

Krzysztof GOCZYŁA, Aleksandra KARPUS  
Politechnika Gdańska, Wydział Elektroniki, Telekomunikacji i Informatyki

## PROBLEMY OCENY JAKOŚCI ONTOLOGII

**Streszczenie.** W artykule podjęto rozważania na temat tego, czym jest jakość ontologii, jak zmierzyć jakość istniejących ontologii i jak tworzyć ontologie wysokiej jakości. Dokonano przeglądu istniejących metryk ontologii, które mogą posłużyć do oceny jakości ontologii. Przedstawiono problem zapewniania i oceny jakości ontologii modularnych jako ważny problem badawczy w obliczu pojawiania się skomplikowanych, trudnych do użycia i modyfikacji płaskich ontologii.

**Słowa kluczowe:** ontologia, metryki, jakość, modularyzacja

## ANALYSIS OF EVALUATION OF ONTOLOGY QUALITY

**Summary.** This paper elaborates on what is the quality of the ontology, how to measure the quality of existing ontologies, and how to create high-quality ontologies. A review of existing ontology metrics that can be used to evaluate the quality of ontology is presented. The paper also discusses the problem of quality assurance and quality evaluation of modular ontologies as an important research problem in the face of the prevailing emergence of complicated, difficult to use and modify flat ontologies common in the Semantic Web.

**Keywords:** ontology, metrics, quality, modularization

### 1. Wprowadzenie

Ze względu na wzrost znaczenia ontologii w inżynierii systemów informatycznych jako środka wyrażania ogólnie akceptowanego modelu świata coraz częściej pojawiają się pytania o jakość ontologii. Tworzenie dobrych jakościowo ontologii wydaje się kluczowe dla idei Semantic Web, aby mogły być one wielokrotnie wykorzystywane przez różnych użytkowników, co przyczyni się do uporządkowania zasobów Internetu.

Ontologie powstały, aby rozumiane nie tylko przez ludzi, lecz także przez komputery. Systemy wnioskujące, które przetwarzają bazy wiedzy, mają ograniczone możliwości. Kluczowe jest tworzenie ontologii w taki sposób, by stworzony model był poprawnie interpretowany.

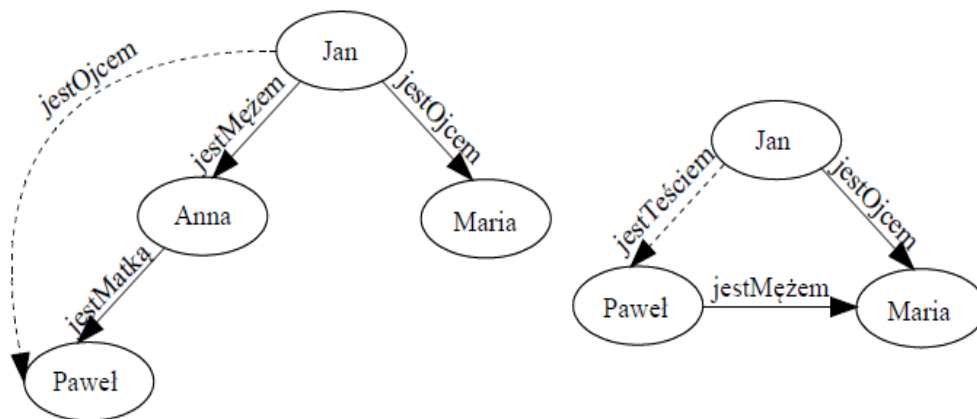
Na przełomie wieków powstało wiele prac poświęconych jakości w inżynierii ontologii. Jednak żaden z autorów nie określił jednoznacznie, czym tak właściwie jest jakość ontologii. W [7] autorzy przedstawili jakość ontologii jako jej adekwatność do realizacji celów. W [8] oraz [10] skupiono się na formalnych aspektach poprawności bazy wiedzy. Zatem jakość można rozpatrywać na kilka sposobów: jako adekwatność do modelowanej rzeczywistości, jako adekwatność do zamierzeń modelującego, a także jako spełnienie wymagań formalnych, takich jak istnienie modelu w sensie logiki pierwszego rzędu czy też brak konceptów niespełnialnych.

W dalszej części artykułu dokonano przeglądu istniejących metryk ontologii, które mogą posłużyć do oceny jakości ontologii. Przedstawiono również problem zapewniania i oceny jakości ontologii modularnych jako ważny problem badawczy w obliczu pojawiania się skomplikowanych, trudnych do użycia i modyfikacji płaskich ontologii.

## 2. Ocena jakości ontologii

### 2.1. Wykrywanie sprzeczności

Jednym z podstawowych i niezaprzeczalnych kryteriów określających, czy ontologia jest dobrej jakości, jest brak wewnętrznych sprzeczności, czyli posiadanie przynajmniej jednego modelu. W [1] zostały omówione problemy sprzeczności w ontologii. Autorzy zwrócili uwagę na to, że nie wszystkie rodzaje sprzeczności da się wykryć bez wiedzy eksperckiej.



Rys. 1. Przykład sprzeczności wywnioskowanych zdań (na podst. [1])

Fig. 1. Example of conflict between simplifications (based on [1])

Na rysunku 1 przedstawiono fragment bazy wiedzy, w której dokonano dwóch z pozoru oczywistych założeń:

- Jeśli  $x$  jest *Mężem*  $y$  i  $y$  jest *Matką*  $z$ , to  $x$  jest *Ojcem*  $z$ .
- Jeśli  $x$  jest *Ojcem*  $y$  i  $z$  jest *Mężem*  $y$ , to  $x$  jest *Teściem*  $z$ .

Wnioski wynikające z tych założeń zostały oznaczone przerywanymi strzałkami. Jeśli przyjrzemy się im dokładniej, to zauważymy, że *Jan* jest jednocześnie ojcem i teściem *Pawła*. Nie jest to prawdą, jednak silnik wnioskujący – bez dodatkowych aksjomatów – nie będzie posiadał takiej wiedzy o świecie jak ludzie i tej sprzeczności nie stwierdzi.

Jako rozwiązanie zaproponowano wykorzystanie reguł języka RuleML, w których miałyby zostać wyrażona wiedza ekspercka niewystępująca w bazie wiedzy. Utworzony przez autorów prototyp aplikacji umożliwił wczytanie lub wprowadzenie reguł opartych na wiedzy eksperckiej po wczytaniu bazy wiedzy w formie dokumentu RDF. Następnie wszystkie asercje wyprowadzone w trakcie procesu wnioskowania zostają dołączone do bazy wiedzy. Ten proces jest powtarzany dopóty, dopóki nie istnieją już żadne nowe wnioski. Wszystkie asercje i ograniczenia zostają przetłumaczone z RDF na reguły i dodane do bazy reguł. Sprzeczne zdania zostaną wykryte przez zapytania do bazy reguł.

## 2.2. Redundantność konceptów

Ważnym czynnikiem obniżającym jakość ontologii jest redundantność konceptów. W [10] zaproponowano metodę wyszukiwania podobnych konceptów, która opiera się na jednoznaczności (ang. *univocality*) w rozumieniu filozoficznym – jako „wspólnej interpretacji natury rzeczywistości”. Wyróżnione zostały dwa rodzaje naruszenia jednoznaczności: wyrażenie tego samego znaczenia na różne sposoby oraz podobieństwo konceptów. Autorzy w swej pracy skupili się na ontologii GO (*Gene Ontology*), która wspiera synonimy, zatem w swych rozważaniach pominęli ten przypadek. Zaobserwowano, że 65,3% wyrażen z GO zawiera inne wyrażenia z GO. Mogą też zawierać zwroty z ontologii ChEBI (*Chemical Entities of Biological Interest*).

Tabela 1

Przykład klastrów w ontologii (za [10])

| { <i>GTERM embryo sac</i> } |   | { <i>CTERM CTERM oxidati</i> } |   |
|-----------------------------|---|--------------------------------|---|
| GO:0009562                  | <b>embryo sac</b> nuclear migration<br>cellularization of the <b>embryo sac</b> | GO:0019327                     | <b>oxidation</b> of lead sulfide              |
| GO:0009558                  |   | GO:0018158                     | protein amino acid <b>oxidation</b>           |
|                             |   | GO:0019604                     | toluene <b>oxidation</b> to catechol          |
|                             |   | GO:0019479                     | L-alanine <b>oxidation</b> to propanoate      |
|                             |   | GO:0019602                     | toluene <b>oxidation</b> via 3-hydroxytoluene |

Zaproponowana metoda opiera się na przekształcaniu zwrotów (nazw konceptów) i na klasteryzacji. Możliwe są trzy rodzaje transformacji:

- a) abstrakcja – identyfikowanie składowych wyrażeń i oznaczanie ich *GTERM*, jeśli są wyrażeniem z *GO*, lub *CTERM*, w przypadku gdy pochodzą z ChEBI;
- b) usuwanie spójników, przyimków i innych słów nieprzenoszących semantyki, np. *the*, *of*<sup>1</sup>;
- c) alfabetyczne sortowanie słów w wyrażeniu.

Po wykonaniu powyższych przekształceń wyrażenia są grupowane w klastry podobnych wyrażeń, na podstawie których ekspert dokonuje ostatecznej oceny redundantności konceptów. Przykłady klastrów w ontologii zostały przedstawione w tabeli 1.

### 2.3. Metryki służące ocenie jakości ontologii

W tym podrozdziale omówiono zaproponowane w [9] metryki, które mogą posłużyć do oceny jakości ontologii, oraz przedstawiono porównanie wartości wybranych metryk dla ontologii SWETO, TAP i GlycO. SWETO i TAP są ontologiami ogólnego zastosowania. Pierwsza, tworzona m.in. przez autorów [9], pokrywa takie dziedziny, jak publikacje, przynależność, geografia i terroryzm. Druga, opracowywana w Stanford University, jest podzielona na 43 dziedziny obejmujące m.in. publikacje, sport i geografę. Trzecia ontologia, rozwijana przez LSDIS Lab, jest bardziej specjalistyczna i dotyczy głównie glikanów.

Bogactwo dziedziczenia (ang. *Inheritance Richness*) jest zdefiniowane jako średnia liczba konceptów bezpośrednio subsumowanych przez inny koncept. Określa ona podział informacji na poziomy w drzewie hierarchii konceptów. Pozwala na rozróżnienie płaskich ontologii od tych, które są budowane w wielopoziomowe drzewo konceptów. Te drugie charakteryzują się niskim wskaźnikiem *IR*. Bogactwo dziedziczenia jest określone następującą formułą:

$$IR = \frac{\sum_{C_i \in C} |H(C_i)|}{|C|}^2, \quad (1)$$

gdzie:  $C$  oznacza zbiór konceptów będących konceptami nadrzędnymi,  $|H(C_i)|$  zaś to liczba konceptów bezpośrednio subsumowanych przez koncept  $C_i$ .

Średnia liczebność (ang. *Average Population*) konceptów pozwala określić, czy w bazie wiedzy znajduje się odpowiednia (względem przyjętego kryterium) liczba osobników. Jest ona zdefiniowana następująco:

$$P = \frac{|I|}{|C|}, \quad (2)$$

gdzie:  $I$  oznacza zbiór wszystkich osobników w bazie wiedzy,  $C$  zaś – zbiór wszystkich konceptów.

<sup>1</sup>Przykłady podano w języku, w którym jest tworzona ontologia *GO*.

<sup>2</sup>Autorzy [9] przedstawili inną postać tego wzoru, jednak ze względu na skomplikowanie zapisu został on przedstawiony tutaj w czytelniejszej postaci.

Ostatnią omówioną metryką jest ważność (ang. *Importance*) konceptu, zdefiniowana jako liczba osobników będących wystąpieniem danego konceptu w stosunku do wszystkich osobników.

$$Imp_i = \frac{|C_i(I)|}{|I|}, \quad (3)$$

gdzie:  $C_i(I)$  oznacza zbiór wszystkich, bezpośrednich i pośrednich, wystąpień pewnego konceptu  $C_i$ ,  $I$  zaś – zbiór wszystkich osobników.

W tabeli 2 zestawiono wartości powyższych metryk wg [9] dla ontologii SWETO, TAP i GlycO. Na podstawie metryki bogactwa dziedziczenia można wywnioskować, że TAP ma najbardziej płaską strukturę konceptów, a co za tym idzie – jest najbardziej ogólna. Im bardziej szczegółowa („wysoka”) jest hierarchia konceptów, tym bardziej specjalistyczne są zastosowania danej ontologii. SWETO jest ontologią z największą średnią liczbą osobników na koncept, z czego większość osobników to wystąpienia konceptów związanych z publikacjami. Ontologia GlycO ma strukturę konceptów o największej liczbie poziomów, jednak większość osobników (aż 90%) to wystąpienia jednego konceptu (N-glycan).

Tabela 2

Porównanie wybranych metryk dla ontologii SWETO, TAP i GlycO

| Ontologia | Bogactwo dziedziczenia | Średnia liczebność | Ważność  |                      |
|-----------|------------------------|--------------------|--|----------------------|
|           |                        |                    | Koncept  | Procent              |
| SWETO     | 4                      | 18482,21           | Publication<br>Scientific_Publication<br>Computer_Science_Researcher | ~60%<br>~60%<br>~40% |
| TAP       | 5,36                   | 21,94              | Musician<br>Athlete  | ~30%<br>~10%         |
| GlycO     | 1,56                   | 5,78               | N-glycan   | ~90%                 |

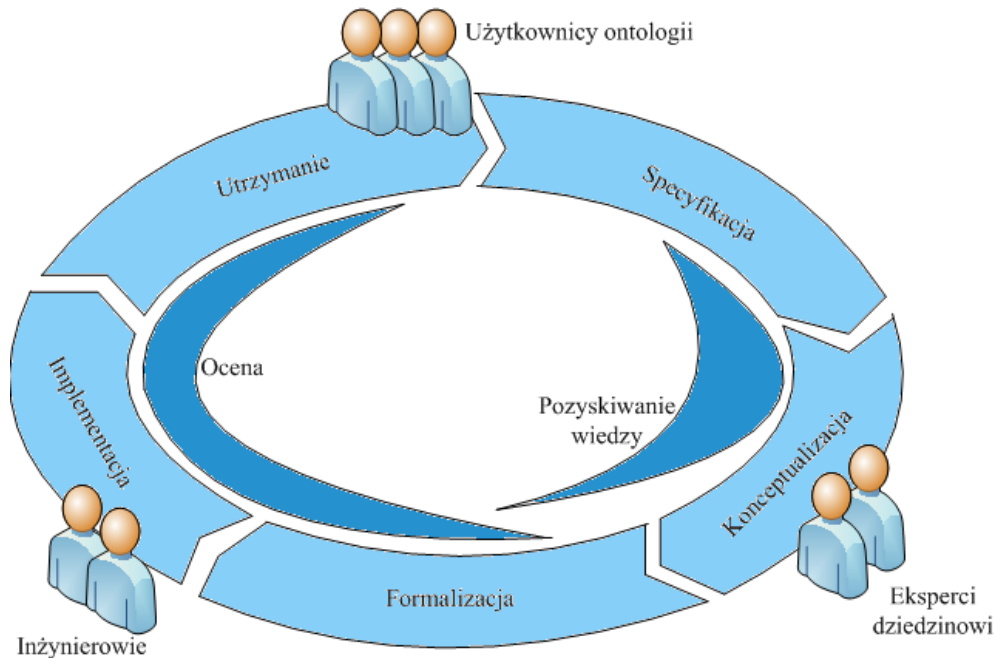
## 2.4. Podobieństwo jako wskaźnik jakości ontologii

W [7] zaproponowano zastosowanie miary podobieństwa (ang. *similarity*) do sprawdzania i zapewniania jakości ontologii w kontekście adekwatności do modelowanej rzeczywistości oraz adekwatności do zamierzeń modelującego. Autorzy podkreślają, że proponowane przez nich rozwiązanie nie zależy od metodyki tworzenia ontologii, jednak bazują ściśle na podejściu zaczerpniętym z inżynierii oprogramowania (*Methontology*). W dalszej części zostało ono krótko omówione.

### 2.4.1. Methontology

W podejściu *Methontology* wyróżnia się pięć etapów wytwarzania ontologii: specyfikacja (ang. *specification*), konceptualizacja (ang. *conceptualization*), formalizacja (ang. *formalization*), implementacja (ang. *implementation*) i utrzymanie (ang. *maintenance*). Podobnie jak w inżynierii oprogramowania specyfikacja oznacza pozyskiwanie wymagań, rozumianych

w tym wypadku jako przeznaczenie bazy wiedzy, zakres dziedzinowy oraz potrzebna ekspresywność. Wiedza dziedzinowa na etapie konceptualizacji jest strukturalizowana, na etapie formalizacji zaś – przekształcana na model formalny. Etapy implementacji i utrzymania są rozumiane tak jak w przypadku oprogramowania.



Rys. 3. Etapy *Methontology* i ich związek z aktorami (na podst. [7])

Fig. 3. Phases of *Methontology* and their relation to actors (based on [7])

W procesie wytwarzania ontologii wyróżniono trzy typy aktorów:

- użytkownicy ontologii – definiują wymagania i oceniają, czy utworzona ontologia je spełnia;
- eksperti dziedzinowi – ustalają, jaka wiedza powinna być zawarta w ontologii;
- inżynierowie – sprawdzają, czy jakaś istniejąca ontologia spełnia wymagania użytkowników i ekspertów, albo implementują nową ontologię.

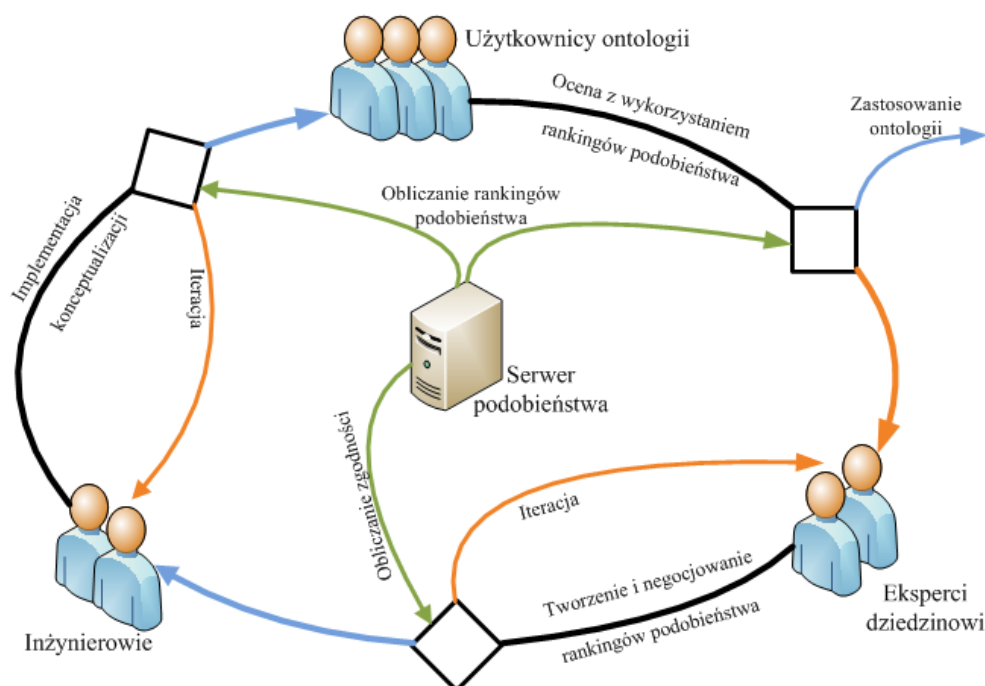
Rysunek 3 ilustruje etapy podejścia *Methontology* i ich związek z aktorami. Przedstawiono na nim również dwa dodatkowe, ważne w tej metodologii, etapy: pozyskiwanie wiedzy (ang. *knowledge acquisition*) i ocenę (ang. *evaluation*). Na etapie pozyskiwania wiedzy można korzystać z różnych metod, takich jak przypadki użycia, technika dwudziestu pytań czy sortowanie kart, znanych z inżynierii oprogramowania. Na etapie oceny ontologia jest walidowana względem zdefiniowanych na wcześniejszym etapie wymagań pod różnymi kątami, np. czy jej zawartość informacyjna jest wystarczająca do spełnienia wszystkich przypadków użycia.

#### 2.4.2. *Miara podobieństwa*

Omawiane przez autorów [7] rozwiązanie SIM-DL jest niesymetryczną miarą, która porównuje badany koncept (ang. *search concept*)  $C_s$  ze zbiorem konceptów docelowych

(ang. *target concepts*)  $\{C_1, \dots, C_m\}$  z ontologii. Podobieństwo jest zdefiniowane jako funkcja  $sim(C, D)$ . Całkowite podobieństwo dwóch konceptów jest znormalizowaną sumą podobieństw wszystkich części definicji konceptu (np. konceptów subsumowanych przez koncept). Wartość podobieństwa równa 1 oznacza, że koncepty są identyczne, 0 zaś, że są zupełnie różne. SIM-DL jako wynik zwraca ranking podobieństwa (ang. *similarity ranking*) konceptów uporządkowanych malejąco względem wartości podobieństwa.

Rysunek 4 przedstawia zastosowanie miary podobieństwa w *Methontology*, zaproponowane w [7]. Główną ideą jest dodanie wskaźnika jakości ontologii względem dopasowania do celów (ang. *fitness for purposes*).



Rys. 4. Rola podobieństwa w procesie inżynierii ontologii (na podst. [7])

Fig. 4. The role of similarity within the ontology engineering process (based on [7])

Na etapie specyfikacji identyfikuje się koncept badany  $C_s$  i zbiór konceptów docelowych oraz ranking podobieństwa  $SR_{de}$  pomiędzy nimi. Można to zrobić np. przez połączenie metod dwudziestu pytań i sortowania kart. Na podstawie wywiadów identyfikujemy koncepty, które później eksperci dziedzinowi sortują, tworząc klastry. Koncept z klastra najczęściej występujący w raportach z wywiadów będzie dla ekspertów najważniejszy, a zatem jest najlepszym kandydatem na stanie się konceptem badanym. Pozostałe będą konceptami docelowymi. Dla tak przygotowanych zbiorów każdy z ekspertów tworzy własny ranking  $SR_{de}^e$ . Jeśli są one wystarczająco zgodne<sup>4</sup>, wyznacza się na ich podstawie uśredniony ranking podobieństwa  $SR_{de}$ . Jeśli nie są zgodne, może to oznaczać różne rozumienie dziedziny przez eks-

<sup>3</sup>Ze względu na duże skomplikowanie stosownych formuł i ich mały wkład merytoryczny w tę pracę, nie będziemy ich tutaj przytaczać. Zainteresowanych czytelników odsyłamy do [6].

<sup>4</sup>Zgodność i sposób jej określania nie są tematem tego rozdziału, dlatego definicja została pominięta.

pertów. Należy powtarzać ten etap tak długo, aż rankingi będą zgodne. Uśredniony  $SR_{de}$  zostaje dołączony do wymagań. Po etapie implementacji inżynierowie, wykorzystując serwer podobieństwa SIM-DL i wtyczkę do edytora ontologii Protégé, obliczają ranking podobieństwa  $SR_{oe}$ . Jeśli jest on zgodny z  $SR_{de}$ , to znaczy, że ontologia jest adekwatna do zamierzeń modelujących. Jeśli tak nie jest, to wymagana jest kolejna iteracja implementacji. W przypadku gdy po kilku iteracjach nie ma zgodności, należy wrócić do etapu specyfikacji. Zamiast implementować ontologię, inżynierowie mogą obliczyć ranking podobieństwa dla istniejących ontologii i sprawdzić, czy któraś z nich spełnia zadane wymagania.

Ostatnim etapem jest weryfikacja przez użytkowników, którzy również obliczają własny ranking podobieństwa dla tych samych konceptów. Podobnie jak dla ekspertów, ranking  $SR_{ou}$  jest uśrednieniem rankingów użytkowników.  $SR_{ou}$  jest porównywany z  $SR_{oe}$  i w przypadku braku zgodności cały proces jest inicjowany od początku. Jeśli rankingi są zgodne, powstała ontologia jest adekwatna do realizacji zamierzonych celów.

### 3. Problemy oceny ontologii modularnych

Inżynieria ontologii wciąż znajduje się na wczesnym etapie rozwoju. Aktualny etap można przyrównać do stanu inżynierii oprogramowania z wczesnych lat 70. ubiegłego stulecia, kiedy to rozumiano, że nie można budować dużych systemów oprogramowania jako tworów monolitycznych i że zwykły podział na procedury rozumiane jako kawałki kodu źródłowego nie wystarcza. Wówczas rozwinęło się podejście modułarne do tworzenia oprogramowania, koncepcyjnie ortogonalne do przyjętego paradygmatu. Podobnie dalszy rozwój inżynierii ontologii i – co za tym idzie – praktycznych systemów z bazami wiedzy jest uzależniony od istnienia i rozwoju sensownych (czyli stosowalnych w praktyce) metod tworzenia ontologii podzielonych na moduły oraz od istnienia profesjonalnych narzędzi wspierających.

Jedną z propozycji, wypracowanych w *Knowledge Management Group at Gdańsk University of Technology* (KMG@GUT), jest podejście kontekstowe do budowy ontologii modularnych. Po szczegóły tej metody odsyłamy Czytelnika np. do [3] i [11]. Na potrzeby niniejszego artykułu skoncentrujemy się na kilku aspektach specyficznych dla tej metody. Przez kontekst rozumiany jest tu zbiór pewnych zmiennych, zwanych zmiennymi kontekstowymi, wraz z ich konkretnymi wartościami. Zbiór ten wyznacza pewien wycinek świata rzeczywistego, który chcemy opisać. Kontekst może być np. wyznaczony przez: obszar geograficzny, okres w historii, funkcję społeczną, organizację itd. Podstawowym modułem kontekstowej bazy wiedzy jest wystąpienie kontekstu, które jest częścią asercjonalną ontologii osadzonej w danym kontekście. Wystąpienia kontekstów wyznaczonych przez ten sam zbiór zmiennych kontekstowych, ale o innych wartościach, są przypisane do jednego typu kontekstu, który zawiera część terminologiczną ontologii. Typy kontekstów tworzą klasyczną hierarchię dziedzic-



czenia, natomiast wystąpienia kontekstów powiązane są ze sobą hierarchią agregacji. Dzięki temu możemy tworzyć i analizować „większe” światy jako złożone z „mniejszych”. W tych „większych”, zagregowanych światach prawdziwe są wnioski wysnute na podstawie asercji zawartych w opisie światów „mniejszych” i aksjomatów zawartych w typach kontekstów. Zauważmy, że ten sam osobnik może znajdować się równocześnie w wielu wystąpieniach kontekstów (np. konkretny pracownik naukowy z wyższej uczelni może być zarazem zawodnikiem drużyny piłkarskiej albo ktoś, kto jest lekarzem w Polsce, może nie być uznawany za lekarza w Niemczech), co pozwala na swoisty przepływ faktów pomiędzy różnymi wystąpieniami kontekstów (różnymi światami). I tak np. mimo że dana osoba nie jest uznawana w danym kraju za lekarza, to fakt, że jest lekarzem w innym kraju implikuje fakt, że ta osoba potrafi reanimować, gdyż własność „umie reanimować” jest wspólna dla wszystkich wystąpień kontekstów danego typu.

Z tego pobieżnego siłą rzeczy opisu widać wyraźnie problemy dotyczące zarówno danej ontologii kontekstowej, jak i problemy określenia samego procesu tworzenia takich ontologii. Ontologia kontekstowa, jako ontologia podzielona na moduły, może być oceniana na dwóch płaszczyznach: na płaszczyźnie pojedynczych modułów i na płaszczyźnie międzymodułowej. Ocena poszczególnych modułów sprowadza się do oceny „klasycznych”, niemodularnych ontologii, gdyż zarówno wystąpienie kontekstu, jak i typ kontekstu są ontologiami niemodularnymi. Znacznie poważniejszym problemem jest ocena strukturalna całej ontologii kontekstowej, gdyż tu pojawiają się zupełnie nowe problemy i pytania. Oto niektóre z nich:

- Czy właściwie dokonano wyboru zmiennych kontekstowych i czy poprawnie określono wartości tych zmiennych?
- Czy poprawnie skonstruowano hierarchie agregacji wystąpień kontekstów i hierarchie dziedziczenia typów kontekstów?
- Czy te hierarchie umożliwiają jak najszerszy przepływ wniosków pomiędzy poszczególnymi kontekstami i czy przepływające między kontekstami wnioski nie mogą być wzajemnie sprzeczne?

Te i inne pytania wskazują w istocie na potrzebę opracowania systematycznej metody tworzenia ontologii kontekstowych. Na przykład już odpowiedź na pierwsze z postawionych pytań nie jest łatwa, zważywszy na to, że zbiór możliwych zmiennych kontekstowych jest nieskończony i wybór tych właściwych dla konkretnej bazy wiedzy może być niełatwy. Konstrukcja hierarchii typów kontekstów przypomina konstrukcję hierarchii dziedziczenia w klasycznych modelach obiektowych, jednak kryteria tworzenia nadtypów i podtypów są zasadniczo różne. Konstrukcja hierarchii agregacji wystąpień kontekstów jest nieco prostszym zagadnieniem, gdyż tu podstawowym kryterium wydaje się uwzględnianie tych samych zmiennych kontekstowych.

Postawione powyżej problemy są specyficzne dla modularnych ontologii kontekstowych. Te jednak są tylko pewnym szczególnym przypadkiem ontologii modularnych. W ramach

prac prowadzonych przez KMG@GUT w projekcie SyNaT<sup>5</sup> opracowano bardziej ogólne podejście do tworzenia modularnych ontologii, oparte na koncepcji *tarsetu* jako podstawowego modułu ontologicznego (w [2] stosowana jest wcześniejsza nazwa: *konglomerat*). W ontologii modularnej tarsety są ze sobą powiązane *sprzęgami* definiującymi, w formie wyrażeń algebry tarsetów, zależności semantyczne pomiędzy nimi (w tym przepływy wniosków). Relacje zachodzące pomiędzy modułami ontologii kontekstowych są szczególnymi przypadkami sprzęgów. Opracowanie systematycznej, inżynierskiej metody tworzenia ontologii podzielonych na tarsety, z określeniem adekwatnych dla danej aplikacji sprzęgów, jest poważnym wyzwaniem badawczym i będzie przedmiotem dalszych prac.

#### 4. Podsumowanie

W niniejszym artykule dokonano przeglądu różnych podejść do zapewniania jakości ontologii, jednak żadne z nich nie gwarantuje stworzenia ontologii wysokiej jakości. Można by pomyśleć, że dobrym rozwiązaniem byłoby połączenie wszystkich lub przynajmniej kilku istniejących metod. Jednak takie podejście doprowadzi do dużej złożoności budowanej ontologii, a w rezultacie – do jej bezużyteczności. Istnieją również inne, sprawdzone w praktyce, jednak nie zawsze wygodne rozwiązania opisane w [5] i [4].

Kierując się wyborem metody, należy przede wszystkim wziąć pod uwagę przeznaczenie bazy wiedzy, gdyż w ten sposób będzie można przewidzieć, jakie jej cechy będą istotne przy jej zastosowaniach. Sytuacja komplikuje się, gdy mamy do czynienia z dużymi ontologiami, które powinny być podzielone na moduły, lub z ontologiami, które z założenia mają być modularne. Opracowanie metod przekształcania ontologii na modularne i dobrych praktyk tworzenia ontologii modularnych, np. ontologii kontekstowych, jest ważnym problemem dla rozwoju praktycznych baz wiedzy. W tym sensie kluczowe wydaje się również zaproponowanie metryk pozwalających zmierzyć jakość tworzonych rozwiązań, zarówno ontologii modularnych, jak i samej modularyzacji.

#### BIBLIOGRAFIA

1. Arpinar I. B., Giriloganathan K., Aleman-Meza B.: Ontology Quality by Detection of Conflicts in Metadata. Proc. of Fourth International EON Workshop: Evaluation of Ontologies for the Web, 2006.

---

<sup>5</sup>„Uniwersalna, otwarta, repozytoryjna platforma hostingowa i komunikacyjna dla sieciowych zasobów wiedzy dla nauki, edukacji i otwartego społeczeństwa wiedzy” (SyNaT), grant NCBiR nr SP/I/1/77065/10.

2. Goczyła K., Waloszek A., Waloszek W., Zawadzka T.: Modularized Knowledge Bases Using Contexts, Conglomerates and a Query Language, [in:] Bembenik R., Skonieczny L., Rybiński H., Niezgódka M. (eds.): Intelligent Tools for Building a Scientific Information Platform, Seria: Studies in Computational Intelligence, Vol. 390, Springer, Berlin / Heidelberg 2012, s. 179÷201.
3. Goczyła K., Waloszek A., Waloszek W.: Contextualization of a DL Knowledge Base, [in:] Calvanese D. et al. (eds.): Proc. of the 20th International Workshop on Description Logics DL'07, Bozen-Bolzano University Press, 2007, s. 291÷298.
4. Gómez-Pérez A.: Ontology Evaluation, [in:] Staab S., Studer R. (eds.): Handbook on Ontologies, Springer-Verlag, 2004, s. 251÷273.
5. Guarino N., Welty C. A.: An Overview of OntoClean. [in:] Staab S., Studer R. (eds.): Handbook on Ontologies, Springer-Verlag, 2004, s. 151÷171.
6. Janowicz K., Kessler C., Schwarz M., Wilkes M., Panov I., Espeter M., Baeumer B.: Algorithm. Implementation and Application of the SIM-DL Similarity Server. Second International Conference on GeoSpatial Semantics (GeoS 2007), Lecture Notes in Computer Science, Vol. 4853, Springer, 2007, s. 128÷145.
7. Janowicz K., Maue P., Wilkes M., Schade S., Scherer F., Braun M., Dupke S., Kuhn W.: Similarity as a Quality Indicator in Ontology Engineering. Proc. of the Fifth International Conference FOIS, IOS Press, Vol. 183, 2008.
8. Kohler J., Munn K., Ruegg A., Skusa A., Smith B.: Quality control for terms and definitions in ontologies and taxonomies. BMC Bioinformatics, BioMed Central, Vol. 7, No. 212, 2006.
9. Tartir S., Arpinar I. B., Moore M., Sheth A. P., Aleman-Meza B.: OntoQA: Metric-Based Ontology Quality Analysis. IEEE Workshop on Knowledge Acquisition from Distributed, Autonomous, Semantically Heterogeneous Data and Knowledge Sources, 2005.
10. Verspoor K., Dvorkin D., Cohen K. B., Hunter L. E.: Term Analysis for Ontology Quality Assurance. Proc. Of the 11th International Protégé Conference, Amsterdam 2009.
11. Waloszek A.: Hierarchiczna kontekstualizacja baz wiedzy. Rozprawa doktorska, Gdańsk 2011.

Wpłynęło do Redakcji 16 stycznia 2013 r.

## Abstract

Due to the increasing importance of ontologies in systems engineering as a means of expressing a generally accepted model of the world, more and more questions arise about the quality of ontologies. Quality can be perceived in several ways: as adequacy of the intended meaning at a specific point in time, as adequacy of the intended application area, as well as avoidance of conflicting data.

We present the problem, addressed in [1], of detection of domain specific, conflicting information. The proposal of detect them is based on rules in RuleML. An example of such a conflict is shown in Figure 1.

Another problem for ontology quality analysis is also occurrence of redundant concepts. It could be identified by applying three transformations of terms (abstraction, stopword removal and reordering) and then creating clusters of similar terms, as proposed in [10].

Several metrics for measuring the quality of ontologie are proposed in [9]. Some of them are: Inheritance Richness (1) which can distinguish a horizontal ontology from a vertical ontology, Average Population (2) wich assesses an average distribution of instances across all concepts, and Importance of a Concept (3) – the numer of instances that belong to a concept.

We also describe an extension of Methontology (presented in Figure 3) proposed in [7]. Autors suggest to use a measure of similarity to assure the quality of ontology being developed. The role of similarity within the ontology engineering proces is shown in Figure 4.

The paper also discusses the problem of quality assurance and quality evaluation of modular ontologies as an important research problem in the face of the prevailing emergence of complicated, difficult to use and modify flat ontologies that are recently common in the Semantic Web.

## Adresy

Krzysztof GOCZYŁA: Politechnika Gdańska, Wydział Elektroniki, Telekomunikacji i Informatyki, ul. G. Narutowicza 11/12, 80-952 Gdańsk, Polska, kris@eti.pg.gda.pl.

Aleksandra KARPUS: Politechnika Gdańska, Wydział Elektroniki, Telekomunikacji i Informatyki, ul. G. Narutowicza 11/12, 80-952 Gdańsk, Polska, aleksandra@karpus.info.