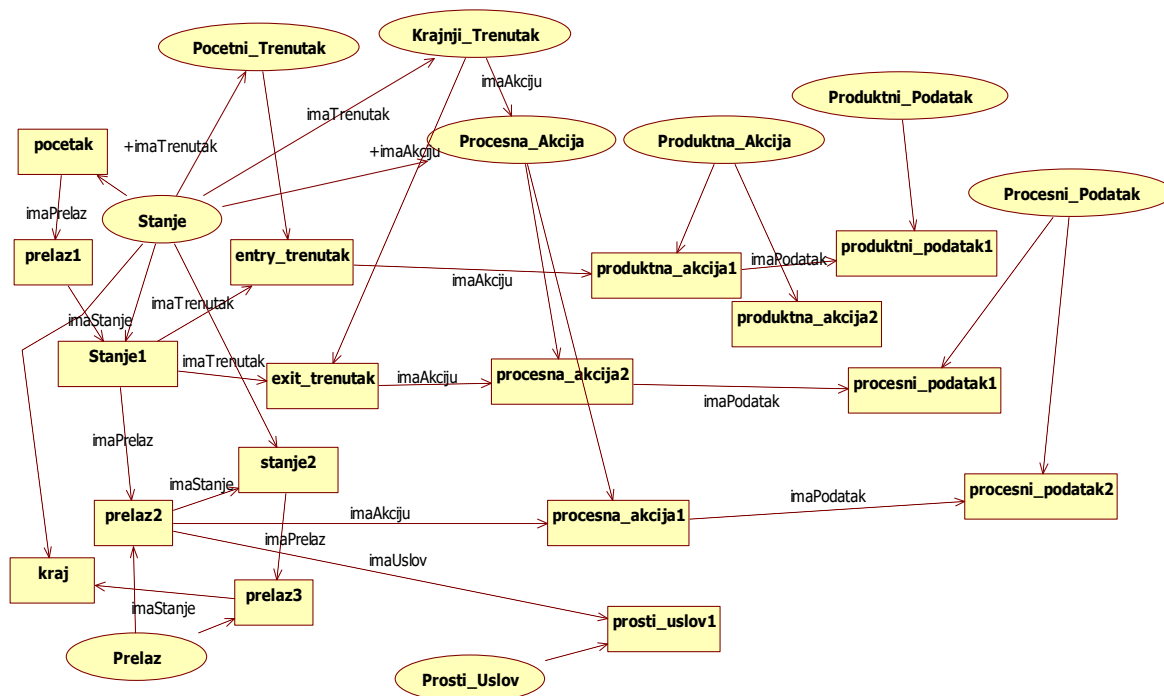
Linije nasleđivanja na slici predstavljaju ontološku relaciju „subClassOf“, a ostale relacije su objekte ontološke relacije. Za svaku relaciju je jasno definisan domen, tj. skup elemenata koji može da ima predmetno svojstvo i opseg, tj. skup vrijednosti koje elementi tog svojstva mogu da imaju. Takođe, relacijama možemo i da dodijelimo potrebna svojstva. Na ovaj način smo predstavili sve pojmove koji se pojavljuju u internom grafičkom jeziku ontološkim konstruktima i definisali smo veze koje su moguće među njima. Veze predstavljaju semantičke relacije između koncepata, koje su takođe i logički određene. Na ovaj način smo semantički opisali naš interni grafički jezik, a samim tim, i sve procese koji se njime mogu realizovati. Svaki konkretan proces predstavlja instancu realizovane ontologije. Na slici 4. Vidimo konkretan proces predstavljen ontologijom. Elipse predstavljaju klase, a pravougaonici konkretne vrijednosti instancirane iz klasa. Klase su sa instancama povezane isprekidanim usmjerenim linijama bez opisa, dok su semaničke relacije između njih predstavljene isprekidanim strelicama sa objašnjenjem koju ontološku relaciju predstavljaju. OWL klase i relacije su realizovane Java klasama sa odgovarajućim osobinama, dok su instance OWL klasa

i relacija realizovane instancama korenspondentnih Java klasa.

V DISKUSIJA

Definisanjem internog grafičkog jezika ontologijom, dobili smo semantički okvir koji nam omogućuje da na jednoznačan način predstavimo sve naše procese, što je posebno bitno u složenoj i heterogenoj kompaniji, koja zapošljava personal različitog nivoa znanja, iskustva i iz različitih oblasti znanja. Takođe, ontologije nam omogućavaju i da procese jasno definišemo, što može da ima prednosti u njihovoj daljoj obradi. Ontologija nam omogućava i jednostavnije povezivanje sa drugim sistemima kompanije, ili drugih kompanija, koja se sada mogu realizovati i na ontološkom nivou, koristeći se mehanizmima povezivanja ontologija.

Takođe, ontologije nam omogućavaju i da procese jasno definišemo, što može da ima prednosti u njihovoj daljoj obradi. Ontologija nam omogućava i jednostavnije povezivanje sa drugim sistemima kompanije, ili drugih kompanija, koja se sada mogu realizovati i na ontološkom nivou, koristeći se mehanizmima povezivanja ontologija.



Slika 4. Reprezentacija procesa ontologijom

Takođe, nad ovako realizovanim procesima je moguće primijeniti i mehanizme rezonovanja, čime se može doći i do novih, dotad neotkrivenih činjenica vezanih za procese. U vezi sa ovim je i mogućnost da upiti prikažu rezultate koje ne bi prikazali ako bi se primijenili na sisteme koji nemaju sposobnost rezonovanja, jer bi prikazali i nove činjenice. Ovakav način reprezentacije omogućava da se svaki od definisanih elemenata dalje detaljnije definiše kroz specijalizacije i ontološke relacije sa drugim elementima, čime se postiže još veća jasnoća i određenost. Na ovaj način možemo izvršiti ranu provjeru da li je proces validan; prije nego što ga pustimo u produkciju i otkriti sve eventualne semantičke greške u implementaciji procesa.

VI ZAKLJUČAK

Prikazani ontološki model predstavlja dobru polaznu osnovu za dalji razvoj. U model bi se mogla implementirati i pravila pomoću nekog jezika za pravila, čime bi ontološki model dobio novu kvalitativnu osobinu. Jezik koji bi se mogao koristiti je npr. Semantic Web Rule Language (SWRL), jer se relativno lako može ugraditi u OWL ontologiju; predstavljanjem SWRL pravila u OWL jeziku. Na ovaj način bi se dobila mogućnost implementacije mnogo naprednijeg rada sa uslovima, jer bi, pored proste provjere uslova, mogli da se realizuju i uslovi u formi implikacije. U tom slučaju bismo procese mogli koristiti na jedan potpuno novi način, jer bi se dobila mogućnost da, pored reaktivne, koristimo i deduktivnu interpretaciju pravila, što, kao posljedicu, ima prepoznavanje kompleksnih pravila i, na osnovu njih, generisanje odgovarajućih akcija ili aktivnosti, kao što bi bilo npr. da se, u zavisnosti od vrste proizvoda koji korisnik kupi i željene lokacije usluge, odredi automatski način tehničke realizacije usluge.

LITERATURA

- [1] Thomas, O.: Joint Reference Modeling: Collaboration Support through Version Management. In: Sprague, R. H. (ed.): Proceedings of the 40th Annual Hawaii International Conference on System Sciences : 3–6 January 2007, Big Island, Hawaii ; Abstracts and CDROM of Full Papers. Los Alamitos, CA : IEEE Computer Society Press, 2007,
- [2] Semantic EPC: Enhancing Process Modeling Using Ontology Languages Oliver Thomas, Michael Fellmann, SBMP 2007, Proceedings of the Workshop on Semantic Business Process and Product Lifecycle

Management held in conjunction with 3rd European Semantic Web Conference, Innsbruck, Austria, June 7, 2007, Volume 251, Paper 9.

- [3] Jenz, D. E.: Strategic White Paper: Ontology-Based Business Process Management; The Vision Statement. Erlensee : Jenz & Partner GmbH, 2003,
- [4] Fox, M. S.: The TOVE Project: A Common-sense Model of the Enterprise. In: Belli, F.; Radermacher, F. J. (eds.): Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, 5th International Conference, IEA/AIE 92 Paderborn, Germany, June 9–12, 1992, Proceedings. London : Springer, 1992 (LNCS), pp.25–34,
- [5] Uschold, M.; King, M.; Moralee, S.; Zorgios, Y.: The Enterprise Ontology. In: The Knowledge Engineering Review 13 (1998), No. 1, pp.31–89. – Special Issue on Putting Ontologies to Use,
- [6] Hepp, Martin and Roman, Dumitru, "An Ontology Framework for Semantic Business Process Management"(2007). *Wirtschaftsinformatik Proceedings 2007*. 27. <http://aisel.aisnet.org/wi2007/27>

ABSTRACT

In this paper, we present proposal of the way of modelling processes of the technical realisation of telecommunications services in telecommunications operator using ontological languages. The modelling of processes is accomplished by modelling their constitutional elements, using internally developed graphical process language and their linkage with ontological languages. Elements of internal language are described by appropriate elements of ontological language. In this way, we improved clarity of model because of inherent formalism property of ontological languages, and of course, we get possibility to check logical consistency of model.

MODELING OF PROCESSES OF TECHNICAL REALISATION OF SERVICES BY ONTOLOGY LANGUAGES

Aleksandar Aladžić
Boško Nikolić