

Pravila na semantičnem spletu

Dejan Lavbič¹, Marko Bajec¹, Marjan Krisper¹

¹ Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, Tržaška 25, 1000 Ljubljana, Slovenija
E-pošta: Dejan.Lavbic@fri.uni-lj.si, Marko.Bajec@fri.uni-lj.si, Marjan.Krisper@fri.uni-lj.si

Povzetek. Jeziki za predstavitev pravil imajo čedalje pomembnejšo vlogo pri uveljavljanju semantičnega spleta. Po eni strani pravila omogočajo zapis deklarativnega znanja pri jezikih za predstavitev ontologij in izražajo omejitve ali pretvorbe, ki so povezane z opisno logiko. Po drugi strani pa lahko s pravili opredelimo uveljavljanje politik, odzivanje na spremembe, specifikacijo delovanja inteligentnih agentov in jih uporabimo še za številne druge namene. Jeziki za predstavitev pravil bodo tako igrali ključno vlogo, saj bodo kot lingua franca za izmenjavo pravil med različnimi sistemi in orodji ter bodo omogočali razvijanje, izvajanje, objavljanje in medsebojno komunikacijo pravil na semantičnem spletu. V članku je predstavljeno ogrodje za predstavitev znanja na semantičnem spletu, kjer je poudarek na integraciji ontologij in pravil. Povezovanje je izpeljano s pomočjo pretvornikov, saj trenutno ni podpore za izvajanje pravil v obliki SWRL, ki je eden najobetavnejših predlogov pravil na semantičnem spletu.

Ključne besede: semantični splet, pravila, RuleML, SWRL

Semantic Web Rule Languages

Extended abstract. Semantic Web rule languages play an important role in making Semantic Web valued and are gaining support. Rules can be used for describing declarative knowledge of ontology languages and to express restrictions or transformations that can include the portion of description logic. On the other hand rules can express policy enforcement, define responses to changes, specify intelligent agent behaviors and be used for other purposes. Therefore, Semantic Web rule languages will play an important role for being lingua franca in interchanging rules among different systems and tools, enabling developing, running, publishing and interacting among Semantic Web rules. It is important to emphasize that reuse and interchange of rules at a higher abstract layer is required due to the fact that during the knowledge acquisition process rules are captured at the business level. To enable rules to be used at the information system level, some translation is required to establish several abstraction layers. In the paper we present a framework for knowledge representation that uses ontologies and rules. Our main focus is on the dynamic, heterogeneous environment and use of the knowledge base where ontologies and rules are combined thus extending OWL DL expressiveness. The merging phase is supported by converters, because there are presently no reasoning engines available to support the latest SWRL proposal. This paper also presents several approaches to rule

representation on the Semantic Web, from SWRL, ORL to RuleML.

Keywords: Semantic Web, rules, RuleML, SWRL

1 Uvod

V ožjem pomenu jezik za predstavitev pravil na semantičnem spletu razumemo kot konkretno sintakso, temelječo na XML.¹ V širšem pomenu pa ga razumemo kot jezik z abstraktno sintakso, ki je namenjen predvsem za skupno podlago pri opredeljevanju različnih izpeljanih jezikov za različne potrebe.

Ti jeziki morajo imeti formalno semantiko, vendar lahko kljub temu obstajajo določeni konstrukti, ki nimajo formalne semantike, izpeljane iz logike prvega reda, jih pa vseeno potrebujemo pri reševanju določenih nalog. V zvezi s tem problemom se moramo zavedati dveh nevarnosti:

- Uporaba praktičnih konstruktov jezika, ki se zdijo pomembni, in sicer imajo določeno intuitivno, a ne formalno semantiko, čeprav imamo na voljo alternative, ki imajo formalno (morda nestandardno) semantiko.
- Prevelika pozornost teoretičnim vprašanjem standardne logike prvega reda, kot je

¹ eXtensible Markup Language – Razširljivi označevalni jezik, ki predstavlja format podatkov za izmenjavo strukturo dokumentov v spletu.

izračunljivost, kompleksnost, odločljivost in jedrnatost.

Pri standardizaciji jezikov za predstavitev pravil na svetovnem spletu je treba upoštevati uporabnike SQL², Prologa³ in drugih sistemov, saj mora obstajati povezava med različnimi pomembnejšimi koncepti omenjenih jezikov.

V prispevku bomo predstavili nekaj načinov, kako lahko sisteme za izvajanje poslovnih pravil povežemo s semantičnim spletom. V razdelku 2 sta najprej podrobneje predstavljena semantični splet in potreba po pravilih. Sledi opis predlaganega ogrodja za predstavitev znanja na semantičnem spletu, ki temelji na integraciji ontologij in pravil. Različni predlogi jezikov za predstavitev pravil na semantičnem spletu so zbrani v razdelku 3, medtem ko v razdelku 4 predstavimo podporo omenjenim jezikom v dejanskih grafičnih orodjih.

2 Semantični splet in pravila

Svetovni splet doživlja velik uspeh, kar nakazuje število uporabnikov in dokumentov, ki se nahajajo na njem. Vendar je dostop do informacij še vedno težaven, saj mora uporabnik vsebino najprej poiskati, nato pa še tolmačiti. Pri tem se postavljajo vprašanja, kaj je vsebina dokumenta in kakšna je semantika povezav med elementi na spletu. Pri iskanju podatkov na svetovnem spletu moramo tako poznati bodisi natančen naslov iskanega dokumenta bodisi ga poiščemo s pomočjo spletnega iskalnika. Problem spletnih iskalnikov pa je v predstavitvi iskalne zahteve vmesniku iskanja, pozneje pa tudi predstavitvi rezultatov iskanja. Zato je Tim Berners Lee v začetku tega tisočletja predstavil idejo semantičnega spleta kot nadgradnjo obstoječega spleta, kjer podatki dobijo pomen, kar omogoča boljše medsebojno sodelovanje ljudi in računalnikov [1, 5, 7]. Moč semantičnega spleta je v združevanju tehnologij za obvladovanje znanja na področju umetne inteligence in spletnih tehnologij.

Veliko pripomb širše javnosti je bilo k vse preveč standardom, ki so na voljo na svetovnem spletu. Očitki so se nanašali predvsem na ne dovolj jasne vizije posameznih predlogov. Verjamemo pa, da uporaba pravil na semantičnem spletu prinaša nove možnosti sodelovanja in souporabe XML, RDF,⁴ OWL⁵ in

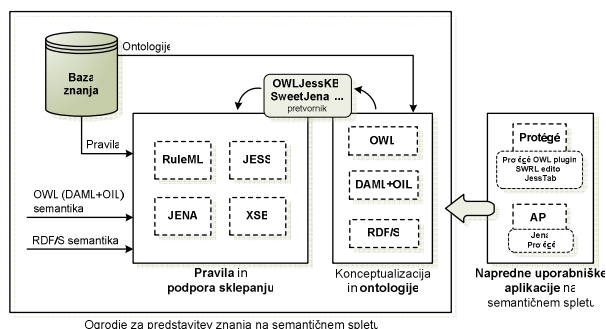
spletnih storitev, da bi odpravili problem integracije podatkov na svetovnem spletu.

2.1 Integracija podatkov

Čedalje več podatkov, ki so na voljo na svetovnem spletu, je v XML obliki. Razlog za to je preprost – enostaven prenos delno strukturiranih podatkov, ki s shemami prinašajo interoperabilnost in z XSLT⁶ omogočajo pretvorbo med različnimi XML formati. Vendar nam XML sam po sebi ne ponuja rešitve integracije podatkov [3]. Številna podjetja še vedno niso razvila svojih shem za poenostavitev izmenjave, pogosto pa izmenjava zahteva tudi združevanje podatkov iz različnih, porazdeljenih virov, brez skupnih shem ali shem brez jasno določenih preslikav. Tako še vedno ostaja problem npr. pri iskanju odgovora na vprašanje, kdaj sta dva podatka približno enaka oz. podobna. Če omogočimo dostop do podatkov na svetovnem spletu prek URI⁷ naslovov, smo naredili šele prvi korak pri integraciji podatkov.

2.2 Uporaba pravil in ontologij

Rešitev prej omenjenega problema lahko najdemo v uporabi pravil na semantičnem spletu. Pravila skupaj z ontologijami prav gotovo prinašajo višjo raven spletnim storitvam, kjer s konvergenco pridemo do semantičnih spletnih storitev [7]. Primere uporabe lahko najdemo v poslovnih okoljih, še zlasti uporaba agentov⁸ v elektronskem poslovanju, ki bo temeljilo na pridobljenem znanju [8]. Vloga agentov bo zbiranje informacij, sklepanje in optimizacija stroškov. Izvajali bodo tudi pogajanja z ustrežno mero avtonomije.



Slika 1: Uporaba pravil in ontologij na semantičnem spletu

najbolj izrazen jezik za predstavitev znanja na področju semantičnega spleta.

² Structured Query Language – Strukturirani poizvedovalni jezik za manipulacijo s podatki v relacijskih podatkovnih bazah.

³ PROgramming in LOGic – Jezik za logično programiranje.

⁴ Resource Definition Framework – S pomočjo jezika RDF predstavimo semantične relacije na ravni primerkov v obliki trojkov (oseba, predikat, predmet).

⁵ Web Ontology Language oz. Ontology Web Language je razvila skupina za spletne ontologije na konzorciju W3C in igra vlogo naslednika DAML+OIL. OWL je v tem trenutku

⁶ Extensible Stylesheet Language Transformations – Jezik za pretvorbo med XML dokumenti.

⁷ Uniform Resource Identifier – Splošen identifikator, ki se uporablja za razlikovanje konkretnih in abstraktnih virov na svetovnem spletu.

⁸ Za agenta kot entiteto velja, da je sposoben delovati v svojem okolju, deluje avtonomno – ima kontrolo nad svojim delovanjem, lahko komunicira z drugimi agenti in je zmožen zaznavanja svojega okolja.

Prvi korak k uresničitvi teh idej pa je zmožnost komunikacije z dovolj veliko mero skupaj določenega pomena, seveda s pomočjo semantičnega spleta, ki vključuje uporabo pravil in ontologij.

Slika 1 prikazuje ogrodje za predstavitev znanja na semantičnem spletu. Glavna ideja se skriva v integraciji pravil in ontologij ter uporabi baze znanja s strani naprednih uporabniških aplikacij [4].

Z uporabo pravil se na semantičnem spletu srečamo z novimi težavami, ki jih prej nismo imeli. Zato se moramo zavedati naslednjih ključnih elementov pri uvajanju in uporabi pravil na tem področju:

1. **Raznolikost.** Obstoječe in nastajajoče ontologije ter pravila uporabljajo heterogene oblike zapisovanja znanja, zato potrebujemo mehanizme, ki bodo sposobni obvladovati te raznolikosti.
2. **Porazdeljenost** obvladovanja pravil in ontologij pri uporabi baz znanj.
3. **Neskladnost.** Pri združevanju znanja pride do konfliktnih situacij, zato moramo imeti ustrezne mehanizme, s katerimi to obvladujemo.
4. **Dinamičnost.** Posodabljanja baz znanj bo vedno več, zato se moramo znati spopasti s to povečano frekvenco.
5. **Zakasnitev.** Razširljivost je odločitvenega pomena pri doseganju integracije znanja, kjer se želimo približati linearnemu času dostopa (podobno kot pri podatkovnih bazah).

Ob upoštevanju zgornjih izhodišč smo se odločili za pristop, kjer je baza znanja produkt integracije ontologij in pravil (glej sliko 1). Pri procesu združevanja so uporabljeni pretvorniki, ki ustrezne koncepte in relacije iz ontologij preslikajo v sisteme za izvajanje pravil, tako da lahko uporabimo obstoječe sisteme za izvajanje poslovnih pravil. Celotno ogrodje je tako na voljo naprednim aplikacijam, ki uporabljajo storitve semantičnega spleta.

3 Jeziki za predstavitev pravil na semantičnem spletu

V zadnjem času je bilo kar nekaj predlogov jezikov za predstavitev pravil na semantičnem spletu in v nadaljevanju si bomo podrobno ogledali naslednje [2, 3, 7]:

- Rule Markup Language (RuleML)⁹
- OWL Rules Language (ORL)¹⁰
- Semantic Web Rule Language (SWRL)¹¹

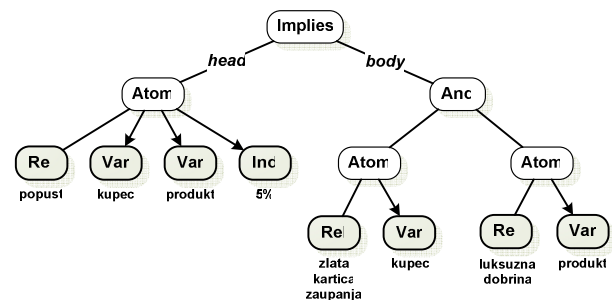
⁹ <http://www.ruleml.org>

¹⁰ <http://www.cs.man.ac.uk/~horrocks/DAML/Rules>

¹¹ <http://www.daml.org/2003/11/swrl>

3.1 RuleML

Jeziki RuleML imajo potencial, da postanejo glavni akterji na področju uporabe pravil na semantičnem spletu in v porazdeljenih sistemih, saj omogočajo kreiranje, procesiranje, objavljanje in medsebojno komunikacijo pravil [7]. Sintaksa RuleML temelji na XML, čeprav obstaja del njegove sintakse tudi v obliki RDF.



Slika 2: Primer RuleML pravila v obliki drevesa

Primer pravila v obliki RuleML si lahko ogledamo na sliki 2, kjer imamo pravilo prikazano v obliki drevesa z ustreznimi gradniki, ki jih definira RuleML. V naravnem jeziku se zgornje pravilo pomeni: Če ima kupec zlato kartico zaupanja in produkt spada med luksuzne dobrine, se kupcu pri nakupu upošteva 5-odstotni popust. Zapis XML omenjenega drevesa pa vidimo na sliki 3.

```
<Implies>
  <head>
    <Atom>
      <Rel>popust</Rel>
      <Var>kupec</Var>
      <Var>produkt</Var>
      <Ind>5</Ind>
    </Atom>
  </head>
  <body>
    <Anc>
      <Atom>
        <Rel>zlata kartica zaupanja</Rel>
        <Var>kupec</Var>
      </Atom>
      <Atom>
        <Rel>luksuzna dobrina</Rel>
        <Var>produkt</Var>
      </Atom>
    </Anc>
  </body>
</Implies>
```

Slika 3: Primer pravila RuleML v obliki XML

Cilj RuleML je zagotoviti ponovno uporabnost in izmenjavo na višji ravni, podobno kot pri MDA.¹² Namesto da ustvarimo še en jezik za opis pravil,

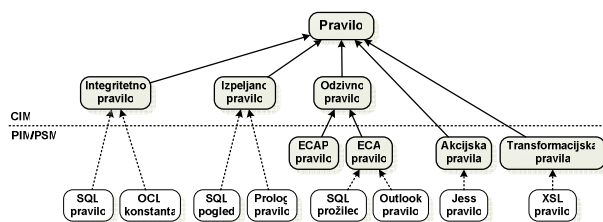
¹² Model-Driven Architecture je arhitektura za razvoj programske opreme, ki so jo predlagali pri Object Management Group (OMG). Funkcionalnosti sistema določimo v modelu, ki je neodvisen od platforme (PIM), s pomočjo ustreznega jezika za modeliranje (npr. UML). Za dejansko implementacijo pa je treba model PIM prevesti v model, ki je specifičen za izbrano platformo (PSM).

RuleML ponuja množico izpeljanih modularnih jezikov, ki temeljijo na skupnem podatkovnem modelu.

Pogled na pravila si lahko analogno pristopu MDA predstavljamo na treh ravneh:

1. Na **poslovni ravni** pravila v obliki izjav opredeljujejo določen del poslovnega procesa ali strukturno omejitev. Za predstavitev se uporablja deklarativna oblika, ki je največkrat naravni jezik ali kakšna grafična opisna oblika. Primeri pravil v naravnem jeziku so:
 - P₁: Oseba, ki želi opravljati voziški izpit, mora biti stara najmanj 18 let.
 - P₂: Dobiček na investicijo ni obdavčen, če je od nakupa delnice minilo že več kot 3 leta.
 - P₃: Ko cena delnice pade za več kot 5% in za investicijo ni treba plačati davka, jo prodaj.
2. Na **ravni neodvisnosti od platforme** so pravila izražena v obliki formalnih izjav, ki jih lahko pozneje povežemo z izvajalnimi izjavami dejanske platforme. Jeziki za zapis pravil na tej ravni so SQL:1999, OCL 2.0, ISO Prolog idr.
3. **Raven platformno odvisnih pravil** vsebuje pravila, zapisana z izjavami v določenem izvajalnem jeziku, kot so prožilci v podatkovni bazi Oracle 9, ekspertnem sistemu Jess ali pravila v orodju Microsoft Outlook.

Zavedati se moramo, da imamo v poslovni domeni opravka z jezikoslovno bogato in zapleteno izrazno obliko, ki jo lahko v formalni matematični obliki zapišemo le v omejenem obsegu [8]. Zato je pri RuleML uporabljen pristop, kjer imamo množico jezikov za opis najpomembnejših tipov pravil.



Slika 4: Kategorizacija pravil

V nadaljevanju si bomo nekoliko podrobneje ogledali osnovne tipe pravil, ki jih predlaga RuleML in jih lahko vidimo na sliki 3: *integritetna*, *izpeljana*, *odzivna*, *akcijska* in *transformacijska pravila*.

3.1.1 Integritetna pravila

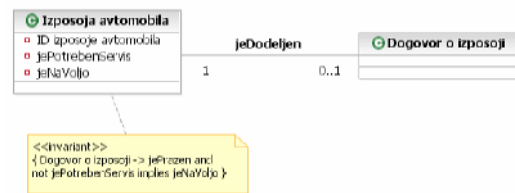
Integritetno pravilo sestavlja logični izraz, zapisan v poljubnem logičnem jeziku (npr. predikatna logika).

Pravilo predstavlja izjavo, ki mora biti resnična v vseh nastalih stanjih in prehodih med stanji v diskretnem dinamičnem sistemu, za katerega je definirano. Pravilo P₁ iz razdelka 2 je primer statičnega integritetnega pravila, medtem ko bi dinamično integritetno pravilo lahko bilo: Potrditvi rezervacije vozila mora slediti rezervacija vozila zahtevanega razreda za dan prej, kot je zahtevan datum.

Priljubljena jezika za izražanje integritete sta SQL in OCL¹³.

3.1.2 Izpeljana pravila

Izpeljano pravilo je sestavljeno iz enega ali več pogojev in sklepa. Na začetku razdelka smo omenili pravilo P₂, ki ga lahko uvrstimo med izpeljana pravila. Zelo znane oblike izpeljanih pravil lahko najdemo v obliki pravil Prolog in pogledov SQL. To vrsto pravil lahko prikažemo tudi v razrednih diagramih UML s pomočjo konstant OCL, kot je prikazano na sliki 4.



Slika 5: Opredelitev izpeljanega pravila s pomočjo OCL

V naravnem jeziku si lahko omenjeno pravilo predstavljamo kot: *Avto lahko najamemo, če ni povezan z nobenim dogovorom o izposoji in zanj ni potreben servis.*

3.1.3 Odzivna pravila

Odzivno pravilo sestavljajo prožilni dogodek, pogoj, odzivna akcija in včasih tudi popogoj. Odzivna pravila delimo glede na prisotnost popogojev v dve skupini: ECA,¹⁴ ki nimajo popogojev in ECAP,¹⁵ kjer so popogoji prisotni. To kategorizacijo prikazuje tudi slika 4, kjer sta omenjena tudi pogosto uporabljena primera teh pravil – prožilec SQL in pravilo Outlook.

3.1.4 Akcijsko pravilo

Akcijsko pravilo preprosto sestavljata pogoj in akcija, ki se ob izpolnjenem pogoj izvede. S to vrsto pravil lahko implementiramo tudi izpeljana pravila. Pri poslovnih pravilih so akcijska pravila ena najpomembnejših in tudi najpogosteje uporabljenih ter so ključna komponenta v družini jezikov RuleML. Primeri znanih izvajalnih

¹³ Object Constraint Language – Jezik za opis pravil nad modeli UML.

¹⁴ Event-Condition-Action

¹⁵ Event-Condition-Action-Postcondition

okolij akcijskih pravil so Jess, iLOG Rules/JRules, Fair Isaac/Baze Advisor itd.

3.2 ORL

Številne pomanjkljivosti jezika OWL izhajajo iz dejstva, da je jezik relativno bogat pri izražanju razredov, medtem ko se veliko slabše odreže pri lastnostih. Preprosto ni konstruktorja, s katerim bi lahko določili relacije med sestavljenimi lastnostmi in drugimi, morda celo sestavljenimi, lastnostmi. Preprost primer, ki opisuje ta problem, je relacija med lastnostmi starš, brat in stric, ki je v osnovni obliki jezika OWL ne moremo zapisati.

Eden od načinov reševanja omenjenega problema je razširitev jezika OWL z bolj izraznim jezikom za opis lastnosti. Dodajanje pravil jezikom za predstavitev znanja, ki temeljijo na opisni logiki, ni ravno nov pristop. Kar nekaj predhodnih sistemov (npr. Classic), ki temeljijo na opisni logiki, je svojo izrazno moč povečalo z uporabo jezika za predstavitev pravil [6]. Vendar so pravila v teh sistemih imela šibkejšo semantično moč, kot jo imamo pri uvedbi relacij med razredi in nadrazredi, saj so bile relacije prisotne le med posameznimi primerki in niso vplivale na sklepanje na podlagi razredne hierarhije.

ORL razširja OWL v sintaktičnem in semantičnem pogledu: osnovna sintaksa pravil ORL je razširitev abstraktne sintakse OWL DL¹⁶ in OWL Lite,¹⁷ pravila pa imajo tudi formalen pomen v povezavi z razširitvijo OWL DL [5].

Ko se odločamo za uporabo pravil OWL, moramo poudariti naslednje prednosti [7]:

- Glede na OWL ne pomenijo bistvenih sprememb, tako da so tudi relativno preprosta za uporabo.
- Pravila OWL imajo veliko izrazno moč, saj temeljijo na izrazni moči OWL pri obeh elementih pravil: vzroku (telo) in posledici (glava)
- Obljubljajo zelo elegantno rešitev raznovrstnega semantičnega procesiranja. Raziskovalnim skupinam, ki se ukvarjajo s semantičnim spletom, je jasno, da pri realizaciji ideje semantičnega spleta ne bo dovolj zgolj ena tehnika, model ali metoda. Potreba po različnih načinih uporabe ne bo pomenila izbire, ampak bo nujno prisotna.

3.3 SWRL

Čeprav z uporabo jezika OWL pridobimo veliko izrazno moč na semantičnem spletu, vseeno obstajajo določene

omejitve pri izražanju. Te omejitve pa lahko odpravimo z uporabo jezika SWRL za predstavitev pravil na semantičnem spletu. SWRL je nastal na temeljih ORL, ki smo ga predstavili v prejšnjem razdelku. Ideja za SWRL je nastala šele leta 2004 in je eden novjših predlogov jezika za opis pravil na semantičnem spletu.

Glavna ideja, ki stoji za SWRL, je razširitev OWL DL s pravili, kjer želimo obdržati čim večjo združljivost nazaj z obstoječo sintakso in semantiko. Pristop uvaja Hornovo obliko pravil, ki razširjajo sintakso OWL. Pravila so preprosto v obliki implikacije med vzroki (telo) in posledicami (glava). Pomen pravila pa lahko preberemo kot: vedno ko je zadoščeno pogojem v telesu, morajo držati tudi posledice v glavi.

SWRL zagotavlja zelo uporabno raven v hierarhiji semantičnega spleta, vendar se moramo zavedati, da bodo nekatere aplikacije potrebovale še večjo izrazno moč. Pri nekaterih aplikacijah je pomembno izražanje aritmetičnih relacij med posameznimi vrednostmi (npr. da osebo, ki ima prihodke enake ali višje od odhodkov, označimo za srečno, medtem ko pri osebah, kjer odhodki presegajo prihodke, načeloma velja, da so nesrečne). Še vedno ni jasno, ali bi bilo mogoče to funkcionalnost doseči z nadaljnjo razširitvijo funkcionalnosti jezika SWRL ali z razširitvijo podatkovnih tipov OWL [6].

4 Podpora jezikom v orodjih

Podpora jezikom za predstavitev pravil na semantičnem spletu v razvojnih orodjih je zelo slaba. Razlogi za to so prav gotovo v problemu zrelosti, saj so rešitve oz. predlogi še zelo novi.

4.1 Protégé

Eno redkih orodij, s pomočjo katerih lahko v grafičnem načinu obvladujemo pravila, zapisana v obliki SWRL, je SWRL editor.¹⁸ Na voljo je kot vključek za Protégé¹⁹ in nam omogoča preprosto obvladovanje pravil v okviru ontologije OWL. Koncepte, lastnosti in primerke ontologije lahko intuitivno vključujemo v pravila, ki jih ustvarjamo ali urejamo, prav tako pa lahko vse dele ustrezno opišemo. V SWRL editorju je prisotno tudi osnovno sintaktično in semantično preverjanje, vendar moramo za integriteto pravil skrbeti sami, saj nam sistem ne more preprečiti pisanja nesmiselnih pravil. Pravila so na koncu shranjena v obliki baze znanja OWL/RDF, tako da je mogoča uporaba teh pravil in spremljajoče ontologije tudi v drugih orodjih. Za večjo razširljivost je na voljo programski vmesnik, s pomočjo katerega lahko iz programskega jezika Java dostopamo do baze znanja in shranjenih pravil. Največja negativna lastnost je, da trenutno ni sistema za izvajanje teh pravil, ampak je na voljo zgolj preprost mehanizem za

¹⁶ Podmnožica jezika OWL z vključeno podporo jeziku opisne logike.

¹⁷ Podmnožica jezika OWL z minimalno semantično izrazno močjo.

¹⁸ <http://protege.stanford.edu/plugins/owl/swrl>

¹⁹ <http://protege.stanford.edu>

sklepanje, ki deluje na podlagi konceptov OWL in medsebojnih relacij, ne pa pravil, zapisanih v obliki SWRL. Vendar pa lahko pravila vseeno uporabimo s pomočjo pristopa, ki smo ga omenili v razdelku 2.2.

4.2 SweetRules

Zanimiv pristop je rešitev SweetRules,²⁰ ki jo avtorji označujejo kot prvo odprtokodno platformo za poslovna pravila na semantičnem spletu. Paket SweetRules prinaša številne uporabne lastnosti:

- pretvorba med različnimi heterogenimi sistemi za izvajanje pravil (forward in backward chaining),
- sklepanje s povezavo z različnimi obstoječimi sistemi,
- obvladovanje in testiranje baze pravil.

S pomočjo SweetRules lahko obvladujemo različne tehnologije na področju pravil in semantičnega spleta, kot so RuleML, Jess, IBM CommonRules, Knowledge Interchange Format (KIF), OWL, Process Handbook, Smodels, Jena-2, SWRL itd.

5 Sklep

Z razvojem semantičnega spleta se je razširila tudi uporaba ontologij kot formalizma za opis znanja in podatkov. Le-te so predstavljene v takšni obliki, da jih lahko souporabljamo na svetovnem spletu. Širša uporaba jezika OWL za predstavitev znanja na semantičnem spletu prinaša nove možnosti za razvoj zahtevnejših aplikacij na različnih problemskih področjih. Vendar je koristnost ontologij omejena z mehanizmi procesiranja, ki so integrirani v takšno obliko predstavitve. Ravno to se veliko odvija na področju formalizacije logičnega sloja ontologij. Jezik za predstavitev pravil na semantičnem spletu (SWRL) je pomemben korak v tej smeri in temelji na predhodnih raziskavah RuleML in ORL. Navsezadnje bomo šele z razpoložljivostjo standardiziranega jezika za zapis pravil lahko učinkovito uporabljali ontologije in pravila pri razvoju inovativnih aplikacij, ki uporabljajo semantični splet. Za uspešen razvoj in uporabo teh tehnologij je ključnega pomena razumevanje zmožnosti in medsebojnega delovanja pravil in ontologij.

6 Literatura

[1] Daconta, M.C., L.J. Obrst, and K.T. Smith, *The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management*. 2003, Indianapolis, USA: John Wiley & Sons.

[2] Groszof, B. and M. Dean, *Semantic Web Rules with Ontologies, and their E-Business Applications*, in *3rd International Semantic Web Conference (ISWC2004)*. 2004: Hiroshima, Japan.

[3] Halpin, H. *Rules: Enabling data integration using Semantic Web*. 2005 [citirano 5.11.2006]; Dostopno na: <http://www.w3.org/2004/12/rules-ws/paper/77/>.

[4] Hatala, M., R. Wakkary, and L. Kalantari, *Rules and ontologies in support of real-time ubiquitous application*. *Journal of Web Semantics*, 2005. **3**(1): p. 5-22.

[5] Horrocks, I. and P.F. Patel-Schneider, *A Proposal for an OWL Rules Language*, in *The 13th International World Wide Web Conference*. 2004: New York.

[6] Horrocks, I., et al., *OWL rules: A proposal and prototype implementation*. *Journal of Web Semantics*, 2005. **3**(1): p. 23-40.

[7] Kendall, G.C. *A Web of Rules*. 2003 [citirano 5.11.2006]; Dostopno na: <http://www.xml.com/pub/a/2003/10/23/iswc.html>.

[8] Taveter, K. and G. Wagner, *Agent-Oriented Enterprise Modeling Based on Business Rules*, in *International conference on Conceptual Modeling (ER2001)*. 2001: Yokohama, Japan.

Dejan Lavbič je leta 2004 diplomiral iz računalništva in informatike na Univerzi v Ljubljani, Fakulteta za računalništvo in informatiko, kjer je trenutno zaposlen kot mladi raziskovalec. Na raziskovalnem področju se ukvarja z inteligentnimi agenti, večagentnimi sistemi, ontologijami, odkrivanjem zakonitosti v podatkih in razvojem mobilnih aplikacij.

Marko Bajec je docent na Fakulteti za računalništvo in informatiko Univerze v Ljubljani. Diplomiral je iz računalništva leta 1996. Isto leto se je vpisal na podiplomski študij računalništva. Magistriral je leta 1998, doktoriral pa leta 2001. V okviru Katedre za informatiko se ukvarja z razvojnimi tehnologijami in razvojem ter prenovitvijo informacijskih sistemov. Marko Bajec je član društva Slovenska informatika, združenja AIS (Association for Information Systems) in član programskega odbora posvetovanja Dnevi slovenske informatike.

Marjan Krisper je docent na Fakulteti za računalništvo in informatiko Univerze v Ljubljani in predstojnik katedre za informatiko. Vodi številne projekte razvoja informacijskih sistemov, elektronskega poslovanja in metodologij razvoja informacijskih sistemov v največjih sistemih v gospodarstvu, na državni upravi in javnem sektorju. Je ustanovni član mednarodnega združenja za informacijske sisteme AIS (Association for Information Systems), član izvršnega odbora Slovenskega društva informatika in član Slovenskega društva za umetno inteligenco.

²⁰ SWEET = Semantic WEb Enabling Tools, <http://sweetrules.projects.semwebcentral.org>