

Zastosowanie ontologii dziedzinowych w semantycznej sieci WWW

Jolanta Cybulka

Politechnika Poznańska, Instytut Automatyki i Inżynierii Informatycznej
Pl. M. Skłodowskiej-Curie 5, 60-965 Poznań

e-mail: Jolanta.Cybulka@put.poznan.pl

Streszczenie. W artykule scharakteryzowano ideę semantycznej sieci WWW oraz przedstawiono propozycję jej realizacji, w której wykorzystuje się metody klasyfikacji pojęć w postaci „ontologii dziedzinowych”, zaproponowanych w obszarze sztucznej inteligencji. Omówiono uniwersalne języki OIL/P-OIL przeznaczone do formułowania ontologii.

Słowa kluczowe: EuroWordNet, metadane, OIL, ontologia dziedzinowa, semantyczna sieć WWW, Top Ontology

1. Semantyczna sieć WWW

Sieć Internet oferuje użytkownikom wiele usług, wśród których do najważniejszych należą: usługi komunikacyjne (poczta elektroniczna, listy dyskusyjne, sieć USENET i in.), usługi związane z gospodarką elektroniczną (sklepy internetowe, reklama, aukcje internetowe itp.) oraz usługi informacyjne. Zwłaszcza te ostatnie, realizujące ideę globalnej „przetwarzania danych”¹ są pożądanymi przez osoby i instytucje zawodowo związane z przetwarzaniem danych. Usługi informacyjne w Internecie są realizowane przez systemy należące do kilku kategorii. Każdy system informacyjny ma architekturę klient/serwer, a różnią się one między sobą protokołami dostępu, sposobami reprezentacji danych i oferowanym zakresem usług. Współcześnie, podstawowy standard internetowego systemu informacyjnego wyznacza system WWW. Do reprezentowania danych w rozpatrywanym systemie zastosowano języki oznaczania (znacznikowania) zawartości dokumentów (ang. *markup languages*). Zdefiniowano kilka języków znacznikowania, w tym HTML (skr. ang. *HyperText Markup Language*) utworzony specjalnie do reprezentowania danych w systemie WWW (w formie dokumentów WWW). Ciekawa jest historia rozwoju tego języka – początkowo zawierał minimalny zestaw znaczników służących do adiacji tekstu, mimo, że od 1986 roku istniał już SGML (skr. ang. *Standard Generalized Markup Language*), stanowiący do dziś najbogatszy język² tej kategorii. Jednakże, autorzy HTML-a (m.in. Tim Berners-Lee) wzięwszy pod uwagę możliwości praktycznego zastosowania języka i aktualne zdolności przepustowe sieci postanowili rozwijać język stopniowo, w miarę potrzeb. Potrzeby narastały, dlatego w ciągu 10 lat (od 1990 r.) powstało kilka wersji języka, w tym ostatnia wersja 4.0. Jednakże, wszystkie wersje języka HTML mają poważny mankament, manifestujący się niemożnością rozszerzania zbioru znaczników, a tym samym definiowania struktury dokumentu (tzw. „typu dokumentu”). W efekcie prac

¹ Pojęcie „informacja” stosujemy intuicyjnie, podobnie terminy „dane” i „wiedza”.

² SGML jest również „metajęzykiem” gdyż umożliwia definiowanie innych języków, w tym np. HTML-a.

projektowych nad nowym, pozbawionym wspomnianych wad językiem znacznikowania, w 1998 roku powstał język XML (skr. ang. *Extensible Markup Language*), stanowiący kompromis między zbyt skomplikowanym SGML-em a zbyt ubogim HTML-em. Język ten, mimo że nie wyparł HTML-a, coraz częściej bywa stosowany do reprezentowania danych w systemie WWW.

W kilkuletniej historii systemu WWW można wyróżnić dwa etapy rozwoju i prognozować powstanie trzeciego ([9]). W początkowym okresie (pierwsza generacja) w systemie dominowały dokumenty HTML-owe zapisywane „ręcznie” o niezmienniej zawartości (dokumenty statyczne). Współcześnie, zapisy w HTML-u (i XML-u) mogą być generowane automatycznie, np. na podstawie zawartości bazy danych w odpowiedzi na zapytanie użytkownika (druga generacja, [8]). Jednakże również w tej wersji system WWW ma poważne ograniczenie, wynikające z braku możliwości automatycznego przetwarzania zawartości znaczeniowej dokumentów. W 1998 r. Tim Berners-Lee zgłosił potrzebę wprowadzenia zmian, które umożliwiłyby przetwarzanie zawartości dokumentów również przez programy, co jest ważne ze względu na ogrom istniejących zasobów informacyjnych i ich szybkie powiększanie się. Do realizacji wspomnianego założenia potrzebna jest wiedza nie tylko o składni, ale także o zawartości semantycznej dokumentów. Praca [4] zapoczątkowała badania zmierzające do powstania „semantycznej sieci WWW” (ang. *Semantic Web* – trzecia generacja), w której mogłyby sprawnie działać: wyszukiwarki, filtry i brokerzy informacji, systemy gospodarki elektronicznej i in. ([9]). Działające współcześnie programy tych kategorii mają liczne ograniczenia, wynikające właśnie z braku oznaczenia zawartości semantycznej dokumentów. W szczególności ([7]):

- wyszukiwanie informacji bazujące na tekstowym dopasowywaniu słów kluczowych często skutkuje dostarczeniem nieodpowiednich wyników w odpowiedzi na zapytania użytkownika; z tego samego powodu nie można znaleźć danych zawierających synonimy słów kluczowych;
- brak wiedzy o znaczeniach pojęć ogranicza możliwości automatycznej ekstrakcji informacji i jej integracji w jedną całość na podstawie różnych źródeł;
- wiedza o znaczeniu danych zawartych w dokumentach jest niezbędna w dużych systemach zarządzania informacją, których celem jest zapewnienie niesprzeczności, pełności i aktualności zgromadzonych danych;
- realizacja procesu automatycznej generacji zawartości dokumentu, w którego treści należy uwzględnić np. właściwości zdefiniowane przez użytkownika, będzie wymagała wiedzy o semantyce źródeł danych.

Narzędzia przetwarzające zawartość internetowych systemów informacyjnych wymagają integracji na kilku poziomach ([13, 14]):

- technicznym (protokoły komunikacji i dostępu do zasobów, np. TCP/IP oraz HTTP),
- syntaktycznym (języki reprezentacji struktury informacji w dokumentach, w szczególności języki znacznikowania, jak HTML, XML i jego odmiany),
- semantycznym.

Pierwsze dwa poziomy integracji są powszechnie stosowane, natomiast trzeci pozostaje w sferze projektów związanych z realizacją semantycznej sieci WWW.

Do definiowania znaczeń w tekstowych źródłach informacji często wykorzystuje się ontologie, stanowiące reprezentacje pojęć (słowniki pojęć) stosowanych w pewnej dziedzinie. Ontologie, wypracowane w obszarze sztucznej inteligencji, są powszechnie wykorzystywane w: inżynierii wiedzy, przetwarzaniu języka naturalnego, kooperacyjnych systemach informacyjnych i in. Najnowsze propozycje realizacji semantycznej sieci WWW również bazują na pojęciu ontologii. Jedną z metod przeznaczonych do zastosowania w systemie WWW jest OIL (skr. ang. *Ontology Inference Layer* [11, 13]) – jej charakterystyka będzie przedmiotem rozważań w punkcie 2. niniejszej pracy.

2. Ontologie dziedzinowe w języku OIL

Język OIL jest przeznaczony do definiowania semantyki dokumentów w systemie WWW. W szczególności, definiuje zestaw bazowych pojęć do modelowania ontologii wraz z ich składnią i formalną semantyką. Projektanci OIL przyjęli ważne założenie o warstwowej budowie i rozszerzalności proponowanego języka: w obecnej wersji składa się on z trzech warstw. Warstwa najniższa *Core OIL*, odpowiada schematom RDF (skr. ang. *Resource Description Framework Schema* [5, 15], proponowany przez grupę W3C internetowy standard metadanych), poziom *Standard OIL* dostarcza pojęć do modelowania ontologii, a warstwa *Instance OIL* umożliwia posługiwanie się przykładami pojęć („obiektami klas”). Zaproponowano także warstwę *Heavy OIL*, która może zawierać dalsze rozszerzenia. Projektując warstwowy model języka i godząc się na ograniczoną „siłę wyrazu” jego trzech warstw niższych, chciano zapewnić efektywność działania bazujących na OIL narzędzi programistycznych, a tym samym możliwość powszechnego ich stosowania w Internecie.

W metodzie i języku OIL wykorzystano pojęcia z zakresu:

- 1) techniki ram (ang. *frames*), z której zaczerpnięto elementarne pojęcia do modelowania ontologii (koncepty/klasy i ich hierarchia, atrybuty klas, relacje między konceptami i ich hierarchie),
- 2) logik deskrypcyjnych (ang. *descripton logics*), służących do formalnego zdefiniowania pojęć określonych w punkcie 1) i dostarczających mechanizmów wnioskowania ([14]),
- 3) formatów reprezentacji danych i metadanych (danych opisujących dane) stosowanych w Internecie (XML i RDF).

Zapis w języku OIL składa się z dwóch części:

- kontenera ontologii (ang. *ontology container*), zawierającego metadane dotyczące definiowanej ontologii (np.: tytuł, dane o twórcy, temat, data publikacji itp.),
- definicji ontologicznej (ang. *ontology definition*) pewnej dziedziny (klasyfikacja pojęć w dziedzinie wraz z zachodzącymi między nimi relacjami).

W definicji kontenera wykorzystano zestaw pojęć zaproponowanych w wersji 1.1 standardu Dublin Core ([10] – standard definiowania metadanych), które są stosowane do opisywania zasobów np. w bibliotekach elektronicznych, muzeach, urzędach itp.

Część zasadnicza definicji ontologicznej może być poprzedzona listą nazw modułów OIL dołączanych do bieżącej definicji (sekcja *import*) oraz opisem ogólnych reguł (aksjomatów, ograniczeń) dotyczących całej ontologii (sekcja *rule-base*). W obecnej wersji języka OIL nie określono postaci wspomnianych reguł. Definicja ontologii jest skończonym (i być może pustym) ciągiem definicji klas (ang. *class-def*), definicji relacji (ang. *slot-def*), aksjomatów i przykładów pojęć (zwanymi dalej „obiektami”).

Definicja klasy (pojęcia, konceptu, terminu) wiąże nazwę klasy z jej opisem. Definicja zawiera:

- typ klasy (ang. *type*); wyróżnia się klasy pierwotne (ang. *primitive*) i zdefiniowane (ang. *defined*), domyślnie klasa jest pierwotną; klasy pierwotne definiują warunki konieczne lecz niewystarczające relacji przynależności konkretnego obiektu do danej klasy, natomiast klasy zdefiniowane określają obydwa warunki przynależności,
- nazwa (ang. *name*) definiowanej klasy,
- opcjonalny opis klasy (ang. *documentation*),
- opcjonalna niepusta lista wyrażeń (ang. *class-expressions*) opisujących klasy, których dana klasa jest podklasą (ang. *subclass-of*),
- lista (być może pusta) więzów nałożonych na relację (rolę, atrybut – ang. *slot-constraints*, patrz dalej); więzy również definiują klasy i dana klasa musi być podklasą wszystkich klas zdefiniowanych za pomocą więzów zawartych na rozpatrywanej liście.

Wyrażenie (ang. *class-expression*) może być nazwą klasy, więzem (ang. *slot-constraint*) lub wyrażeniem złożonym z wymienionych połączonych spójnikami logicznymi *AND*, *OR* i *NOT*.

Więzy odnoszą się do relacji (ang. *slot*), mają charakter list ograniczeń nałożonych na relacje i zawierają:

- nazwę (ang. *name*) relacji, której dotyczą,
- opcjonalną listę wyrażeń typu *has-value*; każdy obiekt definiowanej za pomocą tego więzu klasy musi być powiązany relacją z co najmniej jednym obiektem każdej z klas opisanych na rozpatrywanej liście; więzy typu *has-value* reprezentują kwantyfikator szczegółowy w logice pierwszego rzędu; niech dana będzie relacja binarna *lubi* i klasa *amator*, wtedy definicja:

```
class-def amator
  slot-constraint lubi
  has-value kino, teatr
```

oznacza, że każdy obiekt klasy *amator* „lubi” kino i teatr (czyli jest powiązany relacją z pewnym obiektem klasy *kino* i pewnym obiektem klasy *teatr*); nie wyklucza się przy tym, że obiekty klasy *amator* mogą być powiązane relacją *lubi* z obiektami innych klas, nie wymienionych na liście *has-value*,

- opcjonalną listę wyrażeń typu *value-type*; jeśli obiekt definiowanej klasy jest powiązany relacją z pewnym obiektem, to ten obiekt musi być instancją wszystkich

klas wyszczególnionych na liście (i tylko takich); niech dana będzie klasa `uczelnia` i relacja `zatrudnia`, wtedy definicja:

```
class-def uczelnia
  slot-constraint zatrudnia
  value-type osoba
```

oznacza, że obiekt klasy `uczelnia` jest powiązany relacją `zatrudnia` wyłącznie z obiektami klasy `osoba`,

- opcjonalne określenie maksymalnej (ang. *max-cardinality*) liczby obiektów typów wyszczególnionych na liście (*class-expression*), z którymi obiekt definiowanej klasy może być powiązany za pomocą rozważanej relacji; na przykład:

```
slot-constraint studiuje
  max-cardinality 2 kierunek
```

oznacza, że obiekt definiowanej klasy może być powiązany relacją `studiuje` z maksymalnie dwoma obiektami klasy `kierunek`,

- opcjonalne określenie minimalnej (ang. *min-cardinality*) liczby obiektów typów wyszczególnionych na liście (*class-expression*), z którymi obiekt definiowanej klasy może być powiązany via rozważana relacja; na przykład:

```
slot-constraint studiuje
  min-cardinality 1 kierunek
```

oznacza, że obiekt definiowanej klasy musi być powiązany relacją `studiuje` z co najmniej jednym obiektem klasy `kierunek`,

- opcjonalne określenie liczby (ang. *cardinality*) obiektów typów wyszczególnionych na liście (*class-expression*), z którymi obiekt definiowanej klasy może być powiązany za pomocą rozważanej relacji; więzy tego rodzaju odpowiadają parze więzów *min-cardinality* i *max-cardinality* z tym samym argumentem,

Definicja relacji (ang. *slot-def*) wiąże jej nazwę z opisem. Definicja ta zawiera:

- nazwę relacji (*name*),
- opcjonalny opis (*documentation*),
- opcjonalną listę (ang. *subslot-of*) relacji, których relacja definiowana jest „podrelacją”,
- opcjonalne określenie dziedziny (ang. *domain*) relacji, w postaci listy wyrażeń (*class-expression*); jeśli para obiektów spełnia definiowaną relację, to pierwszy element pary musi być obiektem należącym do wszystkich klas z listy, np.:

```
slot-def lub1
  domain amator,
```

- opcjonalne określenie zakresu (ang. *range*) relacji, w postaci listy wyrażeń (*class-expression*); jeśli para obiektów spełnia definiowaną relację, to drugi element pary musi być obiektem należącym do wszystkich klas z listy, np.:

```
slot-def zatrudnia
  range osoba,
```

- opcjonalne określenie istnienia relacji odwrotnej (ang. *inverse*) do danej, o wskazanej nazwie, np.:

```
slot-def lubi
  inverse jest_lubiane,
```

- opcjonalne określenie właściwości (ang. *properties*) relacji, takich, jak: przechodniość, symetria i bycie funkcją.

Aksjomaty definiują: rozłączność pojęć (ang. *disjoint*), pokrywanie się pojęć (ang. *covered*), rozłączność i pokrywanie się (ang. *disjoint-covered*).

Ontologiczna klasyfikacja terminologii może zawierać definicje przykładów pojęć (ang. *instance-of*) i relacji zachodzących pomiędzy nimi (ang. *related*). Można także definiować pojęcie poprzez wskazanie zbioru jego przykładów (ang. *one-of*). Dopuszcza się stosowanie standardowych typów danych, jak *integer* i *string*.

Przedstawimy przykład, zaczerpnięty z pracy [13], który zilustruje wprowadzone pojęcia. Kontener ontologii definiującej podstawową terminologię związaną z afrykańskimi zwierzętami zawiera następujący tekst w notacji pseudo-XML-owej (pominięto znaczniki końcowe, a znaczniki początkowe zapisano kursywą):

ontology-container

```
title "African animals"
creator "Ian Horrocks"
subject "animal, food, vegetarians"
description "A didactic example ontology describing African
animals"
description.release "1.01"
publisher "I. Horrocks"
type "ontology"
format "pseudo-xml"
format "pdf"
identifier "http://www.cs.vu.nl/~dieter/oil/TR/oil.pdf"
source "http://www.africa.com/nature/animals.html"
language "OIL"
language "en-uk"
relation.hasPart "http://www.ontosRus.com/animals/jungle.onto"
```

Rozpatrywana ontologia definiuje relację *eats* (i odwrotną *is-eaten-by*) oraz przechodnią relację *has-part* (i odwrotną do niej *is-part-of*):

ontology-definitions

```
slot-def eats
  inverse is-eaten-by
slot-def has-part
  inverse is-part-of
properties transitive
```

Definiuje się także klasy (pojęcia): *animal*, *plant*, *tree*, *branch*, *leaf*, *carnivore*, *herbivore*, *giraffe* i *lion*.

Klasa `animal` nie ma żadnych atrybutów, podobnie klasa `plant`, do której nie należą obiekty klasy `animal`. Klasa `tree` jest uszczegółowieniem klasy `plant`.

```
class-def animal
class-def plant
  subclass-of NOT animal
class-def tree
  subclass-of plant
```

Klasa `branch` ma atrybut będący więzem `has-value` nałożonym na relację `is-part-of` (każdy obiekt klasy `branch` jest powiązany relacją `is-part-of` z pewnym obiektem klasy `tree`). Podobnie zdefiniowano klasę `leaf` (z przechodniości relacji `is-part-of` wynika, że każdy obiekt klasy `leaf` jest powiązany z pewnym obiektem klasy `tree`).

```
class-def branch
  slot-constraint is-part-of has-value tree
class-def leaf
  slot-constraint is-part-of has-value branch
```

Klasa `carnivore` jest uszczegółowieniem klasy `animal` i zawiera atrybut będący więzem `value-type` nałożonym na relację `eats`, który określa typ drugiego argumentu relacji (jeśli obiekt klasy `carnivore` jest powiązany relacją `eats` z pewnym obiektem, to ten obiekt musi być instancją klasy `animal`). Dodatkowo, definicja klasy `carnivore` ma kwalifikator `defined`, co oznacza, że wspomniana definicja określa warunek konieczny i wystarczający relacji przynależności obiektu do definiowanej klasy.

```
class-def defined carnivore
  subclass-of animal
  slot-constraint eats value-type animal
```

Klasa `herbivore` jest uszczegółowieniem klasy `animal` i zawiera atrybut będący więzem `value-type` nałożonym na relację `eats`, który określa typ drugiego argumentu relacji (jeśli obiekt klasy `herbivore` jest powiązany relacją `eats` z pewnym obiektem, to ten obiekt musi być instancją klasy `plant` lub klasy wyznaczonej przez nałożenie więzu `has-value` z wartością `plant` na relację `is-part-of`). Dodatkowo, klasy `herbivore` i `carnivore` są rozłączne. Klasa `herbivore`, podobnie jak `carnivore` ma kwalifikator `defined`.

```
class-def defined herbivore
  subclass-of animal
  slot-constraint eats
  value-type plant OR (slot-constraint is-part-of has-value plant)
disjoint herbivore, carnivore
```

Klasa `giraffe` jest uszczegółowieniem klasy `animal` i ma atrybut ograniczający zakres relacji `eats` do obiektów klasy `leaf`. Warto wspomnieć, że program sprawdzający

poprawność powyższej definicji i określający taksonomię pojęć powinien wydedukować, iż klasa `giraffe` jest podklasą klasy `herbivore`.

```
class-def giraffe
  subclass-of animal
  slot-constraint eats
  value-type leaf
```

Klasa `lion` jest uszczegółowieniem klasy `animal` i ma atrybut ograniczający zakres relacji `eats` do obiektów klasy `herbivore`. Z przedstawionych definicji wynika, że klasa `lion` jest podklasą klasy `carnivore`.

```
class-def lion
  subclass-of animal
  slot-constraint eats
  value-type herbivore
```

Umieszczenie następującej definicji klasy `tasty-plant`:

```
class-def tasty-plant
  subclass-of plant
  slot-constraint is-eaten-by has-value herbivore, carnivore
```

doprowadza do pojawienia się sprzeczności, gdyż obiekty klasy `carnivore` mogą być powiązane relacją `eats` wyłącznie z obiektami klasy `animal`.

Klasa `asian-animal` jest uszczegółowieniem klasy `animal` i każdy jej obiekt jest powiązany relacją `comes-from` z obiektem `Asia` klasy zdefiniowanej jako (*one-of* `Asia`).

```
class-def defined asian-animal
  subclass-of animal
  slot-constraint comes-from value-type (one-of Asia)
```

Obiekty `Africa` i `Asia` są przykładami pojęcia `continent`, natomiast `India` jest przykładem pojęcia `country`. Dodatkowo, obiekt `India` jest powiązany relacją `is-part-of` z obiektem `Asia`.

```
class-def continent
class-def country
instance-of Africa continent
instance-of Asia continent
instance-of India country
related is-part-of India Asia
```

3. Wersja P-OIL języka OIL

Język definiowania ontologii w metodzie OIL charakteryzuje się zwięzłością lecz jest niezbyt czytelny dla przeciętnego użytkownika. W założeniu, metoda OIL ma być narzędziem powszechnie użytecznym także dla osób nie będących specjalistami w zakresie informatyki i lingwistyki. Zatem, język klasyfikowania pojęć powinien w maksymalnym stopniu naśladować język naturalny (w niniejszej pracy przyjęto, że

językiem naturalnym jest język polski). W tym kierunku poszły prace nad wersją P-OIL (skr. ang. *Polish Ontology Inference Layer*) języka.

W oryginalnej metodzie OIL relacje semantyczne są wyrażane za pomocą dowolnych fraz językowych (najczęściej czasowników, także fraz rzeczownikowych). Z jednej strony, takie podejście ułatwia definiowanie powiązań zachodzących pomiędzy pojęciami, z drugiej jednak – jest znacząco nieprecyzyjne. W wielu pracach i projektach z zakresu lingwistyki i sztucznej inteligencji ([1, 2, 3, 17]) ogranicza się zbiór relacji wykorzystując do tego celu głównie infrastrukturę „ról semantycznych”, jakie mogą w zdaniu odgrywać różne części mowy. Każda z wyróżnionych relacji ma ściśle określoną semantykę. Wadą podejścia może okazać się to, że posługiwanie się predefiniowanymi relacjami wymaga dokładnej ich znajomości.

W procesie definiowania zbioru relacji wykorzystano propozycje zawarte w projekcie EuroWordNet ([2, 18]), a także pewne pomysły prezentowane w innych źródłach ([3]). Pierwsza spośród zaproponowanych relacji dotyczy aktu (działania) i definiuje „posiadanie przez akt agenta” (*ma_agenta*) wraz z relacją odwrotną definiującą „bycie agentem danego aktu” (*jest_agentem*). Rozważana relacja ma „podrelacje” określające, czy agent jest właścicielem, czy perceptorem (osoba doświadczająca czegoś lub znajdująca się w pewnym stanie psychicznym). Dołączono także relację „bycia współagentem” i relację odwrotną do niej. Następną grupą relacji odnosi się do roli „pacjensa” aktu, czyli osoby lub rzeczy, na którą akt oddziałuje (*ma_pacjensa*, *jest_pacjensem*). Podrelacje wyrażają fakt bycia: odbiorcą, beneficjentem lub obiektem pewnego aktu. Kolejne relacje dotyczą „posiadania lokacji (miejsca)” i „bycia lokacją (miejscem)”. Zawierają takie warianty lokacyjne, jak: miejsce znajdowania się, miejsce pochodzenia, miejsce przeznaczenia, miejsce przejścia. Do zdefiniowania właściwości (kolor, wielkość, kształt, wymiar, pozycja i orientacja w przestrzeni) wykorzystano propozycje zawarte w strukturach Qualii (idea Arystotelesa). Wśród relacji reprezentujących stopień i miarę wyróżniono „posiadanie wieku” i „posiadanie wagi”. Wyróżnia się także trzy relacje (wraz z odwrotnymi do nich) reprezentujące związki typu „całość-część” (*ma_skladnik*, *ma_jako_czesc*, *ma_jako_czlonka*). Zestawienie kończą relacje „posiadania przyczyny” i „posiadania instrumentu/narzędzia”. Zdefiniowany zbiór może podlegać dalszym modyfikacjom.

Język P-OIL zawiera także siatkę predefiniowanych pojęć, które zaczerpnięto w całości z [18]. Wyraża ona tzw. „Top Ontology” będącą klasyfikacją ogólnych relacji semantycznych. Najogólniejszym pojęciem (kategorią semantyczną) jest *top*, z rozłącznymi podklasami: *koncept_I_rodzaju* (byt konkretny, postrzegalny za pomocą zmysłów, umiejscowiony w czasie i przestrzeni), *koncept_II_rodzaju* (sytuacja statyczna lub dynamiczna, zachodząca i umiejscowiona w czasie) oraz *koncept_III_rodzaju* (byt abstrakcyjny, nie rozpatrywany w aspekcie czasu i przestrzeni).

Cechy semantyczne pierwszego rodzaju są wzorowane na strukturach Qualii ([17]) i definiują: pochodzenie (rola agentywna, czyli przyczyna powstania definiowanego bytu – może być naturalna albo sztuczna), formę (rola formalna, oznaczająca bycie substancją albo obiektem fizycznym), kompozycję (rola konstytucjonalna, w której byt może być samodzielnie istniejącą częścią całości albo całością zbudowaną z części) oraz funkcję (rola teliczna, określająca cel istnienia lub działania opisywanego bytu). Pojęcia

drugiego rodzaju opisują typy i składowe sytuacji. Sytuacje mogą być: dynamiczne (w tym ściśle umiejscowione w czasie, bądź nie) lub statyczne – będące właściwościami lub relacjami. Składowe sytuacji można opisywać za pomocą szesnastu aspektów szczegółowych. Pojęcia trzeciego rodzaju nie są na razie uszczegółowione. Zbiór pojęć stanowi jedynie propozycję i może podlegać dalszym zmianom.

Podsumowując, nowe aspekty P-OIL to:

- polska wersja językowa,
- zmodyfikowana składnia wyrażen języka służącego do definiowania ontologii,
- nowe spojrzenie na pojęcie relacji (ang. *slot*), uwzględniające m.in. dokonania lingwistyki i sztucznej inteligencji ([1, 3, 12, 17]) w zakresie relacji i ról semantycznych,
- zastosowanie Top Ontology ([18]) do zdefiniowania zestawu predefiniowanych pojęć.

Na zakończenie rozważań, w celach ilustracyjnych przytoczymy fragment definicji ontologii dotyczącej afrykańskich zwierząt, sformułowanej w języku P-OIL.

ontology-container

title "Zwierzęta afrykańskie i azjatyckie "

creator "J. Cybulka, na podstawie I. Horrocks"

subject "zwierzęta, żywienie, wegetarianizm"

description "Przykładowa ontologia dotycząca zwierząt afrykańskich i azjatyckich z wykorzystaniem ról semantycznych i predefiniowanych pojęć z tzw. Top Ontology, zaproponowanej w projekcie EuroWordNet oraz z zastosowaniem nowej składni języka P-OIL"

description.release "1.0"

publisher "J. Cybulka"

type "ontology"

format "nowy pseudo-xml"

source "<http://www.cs.vu.nl/~dieter/oil/TR/oil.pdf>"

language "P-OIL"

language "pl"

Pojęcia zwierzę i roślina są predefiniowane (Top Ontology). Relacja *jest-częścią* należy do zestawu relacji predefiniowanych. Definicje pojęć: drzewo, gałąź, liść, mięsożerca, roślinożerca, żyrafa i lew przedstawiają się następująco:

definicje-ontologiczne

pojęcie drzewo:

jest-uszczegółowieniem-pojęcia roślina,

koniec-definicji-pojęcia drzewo

pojęcie gałąź:

jest-powiązane-relacją jest_częścią

między-innymi-z-pojęciem drzewo,

koniec-definicji-pojęcia gałąź

pojęcie liść:

jest-powiązane-relacją jest_częścią *między-innymi-z-pojęciem* gałąź,

koniec-definicji-pojęcia liść

pojęcie-zdefiniowane mięsożerca:

jest-uszczegółowieniem-pojęcia zwierzę,

koniec-definicji-pojęcia mięsożerca

pojęcie roślinożerca:

jest-uszczegółowieniem-pojęcia zwierzę,

koniec-definicji-pojęcia roślinożerca

pojęcie żyrafa:

jest-uszczegółowieniem-pojęcia zwierzę,

koniec-definicji-pojęcia żyrafa

pojęcie lew:

jest-uszczegółowieniem-pojęcia zwierzę,

koniec-definicji-pojęcia lew

Dalsze definicje wskażą różnice metodologiczne między językami OIL i P-OIL. Zamiast relacji eats definiujemy zbiór hierarchicznie powiązanych ze sobą pojęć (jedzenie, spożywanie_pokarmu, jedzenie_pokarmu_roślinnego, jedzenie_e_liści, jedzenie_pokarmu_zwierzęcego, jedzenie_każdego_pokarmu) charakteryzujących akt będący „jedzeniem” (spożywaniem pokarmu), jako sytuację nieumiejscowioną w konkretnym czasie, inicjowaną przez „agenta”, dotyczącą pewnego obiektu spożywanego („pacjensa”).

pojęcie jedzenie:

jest-uszczegółowieniem-pojęcia sytuacja_nieumiejscowiona_w_czasie,

jest-uszczegółowieniem-pojęcia pochodzenie/agentywność,

jest-powiązane-relacją ma_agenta dokładnie-z-pojęciem zwierzę,

koniec-definicji-pojęcia jedzenie

pojęcie spożywanie_pokarmu:

koniec-definicji-pojęcia spożywanie_pokarmu

pojęcia-równoważne jedzenie, spożywanie_pokarmu

pojęcie jedzenie_pokarmu_roślinnego:

jest-uszczegółowieniem-pojęcia jedzenie,

jest-powiązane-relacją ma_agenta dokładnie-z-pojęciem roślinożerca,

jest-powiązane-relacją ma_pacjensa/obiekt

dokładnie-z-pojęciem (roślina lub

pojęcie-które jest-powiązane-relacją

jest_częścią dokładnie-z-pojęciem

roślina),

koniec-definicji-pojęcia jedzenie_pokarmu_roślinnego

pojęcie jedzenie_liści:

jest-uszczegółowieniem-pojęcia jedzenie_pokarmu_roślinnego,
jest-powiązane-relacją ma_agenta między-innymi-z-pojęciem żyrafa,
jest-powiązane-relacją ma_pacjensa/obiekt dokładnie-z-pojęciem liść,
koniec-definicji-pojęcia jedzenie-liści

pojęcie jedzenie_pokarmu_zwierzęcego:
jest-uszczegółowieniem-pojęcia jedzenie,
jest-powiązane-relacją ma_agenta dokładnie-z-pojęciem mięsożerca,
jest-powiązane-relacją ma_pacjensa/obiekt dokładnie-z-pojęciem zwierzę,
koniec-definicji-pojęcia jedzenie_pokarmu_zwierzęcego

pojęcie jedzenie_każdego_pokarmu:
jest-uszczegółowieniem-pojęcia jedzenie,
jest-powiązane-relacją ma_agenta dokładnie-z-pojęciem wszystkożerca,
jest-powiązane-relacją ma_pacjensa/obiekt
dokładnie-z-pojęciem (zwierzę lub roślina lub
pojęcie_które jest-powiązane-relacją
jest_częścią dokładnie-z-pojęciem roślina),
koniec-definicji-pojęcia jedzenie_każdego_pokarmu

Na koniec, definicja pojęcia słoń, którego przykłady są roślinożercami o szarym kolorze skóry, a także pojęcia zwierzę_azjatyckie, kontynent i kraj. Przytaczamy także przykłady pojęć (Azja, Afryka i Indie) oraz relacje z ich udziałem.

pojęcie słoń:
jest-uszczegółowieniem-pojęcia roślinożerca,
jest-powiązane-relacją ma_właściwość/kolor z-przykładem kolor_szary
koniec-definicji-pojęcia słoń

pojęcie-zdefiniowane zwierzę_azjatyckie:
jest-uszczegółowieniem-pojęcia zwierzę,
jest-powiązane-relacją ma_miejsce/pochodzenia_z
dokładnie-z-pojęciem ((*pojęcie-zbudowane-z* Azja) lub *pojęcie-które*
jest-powiązane-relacją jest_częścią
z-przykładem Azja),
koniec-definicji-pojęcia zwierzę_azjatyckie

pojęcie kontynent:
jest-uszczegółowieniem-pojęcia miejsce,
koniec-definicji-pojęcia kontynent

pojęcie kraj:
jest-powiązane-relacją jest_częścią między-innymi-z-pojęciem kontynent,
koniec-definicji-pojęcia kraj

Afryka *jest-przykładem-pojęcia* kontynent

Azja *jest-przykładem-pojęcia* kontynent

Indie są przykładem pojęcia kraj

Indie są powiązane relacją jest częścią z przykładem Azja

„Smaczna roślina” o następującej definicji jest pojęciem sprzecznym, dla którego nie można podać żadnego przykładu.

pojęcie smaczna_roślina:

jest-uszczegółowieniem-pojęcia roślina,

jest-powiązane-relacją jest_pacjensem/obiektem

między-innymi-z-pojęciem jedzenie_pokarmu_roślinnego i

jedzenie_pokarmu_zwierzęcego,

koniec-definicji-pojęcia smaczna_roślina

4. Podsumowanie

W artykule prezentujemy jeden z etapów realizacji semantycznej sieci WWW, polegający na zdefiniowaniu języka przeznaczonego do wyrażania ontologii dziedzinowych. Język można poddawać rozszerzeniom i modyfikacjom, zwłaszcza w zakresie predefiniowanych relacji i pojęć, a dla wersji aktualnej zrealizowano edytor strukturalny, będący narzędziem wspomagającym tworzenie definicji ontologicznych ([16]). Kolejne etapy prac zmierzających ku pełnej realizacji sieci semantycznej będą wymagały m.in. wypracowania metody oznaczania zawartości dokumentów zgodnie ze zdefiniowanymi ontologiami, a także utworzenia licznych narzędzi mogących przetwarzać oznaczoną semantycznie zawartość.

Literatura

- [1] Allen J., *Natural Language Understanding*, Benjamin Cummings, Menlo Park.
- [2] Alonge A., Calzolari N., Vossen P., Bloksma L., Castellon I., Marti M. A., Peters W., The Linguistic Design of the EuroWordNet database. *Computers and the Humanities*, 32, pp. 91-115.
- [3] Antoni-Lay M-H., Francopoulo G., Zaysser L., A Generic Model for Reusable Lexicons: The GENELEX Project. *Literary and Linguistic Computing*, 9(1), pp. 47-54.
- [4] Berners-Lee T. Semantic Web road map, <http://www.w3.org/DesignIssues/semantic.html>.
- [5] Brickley D., Guha R. V., Resource Description Framework (RDF) Schema Specification 1.0, <http://www.w3.org/TR/2000/CR-rdf-schema-20000327>.
- [6] Broekstra J., Klein M., Decker S., Fensel D., Horrocks I., Adding formal semantics to the Web: building on top of RDF Schema, 2000, <http://www.ontoknowledge.org/oil/extending-rdfs.pdf>.
- [7] Broekstra J., Klein M., Fensel D., Decker S., Horrocks I., OIL: a case-study in extending RDF-Schema, <http://www.cs.vu.nl/~mcaklein/papers/oil-rdfs.pdf>.
- [8] Cybulka J., Sikorski A., Przetwarzanie transakcji w bezpołączeniowych protokołach komunikacyjnych, *Pro Dialog*, 9, 1999.

- [9] Decker S., D.Fensel, van Harmelen F., Horrocks I., Melnik S., Klein M., Broekstra J., Knowledge Representation on the Web, w: Proceedings of the International Workshop on Description Logics (DL2000), <http://www.ontoknowledge.org/oil/download/DL00-oil.pdf>.
- [10] Dublin Core Metadata, <http://purl.oclc.org/dc/>.
- [11] Fensel D., Horrocks I., Harmelen van F., Decker S., Erdmann M., Klein M., OIL in a Nutshell, w: Knowledge Acquisition, Modeling, and Management, Proceedings of the European Knowledge Acquisition Conference (EKAW-2000), R. Dieng et al. (eds.), *Lecture Notes in Artificial Intelligence*, Springer-Verlag, October 2000.
- [12] Frawley W., *Linguistic semantics*, Lawrence Erlbaum Associates, Inc., Publishers, Hillsdale, New Jersey.
- [13] Horrocks I., Fensel D., Broekstra J., Decker S., Erdmann M., Goble C., Harmelen F. van, Klein M., Staab S., Studer R., Motta E., The Ontology Inference Layer OIL, raport techniczny, Manchester University/Vrije Universiteit Amsterdam, 2000, <http://www.cs.vu.nl/~dieter/oil/Tr/oil.pdf>
- [14] Horrocks I., A Denotational Semantics for Standard OIL and Instance OIL, raport techniczny, <http://www.ontoknowledge.org/oil/download/semantics.pdf>.
- [15] Lassila O., Swick R. R., Resource Description Framework (RDF): Model and Syntax Specification, rekomendacja W3C, <http://www.w3.org/TR/REC-rdf-syntax/>.
- [16] Lenartowska M. – Metadane w internetowych systemach informacyjnych. Środowisko realizacji standardu RDF, praca magisterska, Poznań, 2002.
- [17] Pustejovsky J., The Generative Lexicon, *Computational Linguistics*, vol 17, no 4, pp. 409-441.
- [18] Rodríguez H., Climent S., Vossen P., Bloksma L., Peters W., Alonge A., Bertagna F., Roventini A., The Top-Down Strategy for Building EuroWordNet: Vocabulary Coverage, base Concepts and Top Ontology. *Computers and the Humanities*, 32, pp. 117-152.
- [19] Stuckenschmidt H., Harmelen F.van, Fensel D., Klein M., Horrocks I., Catalogue Integration. A Case Study in Ontology-Based Semantic Translation, May 9, 2000, <http://www.ontoknowledge.org/oil/download/CatIntegr.pdf>.