Vizuelno anotiranje veb sadržaja semantičkim podacima

Nikola Vulićević

***Sadržaj* — Iako su tehnologije semantičkog veba odavno na zrelom nivou, prohvatanje semantičkog veba je do sada bilo vrlo sporo. Da bi bilo povoda za razvoj semantičkih servisa, moraju postojati semantički podaci koje će servisi koristiti. Stvaranje semantičkih podataka automatskim ekstaktovanjem i procesiranjem postojećih podataka sa veba je prvi korak koji je u određenoj meri ispunjen. Drugi korak je da se običnim korisnicima veba koji stvaraju veliki deo sadržaja, obezbede alati za lako stvaranje semantičkog sadržaja. U ovom radu je razvijen i prikazan alat za anotiranje veb stranica semantičkim podacima u microdata formatu. Alat je dodatak za TinyMCE editor tako da se može koristiti u raznim sistemima za upravljanje sadržajem.**

***Ključne reči* — semantički veb, anotiranje, microdata, tinymce**

I. UVOD

Sadržaj na internetu trenutno mogu da razumeju samo ljudi i u vrlo maloj meri računari. Računari u najvećem broju slučajeva razmenjuju i prikazuju informacije korisniku bez razumevanja o njihovom značenju. Semantički veb je tehnologija koja ima cilj da novom i postojećem sadržaju na internetu doda semantiku, što će programerima omogućiti nove i bolje načine za rešavanje problema, a krajnjim korisnicima nove i bolje servise. Da bi semantički veb bio koristan, ljudi moraju da objavljuju semantičke podatke. To je problem jer su trenutno beneficije male za onoga ko ih objavljuje a predstavljaju dodatno opterećenje. Semantičke podatke možemo dobiti ekstraktovanjem podataka iz postojećih relacionih baza podataka i html strana, ili procesiranjem prirodnog govora. Većina semantičkih podataka koji su trenutno na vebu potiče iz ovakvih procesa što nije idealno.

Nikola Vulićević, Računarski fakultet, Srbija (e-mail: nvulicevic07@raf.edu.rs).

Obični korisnici interneta stvaraju veći deo sadržaja na internetu. On može biti u vidu raznih prezentacija, fotografija, blog članaka, mikropostova (twitter), recenzija itd. Da bi se ispunila krajnja svrha semantičkog veba, autori sadržaja moraju da stvaraju i semantičke podatke, tako da je vrlo bitno da im za to budu dostupni što bolji alati kako ne bi bili izloženi tehnologijama namenjenim programerima. Danas, jedini način na koji velika većina autora sadržaja na internet dodaje meta-podatke sadržaju je tagovanje. Tagovanje je brz i lak način za klasifikovanje sadržaja, ali ima dosta ograničenja. Tagovi nisu definisani u dobro poznatim rečnicima tako da za istu stvar više korisnika može da koristi različite tagove a krajnji korisnik koji pretražuje po tagovima može samo da nagađa koje je tagove autor sadržaja koristio.

Tagovi su samo početak ka semantičkom vebu, a sledeći korak je microdata format koji je već zaživeo zahvaljujući podršci velikih pretraživaca i lakoći korišćenja. Microdata format je način za semantičko označavanje html sadržaja direktno u html kodu. Da bi bio koristan microdata mora da se koristi sa ontologijama koje definišu entitete, atribute i njihove relacije u određenom domenu znanja. Jedan od problema semantičkog veba je nepostojanje standardizovanih ontologija. Pri korišćenju uključenih (embedovanih) semantičkih podataka ovaj problem je donekle rešen jer je inicijativom tri najveća pretraživaca (Google, Bing i Yahoo) pokrenut sajt schema.org koji sadrži mnoštvo šema za semantičko označavanje podataka u html stranama, a koji će pretraživacima biti korisni. Neke od dostupnih šema su: ljudi, mesta, kreativna dela, organizacije, događaji i mnoge druge. Sva tri pretraživača koriste dostupne semantičke podatke u html stranama da poboljšaju informacije prikazane u rezultatima pretrage, ali trenutno ne pomažu boljem rangiranju u pretrazi [1]. S obzirom da schema.org i microdata postoje svega par godina, korist za krajnjeg korisnika će u budućnosti biti sve veća.

II. SEMANTIČKI PODACI U HTML STRANAMA

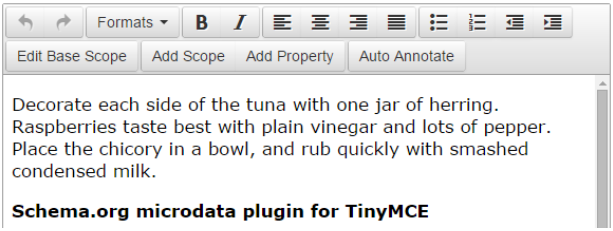
Većina dinamičkih sajtova, koji često serviraju novi sadržaj, zasniva se na sistemima za upravljanje sadržajem (CMS). Iako se ogromna količina informacija na internetu stvara ovim putem, podrška za semantičke podatke u CMS sistemima je veoma slaba ili nepostojeća. Takođe, korisnici koji stvaraju sadržaj tim putem koriste takve sisteme jer ne žele da imaju dodira sa

tehničkim aspektima veba. Naravno, semantički podaci treba i mogu da se uključuju u veb stranice pri generisanju html prezentacija pomoću podataka iz baza podataka, ali smatramo da to nije dovoljno za ispunjenje vizije semantičkog veba.

TinyMCE je daleko najkorišćeniji JavaScript WYSIWYG (What You See Is What You Get) editor i deo je osnovne WordPress instalacije. On u veb browser-ima daje mogućnost vizuelnog formatiranja html sadržaja poput regularnih tekst editora na koje su korisnici navikli.

Alat razvijen u ovom projektu je dodatak (plugin) za TinyMCE editor, pod nazivom Schemantic (schema+semantic), koji mu dodaje funckionalnost za semantičko editovanje sadržaja. S obzirom da WordPress pokreće skoro 23% svih sajtova na internetu, razvoj dodataka za njega nam daje ogroman broj potencijalnih korisnika [2]. Naravno, može se koristiti i u drugim postojećim ili novim sistemima pored WordPressa, ukoliko je u njih moguće integrisati TinyMCE editor. Da postoji potreba za ovakvim alatima pokazuje to da je i Google naprednijim korisnicima obezbedio sličan alat, ali koji pruža samo najosnovniju funkcionalnost i koristi se kao odvojeni alat [3].

Na Slici 1 je prikazan TinyMCE u veb browser-u sa dodatnim toolbar-om koji sadrži opcije Schemantic plugin-a: Edit Base Scope, Add Scope, Add Property i Auto Annotate. Da bismo potpuno razumeli kako Schemantic plugin radi, moramo prvo da razumemo microdata format. Iako je microdata format malo opširniji od donjeg opisa, biće opisani samo oni delovi koji su za sada podržani u alatu.



Slika 1: TinyMCE editor sa dodatnim opcijama za semantičko editovanje

A. Microdata format

Microdata podatke uključujemo u html dodavanjem par ključnih reči kao atribute html elemenata. Html atribut *itemscope* označava određeni html element kao semantički element koji će sadržati svojstva koja ga opisuju. On

se uglavnom koristi zajedno sa html atributom *itemtype* koji, za razliku od *itemscope*, ima URL vrednost koja predstavlja jedinstveni identifikator tog tipa elementa. Schemantic plugin koristi schema.org šeme tako da za URL vrednost *itemscope* atributa koristimo schema.org URL adrese. U sledećem primeru označavamo blok koji opisuje osobu pomoću *Person* šeme sa schema.org:

```
<div itemscope itemtype=”http://schema.org/Person”> … </div>
```

Html elementi koji se nalaze u *itemscope* bloku su njegovi potencijalni atributi. Pod element postaje atribut kada sadrži *itemprop* html atribut. *Itemprop* html atribut za vrednost prima ime atributa iz šeme definisane *itemtype* atributom. Sama vrednost atributa je tekstualni sadržaj elementa, ili vrednost URL atributa ukoliko ga element ima (<img src="">, <a href="">) [4]. Gornji primer možemo da proširimo sa par atributa:

```
<div itemscope itemtype=”http://schema.org/Person”>
        <span itemprop=”givenName”>Nikola</span>
        <span itemprop=”familyName”>Vulićević</span>
</div>
```

Vidimo da microdata format koristi podatke koji se već nalaze na html strani za prikaz korisniku. Ako želimo da na stranici uključimo dodatne podatke, koje korisnik neće videti, koristimo *meta* html element. On je koristan i za dodavanje podataka elementima koji imaju značenje korisniku, ali nemaju tekstualnu reprezentaciju, na primer slike ili video elementi [5]. Sledeći primer pokazuje korišćenje meta elemenata za definisanje sakrivenih atributa:

```
<div itemscope itemtype=”http://schema.org/Person”> …
        <meta itemprop="email" content=”email@gmail.com”/>
        <meta itemprop=”gender” content=”male”/>
</div>
```

Pored scope (*itemscope*) i property (*itemprop*) elemenata, postoje element koji mogu biti oba u isto vreme. Atributi koje smo do sada videli su bili primitivnog tipa, ali microdata atribut može biti tipa bilo kog entiteta koji je definisan u schema.org ontologiji. Ako pogledamo strukturu ‘http://schema.org/Person’ šeme, vidimo da su određeni atributi tipa *Person*,

Place ili *Organization*. Sledeći kôd je primer ovakvih ugneždenih atributa i opisuje osobu koja je član organizacije:

```
<div itemscope itemtype=”http://schema.org/Person”> …
<div itemprop=”memberOf” itemscope itemtype=”schema.org/Organization”>
        <span itemprop=”legalName”>Računarski Fakultet</span>
</div></div>
```

III. OPIS REALIZOVANOG PROJEKTA

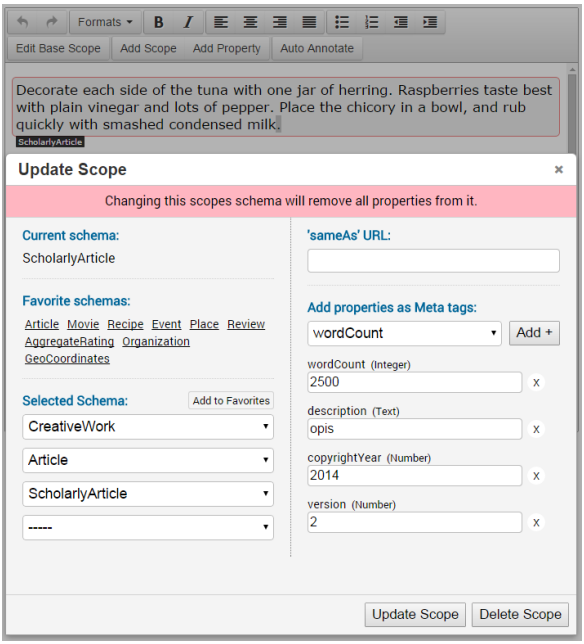
Označavanje sadržaja microdata formatom zahteva bar osnovno znanje html-a koje većina korisnika, koji stvaraju sadržaj preko WordPress-a, nema. Takođe, isplativost anotiranja je trenutno mala, tako da je neophodno da taj proces bude što lakši za korisnika. Schemantic plugin je alat za vizuelno anotiranje sadržaja microdata formatom koji u pozadini generiše validan html kôd i ne zahteva znanje html-a. Korišćenje se zasniva na istim interakcijama sa kojima su korisnici već upoznati, a semantički entiteti u dokumentu imaju poseban vizuelni stil koji nije vidljiv u krajnjem dokumentu. Pored toga, sadrži i mehanizame koji pomažu pri anotiranju: auto sugestije, auto anotiranje i individualno prilagođavanje (*eng. customization*). Izvorni kôd alata je javno dostupan na GitHub-u tako da svako može da doprinese njegovom daljem razvoju [6].

Da bismo bolje razumeli i stekli pravu sliku o funkcionalnosti Schemantic plugin-a, pre nego što detaljno opišemo sve opcije, prvo ćemo ih objasniti ukratko. Schemantic plugin sadrži četiri opcije:

- Add Scope: ‘Scope’ je html element koji sadrži *itemscope* i *itemtype* atribute. On je entitet koji za tip ima jednu od schema.org šema. Ova opcija prikazuje dijalog sa opcijama i označava selektovani sadržaj u editoru kao scope element. Ovako napravljen scope je uokviren punom crvenom linijom.
- Edit Base Scope: Preko ove opcije označavamo ceo dokument kao scope (bazni scope). Svrha ove opcije je da ovakav scope nema vizuelnu reprezentaciju u editoru tako da ne smeta pri daljem editovanju. Takođe je prečica za brz početak rada na anotiranju dokumenta.
- Add Property: ‘Property’ je html element koji sadrži *itemprop* atribut i nalazi se u scope elementu. Ova opcija označava trenutno selektovani sadržaj editora kao property element i on je predstavljen uokviren

isprekidanom linijom nasumične boje.

- Auto Annotate: Ceo dokument će biti analiziran dbPedia Spotlight API-em i automatski anotiran prepoznatim entitetima. Automatski napravljeni scope-ovi će, u skladu sa Linked Data inicijativom, imati identifikacioni atribut *sameAs* koji za vrednosti ima URL do dbPedia entiteta. Semantički editori i tehnologije za procesiranje prirodnog govora (NLP) su prirodan spoj. NLP može pomoći pri anotiranju dobro poznatih entiteta dok će korisnik i dalje imati potpunu kontrolu nad anotacijama u dokumentu.



Slika 2: Add Scope dijalog

Postojeće scope i property blokove možemo izmeniti duplim klikom na njih čime će se otvoriti prozor za editovanje iz kog možemo i da ih izbrišemo. Prelazak miša preko njih će prikazati kog su tipa.

A. Add Scope

Na *slici 2* je prikazan dijalog za editovanje postojećeg scope-a koji je isti kao i dijalog za pravljenje novog. Scope koji editujemo se može videti iznad

dijaloga uokviren punom crvenom linijom. U levoj polovini dijaloga se nalaze kontole za selektovanje tipa scope-a. Šeme sa schema.org su organizovane hijerarhijski gde se korenska šema koju sve ostale nasleđuju naziva *Thing*. Na primer, šema *CreativeWork* je pod-šema *Thing* šeme, a šema *Article* je pod-šema *CreativeWork* šeme. Pošto broj šema nije mali a njihova hijerarhija je duboka do četiri nivoa, postoji opcija za pravljenje prečica do najkorišćenijih šema.

U desnoj polovini dijaloga se nalaze kontrole za dodavanje *meta* (nevidljivih) atributa. Svaka šema ima svoje jedinstvene atribute a dostupni su joj i atributi svih roditeljskih šema. Meta podaci mogu biti samo atributi primitivnih tipova (tekst, brojevi) dok se za dodavanje atributa svih tipova koristi poseban dijalog. Atribut *sameAs* je istaknut kao posebna opcija zbog njegovog značaja koji je objašnjen kasnije u tekstu.

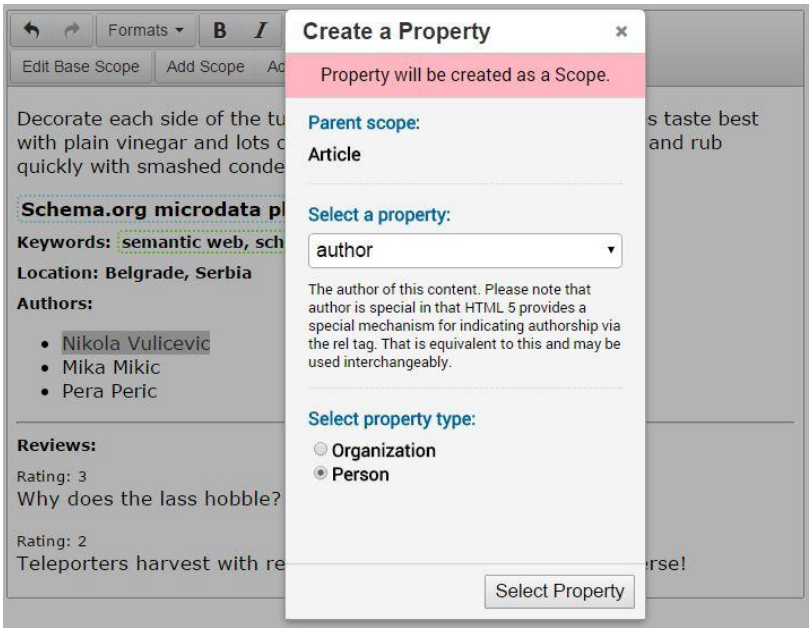
B. Add Property

Na slici 3 je prikazan dijalog za dodavanje i editovanje atributa. Atribut će imati vrednost selektovanog teksta koji se mora nalaziti u postojećem scope-u. Vidimo da se selektovani tekst "Nikola Vulicevic" ne nalazi u scope-u, ali je dodavanje atributa dozvoljeno jer je ceo dokument definisan kao bazni scope tipa *Article* preko 'Edit Base Scope' opcije. Dijalog za dodavanje atributa će prikazati sve dostupne atribute scope-a u kome se nalazi kao i njegovih roditeljskih scope-ova. Atributi mogu imati više potencijalnih tipova i korisnik tada ima izbor kao što je prikazano na slici 3. Ukoliko je izabrani atribut primitivnog tipa (tekst, broj) on će biti dodat kao običan atribut uokviren isprekidanom linijom i njegov tip ćemo kasnije moći da promenimo. Ukoliko atribut predstavlja neki drugi entitet on će biti dodat kao ugnježđeni scope čiji tip kasnije nije moguće promeniti, ali mu je moguće dodavati atribute i još ugnježđenih scope-ova. Na primer, na slici 3 dodajemo atribut *author* koji je tipa *Person*, tako da će ovaj atribut istovremeno biti i scope i atribut.

C. sameAs atribut

Korenski entitet *Thing* iz schema.org ontologije sadrži *sameAs* atribut URL tipa. Njegova namena je da anotirane entitete iz našeg dokumenta poveže sa spoljnim entitetima identifikovanim, na primer, Freebase ili dbPedia adresom [7]. Korišćenje URL identifikatora za naše entitete i pravljenje relacija ka spoljnim entitetima su glavni principi Linked Data inicijative. Stoga je ovaj atribut tretiran kao najvažniji atribut i stavljen je u prvi plan. Polje za *sameAs*

atribut prihvata bilo koju URL adresu, ali sadrži i auto suggest funkcionalnost koja će za uneti pojam predložiti listu Freebase entiteta sa njihovim odgovarajućim URL adresama (Slika 4).



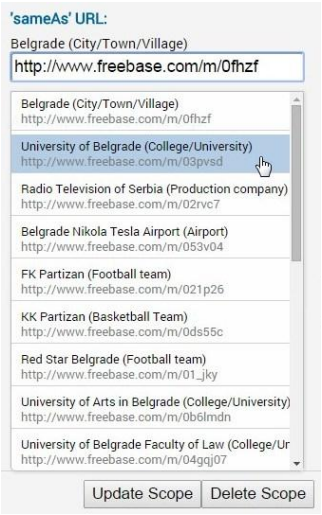
Slika 3: Add Property dijalog

U *sameAs* polju je preporučeno korišćenje Freebase adresa iz više razloga. Freebase je trenutno najveća baza semantičkih podataka i Google je koristi kao glavni izvor podataka u Knowledge Graph projektu [8]. Naravno nije čudno da Google preporučuje tehnologije koje poseduje, ali njihovim korišćenjem krajnji korisnik, kome je ovaj alat namenjen, trenutno ima najviše koristi. Pored toga, jedina prava alternativa Freebase-u je dbPedia a većina entiteta u Freebase-u je mapirano na entitete iz dbPedie preko *owl:sameAs* atributa i obrnuto [9]. Podrškom za Freebase u Schemantic alatu je na neki način upotpunjena podrška za trojstvo semantičkih tehnologija (microdata, schema.org, freebase) koje Google preporučuje i sam koristi za poboljšanje pretrage.

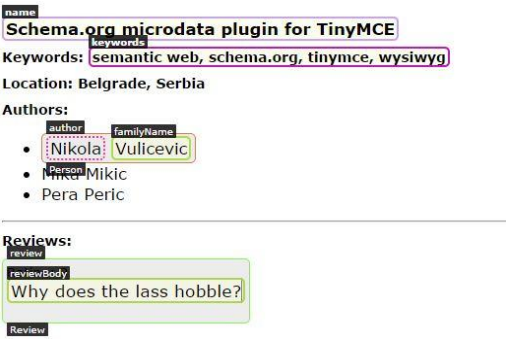
D. Primer

Na Slici 5 je prikazan primer anotiranog dokumenta Schemantic alatom. Ovaj dokument je tipa *Article* i ima definisan naziv i ključne reči. Takođe

ima ugnježdeni atribut *autor* tipa *Person* koji ima atribute *givenName*, *familyName* kao i sakrivene *meta* atribute.

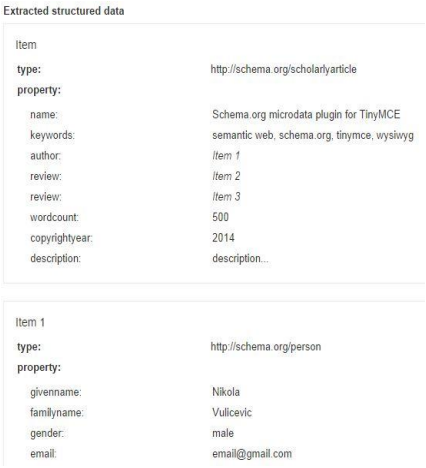


Slika 4: Freebase auto suggest u Add Scope dijalogu



Slika 5: Primer anotiranog dokumenta

Da bi potvrdili da je generisani html kôd validan, možemo koristiti Google Structured Data Testing Tool [10] koji je jedan od mnogih validatora embedovanih semantičkih podataka. U svim testovima validator je pravilno pročitao sve semantičke podatke iz anotiranog dokumenta, a rezultat jednog od testova se može videti na Slici 6.



Slika 6: Validacija dokumenta Google Structured Data Testing alatom

E. Poređenje sa alatima sličnog tipa i buduća unapređenja

Za kraj, napravićemo poređenje sličnih alata za vizeuelno anotiranje i diskutovati o nedostacima i mogućim unapređenjima Schemantic plugin-a.

Google Structured Data Markup Helper [3] pruža mogućnost anotiranja proizvoljnih html strana ili blokova html koda. Ne integriše se u postojeće editore dokumenata tako da je neophodna ručna zamena postojećeg ne-anotirnanog html-a sa anatiranim html-om koji generiše ovaj alat. Za anotiranje je dostupno 10 šema iz schema.org ontologije, i to su šeme koje Google koristi za poboljšanje prikaza u rezultatima pretrage.

RDFaCE [11] je dodatak za TinyMCE editor namenjen naprednijim korisnicima. Pored podrške za schema.org ontologiju on podržava i anotiranje proizvoljnim semantičkim podacima u RDFa formatu. Posebno zanimljiva mogućnost je editor RDF trojki koji pruža pregled svih RDF trojki u dokumentu kao i dodavanje novih.

Domeo [12] je web aplikacija za anotiranje XML dokumenata sa fokusom na naučno istraživačke radove i deljenju anotacija među istraživačima. Anotacije napravljene ovim alatom nisu uključene u dokumentima već se čuvaju u RDF formatu. Pošto su dokumenti i anotacije odvojeni, Domeo dozvoljava učitavanje i anotiranje postojećih dokumenata nad kojima nemamo mogućnost editovanja već samo čitanja [13].

Glavni cilj Schemantic plugin-a je da običnim korisnicima da funkcionalnost za semantičko anotiranje u prilagođenom interfejsu koji ih neće odbiti od tehnologije. Iako je u njemu već ugrađeno više sistema koji

pomažu korisniku, postoji prostor za još unapređenja. Iako trenutno nisu očigledna, moguća proširenja će postati jasnija iz potreba individualnih korisnika tokom korišćenja alata. Proširenja koja su trenutno u razvoju su:

- Ugrađeni alat za validaciju i vizuelizaciju trenutnih anotacija
- Šabloni (templates) za brzo dodavanje anotacija
- Čuvanje (export) anotacija u JSON-LD i RDF formatima
- Bolja podrška za Linked Data principe

Nakon što je podrška za obične korisnike usavršena moguće je uvođenje podrške za napredne korisnike gde je bitno naći način da se u procesu ne naruši lakoća korišćenja originalnog interfejsa.

LITERATURA

[1] https://support.google.com/webmasters/answer/1211158?hl=en
[2] http://w3techs.com/technologies/overview/content_management/all/
[3] https://www.google.com/webmasters/markup-helper
[4] http://html5doctor.com/microdata/
[5] http://schema.org/docs/gs.html#advanced_missing
[6] https://github.com/substrance/schemantic
[7] http://www.w3.org/wiki/WebSchemas/sameAs
[8] http://googleblog.blogspot.co.uk/2012/05/introducing-knowledge-graph-things-not.html
[9] http://wiki.freebase.com/wiki/DBPedia
[10] http://www.google.com/webmasters/tools/richsnippets
[11] http://rdface.aksw.org/
[12] http://www.annotationframework.org/
[13] http://vimeo.com/33057828

ABSTRACT

Although the technologies of the semantic web have long been at the mature level, the adoption of the semantic web has been slow. For there to be a reason for development of semantic web services there needs to exist a lot of semantic data that those services could use. Creating semantic data by means of automatic extracion and processing of existing data on the web is the first step that is mature enough. Second step is to provide the tools, for regular users who create most of the content on the web, for easy creation of the semantic data. In this paper, we have developed and presented a tool for visual semantic annotation of web pages using microdata format. This tool is a plugin for TinyMCE rich-text editor so it can be used in a variety of content management systems (CMS).

Visual web content annotation with semantic data

Nikola Vulićević