Primjena OWL ontologija u tehničkim sustavima

Jukić, Tomislav

Master's thesis / Diplomski rad

2018

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:330313

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-04-23**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU

FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA

Sveučilišni studij

PRIMJENA OWL ONTOLOGIJA U TEHNIČKIM SUSTAVIMA

Diplomski rad

Tomislav Jukić

Osijek, 2018.

SADRŽAJ

1.UVOD 1

1.1 Zadatak diplomskog rada 1

2. RAZVOJ SEMANTIČKOG WEBA 2

2.1. World Wide Web Consortium 3

2.2. Semantički web 3

2.2. Arhitektura semantičkog weba 5

2.3. HTML 7

2.4. XML 8

2.5. RDF 8

2.6. RDFS 9

2.7. OWL 9

3. ONTOLOGIJA 11

3.1.Vrste ontologije 13

3.2. Operacije u ontlogijama 14

3.4. Ontološki jezici 16

4. PRIMJENA OWL ONTOLOGIJE U TEHNIČKIM SUSTAVIMA 18

4.1. Protégé 19

4.2. OWL ontologija kod poslovne montaže 20

4.3. Primjena OWL ontologije u operacijskoj sali 24

4.4. Ograničenja OWL ontologije 26

5. ZAKLJUČAK 27

LITERATURA 28

SAŽETAK 31

ŽIVOTOPIS 33

1.UVOD

U skladu s današnjim ubrzanim razvojem tehnologija, u računarstvu i informatici sve se više spominje pojam „ontologije“. Uz navedeni pojam često se veže više značenja, ali u području računarstva pojam „ontologije“ predstavlja dokument koji sadrži iskaze o međusobnim odnosima određenih pojmova od kojih se nekih sustav sastoji. Ontologija kao pojam svoje podrijetlo ima u filozofiji te u njoj predstavlja fundamentalnu granu metafizike koja se bavi analizom različitih tipova bivanja, a posebno se usmjerava na odnose između onog pojedinačnog i općeg te između esencije i egzistencije kao filozofskih pojmova. Područja primjene i istraživanja ontologije su područja poput inženjerstva znanja, obrade prirodnog jezika, informacijskih sustava i tako dalje. Ontologije su kao grane metafizike usmjerene na identificiranje stvari koje postoje u općenitim terminima. [1][2]

Osim navedenih područja, ontologija se intenzivno koristi u području semantičkog weba koji je opširnije opisan u drugom poglavlju ovog rada. Semantički web predstavlja proširenje trenutnog weba, a informacije koje obuhvaća dobivaju u njemu kvalitetno definirano značenje. Tako se ujedno omogućuje i kvalitetnija suradnja računala i ljudi. Izazov Semantičnog weba je pružanje jezika koji sadržava podatke i pravila za obrazloženje o podacima te dopušta izvoz pravila iz bilo kojeg postojećeg sustava za zastupanje znanja na web. [3]

Povećanjem veza i različitih međudjelovanja računala, ontološki jezici su postajali sve potrebniji jer su njihovi zapisi razumljivi svim računalima. Poglavlje o razvoju semantičkog weba prikazuje i opis značajnih ontoloških jezika poput HTML-a, XML-a, RDF-a, RDFS-a i OWL jezika.

Nakon opisane ontologije i njenih najistaknutijih svojstava te ontoloških jezika, slijedi poglavlje o primjeni OWL ontologije u tehničkim sustavima. Iako OWL ontologije nije značajno korištena u tehničkim sustavima, poglavlje prikazuje dva primjera moguće primjene; usko povezane s robotikom. U poglavlju su opisana ograničenja primjene OWL ontologije, a potom slijedi zaključak ovoga rada.

1.1 Zadatak diplomskog rada

Zadatak diplomskog rada potrebno je opisati što je to ontologija, uloga semantičkog weba za OWL ontologiju i navesti neke primjere OWL ontologije za tehničke sustave.

2. RAZVOJ SEMANTIČKOG WEBA

Informacija kao objekt računarstva podrazumijeva obradu informacija, koja jednako uključuje transformaciju i prijenos podataka. Naravno, informacija podrazumijeva komunikaciju, jer ona predstavlja glavni smisao postojanja informacija. Računala su predstavlja informatičke otoke, a unutar njih su tekle informacije koje su završavale na papirima, magnetskim trakama ili terminalskim zaslonima. Povećanjem potreba čovječanstva, napretkom društva i tehnologije polako se razvijao i internet.

Internet je globalni informacijski sustav logički povezan jedinstvenim sustavom adresiranja putem internet protokola koji povezuje računala i računalne mreže. Internet je nastao 1969. godine u SAD-u u sklopu istraživanja, a zvao se ARPANET (engl. *Advanced Research Project Agency*). ARPANET predstavlja glavni katalizator razvoja drugih mreža i konačno današnjeg Interneta. [4]

Računalni web imao je svoje početke u projektu iz 1990.godine, gdje je Tim Bernes-Lee (Sl. 2.1.), engleski inženjer i znanstvenik je predložio projekt razvoja weba. U sklopu weba postoje i standardi koji predstavljaju skup smjernica i preporuka za ispravno kreiranje HTML, CSS i XML koda s ciljem nesmetanog i jednostavnijeg pristupa internet tehnologijama, a za međusobnu komunikaciju računala koriste komunikacijske protokole. [5]

Sl. 2.1. *Tim Bernes-Lee* [5]

2.1. World Wide Web Consortium

Tijelo koje nadgleda razvoj standarda i brine se da stranice izrađene u tom obliku budu što dostupnije korisnicima je World Wide Web Consortium (W3C). Uz već spomenute tehnologije HTML, CSS, XML W3C je razvio i OWL standard koji omogućuje daljnji razvoj weba, a teži semantičkom webu. Temeljne osobine semantičkog weba su pristupačnost i jednostavnost jer je i sam cilj weba razvoj svijeta informacija kojemu svatko može doprinositi vlastitim idejama i informacijama. Tako nastaje i velika mreža informacija koja je mješavina teksta, analiza, različitih prezentacija i multimedijskih datoteka koje će određene osobe imati priliku objavljivati na web stranicama. Naravno, unatoč pristupačnosti i jednostavnosti dolaska do informacija putem web stranica, postoji i nedostatak koji se krije u nedosljednosti informacija, neintegriranosti i slabog povezanosti. Upravo se zbog toga postavlja i važno pitanje razlikovanja točnih i netočnih podataka, potrebnih i nepotrebnih informacija. [3]

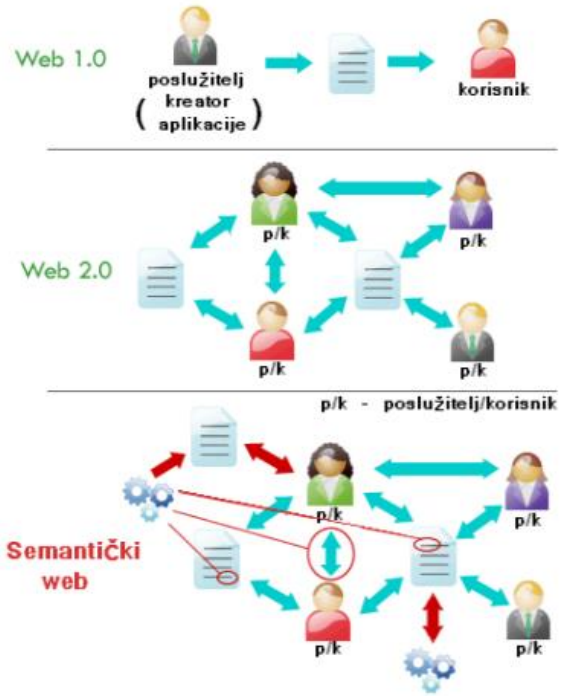
Inovacije u sklopu web stranica razvijaju se svakodnevno. Gotovo svaku informacijsku tehnologiju moguće je implementirati u sklopu web stranice. Povećanjem korisnika internetskih stranica te razvojem dostupnosti Interneta širom svijeta postavlja i potrebu za izgradnjom infrastrukture weba koje će moći odgovoriti sve većim zahtjevima tehnologije i društava. Budući da je to teško ostvarivo, iz dana u dan se razvijaju inteligentnije aplikacije. [4]

2.2. Semantički web

World Wide Web je tehnologija koja ima samo nekoliko godina, ali rast i utjecaj na društvo unutar kojega je ugrađen bili su zapanjujući. Semantički web nije zasebni web već proširenje trenutnog, u kojem se informacijama daje kvalitetno definirano značenje, što ujedno omogućuje i kvalitetniju suradnju računala i ljudi. Izazov Semantičnog weba je pružanje jezika koji sadržava podatke i pravila za obrazloženje o podacima te dopušta izvoz pravila iz bilo kojeg postojećeg sustava za zastupanje znanja na web. Logika sustava treba biti dovoljno moćna za opisivanje složenih svojstava predmeta. [3]

Dvije važne tehnologije za razvoj semantičkog weba su već na snazi: XML (engl. *eXtensible Markup Language*) i RDF (engl. *Resource Description Framework*). Izvorni (klasični, sintaktički), odnosno web kakav je u najvećoj mjeri poznat i danas, predstavlja tek jednu od usluga na Internetu. Zamišljen je kao informacijsko-komunikacijski medij za prezentaciju, odnosno razmjenu informacija. Napravljen je pomoću tehnologija kao što su HTML, CSS, JavaScript, PHP, MySQL i sl. Sve te informacije bile su kreirane i kreiraju se još i danas s ciljem da ih koriste primarno ljudi.

U tome leži jedan od glavnih nedostataka sintaktičkog weba: informacije su prilagođene ljudima i kao takve nisu razumljive računalima. Nasuprot tome, semantička tehnologija omogućuje smještaj informacija na mreži u formi djelomično razumljivoj računalima. Zamišljeno je da semantički web bude ekstenzija sintaktičkog, zadržavajući pritom sva ranije implementirana svojstva, standarde i funkcionalnosti. Semantička tehnologija zasniva se na semantici podataka. Semantika (grčki *semantikos*, koji daje znakove, značajan, znak) se odnosi na aspekte značenja koji su izraženi u jeziku, odnosno kodu ili nekom drugom obliku predstavljanja. [6] [7]

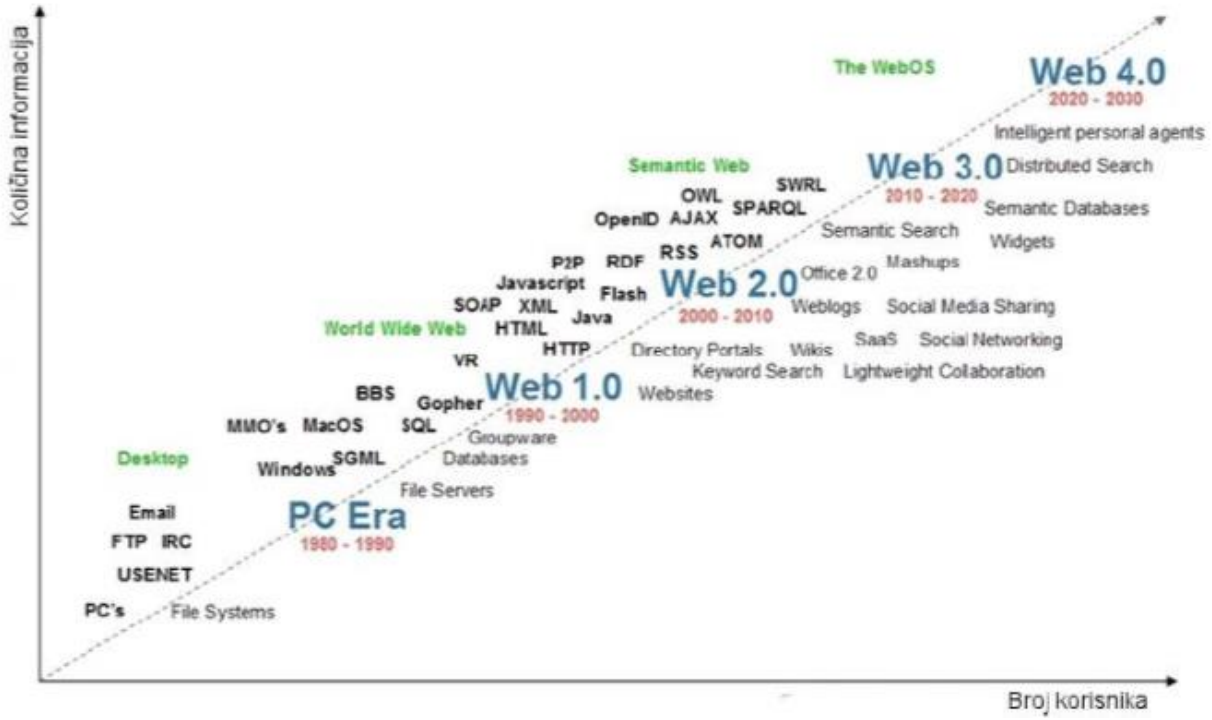


Sl. 2.2. *Usporedba web generacija* [37]

Na slici iznad (Sl. 2.2.) grafički je predočena razlika između web generacija i semantičkog weba što se tiče čitanja i obrade podataka. Prva web generacija ima poslužitelja koji postavlja podatke ili dokumente na web koje samo on može mijenjati ili nadograđivati, a korisnici navedenog weba imaju samo uvid u podatke (engl. *read only*).

Drugu generaciju možemo definirati kao revoluciju na koji način koristimo web. Sve aplikacije kao što su blogovi, podcastovi ili RSS feedovi spadaju pod novu generaciju aplikacija na Internetu na koje se može odnositi kao na Web 2.0 aplikacije. Dolaskom druge generacije weba postalo je moguće da svatko ima uvid u postavljene podatke koje ovisno o funkciji mogu mijenjati i obrađivati, time je svatko postao poslužitelj i korisnik.

Temelj Berners-Leeove ideje semantičkoga weba je u tome da će strojno obradivi metapodaci biti osnova za sljedeći naraštaj te da će se na taj način olakšati pristup informacijama i pristup komunikacijama. Smisao semantičkog weba je to da se Internet može pretraživati ne samo korištenjem riječi nego i kroz upotrebu značenja. Kod semantičkog weba, uz navedena svojstva druge web generacije i to što su podaci međusobno povezani, podaci koji su postavljeni na web su i strojno čitljivi te ih računala mogu samostalno mijenjati ili nadopunjivati. [8] [9]

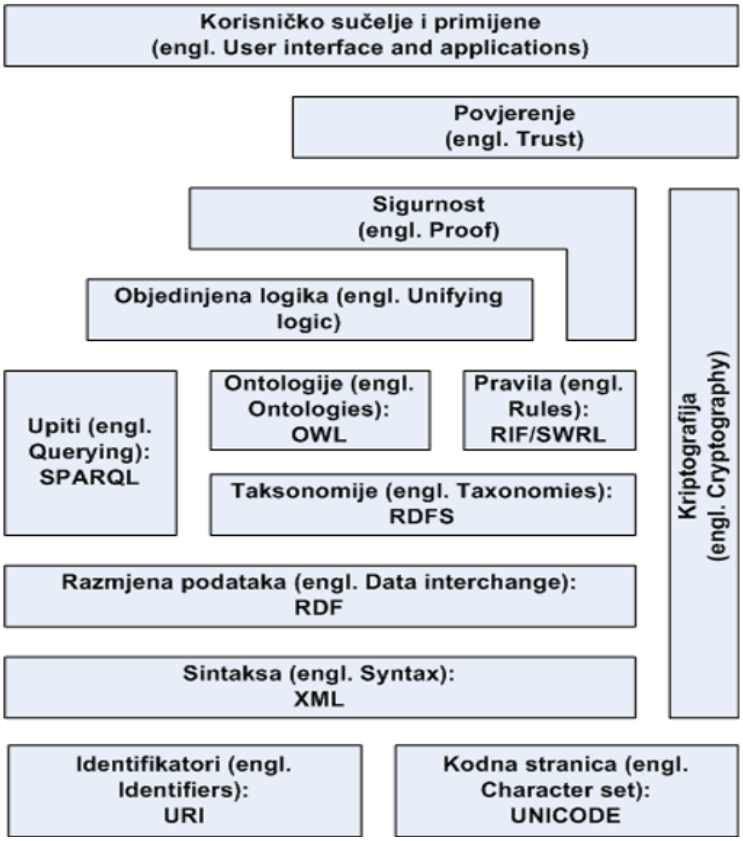


Sl. 2.3. *Razvoj weba* [37]

2.2. Arhitektura semantičkog weba

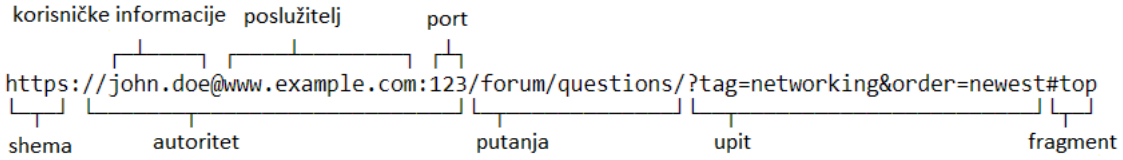
Osnovna ideja semantičkog weba je boljom standardizacijom metapodataka pomoći korisnicima u pronalaženju podataka na webu. Principi koje koristi semantički web su implementirani u slojeve na način da se svaki novi sloj gradi na već postojećem. Slojevi sadržavaju standarde i regulative koje definiraju različite aspekte tehnologije semantičkog weba, a prilikom razvoja jednog sloja koriste se dva osnovna principa.

Prvi princip podrazumijeva da će programi koji koriste određeni razvijeni sloj biti u mogućnosti interpretirati informacije koje su pisane za sloj koji se nalazi na razini ispod. Drugi princip govori o djelomičnom razumijevanju slojeva koji se u hijerarhiji nalaze iznad; odnosno postavlja se zahtjev djelomičnog razumijevanja informacije koje postoje na tim slojevima. Na taj se način zadržava kompatibilnost te se ostvaruje povezanost udaljenih slojeva. [10][11]



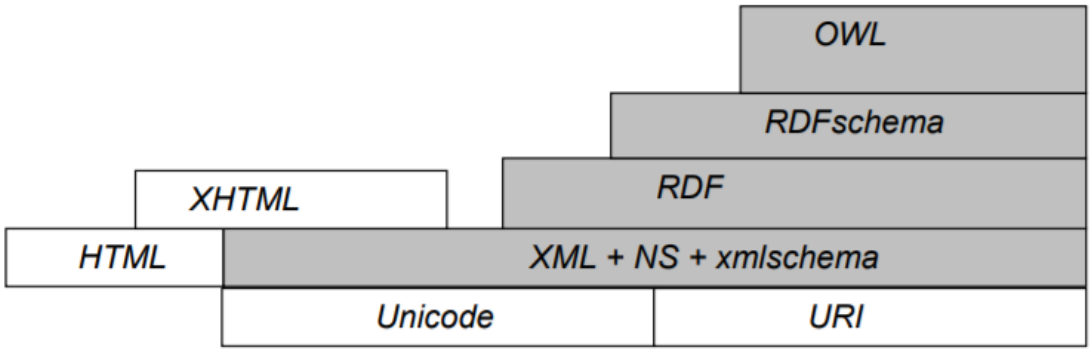
Sl. 2.4. *Slojevi semantičkog weba s pripadajućim tehnologijama* [7]

Na dnu prikazane strukture slojeva (Sl. 2.3.) nalaze se UNICODE i URI (engl. *Uniform Resource Information*) sloj te oni brinu da korisnici koriste međusobno definirane postave znakova te pomažu u identifikaciji objekata unutar semantičkog weba, a predstavlja ujedno i njegove temelje. Na slici (Sl. 2.4.) ispod je prikazan primjer URI-ja s njegovim dijelovima. [7][10][11][12]



Sl. 2.5. *Primjer jedinstvenog identifikatora resursa i njegovih dijelova* [12]

Osim slojeva, postoje i jezici u sklopu semantičkog weba (Sl. 2.6.). Do razumijevanja semantike dolazi se korištenjem programskih jezika kojima se određuje semantika podataka i njihovih izvora.



Sl. 2.6. *Prikaz i odnos jezika semantičkog weba* [38]

2.3. HTML

HTML je kratica za HyperText Markup Language. On nam omogućava predstavljanje podataka i njihov izgled na webu te opisuje kako neka informacija izgleda na web stranici, tj. označava prezentacijski jezik za izradu web stranica. Hipertekst dokument se stvara pomoću HTML jezika, a zatim se HTML jezikom oblikuje sadržaj te se stvaraju hiperveze hipertext dokumenta. Temeljna pozitivna značajka HTML-a su jednostavnost upotrebe, lako učenje, te široka dostupnost korisnicima i upravo te značajke su razlog njegove opće prihvaćenosti i popularnosti. Temeljna zadaća HTML jezika jest uputiti web preglednik kako prikazati hipertekst dokument, a teži se da taj dokument izgleda jednako bez obzira o kojemu je web pregledniku, računalu i operacijskom sustavu riječ. [11] [13]

2.4. XML

XML (engl. *Xtensible MarkUP Language)* je proširivi jezik za označavanje koji određuje strukturu podataka. XML omogućava autorima da kreiraju vlastito označavanje, sintaksu, koji u sebi nosi dio semantike. Ideja stvaranja XML-a bila je stvoriti jedan jezik koji će biti jednostavno čitljiv i ljudima i računalnim programima. Princip realizacije je vrlo jednostavan: odgovarajući sadržaj treba se uokviriti odgovarajućim oznakama koje ga opisuju i imaju poznato, ili lako shvatljivo značenje. Format oznaka u XML-u vrlo je sličan formatu oznaka u npr. HTML jeziku. XML jezik se danas koristi i različite svrhe poput odvajanja podataka od prezentacije, razmjenu podataka, pohranu podataka, povećavanje dostupnosti podataka te izradu novih specijaliziranih jezika za označavanje. [11][14] [15]

2.5. RDF

RDF (*eng. Resource Description Framework*) je centralni protokol na webu zasnovan kao W3C standard, opisuje semantičke veze između elektronskih izvora. Treba naglasiti da RDF nije jezik već model podataka s mogućnošću predstavljanja podataka na eksplicitniji način nego što je moguće XML-u. Osnovna konstrukcija RDF-a je izjavna rečenica koja se sastoji od tri dijela, subjekta, predikata i objekta tj. njihov zajednički naziv je triplet. Triplet nam služi za opis semantičke veze između različitih resursa. Jedna od osnovnih točaka prednosti u odnosu na XML su izjavne rečenice koje omogućuju opisivanje bilo koje domene, a mogu se podijeliti po poslužiteljima te se mogu razmjenjivati između heterogenih aplikacija bez gubitka značenja. Upravo tako se stvara velika virtualna baza znanja. Subjekt podrazumijeva element na koji se u rečenici ukazuje, a predstavlja osnovni blok RDF-u. Predikat je svojstvo koje je pripisano subjektu, a podrazumijeva poseban tip resursa kojemu je temeljna svrha opisivanje atributa nekog resursa. Pri tome atribut označava odnos između dva resursa. Ostaje nam još objekt, on je zaslužan za predstavljanje vrijednosti koja je dodijeljena subjektu, te kao takav može sadržavati neke od dostupnih jezičnih identifikatora kao i dodatne identifikatore tipa podatka.[11][13][16] [17]

2.6. RDFS

RDFS (engl. *Resource Description Framework Schema*) predstavlja osnovni alat za oblikovanje hijerarhijske strukture web objekata, a služi nam za opisivanje veza za razne tipove objekata (engl. *Classes*), odnosno hijerarhijske odnose između njih (engl. *subClasses*), svojstva koji opisuju objekt (engl. *Properties*), odnose između njih (engl. *subProperty*) te restrikcije domena (*engl. domains*) i raspona (engl. *ranges*). Sustav klasa u RDFS-u obuhvaća jednostavan princip nasljeđivanja na temelju uključenih podataka. Ukoliko je jedna klasa podklasa druge znači da je instanca jedne klase ujedno i instanca druge klase. RDFS jezik se ispisuje pomoću RDF-a, koji je osnova ostalim ontološkim jezicima što znači da se sve shematske informacije prikazuju u obliku RDF trojki. Prema navedenom sve shematske informacije klase, podklase, svojstva, podsvojstva, domena, raspon prikazuju se pomoću RDF trojki. Unutar njegovog okvira, imamo mogućnost deklariranja nove klase i naseliti ih primjerima, ali se ne može reći ništa dodatno o tim klasama i primjerima. Ne mogu se postaviti aksiomi ili pravila zaključivanja koja se na njih odnose. Iz svega navedenog možemo zaključiti da je RDFS jezik proširenje RDF jezika dodatnim unaprijed definiranim tipovima i svojstvima, te pomoću njih se stvaraju izjave koje opisuju nove tipove, njihova svojstva i relacije. Međutim, prilikom opisivanja kompleksnijih odnosa između web objekata, danas se koristi zaključivanje temeljeno na logici što je glavna karakteristika OWL-a (engl. *Ontology Web Language*). [11][18] [19]

2.7. OWL

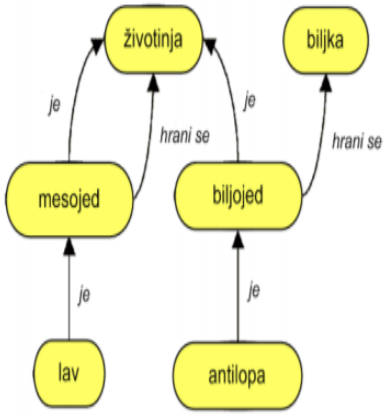
OWL (engl. *Web Ontology Language*) je jezik na najvišoj razini, a njegova je uloga formalno opisati značenje terminologije koja se koristi u Web dokumentima. Razvijen je kao sljedbenik iz RDF i RDFS-a, a baziran je na formalnoj semantici XML-a. OWL je standardni Web jezik za opisivanje ontologija koji je razvijen kao evolucija jezika OIL, DAML i OIL+DAML. OWL jezik, slično kao i RDF Schema jezik, omogućava definiranje tipova resursa, njihovih svojstava i veza te u odnosu na RDF Schema jezik, OWL također može ostvarivati definiranje karakteristika svojstava, ograničavanje vrijednosti svojstva putem tipova resursa, ograničavanje broja pojavljivanja svojstva, izražavanje jednakosti svojstava, definiranje novih tipova resursa operacijama nad skupovima postojećih resursa, nabrajanjem resursa, istovjetnošću te razlikom tipova resursa. Osim toga OWL definira i nekoliko predikata za izražavanje jednakosti te različitosti pojedinih resursa, a definira i način dohvata OWL-om izražene ontologije. Navedene dodatne mogućnosti OWL jezika također pružaju i priliku izgradnje detaljnijih i složenijih

ontologija koje će moći bolje opisivati svojstva resursa, tipove i veze među resursima te postupke zaključivanja koji koriste ontologiju. OWL je definiran kao tri različita podjezika: OWL Full, OWL DL i OWL Lite. [20]

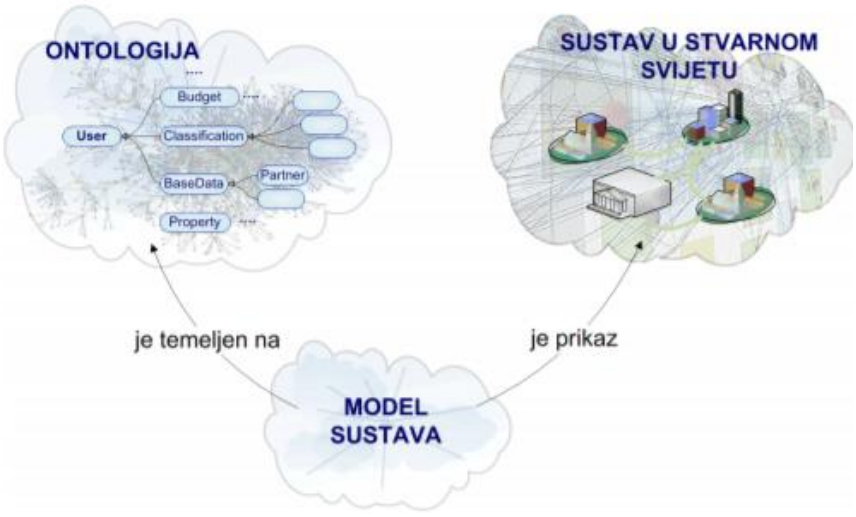
- OWL Full je najizraženiji jezik od navedena tri koji upotrebljava osnovni OWL jezik te također omogućuje kombiniranje s RDF-om i RDFS-om. Najveća prednost ovog jezika je njegova kompatibilnost s RDF-om i RDFS-om, no njegov najveći nedostatak je njegova kompleksnost koja dovodi do nemogućnosti odlučivanja. [20]
- OWL-DL je podjezik OWL Full-a baziran na deskriptivnoj logici te je mnogo elokventniji od OWL Lite-a. Deskriptivna logika je temeljni element logike prvog reda te je kao takva podložna automatskom zaključivanju, a upravo zbog toga je omogućena izrada klasifikacijske hijerarhije i ostvarenje provjere dosljednosti ontologije prema OWL-DL-u. On sadržava ograničenja u smislu upotrebe konstrukata iz OWL-a i RDF-a. Njegova temeljna prednost je ostvarenje efikasne podrške razumijevanja, dok je njegov temeljni nedostatak nepotpuna kompatibilnost s RDF-om. [21][22]
- OWL Lite predstavlja jezik niže složenosti, a njegova osnovna namjena je korištenje u situacijama s jednostavnim hijerarhijama pojmova i jednostavnim odnosima među pojmovima. On ima veliko ograničavanje zbog pojednostavljenosti sintakse, a koristi se većinom u aplikacijama s dovoljnom minimalnom ontologijskom ekspresivnošću. Prednost je jezik koji je lakši za dosegnuti korisnicima i lakša implementacija. [11][21][22]

3. ONTOLOGIJA

Ontologija kao pojam svoje podrijetlo ima u filozofiji te u njoj predstavlja fundamentalnu granu metafizike koja se bavi analizom različitih tipova bivanja, a posebno se usmjerava na odnose između onog pojedinačnog i općeg te između esencije i egzistencije kao filozofskih pojmova. Ontologije kao pojam potječe iz grčkih riječi *ontos* („biti“, „biće“, „bivstvujuće“) i *logos* („riječ/govor“), a nastao je u 17. stoljeću kao dio filozofske discipline. Navedena filozofska disciplina veže se uz Aristotela koji je prvi iznio definiciju o ontologiji kao metafizici koja se bavi različitim tipova bivanja, odnosima prirode stvari i postojanja. Ontologija osigurava precizan rječnik koji može biti korišten prilikom predstavljanja znanja. Predstavljanje znanja podrazumijeva formalizaciju sakupljenog znanja u obliku u kojem je ono pogodno za uporabu. [1][2]



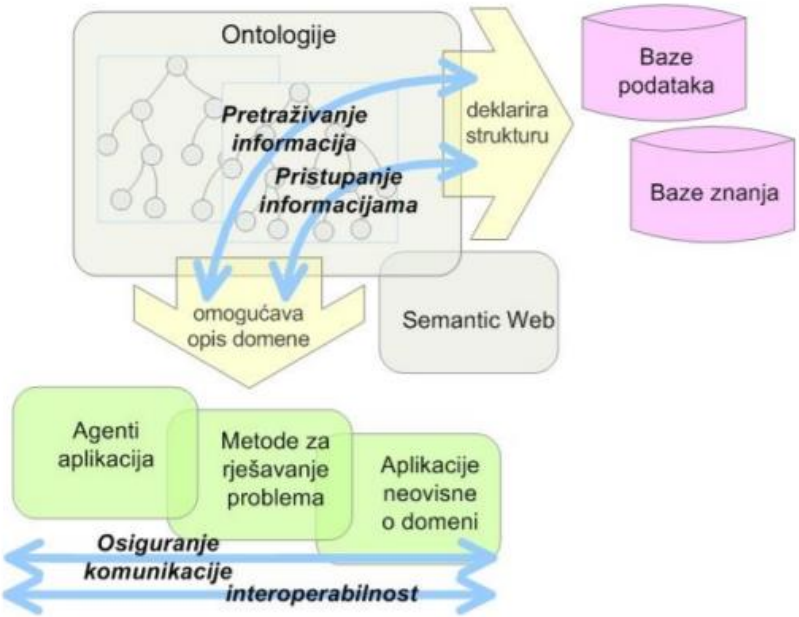
Sl. 3.1. *Jednostavni primjer modela ontologije živih organizama* [24]



Sl 3.2. *Model sustava temeljen na ontologiji* [24]

Ontologija ima svoje ključne pojmove; individue kao početne razine i osnovne objekte; klase kao zbirke ili tipove objekata; atribute kao pripadajuća svojstva, pojave, karakteristike ili parametre koje objekt može imati ili distribuirati; te odnose kao način na koji se objekti odnose jedni prema drugima. Individue su osnovana, početna razina sastavnica ontologije te one mogu obuhvaćati konkretne objekte poput ljudi, životinja, molekula, planeta, kao i apstraktne individue poput brojeva i riječi. Klase kao apstraktne grupe, skupine ili zbirke objekata mogu obuhvaćati individue, druge klase ili kombinaciju jednih i drugih. Atributi se mogu pridruživati objektima u ontologijama te pri tome atribut ima određeno ime i vrijednost te se koristi za pohranu informacija koje su specifične za objekte na koje prianjaju. Korištenje atributa omogućava opisivanje odnosa ili relacije između objekata u ontologijama, a odnos ili relacija pak predstavlja atribut čija je vrijednost drugi atribut u ontologijama. U tome je sadržana glavna važnost ontologije jer tako možemo točno opisati te odnose. [2]

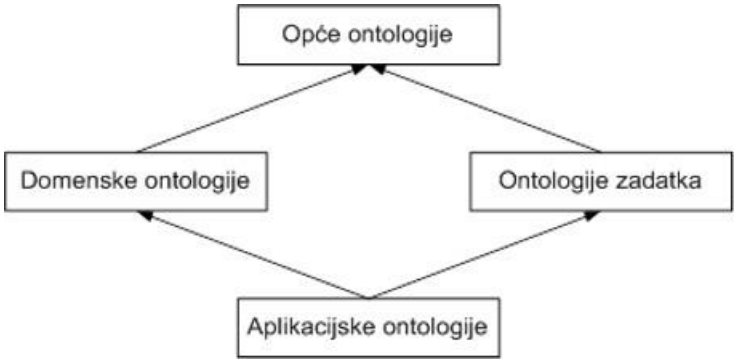
Danas stvarni praktični doprinos ontologije treba sagledati u internet okruženju gdje međudjelovanje korisnika usluga web-a i pružatelja usluga preko web-a treba dostići razinu na kojoj je informacijski prostor web-a dovoljno računalno čitljiv da pruži korisniku precizan prikaz onoga što korisnik traži. [23]



Sl. 3.3. *Uloga ontologije i semantičkog Weba* [24]

3.1.Vrste ontologije

Ontologije se mogu podijeliti na vrste s obzirom na uloge koje ispunjavaju u radu i s obzirom na razine općenitosti kojima su određene. Tako ontologije možemo podijeliti na: domenske ontologije, meta ontologije, općenite ontologije, reprezentacijske ontologije te metodološke ontologije, ali uobičajena podjela ontologija je prema stupnju ovisnosti o zadatku kako je prikazana na slici ispod (Sl. 3.4.).



Sl. 3.4. *Podjela ontologija prema stupnju ovisnosti o zadatku* [40]

Opće ontologije (engl. *top-level*) omogućuju pohranjivanje znanja o svijetu, te definiranje općenitih koncepata i oznaka za objekte poput prostora, stanja, vremena i događaja su zbog svoje visoke razrađenosti dijele se na nekoliko općih ontologija koje se često koriste u istraživanjima BFO (Basic Formal Ontology), SUMO (Sugested Upper Merged Ontology), PROTON (PROToONtology), DOLCE (Descriptive Ontology for Linguistic and Cognitive Engineering), Sowa i GFO (General Formal Ontology). [11][24][25]

Domenske ontologije (engl. *domain ontology*) njihova glavna značajka je prikupljanje znanja iz određene uske domene te se oblikuje specifična domena ili dio svijeta. Domenske ontologije mogu se ponovno upotrebljavati samo u domeni koju obuhvaćaju, kao na primjer u medicini ili računarstvu. Prilikom izmjene u samoj ontologiji, navedena izmjena imat će utjecaj na sve komponente koji su usko povezani s traženim podatcima zbog preciznosti domenske ontologije. [11][24][25]

Ontologije zadatka (eng. *task ontology*) su ontologije nastale specijalizacijom koncepata opisivanjem skupa definicija ili koncepata koji se tiču zadatka iz općih ontologija te se na taj način ostvaruju predlošci koji su potrebni za stvaranje modela procesa. Ontologije zadatka mogu biti

vezane uz ontologiju domene jer ontologija zadatka sadržava znanje potrebno za ostvarenje određenog zadatka, a ontologija domene opisuje znanje o primjeni određenog zadatka [26]

Aplikacijske ontologije su ontologije koje ovise o aplikacijama koje ih upotrebljavaju. Sadrže sve definicije potrebne za oblikovanje modela znanja nužnog za određenu aplikaciju. Aplikacijske ontologije često proširuju ili specijaliziraju koncepte iz vokabulara domenskih ontologija ili ontologija specifičnih zadataka. [26]

Prema Đuraković i Verić [27] jedna od temeljnih prednosti ontologija je koherentna navigacija koja omogućuje kretanje od jednog koncepta do drugog. Također, ontologiju odlikuju i fleksibilne ulazne točke, a podrazumijeva praćenje i povezivanje određenih perspektiva s povezanim konceptima. Ontologije obuhvaćaju i veze koje ističu relevantne informacije bez obzira na prethodno znanje o domeni i terminologiji domene. Osim toga, ontologije mogu predstavljati bilo koje oblike informacija od nestrukturiranih do polustrukturiranih i strukturiranih podataka. Jedna od prednosti ontologija je i povezivanje koncepta te integracija sadržaja pravilnim povezivanjem i mapiranjem koncepata. Navedeni autori naglašavaju i nedostatke ontologija, a to su; nepoznavanje novih koncepata te sukladno tomu i veći broj grešaka, zatim problemi mapiranje te problemi gubitka semantike nakon transformacije.

3.2. Operacije u ontologijama

Moguće je da jedna aplikacija koristi više ontologija, najčešće u slučaju kada se koristi modularni dizajn ontologija ili kada slijedi integracija sa sustavima koji koriste neke druge ontologije. [28]

Spajanje ontologija (engl. *merge*) označava oblikovanje nove ontologije povezivanjem postojećih ontologija. Uobičajeni uvjet je da nova ontologija obuhvaća sva znanja iz originalnih ontologija, ali taj uvjet ne mora biti u potpunosti zadovoljen jer postoji mogućnost da izvorne ontologije nisu potpuno konzistentne. Tada nova ontologija uvodi nove koncepte i relacije koji će poslužiti a povezivanje pojmova iz izvornih ontologija. [28]

Mapiranje (engl. *Mapping*) jedne ontologije u drugu je izražavanje načina na koji se prevodi izjava s jedne ontologije na drugu, a to najčešće označava prevođenje između koncepata i relacija. Najjednostavniji slučaj navedenoga podrazumijeva mapiranje iz jednog koncepta prve ontologije na koncept druge ontologije, te nije uvijek moguće ostvariti takav način mapiranja jer se mogu izgubiti određene informacije. [28]

Usklađivanje (engl. *Alignment*) je proces mapiranja ontologija u oba smjera; dok je moguće modificirati izvorne ontologije tako da postoji odgovarajući prijevod koji će ujedno smanjiti i mogućnost gubitka informacija. Usklađivanje je ostvareno kada ontologije pokrivaju domene koje su komplementarne jedna drugoj. [28]

Sljedeća operacija nosi naziv rafiniranje (engl. *Refinement*), a predstavlja mapiranje ontologije A u ontologiju B. To se ostvaruje na način da svaki koncept ontologije A ima ekvivalent u ontologiji B, no primitivni koncepti s ontologije A mogu odgovarati neprimitivnim, to jest određenim konceptima ontologije B. [28]

Ujedinjenje (engl. *Unification*) je operacija koja podrazumijeva usklađivanje svih koncepata i relacija u ontologijama. Pri tome zaključivanje u jednoj ontologiji može biti mapirano na zaključivanje u drugoj ontologiji i obrnuto. Ujedinjenje se obično ostvaruje kao proces usavršavanja ontologija u oba smjera. [28]

Integracija (engl. *Integration*) je proces traženja istih dijelova dviju različitih ontologija A i B. Tako se omogućuje razvijanje nove ontologije C koja omogućuje prevođenje između ontologija A i B. Upravo to stvara i interoperabilnost između dva sustava gdje jedna koristi ontologiju A, a drugi ontologiju B. Nova ontologija C može zamijeniti ontologije A i B, a može se koristiti i kao umjetni jezik između ove dvije ontologije. [28]

Nasljeđivanje (engl. *Inheritance*) znači da ontologija A nasljeđuje sve od ontologije B, sve koncepte, relacije i ograničenja ili aksiome, a pri tome ne postoji nekonzistentnost uvedena dodatnim znanjem koje je sadržano u ontologiji A. Ova operacija djelomično definira poredak između ontologija. [28]

Sve navedene operacije nije moguće napraviti za sve ontologije jer su to vrlo teški zadaci koji se često ne mogu automatski riješiti zbog određenih razloga, kao na primjer zbog neodlučnosti pri korištenju ekspresivnih logičnih jezika ili pak zbog nedovoljne specifikacije ontologije koja nije dovoljna za utvrđivanje sličnosti s nekom drugom ontologijom. Upravo zbog toga se zadaci rade ručno ili poluautomatski. Na taj način stroj pomaže u pronalaženju mogućih odnosa između elemenata različitih ontologija. Naravno, konačnu potvrdu odnosa donose ljudi koji odlučuju na temelju prirodnog jezika opisa ontoloških elemenata ili samo na temelju prirodnih jezika imena ontoloških elemenata. [28]

3.4. Ontološki jezici

Prema Juriću [26] osnovno učenje ontologije predstavlja automatsko kreiranje i otkrivanje ontologija pomoću tehnika iz područja strojnog učenja ili otkrivanja znanja. Jezici za prezentacije ontologija nalaze se između prirodnog jezika koji je razumljiv ljudima i formalnog jezika koji je razumljiv strojevima te je u svrhu rješavanja navedenog problema nužno omogućavati ljudima daljnje korištenje prirodnog jezika na način da se automatski mapiraju riječi i izrazi iz prirodnog jezika koji je razumljivi ljudima u ontološke koncepte.

Jezici koji su temeljni na logici se mogu podijeliti na jezike temeljene na predikatnoj logici prvog reda, jezike temeljene na logici baziranoj na pravilima te jezike temeljene na deskriptivnoj logici.

Jezici su zasnovani na predikatnoj logici prvog reda, čije proširenje je i način gradnje ontoloških jezika. Primjer ontoloških jezika prvog reda bliskih predikatnoj logici su CycL i KIF.

CycL je formalni jezik s sintaksom naslijeđenom od predikatne logike prvog reda te on kao takav proširuje proširuje predikatnu logiku prvog reda konceptima drugog reda: tipiziranjem, reifikacijom i mikroteorijama. [29]

KIF (engl. *Knowledge Interchange Format*) je jezik razvijen s ciljem omogućavanja razmjene znanja među udaljenim računalnim sustavima te se koristi za izražavanje i razmjenu ontologija, a sastoji se od konstanti, definicija i rečenica. Ono što je karakteristično za KIF je korištenje četiri različita tipa konstanti: objektni, funkcijski, relacijski i logički. [29]

Jezici temeljni na deskriptivnoj logici obuhvaćaju jezike koji su dodatno podijeljeni na jezike temeljene isključivo na deskriptivnoj logici, hibridne jezike te jezike semantičkog web-a. Jezici temeljeni isključivo na deskriptivnoj logici sadržavaju dvije temeljne komponente, Tbox, Abox i RBox. TBox je dio baze znanja koja sadržava definicije koncepata i uloge, a ABox pak sadržava tvrdnje o individualnim pojedincima i relacijama te predstavlja dio baze znanja koji sadržava informacije o konkretnim entitetima i odnosima koje trenutačno promatramo, dok je u TBox pohranjeno znanje o nekoj domeni koju promatramo. Rbox obuhvaća znanje o svojstvima definiranih svojstava. [11]

Jezici semantičkog web-a su primjerice OIL (engl. *Ontology Inference Layer*), DAML-OIL (engl. *DARPA Agent Markup Language- OIL*) te OWL kao jedan od najpopularnijih jezika. OWL jezik smo spomenuli u prethodnom poglavlju i on je ostvaren s ciljem formalnog pisanja značenja terminologije upotrebljene u web dokumentima te teorijsku osnovu za OWL čine deskriptivne logike kao formalni jezici za predstavljanje znanja. [11][20]

3.5. Ontološke veze

U otvorenom okruženju agenti mogu imati korist u nekim aplikacijama na temelju prikupljenog znanja o postojećim vezama (engl. *Relationships*) između nekih ontologija. Znanje o postojećim vezama tako može koristiti u donošenju odluke o načinu i početku komunikacije s drugim agentima. No, čak i ako u svaki agent može vjerovati takvim vezama, ontologijski agent ima najadekvatniju ulogu za razumijevanje toga te tako mogu biti ispitane vrijednosti veza te se može koristiti za prijevod ili efikasniji izbor zajedničke ontologije za komunikaciju između agenata.

U tu svrhu se koristi sljedeći predikat:

onto-veza ?O1 ?O2 ?razina

koji je definiran kao istinit što ujedno znači da kada postoji veza razine (*eng. level*) između dvije ontologije u argumentima O1 i O2. Tada razina argumentacije može pretpostaviti jednu od vrijednosti navedenih u tablici ispod.

Ekstenzija (engl. *extension*)	kada O1 proteže ontologiju O2
Identičan (engl. *identical*)	kada su obje ontologije identične
Ekvivalent (engl. *equivalent*)	kada su obje ontologije ekvivalentne
Jako-predvidiva (engl. *strongly-translatable*)	kada je izvorna ontologija O1 jako predvidiva na ciljanu O2 ontologiju
Slabo-predvidiva (engl. *weakly-translatable*)	kada je izvorna ontologija O1 slabo predvidiva na ciljanu O2 ontologiju
Približno-predvidiva (engl. *approx-translatable*)	kada je izvorna ontologija O1 približno predvidiva na ciljanu O2 ontologiji

Tab. 3.1. *Ontološke veze* [30]

Kada je odnos utvrđen, možemo ga koristiti za određivanje ontologije u ovisnosti o svrsi. [30]

4. PRIMJENA OWL ONTOLOGIJE U TEHNIČKIM SUSTAVIMA

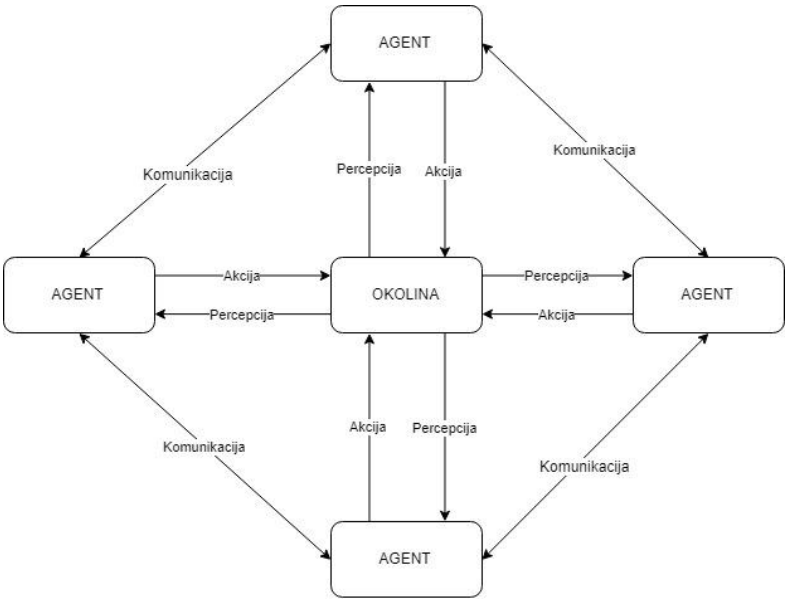
OWL ontologije u tehničkim sustavima nisu značajno zastupljene niti korištene, ali postoje određene iznimke i područja ljudske djelatnosti u kojima je primjenjiva, a neki od primjera navedenoga su opisani u ovom dijelu rada. Budući da je ljudski rad sve više prožet idejom stvaranja zamjene čovjeka u određenim područjima ljudskih djelatnosti, OWL ontologija svoju primjenu pronalazi u području robotike vezano uz tehničke sustave. Razvoj robotike potaknut je društvenim nastojanjem stvaranja zamjene čovjeka koja će moći oponašati ljudska svojstva u međudjelovanju s ljudskom okolinom.

Robotika je primijenjena tehnička znanost koja predstavlja spoj strojeva i računarske tehnike. Ona je interdisciplinarna znanost koja obuhvaća područja poput mehanike, elektronike, informatike i automatike. Danas je robotika brzo rastuće znanstveno polje te je posebno važno stvaranje agentske tehnologije koja je ujedno usko povezana s područjem umjetne inteligencije koja je prvenstveno vezana za agentsku tehnologiju i višeagentske sustave. [31]

Agentska tehnologija od ključne je važnosti u pogledu interakcije čovjeka i računala, a osim pojma agent često se upotrebljavaju i pojmovi poput: autonomni agent, softverski agent, inteligentni agent, ili nešto specifičniji pojmovi virtualni agent, mobilni agent, informacijski agent te agent za korisničko sučelje. [38] Temeljno svojstvo agenta je autonomnost koja uključuje i socijalnu sposobnost (interakciju s drugim agentima), reaktivnost (agenti mogu opažati okolinu i odgovarati na promjene okoline) i proaktivnost (djelovanje u smislu ispunjavanja ciljeva); mentalne koncepte koji se obično primjenjuju na ljude (vjerovanje, želja, namjera, znanje, emocije). [32]

Sl. 4.1. *Automatizirana montažna linija* [7]

Interakcija agenta i okoline uključuje prikupljanje informacija iz okoline koje se u nekom određenom vremenu mijenjaju, pri tome agent snima trenutačno stanje. Informacije se prikupljaju pomoću dostupnih senzora, na temelju prikupljenih informacija vrši se obrada i zaključivanje te generiranje prikladne akcije agenta u skladu sa zadanim ciljem. Akcija se šalje izvršnom elementu (akcijski element) koji vrši generirane akcije u okolini. Agent pomoću subjekta zaključivanja, učenja, planiranja, prikupljanja znanja i drugih metoda umjetne inteligencije u određenoj mjeri nastoji optimizirati svoje djelovanje. Kod višeagentnih sustava agenti uz to što prikupljaju podatke iz okoline, komuniciraju i međusobno kao što je priloženo na slici. Donošenje odluka kod agenata unutar grupe i za višeagentski sustav kao cjelinu rješava se uzimajući u obzir postavljene kriterije i trenutno stanje okoline. [33]



Sl. 4.2. *Višeagentni sustav: model interakcije* [7]

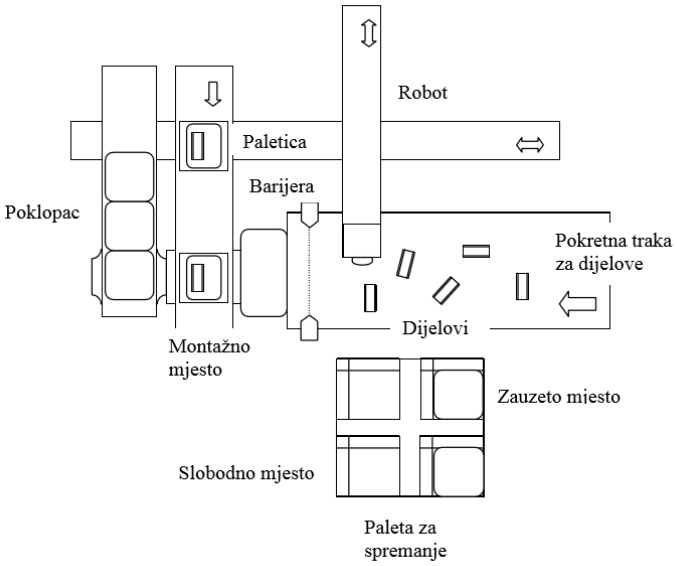
4.1. Protégé

Protégé je besplatni alat otvorenog koda koji se koristi za stvaranje ontologije, a razvija se na Stanford University u suradnji s University of Manchester. Sami modeli domene i aplikacije koje su temeljene na znanju razvijene su u Javi što ujedno znači da je i prenosiv te je njegovo izvršavanje omogućeno na svim računalima koja imaju Java Virtual Machine. Radna okolina ovog alata je okolina na koju se mogu povezati različiti softverski alati jer je Protégé pisan kao framework. Podržava dva osnovna načina modeliranja, Protégé-frames i Protégé-OWL editor. Protégé-OWL editor namijenjen je stvaranju ontologija za semantički web, te radi s različitim

oblicima ontologija poput RDF-a, Owl-a i XML Schema. Osim toga, Protégé nudi i širok skup struktura te akcija za modeliranje znanja koje čine temeljne elemente za stvaranje, vizualizaciju i manipulaciju ontologija u različitim formatima. Također, omogućuje učitavanje i spremanje OWL ontologija, uređivanje semantičkog jezika pravila, svojstava i vizualiziranih klasa. [34][35]

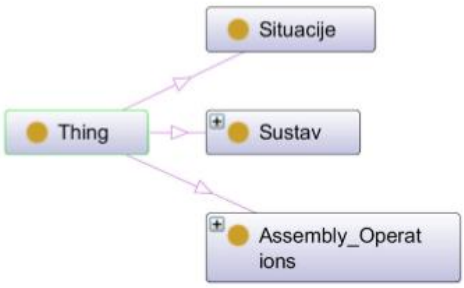
4.2. OWL ontologija kod poslovne montaže

Prema Ranogajec [29] jedna od primjena OWL ontologija je u poslovnoj montaži. Sukladno navedenome u ovom poglavlju će biti prikazan autorov primjer što je moguće ostvariti povezivanjem ontologije i robota. OWL ontologija u poslovnoj montaži podrazumijeva korištenje robota za obavljanje montažnih operacija te senzora za prikupljanje informacija iz okoline. U tom procesu se dijelovi s pokretne trake postavljaju u kućište, te zatim kućište dolazi drugom pokretnom trakom. Dijelovi dolaze pojedinačno do optičke barijere koja se dolaskom dijelova prekida, a pokretna traka se tada zaustavlja. Nakon što paleta s kućištem dolazi na mjesto montaže robot počinje izvršavati operacije. Prvo izvršava testiranje kojim se utvrđuje spremnost paletice za umetanje dijela s pokretne trake. Potom se ostvaruje ″inspection″ dijela na pokretnoj traci, odnosno utvrđuje se da li je dio taj dio u odgovarajućoj orijentaciji za prihvat. Zatim robot operacijom ″pick up″ uzima dio s pokretne trake i postavlja ga u kućište. Nakon prethodne operacije, robot preuzima poklopac i zatvara kućište. ″Test″ palete je sljedeća operacija kojom se detektira slobodno mjesto za posljednju operaciju storage kućišta. Ukoliko postoji slobodno mjesto, robot preuzima kućište s postavljenim poklopcem te ga sprema na veliku paletu. Rad robota završava nakon što se popuni prostor velike palete. [29]



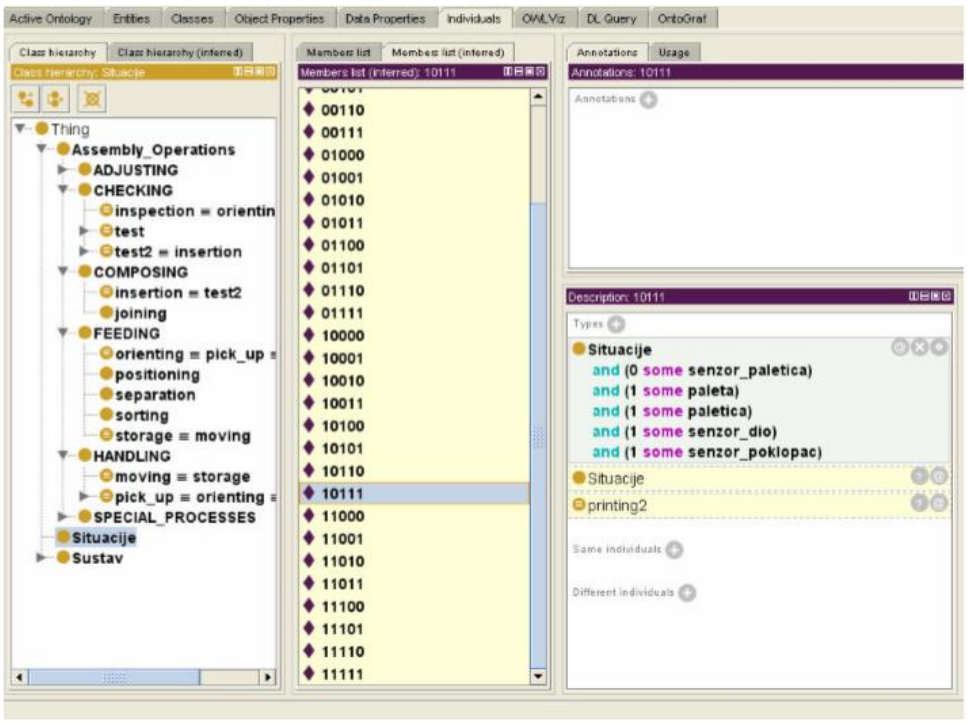
Sl. 4.3. *Shematski prikaz radne okoline* [29]

U ontologiji su tri glavna koncepta koja se nalaze odmah ispod koncepta Thing, kao što je vidljivo na sljedećoj slici.



Sl. 4.4. *Ontološke klase* [29]

Na slici su prikazane pojedine klase. Prva je *Assembly_operations* koja predstavlja klasu koja se koristi kao baza montažnih operacija. *Situacije*, klasa koja obuhvaća moguće kombinacije stanja senzora sustava (individua). Individue predstavljaju stanja senzora sustava koje upućuju na samu svrhu ontologije, to jest ako nam je poznato stanje senzora sustava, ontologija će odabrati montažne operacije koje je moguće ostvariti. Potrebno je naglasiti da su moguća dva stanja; aktivno i neaktivno stanje, odnosno 0 ili 1. Posljednja prikazana klasa je *Sustav* kao klasa koja obuhvaća listu svih senzora koji se nalaze u sustavu. [29]



Sl. 4.5. *Hijerarhija ontoloških klasa* [29]

Rasuđivanje je važan element stvaranja ontologija te ima dvije temeljne uloge. Prva uloga je osiguranje da je do sada izgrađena ontologija logički konzistentna, odnosno da ne sadržava kontradiktorne opise. Druga uloga je olakšavanje stvaranja ontologija; to jest proces rasuđivanja može automatski iz već oblikovane hijerarhije klasa, stvoriti novu hijerarhiju u kojoj su izvedeni svi mogući zaključci. Jedan od primjera je sljedeći:

Stanje na ulazu je sljedeće:

Senzor_dio – aktivan, 1

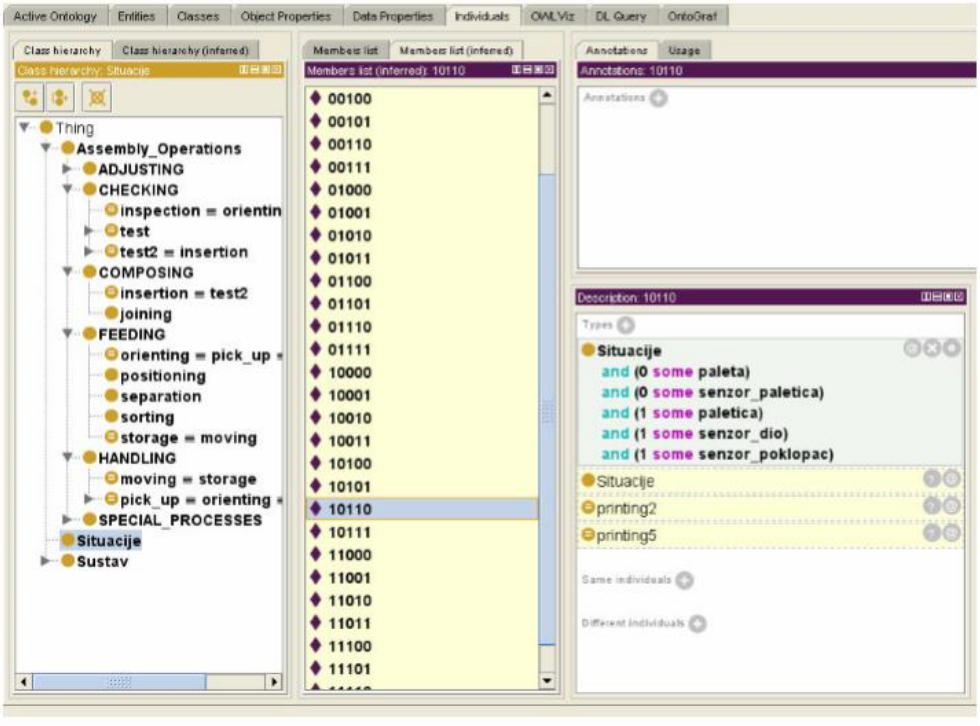
Senzor_paletica – nije aktivan, 0

Senzor_poklopac – aktivan, 1

Paletica – vizija ok, 1

Paleta – vizija nije ok, 0

Prikazani primjer znači da je stanje sustava zapisano s Individual 10110. Rezultat prikazanog stanja na izlazu prikazuje sljedeća slika. [29]



Sl. 4.6. *Rezultat prvog rasuđivanja* [29]

Slika prikazuje rezultat stanja senzora, odnosno rješenje montažnih operacija printing2 i rješenje printing5, što nam govori da senzor i vizija palete nisu aktivni. [29]

Drugi primjer se odnosi na rasuđivanje s rezultatima senzora:

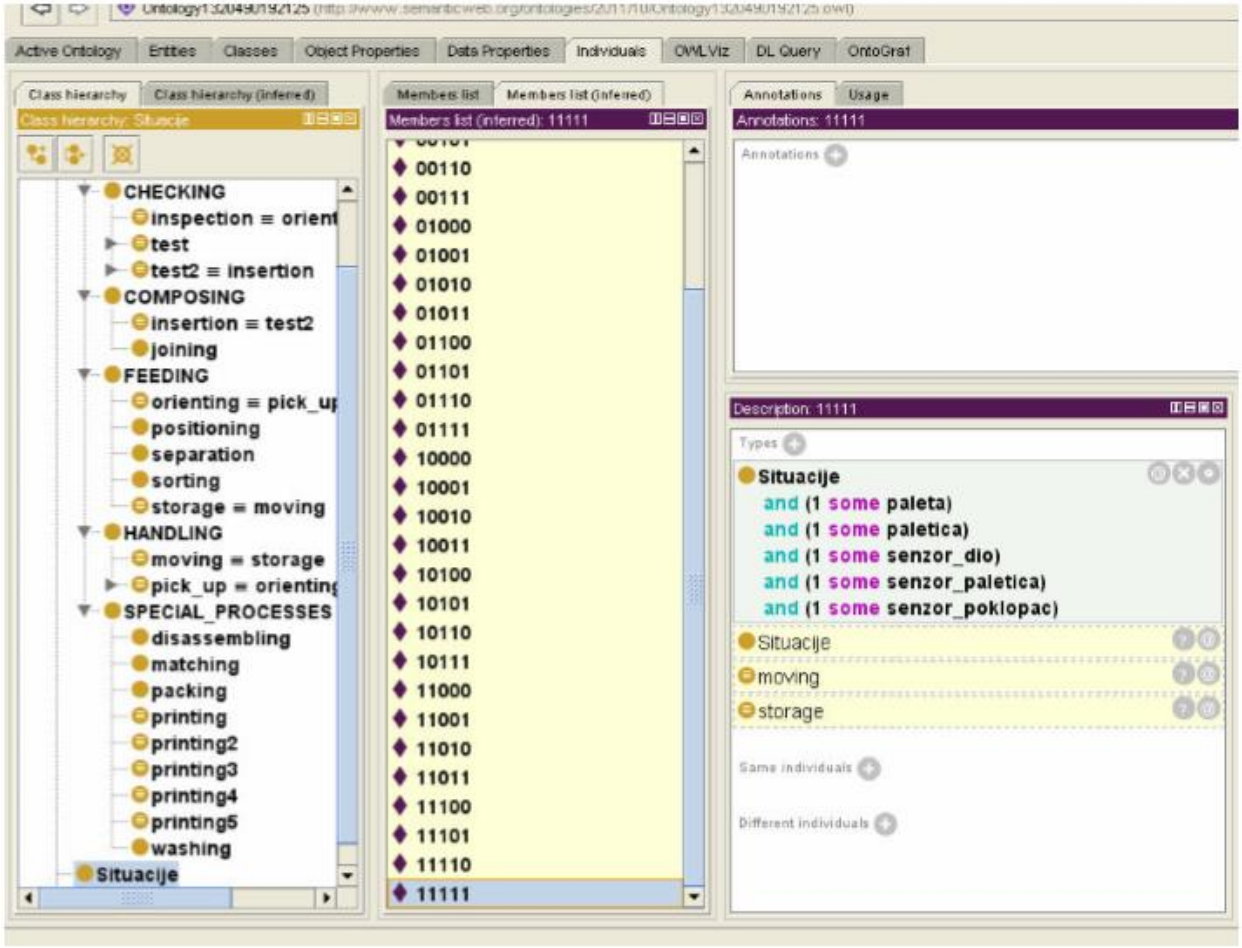
Senzor_dio – aktivan, 1

Senzor_paletica – aktivan, 1

Senzor_poklopac – aktivan, 1

Paletica – vizija ok, 1

Paleta – vizija ok, 1.

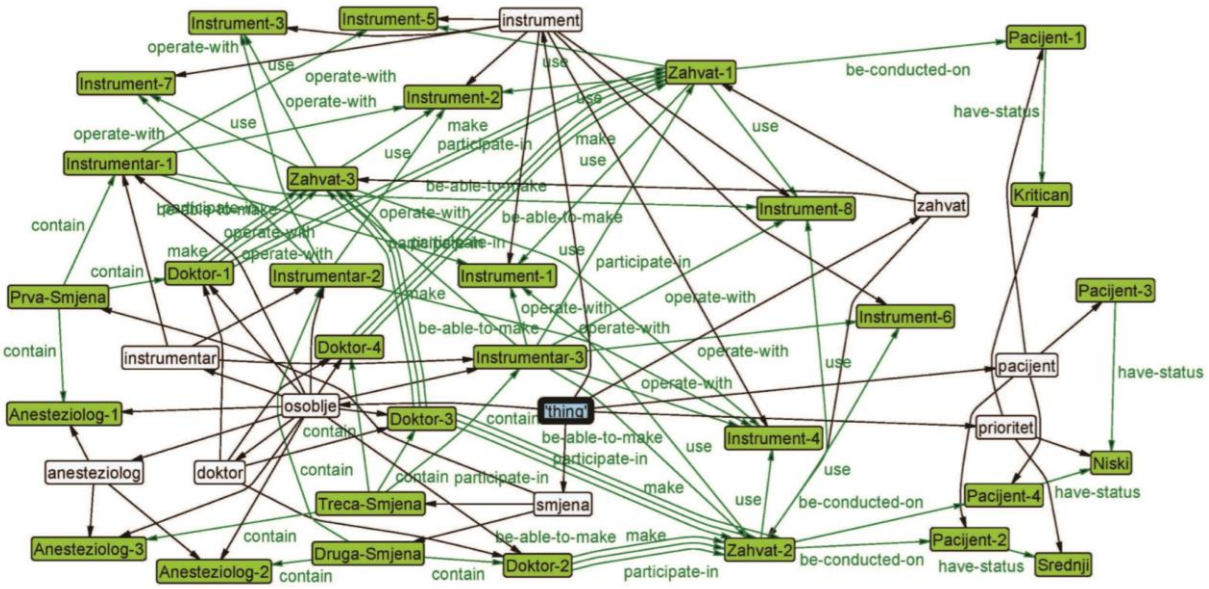


Sl. 4.7. *Rezultat drugog rasuđivanja* [29]

Slika prikazuje rješenje ontologije u slučaju kada su svi senzori aktivni. Vidljivo je da dobivanje dobrih informacija omogućuje izvršavanje odgovarajućih montažnih operacija, odnosno robot izvršava posljednje operacije moving i storage kućišta s umetnutim dijelom i postavljenim poklopcem na paletu.

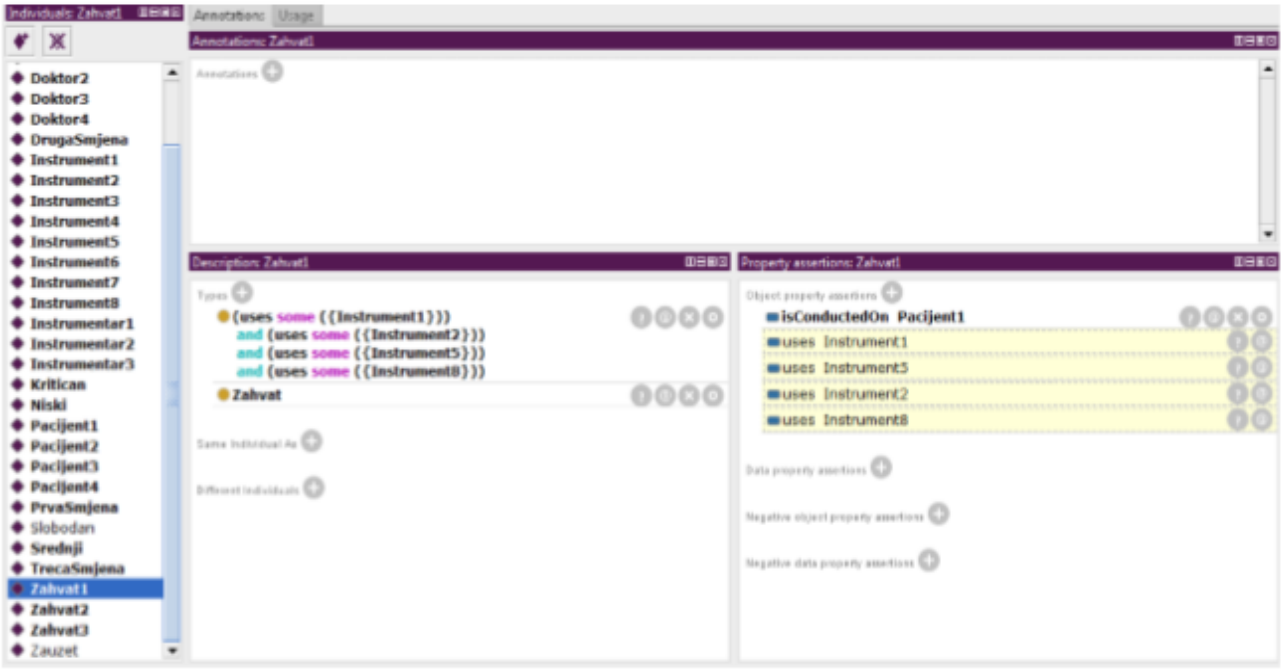
4.3. Primjena OWL ontologije u operacijskoj sali

Petljak [22] navodi kao jedan od primjera primjene OWL ontologije, primjenu u operacijskoj sali. Budući da je temeljna svrha ontologije predstavljanje znanja za promatranu domenu interesa, važno je opisati domenu, njene komponente te definirati njihove ovisnosti. Operacijska sala obuhvaća osoblje, instrumente koji se koriste za operacijske zahvate, operacijske zahvate kao takve, prioritete koji su vezani uz određene zahvate te smjene. Također, važno je navoditi i relacije koji povezuju određene komponente ontologije. Petljak [32] navodi relacije: conduct, be-able-to-make, operate-with, use i tako dalje. Nakon procesa uvođenja ontologije u Protégé program, dobiva se vizualno kvalitetniji prikaz ontologije, prikazano na slici ispod. Na slici je vidljiva kompleksnost veza između određenih komponenata.



Sl. 4.8. *Grafički prikaz ontologije* [22]

Izradom ontologije omogućuje se i kreiranje baze znanja za operacijsku salu te će se na temelju upita moći dobivati potrebni odgovori. Slika prikazuje klase kao što su *instrument, zahvat, pacijent, prioritet* i tako dalje. Također, prisutne su potklase i relacije između njih.



Sl. 4.9. *Opis za Zahvat1* [22]

Protégé program pruža opciju zaključivanja (eng. Reasoner). Opcija se ostvaruje na način da upišemo određeno pitanje u zaključivač, na primjer pitanje vezano uz instrumente koji su nam potrebno u operacijskoj sali „ Who-Or-What is instrument that is used by Zahvat-1 ? “. Zaključivač nam na temelju postavljenog pitanja daje odgovor, a primjer odgovora je vidljiv na slici ispod.

Sl. 4.10. *Primjer odgovora zaključivača* [22]

4.4. Ograničenja OWL ontologije

Ontologije se sve više koriste u inženjerskoj akademskoj zajednici, ali još uvijek postoji značajna razlika između znanstvenih istraživanja i industrijskih primjena, čime se onemogućava potpuno iskorištavanje ontoloških potencijala. I dalje postoji značajna potreba za učinkovitijim softverskim okruženjima koja omogućuju različite faze životnog ciklusa softverskog alata temeljenog na ontologiji. Sam je razvoj i korištenje softverskog alata koji je utemeljen na ontologije iznimno zahtjevan jer mu je potrebna suradnja stručnjaka iz različitih područja znanja poput ontoloških inženjeringa, računalnih znanosti, sustava za upravljanje bazama podataka i softverskog inženjeringa. Jedan od aktualnih problema je što industrijske aplikacije zahtijevaju upravljanje velikim količinama podataka, a to se često rješava korištenjem relacijskih baza podataka visokih performansi. U slučaju OWL ontologija dostupna rješenja predstavljaju ograničenja koja se odnose na jednu ili više od sljedećih karakteristika: licenca, operativni sustav, jezik implementacije , mehanizam spremišta i performanse, programski jezik za klijentske priključke, jezik upita, sigurnost, verzije i rukovanje binarnim podacima. [36]

5. ZAKLJUČAK

Semantički web danas ima veliki značaj jer omogućuje brz pronalazak informacija koje su potrebne korisnicima. Stvaranje semantičkog weba zahtijeva stvaranje standardnog jezika za izražavanje ontologije te stvaranje ontologija koje su potrebne. Ontologije su osnovna tehnologija koja omogućuje nadogradnju običnog weba. One predstavljaju rječnike termina koji su iskoristivi u različitim područjima ljudske djelatnosti te obuhvaćaju znanja kojima se ti termini povezuju. Područja primjene i istraživanja ontologije su područja poput inženjerstva znanja, obrade prirodnog jezika, informacijskih sustava i tako dalje. Ontologije su kao grane metafizike usmjerene na identificiranje stvari koje postoje u općenitim terminima. [1][2]

Važna značajka stvaranju ontologije su ontološki jezici koji pružaju različite sadržaje, a jedan od novijih standardnih jezika za ontologije je OWL (engl. *Ontology Web Language*). Navedeni jezik je baziran na logičkom modelu koji omogućava definiranje i opisivanje pojmova. OWL ontološki jezik se razvio kao jezik za predstavljanje znanja zahvaljujući mogućnosti izvođenja činjenica koje nisu eksplicitno navedene.

U ovome radu su opisani neki primjeri primjene OWL ontologije u tehničkim sustavima, posebno vezanim uz robotiku. Prvi kao onaj koji opisuje primjenu OWL ontologije u poslovnoj montaži te podrazumijeva korištenje robota za obavljanje montažnih operacija te senzora za prikupljanje informacija iz okoline; a drugi primjer vezan uz primjenu u medicini. No, medicina i poslovna montaža nisu jedina područja ljudskoga rada u kojima se koriste ontologije. Primjena ontologije je prisutna i u područjima umjetne inteligencije, semantičkoga weba, softverskog inženjeringa te informacijskim sustavima. Unatoč navedenim primjerima, OWL ontologija u tehničkim sustavima nije uhvatila potpuni zamah zbog ograničenja poput operativnih sustava, jezika implementacije, jezika upita, sigurnosti i mnogih drugih koji su već navedeni i opisani u prethodnom poglavlju.

Ipak, smatram da će se daljnjim razvojem tehnologije razvijati nove i šire mogućnosti primjene OWL ontologije. Postoji mnogo domena ontologije te će ih u budućnosti biti još više što će ujedno omogućiti i značajniju primjenu u raznovrsnim područjima i oblicima ljudske djelatnosti. Budući da je jedna od temeljnih težnji današnjeg čovječanstva olakšati ljudski rad, ubrzati proizvodnju resursa, smanjiti troškove i stvoriti zamjenu za ljudski rad, sve je veći potencijal za upotrebu OWL ontologije upravo u području robotike.

LITERATURA

[1] B.Smith, C.Welty, „Formal Ontology in Information Systems“, 2001.

[2] L. Veljak, „Principi ontologije i pitanje o primatu ontologije nad logikom i gnoseologijom“, 2011.

[3] T.Berners-Lee, W. Hall, J. A. Hendler, K. O'Hara, N.Shadbolt, D. J. Weitzner, „A Framework for Web Science“, 2006

[4] Članak s interneta, http://hr.wikipedia.org/wiki/Internet, 10.9.2018.

[5] Članak s interneta, https://home.cern/topics/birth-web, 10.9.2018.

[6] T. Berners-Lee, J. Hendler, O. Lassila, „The Semantic Web“, https://www-sop.inria.fr/acacia/cours/essi2006/Scientific%20American_%20Feature%20Article_%20The%20Semantic%20Web_%20May%202001.pdf

[7] T. Stipančić, Spoznajni model upravljanja grupom industrijskih robota, Doktorska disertacija, Fakultet strojarstva i brodogradnje, University of Zagreb, 2013

[8] T. O'Reilly, What Is Web 2.0: Design Patterns and Business Models for the Next Generation of Software, 2007.

[9] N. Nađ, Integracija Web 3.0 sadržaja i usluga u obliku Mashup aplikacija, diplomski rad, Fakultet organizacije i informatike, Sveučilište u Zagrebu, 2013.

[10] Članak s interneta, https://epdf.tips/a-semantic-web-primer9ce82ef7b0ed18e1f336b885e912a1ef16729.html, 10.9.2018.

[11] T. Galba, Algoritam transformacije ontologije u strukturu taksonomije za evidencijsko zaključivanje, Doktorski rad, Fakultet elektrotehnike, računarstva i informacijskih tehnologija, Sveučilište u Osijeku, 2016.

[12] Članak s interneta, https://en.wikipedia.org/wiki/Uniform_Resource_Identifier, 10.9.2018.

[13] Članak s interneta, https://hr.wikipedia.org/wiki/HTML, 10.9.2018.

[14] Članak s interneta, https://en.wikipedia.org/wiki/XML, 10.9.2018.

[15] Članak s interneta, https://www.w3.org/XML/, 10.9.2018.

[16] Članak s interneta, http://www.xml.com/pub/a/2001/01/24/rdf.html, 10.9.2018.

[17] W3C, „Resource Description Framework (RDF) Model and Syntax Specification“, https://www.w3.org/TR/1999/REC-rdf-syntax-19990222/, 10.9.2018.

[18] C. Legg, „Ontologies on the semantic web“, University of Waikato, 2007.

[19] I. Horrocks, P. Patel-Schneider, F. Van Harmelen, „From SHIQ and RDF to OWL: the making of a Web ontology language“, 2003.

[20] J.Cardoso, A. M. Pinto, „The Web Ontology Language (OWL) and its Applications“, 2015.

[21] Y. Alsafi, V. Vyatkin, „Ontology-based Reconfiguration Agent for Intelligent Mechatronic Systems in Flexible Manufacturing“

[22] D. Petljak, „Baza znanja temeljena na ontologijama“, Završni rad, Fakultet strojarstva i brodogradnje, Sveučilište u Zagrebu, 2015.

[23] D. Radovanović, „Semantički web i elektronski izvori informacija“, Infoteka 4 vol. 2, 2003.

[24] K. Klarin, „Korištenje ontologija pri razvoju informacijskog sustava“, Ph.D. thesis, Fakultet elektrotehnike, strojarstva i brodogradnje, Sveučilište u Splitu, 2013.

[25] N. Guarino, „Formal Ontology and Information Systems“, 1998.

[26] D. Jurić, „Metode učenja ontologija, trenutno stanje i problemi“, Fakultet elektrotehnike i računarstva, Sveučilište u Zagrebu, 2008.

[27] A. Đuraković, Verić V, „Semantički web“, s interneta, http://www.mathos.unios.hr/~adjurako/semantickiweb/Semanti%C4%8Dki%20web.pdf, 09.09.2013

[28] Članak s interneta, „Operations on Ontologies“, http://www.obitko.com/tutorials/ontologies-semantic-web/operations-on-ontologies.html

[29] A. Ranogajec, Diplomski rad, Fakultet strojarstva i brodogradnje, Sveučilište u Zagrebu, 2011.

[30] Članak s interenta, „FIPA Ontology Service Specification“, 2010, http://www.fipa.org/specs/fipa00086/XC00086D.pdf

[31] Z. Balaž, K. Meštrović, „Učenje i poučavanje iz umjetne inteligencije“, Polytechnic & design, Vol. 2, No. 1, 2014.

[32] M. Mihanović, „Uporaba inteligentnih agenata u dinamičkom generiranju courseware-a“, Diplomski rad, Sveučilište u Splitu, 2009.

[33] M. Rukavina, „Agent u 3d okolini“, Diplomski rad, Fakultet organizacije i informatike, Sveučilište u Zagrebu, 2015.

[34] Članak s interneta, https://protege.stanford.edu/, 15.9.2018.

[35] Članak s interneta, https://en.wikipedia.org/wiki/Prot%C3%A9g%C3%A9_(software), 15.9.2018.

[36] S. El Kadiri, W. Terkaj, E. N. Urwin, C. Palmer, D. Kiritsis, R. Young, „Ontology in Engineering Applications“, 2015.

[37] D. Starčević, Diplomski rad, Fakultet strojarstva i brodogradnje, Sveučilište u Zagrebu, 2013.

[38] I. Ćurak, „Višeagentski sustav za optimizaciju procesa izrade vremenskog rasporeda“, Doktorska disertacija, Fakultet elektrotehnike, strojarstva i brodogradnje, Sveučilište u Splitu, 2012.

[39] K. Pavić, „Semantički web“, Seminarski rad

[40] M.Banek, „Ontologije i semantički web“, Fakultet elektrotehnike i računarstva, Sveučilište u Zagrebu, 2013.

SAŽETAK

U skladu s današnjim ubrzanim razvojem tehnologija, u računarstvu i informatici sve se više spominje pojam „ontologije“. U području računarstva pojam „ontologije“ predstavlja dokument koji sadrži iskaze o međusobnim odnosima određenih pojmova od kojih se nekih sustav sastoji. Područja primjene i istraživanja ontologije su područja poput inženjerstva znanja, obrade prirodnog jezika, informacijskih sustava i tako dalje. Ontologije su kao grane metafizike usmjerene na identificiranje stvari koje postoje u općenitim terminima, a intenzivno se koristi u području semantičkog weba koji predstavlja proširenje trenutnog weba. Povećanjem veza i različitih međudjelovanja računala, stvaranje ontologija je postajalo sve važnije, a ontološki jezici sve potrebniji jer su njihovi zapisi razumljivi svim računalima. Ontološki jezici pružaju različite sadržaje, a jedan od novijih standardnih jezika za ontologije je OWL (Ontology Web Language). OWL ontološki jezik se razvio kao jezik za predstavljanje znanja zahvaljujući mogućnosti izvođenja činjenica koje nisu eksplicitno navedene. Budući da se društveni napredak danas sve više svodi na razvoj računalnih i informacijskih sustava, OWL ontologija stječe sve veći potencijal primjene u tehničkim sustavima, posebno vezanim uz područje robotike.

Ključne riječi: ontologije, OWL, semantički web, tehnički sustavi

ABSTRACT

Application of OWL ontologies in technical systems

In line with today's global development of technology, computing and information technology, the term "ontology" is increasingly mentioned. In the field of computing, the term "ontology" is a document that contains statements about the relationships between certain concepts that some systems contain. Fields of application and research on ontology are areas such as engineering, natural language processing, information systems and so on. Ontologies are metaphysical branches focused on identifying things that exist in general terms. They are intensively used in the field of semantic web, which is the extension of the current web. The development of ontology creation became more powerful, thanks to increasing connections and different computer interactions, and ontological languages more necessary because their records are understandable to all computers. Ontologies languages provide different contents, and one of the new standard languages for ontology is OWL (Ontology Web Language). OWL Ontology Language has developed as a language for the presentation of knowledge thanks to the possibilities of performing facts that are not explicitly mentioned. Since social progress increasingly leads to the development of computer and information systems, OWL ontology gains increasing potential in technical systems, particularly in the field of robotics.

Keywords: ontology, OWL, semantic web, technical systems

ŽIVOTOPIS

Tomislav Jukić, rođen 18.6.1991.godine u Požegi. Osnovnu školu Zdenka Turković pohađao u Vetovu i Kutjevu u razdoblju od 1998. do 2006. godine. Nakon osnovnoškolskog obrazovanja upisuje Tehnička škola u Požegi i uspješne obrane maturalnog rada 2010. godine stječe zvanje tehničar za računalstvo. Iste godine upisuje Fakultet elektrotehnike, računarstva i informacijskih tehnologija u Osijeku, smjer računarstvo. Nakon toga slijedi Sveučilišni diplomski studij smjer Računarstva, Izborni blok DRD - Informacijske i podatkovne znanosti kojeg trenutno pohađa.