

GRZEGORZ KOLARZ*

SYSTEM DO GENEROWANIA ONTOLOGII NA PODSTAWIE DIAGRAMÓW UML

SYSTEM TO ONTOLOGY GENERATION FROM UML DIAGRAMS

Streszczenie

Zastosowanie sieci semantycznych daje możliwość precyzyjnego przetwarzania danych. Jednak ciągle obserwujemy niedostatek ontologii związanych z przetwarzanymi zagadnieniami. Z drugiej strony wiele projektów ma specyfikacje stworzone w notacji UML. W artykule przeanalizowano technologie wykorzystywane do definiowania ontologii, notację UML oraz zaprezentowano program do tworzenia ontologii na podstawie diagramów UML.

Słowa kluczowe: sieci semantyczne, rdf, owl, uml

Abstract

Application of semantic web enables computers to more efficient and more precise data processing. However, we are still facing lack of ontologies that covers some particular domain. On the other hand there are projects containing UML specification. The paper characterized technologies used to ontology creation. Program to ontology generation from UML diagrams was presented.

Keywords: semantic web, rdf, owl, uml

* Mgr inż. Grzegorz Kolarz, Instytut Informatyki Stosowanej, Wydział Mechaniczny, Politechnika Krakowska.

1. Wstęp

Dzięki rozwojowi Internetu mamy dostęp do ogromnej ilości informacji. Treści te są łatwe do odczytania dla człowieka, natomiast często stosunkowo trudne do zinterpretowania przez maszynę. Przyczyną takiego stanu rzeczy jest sposób zapisu dokumentów internetowych. Jego istotą jest graficzna reprezentacja tekstu. Metoda wygodna dla człowieka bywa bardzo kłopotliwa do przetwarzania przez komputery. Na przykład w formatowaniu pliku html oznacza się, że dany tekst ma być wyświetlony kursywą, ale podczas przetwarzania maszynowego trudno jest zrozumieć cel i znaczenie takiego formatowania.

Wyszukiwarki internetowe stosują skomplikowane algorytmy w celu określenia istotności i trafności informacji przeszukiwanych stron [1]. Algorytmów oceny treści jest wiele, nie sposób je tutaj wymienić. Ich wspólną cechą jest syntaktyczna ocena wyników wyszukiwania. Powoduje to pojawienie się wyników, których użytkownik nie chciał wyszukiwać. Na przykład po wpisaniu w wyszukiwarkę internetową frazy "uml", pojawiają się wyniki wyszukiwania kierujące równocześnie do stron związanych z językiem UML, jak i strony Urzędu Miasta Łodzi.

Błędne wyciąganie wniosków to nie tylko domena wyszukiwarek internetowych. Internet jest potężnym zasobem informacji, dlatego też problem ten jest tam szczególnie widoczny, jednak zjawisko to pojawia się również w systemach podejmowania decyzji, tzn. wszędzie tam, gdzie używane są heurystyczne metody drażenia danych. Dzieje się tak ze względu na brak dostarczenia kompletnego kontekstu dla poszukiwanej informacji oraz braku możliwości zdefiniowania w systemie eksperckim pełnej możliwej bazy wiedzy (w taki sposób, aby każdy z jego elementów oraz elementów pochodzących był w pełni zdefiniowany).

Sposobem rozwiązania tego typu problemów są sieci semantyczne. W przeciwieństwie do systemów omawianych powyżej opierają się na semantycznej analizie danych. Do danych, poza istotnymi informacjami dla użytkownika, dołączone są również metadane. Metadane opisują znaczenie semantyczne poruszanych zagadnień oraz mapują dane z danymi położonymi w innym miejscu (mającymi związek z inną dziedziną wiedzy) [2]. Dzięki takiemu podejściu sieci semantyczne są w stanie bardziej precyzyjnie ocenić znaczenie przetwarzanej informacji, odfiltrować nieistotne dane i udzielić użytkownikowi lepszej jakościowo odpowiedzi. Jednak, żeby to było możliwe, analizowane dokumenty muszą zawierać kompletny zbiór metadanych prezentowanej treści oraz w sieci musi istnieć wiele definicji (ontologii) w celu ustalenia wzajemnych relacji pomiędzy przetwarzanymi danymi. Niestety tworzenie ontologii od początku nie jest rzeczą trywialną, ale wymaga dużo wysiłku.

Mając na uwadze powyższe, można stwierdzić, że szersze wykorzystanie sieci semantycznych poprawi jakość przetwarzanych danych. Jakość polepszy się proporcjonalnie do ilości modeli dostępnych w sieci. W związku z tym istnieje potrzeba dostarczenia narzędzi umożliwiających szybsze i proste definiowanie nowych definicji (ontologii).

Z drugiej strony należy zauważyć, że powszechnym językiem modelowania rzeczywistości jest język UML. Pozwala on na modelowanie systemów nie tylko związanych z informatyką. Stworzono wiele modeli za pomocą diagramów UML, w związku z tym możliwość przekształcenia ich w ontologie może przynieść duże efekty w tej dziedzinie.

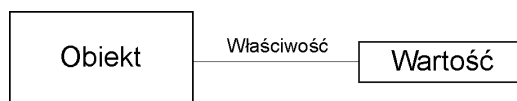
2. Analiza koncepcji zastosowanych w sieciach semantycznych

Podstawowym elementem opisującym sieć semantyczną jest ontologia. Ontologia to formalna definicja koncepcji pewnego obszaru wiedzy czy obiektów rzeczywistości. Składa się z klas obiektów, relacji między nimi i norm, które mają zastosowanie w danej dziedzinie wiedzy[2]. Do definiowania ontologii używa się języka OWL (*ang. Ontology Web Language*), który używa formatu opisu zasobów RDF (*ang. Resource Definition Format*).

2.1. Format opisu dokumentu RDF

Język RDF (*ang. Resource Description Framework*) pozwala na opisywanie zasobów w sieci Web. Celem RDF jest opis wiedzy zawartej w tych zasobach w postaci łatwo przetwarzanej przez programy komputerowe, dlatego też zdecydowano, że RDF jest językiem opartym o język XML [3]. Dzięki jego wykorzystaniu i dostępności parserów na praktycznie każdej platformie komputerowej może być łatwo przetwarzany przez różne systemy komputerowe.

Formalnie rzecz ujmując, wiedzę w RDF formalnie zapisuje się jako trójkę: obiekt (musi być zdefiniowany w sposób jednoznaczny, a jego definicja powinna zapewniać dostęp do obiektu z każdego miejsca; najczęściej stosowany jest tutaj adres URI), właściwość i wartość [4]. Schematyczny zapis trójki został zaprezentowany na rysunku 1.



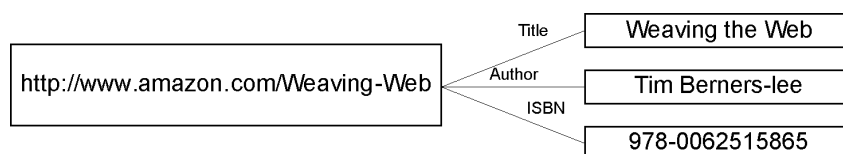
Rys. 1. Schematyczny zapis trójki RDF
Fig. 1. Schematic notation of the RDF triple

Przykładowy opis książki *Weaving the Web* [1] z użyciem RDF został zaprezentowany na listingu 1. Natomiast rysunek 2 prezentuje jego schematyczną reprezentację.

```

1: <?xml version="1.0"?>
2: <rdf:RDF
3:   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
4:   xmlns:si="http://www.w3schools.com/rdf/">
5:   <rdf:Description rdf:about="http://www.amazon.com/Weaving-
6:   Web">
7:     <si:title>Weaving the Web</si:title>
8:     <si:author>Tim Berners-lee</si:author>
9:     <si:isbn>978-0062515865</si:isbn>
10:  </rdf:Description>
11: </rdf:RDF>

```



Rys. 2. Opis przykładowej książki z wykorzystaniem RDF
Fig. 2. Example book description using RDF

W linii 1 została zawarta deklaracja dokumentu zgodnie ze specyfikacją XML. Linie 2,3,4 to utworzenie elementu RDF oraz deklaracja przestrzeni nazw dokumentów RDF. Jest to informacja dla parsera, że zawartość tego elementu jest zgodna ze schematem rdf zdefiniowanym w schematach rdf oraz si. W liniach 5,6 został utworzony element Description z przestrzeni nazw zadeklarowanej jako rdf, wraz z atrybutem about. Wartość tej deklaracji precyzyjnie definiuje element będący przedmiotem opisu. Linie 7, 8, 9 definiują kolejne dane związane z przetwarzanym elementem: tytuł autor, numer ISBN. Linie 10 i 11 zamykają deklarację elementów Description oraz RDF.

2.1. Język OWL

Użycie języka RDF pozwala jedynie na zdefiniowanie opisu pojedynczych elementów. Do stworzenia gramatyki danej dziedziny konieczne jest zdefiniowanie relacji pomiędzy wieloma elementami RDF [4]. Zatem ontologie to zestawy stwierdzeń zapisanych w RDF definiujące relacje pomiędzy pojęciami będącymi przedmiotem ontologii oraz opisujące reguły wnioskowania [5]. Najciekawszymi elementami do definiowania języka Owl są:

- Annotacje: *rdfs:label* (etykieta), *rdfs:comments* (komentarz), *rdfs:seeAlso* (zobacz również), *rdfs:isDefinedBy* (jest zdefiniowany przez), *owl:deprecated* (przestarzały), *owl:priorVersion* (poprzednia wersja), *owl:incompatibleWith* (niezgodny z),
- Element klasy: *rdf:type* (typ), *owl:sameAs* (taki jak), *owl:differentFrom* (różny od),
- Klasy: *rdfs:subClassOf* (podklasa, dziedziczenie), *owl:equivalentClass* (klasa analogiczna), *owl:Thing* (rzecz, najwyższy element w definicji owl), *owl:intersectionOf* (relacja przecięcia), *owl:unionOf* (relacja złączenia), *owl:complementOf* (relacja dopełnienia), *owl:disjointWith* (relacja rozłączności).

3. Analiza języka UML pod kątem przydatności do tworzenia OWL

UML to zunifikowany język modelowania służący do modelowania otaczającej nas rzeczywistości, tzw. dziedziny problemu, z wykorzystaniem pojęć obiektowych [6]. Do modelowania wykorzystuje się 13 rodzajów diagramów. Najciekawszym diagramem z punktu widzenia transformacji do formatu OWL jest diagram klas. W diagramie klas możemy wyróżnić klasy oraz relacje zachodzące pomiędzy nimi: dziedziczenie, kompozycja, a także liczebność elementów. Te elementy umożliwiają tworzenie statycznych modeli dużej części systemów informatycznych. Elementami odpowiadającymi im z języka OWL są kolejno: *rdfs:Class*, *rdfs:subClassOf*, *owl:ObjectProperty*, *owl:cardinality*. Dodatkowo wszystkie właściwości mogą zostać wyeksportowane, tak jak są zdefiniowane, gdyż będzie to poprawne z punktu widzenia poprawności schematu RDF.

4. Opis sposobu tworzenia ontologii na podstawie diagramów UML

Do generowania ontologii na podstawie diagramów UML wybrano program StarUML. O wyborze tego oprogramowania zdecydowało kilka faktów:

- oprogramowanie jest całkowicie darmowe,
- dostarcza programistom system tworzenia własnych rozszerzeń (kluczowa cecha),

- jest to oprogramowanie typu open source, co jest ważne przy tworzeniu niestandardowych rozszerzeń,
- umożliwia import formatu plików Rational Rose. Wiele programów do modelowania UML umożliwia zapis w tym formacie plików, w związku z tym istnieje możliwość otwarcia takiego diagramu w programie StarUML, a następnie utworzenie na jego podstawie ontologii.

Sam proces generowania ontologii nie jest procesem skomplikowanym. Po kliknięciu prawym przyciskiem na diagramie klas pojawia się dodatkowa opcja „Export as ontology” umożliwiająca wyeksportowanie wybranego diagramu klas jako ontologii.

Konwerter wybiera z diagramu klas tylko te elementy, które można wykorzystać w tworzonej ontologii. Wszystkie pozostałe informacje zawarte w diagramie klas (np. nazwy funkcji) są ignorowane.

5. Wnioski

Stworzone rozszerzenie znacznie przyspiesza proces generowania ontologii. Należy jednak pamiętać o tym, że wygenerowane pliki nie zawierają pełnej ontologii posiadającej wszystkie elementy. Wynika to z faktu, że OWL oferuje większe możliwości definiowania struktur niż UML. Znając tylko diagram klas, nie posiadamy wszystkich niezbędnych informacji do zdefiniowania ontologii. Wygenerowane dane należy zatem traktować, jako zdefiniowane encje w sensie RDF (z małymi dodatkami pochodzącymi z OWL). Dlatego też po wygenerowaniu szkieletu należy uzupełnić brakujące informacje, edytując wyeksportowane pliki. Należy dodać brakujące elementy ontologii oraz zmapować wygenerowane obiekty na obiekty innych ontologii (np. *FoF – Friend of Friend*[4]). Można do tego celu użyć jednego z wielu narzędzi przeznaczonych do tworzenia i edycji ontologii, np. Protégé OWL.

Literatura

- [1] Berners-Lee T., *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web*, Harper Paperbacks, 2000.
- [2] Hebel J., Fisher M., Blace R., Perez-Lopez A., *Semantic Web Programming*, John Wiley & Sons, 2009.
- [3] Ma Z., Wang H., *The Semantic Web for Knowledge and Data Management: Technologies and Practices*, IGI Global, 2009.
- [4] Pollock J.T., *Semantic Web For Dummies*, Wiley & Sons, 2009.
- [5] Allemang D., Hendler J., *Semantic Web for the Working Ontologist: Modeling in RDF, RDFS and OWL*, Morgan Kaufmann Publishers, 2008.
- [6] Fowler M., *Uml w kropelce, wersja 2.0*, LTP Oficyna Wydawnicza, 2005.