

Generisanje OWL ontologija iz relacionih baza podataka

Aleksandar Damljanović, Slavko Marić

Sadržaj — Relacione baze podataka predstavljaju najvažniji izvor podataka na Internetu, međutim nedostaje im semantička perspektiva koja je neophodna za postizanje globalnog cilja kao što je Semantički Web. Transformisanjem u ontologije, struktura i sadržaj baza podataka postaju dostupni aplikacijama Semantičkog Web-a, koje koriste ontologije da identifikuju semantiku korištenih podataka. U ovom radu je predstavljen jedan pristup generisanja OWL ontologija iz relacionih baza podataka. Pristup je implementiran korišćenjem Protégé softverskog alata i Oracle relacione baze podataka.

Ključne reči — Ontologija, Relaciona baza podataka, Semantički Web.

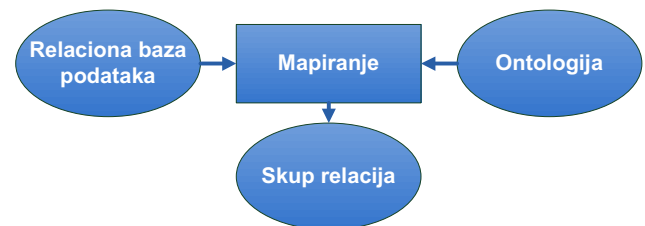
I. UVOD

VIZIJA Semantičkog Web-a ogleda se u konceptu nastanka Web-a podataka [1], gde dostupni izvori informacija mogu biti organizovani i korišteni semantičkim, a ne samo sintaksnim i strukturnim metodama, omogućavajući bolje razumevanje i ispunjavanje potreba ljudi i mašina za Web sadržajem. Ontologija kao ključna komponenta Semantičkog Web-a, predstavlja formalizovani rečnik termina koji definiše koncepte i relacije za opisivanje i prezentovanje određene oblasti interesovanja, deljiv od strane članova zajednice [2]. Za definisanje ontologija na Semantičkom Web-u koriste se standardni ontološki jezici poput Web Ontology Language (OWL). OWL je razvijen od strane W3C Web Ontology Working Group-e sa ciljem da olakša razvoj i dostupnost ontologija putem Web-a, a samim tim i pristupačnost Web sadržaja mašinama [3].

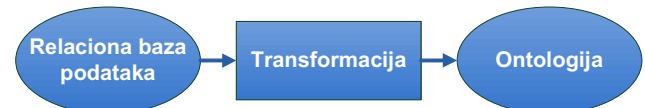
Jedan od kritičnih koraka u evoluciji trenutnog Web-a dokumenata u Web podataka, predstavlja uključivanje ogromne količine podataka uskladištenih u bazama podataka u koncept Semantičkog Web-a [4]. Baze podataka, koje čine integralni deo Deep Web-a, sadrže preko 500 puta više podataka nego statičke Web stranice, a 77% tih baza podataka predstavljaju upravo relacione baze podataka [5]. Integracija struktura i podataka relacionih baza podataka sa Semantičkim Web-om, postala je predmet intenzivnog istraživanja u naučnoj zajednici što je dovelo do osnivanja W3C RDB2RDF Group-e 2009. godine, sa ciljem standardizacije jezika za mapiranje

strukture relacionih baza podataka u RDF i OWL.

Ontologije omogućuju konceptualnu viziju heterogenih izvora podataka distribuiranih na velikom broju baza podataka [6]. One predstavljaju interfejs ka bazama podataka koji je u stanju da predstavi njihovu bogatu semantiku, poveća integraciju i ekstrakciju znanja iz podataka, istovremeno smanjujući redundantnost i nekozistentnost istih. Pristupi integraciji relacionih baza podataka i ontologija mogu se klasifikovati u dve glavne kategorije: pristupi koji mapiraju baze podataka sa postojećim ontologijama (domenskim ili lokalnim), i pristupi koji direktno transformišu sadržaj i strukturu relacionih baza podataka u odgovarajuću ontologiju [7],[8], kao što je prikazano na slici 1.



Mapiranje između relacione baze podataka i ontologije



Transformacija relacione baze podataka u ontologiju

Sl. 1. Pristupi integraciji rel. baza podataka i ontologija

Osnovna motivacija ovog rada je da se kritički analiziraju dosadašnja istraživanja koja se odnose na transformaciju relacionih baza podataka u odgovarajuće ontologije imajući u vidu mogućnost realne primene uz eventualna unapređenja dosad predloženih pristupa. Pristup podrazumeva automatsko generisanje ontologija, čime bi se izbegao vremenski zahtevan, neefikasan i naporan postupak ručnog kreiranja. Cilj je da se dođe do postupka za generisanje ontologija koji bi predstavljao dobar početak za izgradnju potpunih i standardnih ontologija u kontekstu Semantičkog Web-a.

U sekciji 2. su izložena srodna istraživanja na temu transformacije relacionih baza podataka u ontologije. Sekcija 3. sadrži opis predloženog pristupa transformacije. U sekciji 4. opisani su alati koji su korišteni prilikom implementacije. Sekcija 5. sadrži zaključak i naznačava pravce daljnjih istraživanja.

II. SRODNA ISTRAŽIVANJA

U literaturi postoji više radova koji se odnose na

Aleksandar Damljanović, LANACO d.o.o., Knjaza Miloša 15, 78000 Banjaluka, BiH; (e-mail: aleksandar.damljanovic@lanaco.com).

Slavko Marić, Elektrotehnički fakultet Banjaluka, Patre 5, 78000 Banjaluka, BiH; (e-mail: ms@etfbf.net).

mapiranje relacionih baza podataka sa postojećim ontologijama [9],[10]. Ovaj pristup se uglavnom zasniva na izradi wrapper sistema koji ima ulogu povezivanja relacione baze podataka i domenske ili lokalne ontologije. Njegova prednost je korišćenje već definisanih domenskih ontologija, koje su opšteprihvaćene i koje se mogu ponovo upotrebljavati, a glavni nedostatak se ogleda u često veoma napornom poslu mapiranja. Mapiranje kao rezultat daje skup relacija između ontologija i baza podataka.

Sa druge strane, transformacija strukture i podataka relacionih baza podataka u ontologiju, pretpostavlja da postoji samo baza podataka, dok se kao rezultat dobija odgovarajuća ontologija. Dobijena ontologija kasnije takođe može biti mapirana sa nekom postojećom ontologijom, a često se podvrgava i procesu validacije i prečišćavanja, radi dobijanja optimalnog rezultata. Transformacije relacionih baza podataka u ontologije opisana je u literaturi u više radova [6],[8], [11]-[13].

Buccella i dr. u radu [11] predstavljaju pristup koji koristi SQL-DDL (Data Definiton Language) kod za transformaciju baze podataka u OWL ontologiju. Pravila transformacije nisu formalizovana, već su zasnovana na objašnjenima kroz primere i obuhvataju tabele bez ograničenja, tabele sa jednim stranim ključem, tabele sa dva strana ključa i tabele sa više od dva strana ključa. Pored primarnog i stranog ključa, predložena je transformacija NOT NULL i CHECK ograničenja.

Li i dr. [12] su predložili skup polu-formalnih pravila za kreiranje OWL ontologije upotrebom SQL-DDL koda. Pravila su definisana kombinacijom formalne notacije i engleskog jezika, i namenjena su za generisanje ontoloških klasa, atributa, hijerarhije, kardinalnosti i instanci. Predložena je transformacija i UNIQUE ograničenja.

Astrova i dr. [8] su predstavili pravila i primere za definisanje automatskog sistema za transformaciju relacione šeme u OWL Full ontologiju. Prvi put je predložena transformacija stranog ključa koji referencira istu tabelu putem atributa simetričnosti. SQL tipovi podataka su transformisani u odgovarajuće XML Schema tipove podataka.

Tirmizi i dr. [13] su definisali pristup za transformaciju normalizovanih relacionih baza podataka u OWL ontologiju pomoću SQL-DDL koda. Ovaj pristup predstavlja konsolidovani sistem zasnovan na prethodno opisanim pristupima, i definiše skup transformaciona pravila u logici prvog reda, sa ciljem da se izbegnu sintaksne i semantičke dvosmislenosti.

Alalwan i dr. [6] su predložili pristup koji pored transformacije normalizovane relacione šeme baze podataka upotrebom SQL-DDL koda, takođe vrši validaciju dobijene ontologije pomoću konceptualnog modela (E/R dijagram). Omogućena je i migracija podataka u dobijenu ontologiju.

Pristup predložen u ovom radu ima za cilj da unapredi postojeće pristupe u domenu transformisanja kompozitnih ograničenja. U većini radova opisana je transformacija jednostavnih ograničenja, koja se sastoji od jedne kolone. Transformisanjem kolona i kompozitnih ograničenja u

klase, dobijamo mogućnost njihovog jasnijeg definisanja i upotrebe u ontologiji, kao i mogućnost ponovne upotrebe tako opisanih klasa u različitim relacijama (npr. kolona često sadrži više ograničenja).

III. PREDLOŽENI PRISTUP

Predloženi pristup kreira ontologiju na osnovu SQL-DDL koda dobijenog iz relacione baze podataka, i nakon toga migrira sadržaj baze podataka u OWL ontologiju. Transformacija predstavlja preslikavanje komponenti šeme baze podataka (tabele, kolone, ograničenja itd.) u odgovarajuće ontološke komponente (klase, attribute, restrikcije itd.).

A. Tabele

Svaka tabela relacione baze podataka transformiše se u odgovarajuću OWL klasu. Klase dobijaju ime na osnovu imena tabele, a opis tabele postaje komentar u anotacijama klase.

```
<owl:Class rdf:about="#Countries">
  <rdfs:comment>Country table.</rdfs:comment>
</owl:Class>
```

Klase tabele označavaju se kao disjoint klase, tako da individue ne mogu biti instance više od jedne klase.

B. Kolone i tipovi podataka

Sve kolone koje ne predstavljaju strani ključ, transformišu se u attribute tipa podataka (datatype property). Atributi dobijaju ime po imenu kolone u tabeli, a opis kolone postaje komentar u anotacijama atributa.

```
<owl:DatatypeProperty rdf:about="#Country_name">
  <rdf:type
    rdf:resource="#owl:FunctionalProperty"/>
  <rdfs:comment>Country name</rdfs:comment>
  <rdfs:domain rdf:resource="#Countries"/>
  <rdfs:range rdf:resource="#xsd:string"/>
</owl:DatatypeProperty>
```

OWL specifikacija [3] preporučuje upotrebu XML Schema tipova podataka te stoga opseg atributa predstavlja XML Schema tip podatka koji odgovara SQL tipu podatka kolone, vidi tabelu 1.

TABELA 1: PREGLED NEKIH KORESPONDENTNIH SQL/XSD TIPOVA PODATAKA

SQL tip podatka	XSD tip podatka
decimal	xsd:decimal
varchar	xsd:string
date	xsd:date

Kolona koja ne pripada ograničenju koje je opisano klasom, ima za domen klasu tabele kojoj ta kolona pripada. U suprotnom, domen kolone predstavlja klasa koja opisuje ograničenje kojem kolona pripada. O transformaciji ograničenja više će biti govora u sekciji C.

Ako kolona poseduje nekoliko ograničenja, dodatno kreiramo klasu koja predstavlja tu kolonu. Domen atributa tipa podatka te kolone postaje klasa te kolone. Klasa uspostavlja relacije sa klasama putem objektnih atributa (object property) koje dobijamo transformacijom tih

ograničenja. Klase kolona označavaju se kao disjoint klase.

Pošto određena kolona može imati samo jednu vrednost za određeni red u tabeli, svaki atribut vrste podataka označićemo kao funkcionalni atribut, tj. svaka individua može imati samo jednu vrednost tog atributa.

C. Ograničenja

Pristup opisan u ovom radu opisuje transformaciju sledećih ograničenja: NOT NULL, UNIQUE, PRIMARY KEY, FOREIGN KEY i CHECK (delimičan opis).

Ograničenje NOT NULL transformiše se u restrikciju minimalne kardinalnosti 1 za dati atribut tipa podatka.

```
<owl:Restriction>
  <owl:onProperty rdf:resource="#End_date"/>
  <owl:minQualifiedCardinality
    rdf:datatype="&xsd;nonNegativeInteger">1
  </owl:minQualifiedCardinality>
</owl:Restriction>
```

Ograničenje UNIQUE transformiše se u odgovarajuću klasu i prateći objektni atribut. Ime klase i ime atributa označavamo imenom ograničenja iz baze podataka i dodatnim sufiksima. Domen atributa je klasa tabele kojoj ograničenje pripada, a opseg je klasa ograničenja. Atribut je inverzno funkcionalan, čime se ostvaruje jedinstvenost ograničenja, i ima restrikciju kardinalnost jednaka 1.

```
<owl:ObjectProperty rdf:about="#Emp_email_uk">
  <rdf:type
    rdf:resource="&owl;InverseFunctionalProperty"/>
  <rdfs:range rdf:resource="#Emp_email_uk_uq"/>
  <rdfs:domain rdf:resource="#Employees"/>
</owl:ObjectProperty>
```

PRIMARY KEY se transformiše u odgovarajuću klasu i prateći objektni atribut koji ima za domen klasu tabele kojoj primarni ključ pripada i kao opseg klasu primarnog ključa. Objektni atribut je inverzno funkcionalan, čime se ostvaruje jedinstvenost primarnog ključa, i ima restrikciju kardinalnost jednaka 1. Klasa primarnog ključa sadrži relacije prema atributima tipa podataka ili klasama transformisanih kolona. Klasa primarnog ključa poseduje minimalnu kardinalnost 1 na attribute ili klase kolona. Ime klase i ime objektnog atributa označavamo imenom ograničenja iz baze podataka i dodatnim sufiksima.

```
<owl:Class rdf:about="#Departments_pk">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty
        rdf:resource="#Department_id"/>
      <owl:minQualifiedCardinality
        rdf:datatype="&xsd;nonNegativeInteger">1
      </owl:minQualifiedCardinality>
      <owl:onDataRange
        rdf:resource="&xsd;integer"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

FOREIGN KEY se transformiše u funkcionalni objektni atribut koji ima za domen klasu tabele kojoj pripada i kao opseg klasu referencirane tabele. Ako je kolona deo i

primarnog ključa, tada je domen atributa klasa tog primarnog ključa. Ako se tabela referencira na samu sebe, onda je atribut simetričan, a domen i opseg su tabela kojoj strani ključ pripada. Ime atributa označavamo imenom ograničenja iz baze podataka.

```
<owl:ObjectProperty rdf:about="#Loc_c_id_fk">
  <rdf:type
    rdf:resource="&owl;FunctionalProperty"/>
  <rdfs:range rdf:resource="#Countries"/>
  <rdfs:domain rdf:resource="#Locations"/>
</owl:ObjectProperty>
```

Kada je CHECK ograničenje u pitanju, OWL nije u stanju da izrazi svu semantiku ovog ograničenje pruža. Ograničenja koje možemo izraziti su vrednosno ograničenje, koje možemo predstaviti putem owl:hasValue restrikcije ili XML Schema restrikcija, i nabrojivo ograničenje, koje možemo predstaviti kroz opseg atributa tipa podatka korišćenjem nabrojivih tipova podataka ili klase.

```
<xsd:minExclusive rdf:datatype="&xsd;integer">0
</xsd:minExclusive >
```

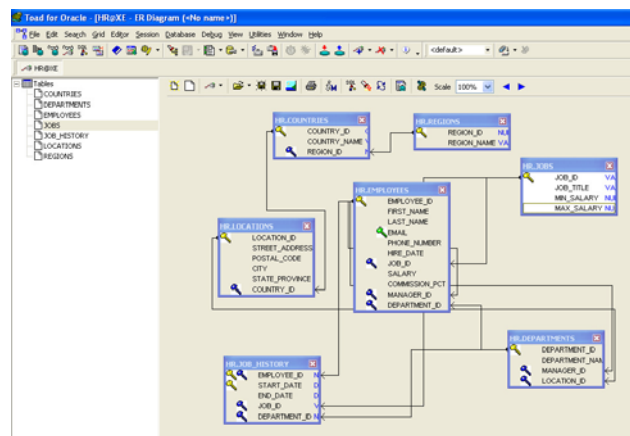
Klase ograničenja označavaju se kao disjoint klase.

D. Transformacija sadržaja

Zapisi iz tabela transformišu se u individue. Prvo se kreiraju klase kolona koje imaju više ograničenja i dodeljuju im se vrednosti atributa tipova podataka. Zatim se kreiraju individue klasa primarnih ključeva, UNIQUE ograničenja i klasa tabela, a potom im se dodeljuju vrednosti atributa tipova podataka i kreiraju se relacije između njih. Sledi kreiranje ostalih ograničenja osim stranih ključeva, i na kraju sledi kreiranje objektnih atributa koje predstavljaju strane ključeve.

IV. IMPLEMENTACIJA

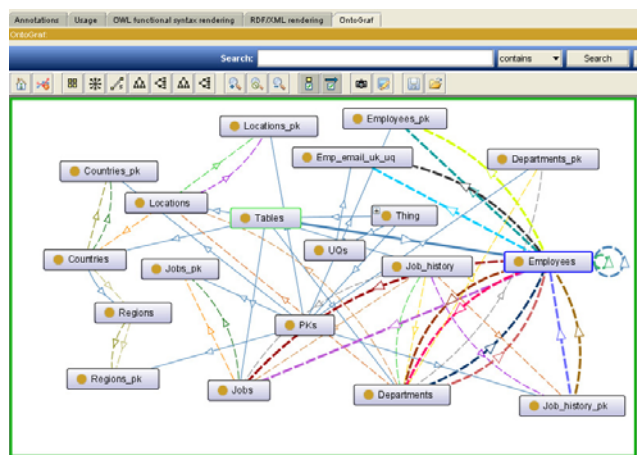
U ispitivanju pristupa korištena je HR baza podataka prikazana na slici 2, dobijena nakon instalacije Oracle Database 10g Express Edition.



Sl. 2. E/R dijagram HR scheme prikazan u TOAD for Oracle alatu za rad sa bazama podataka

Pristup je implementiran pomoću Protégé softverskog alata za rad sa ontologijama [14]. Kreiranu ontologiju evaluirali smo pomoću Pellet rasuđivača [15]. Rezultat evaluacije nije prikazao konflikte i pokazao je

konzistentnost ontologije. Relacije između klasa generisane ontologije prikazane su na slici 3.



Sl. 3. Prikaz relacija između klasa pomoću OntoGraf plugin-a u Protégé softverskom alatu

Nakon evaluacije ontologije upotrebom DL Query i SPARQL-a u mogućnosti smo pisati upite kojima se može izvršiti ekstrakcija znanja.

V. ZAKLJUČAK I BUDUĆIRAD

Semantički web ima za cilj stvaranja Web-a podataka sa jasno definisanim značenjem. Pošto se veliki deo podataka nalazi u relacionim bazama podataka, uspeh Semantičkog Web-a zavisi od njene sposobnosti za efikasnu integraciju sa relacionim bazama podataka. Jedan od pristupa toj integraciji predstavlja i transformisanje relacione šeme baze podataka u rezultujuću OWL ontologiju.

U ovom radu predložili smo unapređeni pristup transformacije relacionih baza podataka u odgovarajuće OWL ontologije. Metod obuhvata transformisanje različitih elemenata relacione baze podataka kao što su tabele, kolone i ograničenja, u odgovarajuće elemente ontologije. Sadržaj baze podataka se takođe transformiše u individue ontologije. U odnosu na druge radove iz ove oblasti, predloženi pristup omogućava definisanje kompozitnih ograničenja. Transformisanjem kolona u klase, omogućena je njihova ponovna upotreba u različitim ograničenjima. Takođe transformisanjem ograničenja u klase, omogućena je validna upotrebu inverzne funkcionalnosti koja garantuje njihovu jedinstvenost. Pristup je testiran na praktičnom primeru Oracle baze podataka u Protégé softverskom alatu. Prilikom evaluacije nisu identifikovani konflikti u generisanoj ontologiji.

Fokus budućeg rada biće na sledećim problemima: implementacija modela nasleđivanja, prepoznavanje i implementacija simetričnih i tranzitivnih relacija, okidača i referencijalnih akcija, implementacija potpune semantike CHECK ograničenja, aspekti praktične upotrebljivosti generisanih ontologija itd. Takođe, pošto smo fokus problema integracije baza podataka pomerili na mapiranje dobijene ontologije sa postojećim domenskim ontologijama, budući rad će obuhvatati problem mapiranja između tih dveju ontologija.

LITERATURA

[1] <http://www.w3.org/2001/sw/>

[2] <http://www.w3.org/standards/semanticweb/ontology>
 [3] *OWL 2 Web Ontology Language Document Overview*, W3C Recommendation, Oct. 2009. Available: <http://www.w3.org/TR/2009/REC-owl2-overview-20091027/>
 [4] S. Sahoo, W. Halb, S. Hellmann, K. Idehen, T. Thibodeau, S. Auer, J. Sequeda and A. Ezzat, "A Survey of Current Approaches for Mapping of Relational Databases to RDF". W3C RDB2RDF XG Incubator Report, 2009. Available: http://www.w3.org/2005/Incubator/rdb2rdf/RDB2RDF_SurveyReport.pdf
 [5] B. He, M. Patel, Z. Zhang and K. Chang, "Accessing the deep web" *Communications of the ACM*, vol. 50, issue 5, pp. 94–101, May 2007.
 [6] N. Alalwan, H. Zedan and F. Siewe. "Generating OWL ontology for database integration", presented at the Third International Conference on Advances in Semantic Processing (SEMAYRO '09), pp. 22–31, Oct. 2009.
 [7] J. Sequeda, S. Tirmizi and D. Miranker, "SQL Databases are a Moving Target", in Position Paper for W3C Workshop on RDF Access to Relational Databases, Cambridge, USA, 2007. Available: <http://www.w3.org/2007/03/RdfRDB/papers/sequeda.pdf>
 [8] I. Astrova, N. Korda, Nahum and A. Kalja, "Rule-based transformation of SQL relational databases to OWL ontologies", in *Proceedings of the Second International Conference on Metadata and Semantics Research*, Corfu, Greece, Oct. 2007.
 [9] C.P. de Laborda, and S. Conrad, "Relational OWL A Data and Schema Representation Format Based on OWL", in *Proceedings of the Second Asia-Pacific Conference on Conceptual Modelling (APCCM2005)*, Australian Computer Society, vol. 43, pp. 89–96, 2005.
 [10] M. Laclavik, "RDB2Onto: Relational Database Data to Ontology Individuals Mapping", in *Tools for Acquisition, Organisation and Presenting of Information and Knowledge*. P.Navrat et al. (Eds.), Vydavatelstvo STU, Bratislava, pp.86–89, 2006.
 [11] A. Buccella, M. Penabad, F. Rodriguez, A. Farina and A.Cechich, "From Relational Databases to OWL Ontologies", in *Proceedings of the Sixth National Russian Research Conference*, Pushchino, Russia, 2004.
 [12] M. Li, X. Du and S. Wang. "Learning ontology from Relational Database". *Proceedings of the Fourth International Conference on Machine Learning and Cybernetics*, pp.3410–3415, Guangzhou, China, 2005.
 [13] S.H. Tirmizi, J. Sequeda and D. Miranker, "Translating SQL Applications to the Semantic Web", in *Proceedings of the Nineteenth International Databases and Expert Systems Application Conference (DEXA2008)*, Turin, Italy, 2008.
 [14] <http://protege.stanford.edu/>
 [15] <http://clarkparsia.com/pellet/>

ABSTRACT

Relational databases are the most important source of Internet content, but they lack the semantic perspective that is essential for achieving global goal such as the Semantic Web. Transforming into ontologies, database structure and content become available to the Semantic Web applications, which use ontologies to identify the semantics of used data. This paper presents an approach to generate OWL ontologies from a relational databases. Approach is implemented using Protégé tool and Oracle relational database.

GENERATING OWL ONTOLOGIES FROM A RELATIONAL DATABASES

Aleksandar Damljanović, Slavko Marić