Nierówności Społeczne a Wzrost Gospodarczy, nr 44 (4/2015), część 1
DOI: 10.15584/nsawg.2015.4.1.8 **ISSN 1898-5084**

mgr inż. Dariusz Dobrowolski[1]

Instytut Informatyki, Wydział Matematyki, Fizyki i Informatyki
Uniwersytet Marii Curie-Skłodowskiej w Lublinie

dr Paweł Kapłański[2]

Katedra Zastosowań Informatyki w Zarządzaniu, Wydział Zarządzania i Ekonomii
Politechnika Gdańska

dr hab. Zdzisław Łojewski[3]

Instytut Informatyki, Wydział Matematyki, Fizyki i Informatyki
Uniwersytet Marii Curie-Skłodowskiej w Lublinie

dr hab. Andrzej Marciniak[4]

Zakład Logistyki i Systemów Transportowych, Wydział Transportu i Informatyki
Wyższa Szkoła Ekonomii i Innowacji w Lublinie

Ontologiczna inżynieria wiedzy

WPROWADZENIE

Wiedza i informacja stają się podstawowymi czynnikami produkcji odnoszącymi się do funkcjonowania przedsiębiorstw, gospodarstw rolnych oraz organizacji publicznych i pozarządowych. Postępujący w szybkim tempie proces globalizacji gospodarki światowej oraz coraz szersze upowszechnianie się technologii

[1] Adres korespondencyjny: Instytut Informatyki, Wydział Matematyki, Fizyki i Informatyki, Uniwersytet Marii Curie-Skłodowskiej w Lublinie, ul. Akademicka 9, 20-033 Lublin, e-mail: dariusz.dobrowolski@umcs.lublin.pl, tel. 81 537 29 32.

[2] Adres korespondencyjny: Katedra Zastosowań Informatyki w Zarządzaniu, Wydział Zarządzania i Ekonomii, Politechnika Gdańska, ul. Narutowicza 11/12, 80-233 Gdańsk, e-mail: zzti@zie.pg.gda.pl, tel. 58 347 14 28.

[3] Adres korespondencyjny: Instytut Informatyki, Wydział Matematyki, Fizyki i Informatyki, Uniwersytet Marii Curie-Skłodowskiej w Lublinie, ul. Akademicka 9, 20-033 Lublin, e-mail: zdzislaw.lojewski@umcs.lublin.pl, tel. 81 537 62 42.

[4] Adres korespondencyjny: Zakład Logistyki i Systemów Transportowych, Wydział Transportu i Informatyki, Wyższa Szkoła Ekonomii i Innowacji w Lublinie, ul. Projektowa 4, 20-209 Lublin, e-mail: andrzej.marciniak@yahoo.com.

informatycznych i nowoczesnych środków komunikacji (*Internet, Semantic Web*) decydują w dużym stopniu o konkurencyjności wszystkich sektorów gospodarki w skali światowej.

Wiedza w dzisiejszym znaczeniu jest nie tylko atrybutem mentalnym człowieka, ale również zasobem produkcyjnym i to zasobem szczególnym – *im więcej go używamy, tym więcej go posiadamy*, a na dodatek wiedza i informacja są idealnymi substytutami dla kolejnych dwóch fundamentalnych w każdym procesie produkcyjnym zasobów – energii i czasu. Im większe mamy zasoby informacji, tym mniej czasu i energii potrzebujemy na osiągnięcie określonego efektu produkcyjnego.

Wiedza tworzona w procesie produkcji jest najbardziej wartościowym produktem ubocznym każdego procesu produkcyjnego. Współczesna inżynieria wiedzy dostarcza metod i narzędzi umożliwiających skuteczną akwizycję danych procesowych, wydobywanie z nich informacji faktograficznych i wykrywanie struktur relacyjnych będących podstawą do tworzenia organizacyjnych i technologicznych procedur zarządzania i sterowania procesami produkcji.

Jednym z podstawowych zagadnień w budowaniu systemów opartych na wiedzy jest wybór systemu reprezentacji wiedzy. Teoretyczne podstawy budowy ontologicznych systemów reprezentacji wiedzy opracował John Sowa [Sowa, 2000].

Sieci semantyczne są najstarszym i najbardziej ogólnym typem reprezentacji wiedzy. Sieć semantyczna to przykład sieci, w której dane są przechowywane, opisywane i powiązane w taki sposób, aby mogły być wykorzystywane nie tylko przez ludzi, ale także przez maszyny (programy, inteligentnych agentów softwarowych). W tym kontekście kluczowe znaczenie ma ontologia w znaczeniu zdefiniowanym przez Toma Grubera jako „formalna, jawna specyfikacja współdzielonej konceptualizacji".

Ontologia wyrażona za pomocą języka OWL stanowi repozytorium wiedzy, z którego można wydobywać informacje. Wydobywanie informacji wyrażonych w postaci ontologii ma cechy zarówno wydobywania informacji zgromadzonych w bazach danych, jak i pozyskiwania informacji ze stron internetowych z wykorzystaniem wyszukiwarki internetowej.

Minimalna, wystarczająca ontologia, powstaje poprzez odwzorowanie węzłów i łuków grafu reprezentującego zastosowany schemat pojęciowy (po jego odpowiednim uszczegółowieniu) w węzły i łuki grafowej semantycznej bazy danych, gdzie węzły reprezentują klasy (kategorie pojęciowe), a łuki – relacje między egzemplarzami (wystąpieniami) klas. Ten sposób budowy ontologii zjawisk i procesów umożliwia niemal bezpośrednią integrację (fuzję) gromadzonych danych procesowych do uogólnionej w formie generatywnego modelu formalnej i wykonywalnej reprezentacji procesu, z którego pochodziły użyte dane procesowe, poprzez zastosowanie typowych metod uczenia maszynowego i automatycznego wnioskowania.

BUDOWA ONTOLOGII PRZY WYKORZYSTANIU FLUENTEDITOR™

Język reprezentacji wiedzy jest naprawdę przydatny tylko wtedy, gdy istnieje środowisko pozwalające na jego wykonanie w postaci procesu obliczeniowego. O zaletach RDF/RDFs czy OWL ich twórcy mogliby przekonywać godzinami, ale nic by nie osiągnęli bez narzędzi będących w stanie te języki zaimplementować. Obecnie, informatyczne narzędzia do projektowania systemów opanowało podejście wizualne, przenoszące znaczną część pracy z kodowania na graficzne łączenie pewnych modułów. Dotyczy to również aplikacji do tworzenia ontologii. Spośród dostępnych, najczęściej wykorzystywanym narzędziem w tworzeniu ontologii jest zaawansowane środowisko opensource'owe Protégé [Protégé].

Najnowsza wersja Protégé (5.0) dostarczana jest z maszyną wnioskującą HermiT i Pellet, pozwalającą na automatyczną klasyfikację obiektów, obsługę zapytań oraz wnioskowanie.

Innym, rewelacyjnym edytorem ontologii jest FluentEditor 2014 [FluentEditor, 2014], stworzony i rozwijany przez polską firmę Cognitum. Jest to kompleksowe narzędzie wykorzystujące kontrolowane języki naturalne (ang. *Controlled Natural Language* – CNL) do edycji i zarządzania złożonymi ontologiami. Kontrolowane języki naturalne (CNL) to podzbiory języków naturalnych, które tworzy się poprzez redukowanie gramatyki i słownictwa w celu ograniczenia lub wyeliminowania dwuznaczności bądź złożoności. Tradycyjny podział języków kontrolowanych jest następujący: 1) ułatwiające człowiekowi czytelność ontologii, 2) wspomagające zautomatyzowaną analizę języka.

Pierwszy typ języków (często nazywanym *uproszczonym* lub *technicznym*, jak na przykład *ASD Simplified Technical English, Caterpillar Technical English, IBM's Easy English*) stosowany jest przeważnie w przemyśle w celu zwiększenia jakości dokumentacji technicznej i ewentualnym uproszczeniu automatycznego (ew. półautomatycznego) tłumaczenia dokumentacji [O'Brien, 2003].

Drugi typ języków ma formalne podstawy logiczne – formalną syntaktykę i semantykę opartą na podzbiorze logiki predykatów I-szego rzędu. Języki te mogą być używane jako języki reprezentacji wiedzy. FluentEditor 2014 jest efektywną alternatywą dla najbardziej złożonych edytorów OWL opartych na XML. Jego główną zaletą jest wykorzystanie kontrolowanego języka angielskiego jako języka reprezentacji wiedzy. Dzięki wykorzystaniu edytora predykatów niemożliwe jest napisanie jakiegokolwiek zdania, które byłoby niepoprawne syntaktycznie, przy tym aktywnie wspomagając użytkownika podczas pisania zdań (tabela 1).

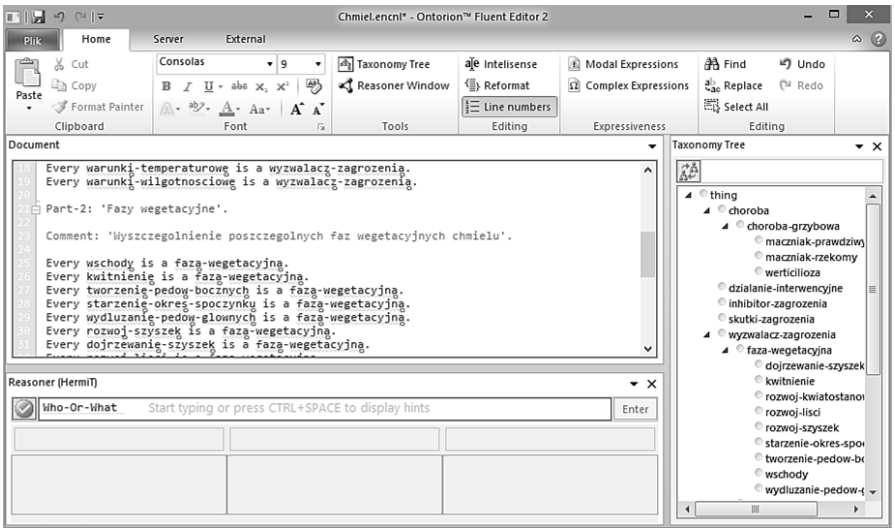
Edycję ontologii dla przykładu zagrożeń uprawy chmielu pokazanego w tabeli 1 ilustruje rys. 1.

Ontologię stworzoną za pomocą FluentEditor 2014 można wyeksportować do formatu owl i przedstawić w formie graficznej za pomocą zawartego w Protege programu GraphViz (rys. 2).

Tabela 1. Najprostsza koncepcja subsumpcji w językach CNL i OWL2

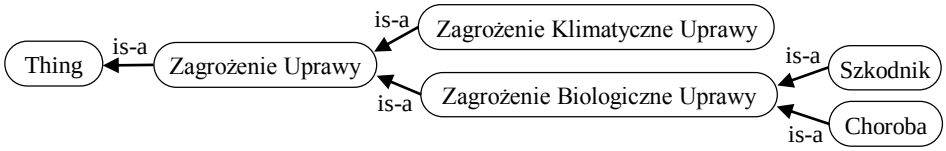
CNL	OWL
Every zagrozenie has-wyzwalacz a wyzwalacz-zagrozenia.	<SubClassOf xmlns="http://www.w3.org/2002/07/owl#"> <Class IRI="Zagrozenie" /> <ObjectSomeValuesFrom> <ObjectProperty IRI="hasWyzwalacz" /> <Class IRI="WyzwalaczZagrozenia" /> </ObjectSomeValuesFrom></SubClassOf>

Źródło: opracowanie własne.



Rysunek 1. Przykład edycji ontologii za pomocą FluentEditor2014

Źródło: opracowanie własne.



Rysunek 2. Graficzna reprezentacja definicji zagrożenia uprawy chmielu

Źródło: opracowanie własne.

ARCHITEKTURA ŚRODOWISKA

Projektowany system z założenia ma funkcjonować w intensywnym strumieniu danych procesowych uzyskiwanych nie tylko w tradycyjny sposób „człowiek – system", ale również poprzez coraz większe pola sensorowe wbudowane w pro-

cesy produkcyjne. Te strumienie danych muszą być gromadzone w informatycznej infrastrukturze procesu produkcji i jego otoczenia w sposób umożliwiający ich semantyczną analizę i integrację. Stąd bierze się potrzeba stosowania technologii semantycznych.

Technologie te implikują bazowe cechy funkcjonalne i operacyjne projektowanego systemu. Semantyczny system akwizycji danych implikuje wykorzystanie agentów software'owych – czyli całkowicie zautomatyzowany proces wydobywania informacji poprzez zastosowanie algorytmów wyszukiwania znaczeń, a w dalszej kolejności zastosowanie algorytmów uczenia maszynowego umożliwiających wykrywanie zależności strukturalnych i relacji przyczynowo-skutkowych tkwiących w danych procesowych.

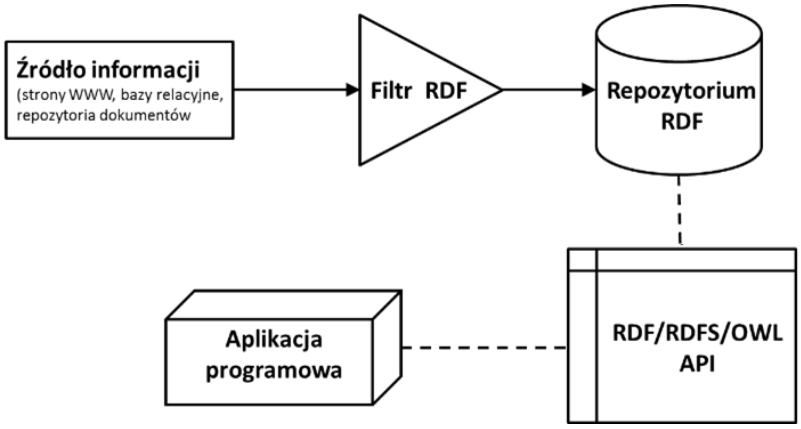
Zależności te reprezentowane w jednym z formalnych języków reprezentacji wiedzy są formalnymi i wykonywalnymi modelami procesów umożliwiającymi zautomatyzowane wnioskowanie wszelakiego rodzaju (indukcja, dedukcja, abdukcja) oraz wspomaganie zarządzania i sterowania procesami w oparciu o formalne, wystarczająco szczegółowe i adaptacyjnie uaktualniane modele. Takie zautomatyzowanie wnioskowania jest niezbędne do aktywnego ale również do proaktywnego zarządzania procesami (predykcja) oraz zarządzania bezpieczeństwem procesów, produktów i ich otoczenia (rozumowanie diagnostyczne, hipotetyczno-dedukcyjne).

Rozumowanie indukcyjne jest podstawą ciągłej aktualizacji wiedzy procesowej na wszystkich poziomach strukturalnych i funkcjonalnych. To wszystko jest podstawą paradygmatu innowacyjnej gospodarki opartej na wiedzy i informacji. Punktem wyjścia do stworzenia modelu systemu zarządzania wiedzą produkcyjną jest schemat przedstawiony na rys. 3.

Źródłem informacji są tu zarówno strony internetowe, bazy danych – relacyjne i nierelacyjne (NoSQL) – jak i repozytoria dokumentów. Zastosowanie filtra danych RDF umożliwia importowanie danych zapisanych w formacie RDF, przetwarzanie i zapisywanie w repozytorium RDF, a następnie odpytywanie metadanych zasobów (internetowych oraz dokumentów) za pomocą aplikacji zawartych w repozytorium API.

Przetwarzając ten schemat na układ komponentów modułowych, zawierających założone dla projektowanego systemu elementy struktury zaproponowano projekt przedstawiony na rys. 4.

Elementami wejściowymi dla projektowanego systemu są m.in. udostępnione dane relacyjne, należące do instytucji naukowych, samorządowych, jak i firm handlowych, stanowiące zalążek systemu. Obok tych danych, informacja (wiedza) dostarczana jest również z innych źródeł takich jak np.: systemy zarządzania treścią (ang. *Content Management System* – CMS) [Drupal], dzięki czemu kształtowanie treści i sposobu ich prezentacji będzie miało miejsce za pomocą prostych w obsłudze interfejsów użytkownika. Zwykle odbywa się to w postaci stron WWW zawierających rozbudowane formularze i moduły.



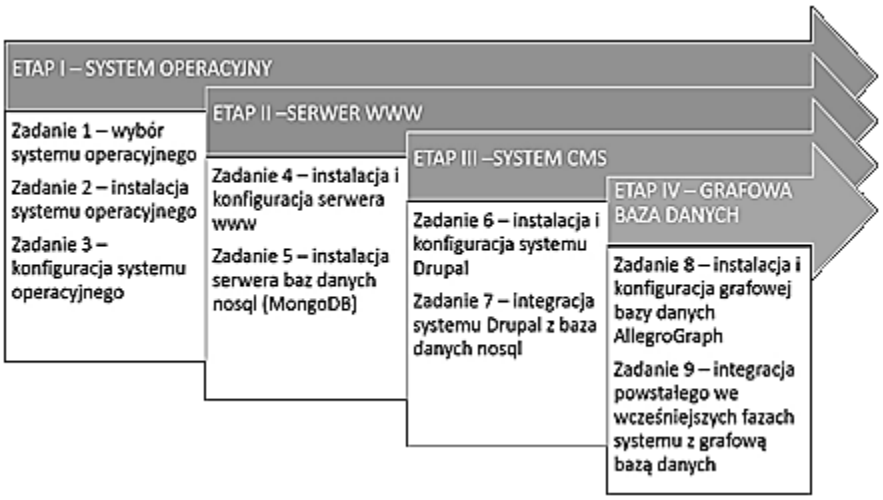
Rysunek 3. Schemat modelu zarządzania wiedzą

Źródło: opracowanie własne na podstawie [Watson, 2011].

Rysunek 4. Przenikanie zależności strukturalnych i funkcjonalnych projektowanego systemu

Źródło: opracowanie własne.

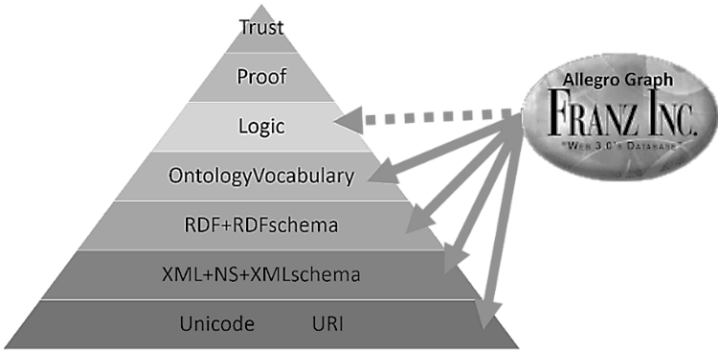
Takie podejście umożliwia kształtowanie i edycję treści przez szerokie grono osób, producentów i firm dostarczających materiały do prowadzonej produkcji i nieposiadających przy tym szczególnej wiedzy informatycznej. Elementem scalającym zarówno dostęp do danych jak i systemów zarządzania treścią jest w tym przypadku Internet, jako globalna sieć wymiany informacji i wiedzy. Zawarte informacje pozwalają na zainstalowanie i uruchomienie sieci semantycznej do celów zarówno naukowych, jak i produkcyjnych. Etapy niezbędne do przeprowadzenia tego procesu zostały przedstawione na rys. 5.



Rysunek 5. Etapy implementacji projektu

Źródło: opracowanie własne.

Kluczową rolę w architekturze systemu pełni program AllegroGraph [AllegroGraph], na rys. 6 pokazano relacje funkcjonalne AllegroGraph z podstawowymi warstwami semantycznymi obsługiwanymi przez system.



Rysunek 6. Allegrograph jako baza wiedzy

Źródło: opracowanie własne.

Wymaga to uruchomienia serwera usług w celu pełnego wykorzystania jego funkcjonalności. AllegroGraph posiada interfejsy klienckie do programowania aplikacji w takich językach jak Java, Python, Ruby, Perl, C#, Clojure oraz Lisp. Możliwe jest wczytywanie baz wiedzy w formacie N-Triples oraz RDF/XML. AllegroGraph zapewnia możliwość wykonywania zapytań w języku SPARQL, a także wspiera silniki wnioskowania takie jak: RDFS++ Reasoning oraz Prolog.

FUNKCJONALNOŚĆ

Dane wyrażone w postaci ontologii można wydobywać tak, jak dane z bazy danych oraz tak, jak wydobywane są informacje przy użyciu wyszukiwarki internetowej. Dane zawarte w ontologiach mogą być znacznie bardziej zróżnicowane niż dane zgromadzone w bazach danych. Możliwe jest jednak wydobywanie informacji z ontologii przez bezpośrednie formułowanie zapytań, podobnie jak w przypadku wyszukiwania za pomocą wyszukiwarek internetowych. Istnieją liczne języki przeznaczone do tworzenia ontologicznych zapytań. Część z nich ma podobne zastosowanie jak język SQL. Za ich pośrednictwem aplikacje komunikują się z systemami zawierającymi dane w postaci ontologii. Z pomocą tych języków definiuje się, jakie dane mają zostać wydobyte z ontologii.

Tabela 2. Zapytanie:

select ?s ?p ?o {?s ?p ?o}

Źródło: opracowanie własne.

Results (16.612 ms) | Information

s	p	o
Uprawa3	hasSymbolUprawy	"PL-L-06-09072-1176"
Uprawa3	hasOdmiana	"Magnum"
Uprawa3	rdf:type	Uprawa
Uprawa3	rdf:type	owl:NamedIndividual
Uprawa2	hasSymbolUprawy	"PL-L-06-09072-1175"
Uprawa2	hasOdmiana	"Magnum"
Uprawa2	rdf:type	Uprawa
Uprawa2	rdf:type	owl:NamedIndividual
Uprawa1	hasSymbolUprawy	"PL-L-06-09072-1174"
Uprawa1	hasOdmiana	"Lubelski"
Uprawa1	rdf:type	Uprawa
Uprawa1	rdf:type	owl:NamedIndividual
Plantator2	hasMiejscowosc	"Panienszczyzna"
Plantator2	hasNazwisko	"Nowak"
Plantator2	hasImie	"Jozef"

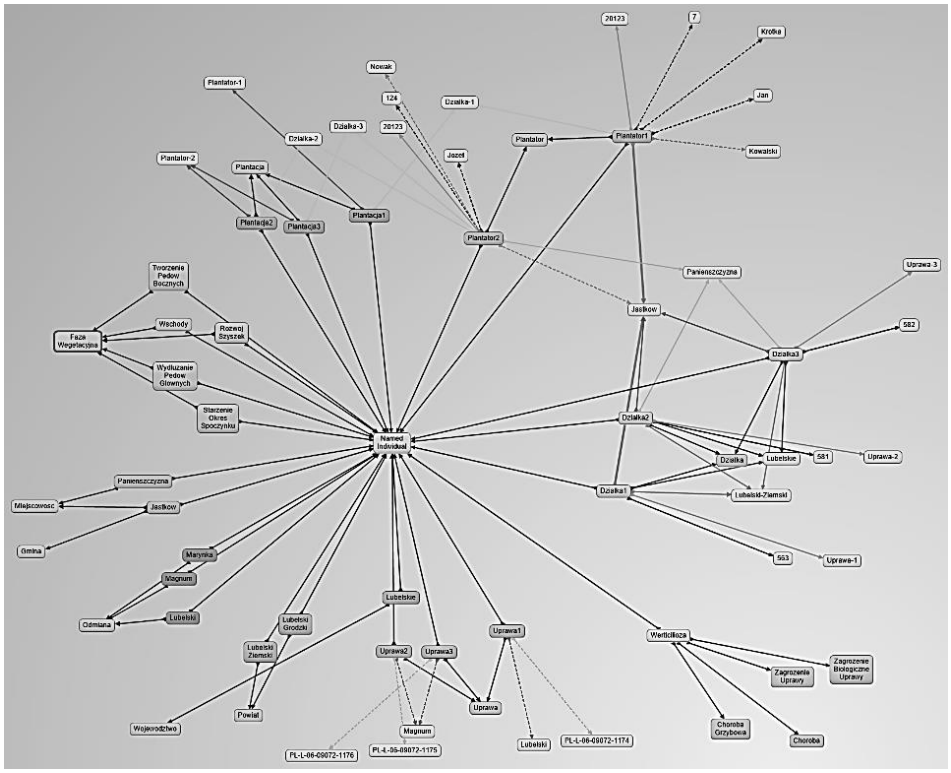
Download using format SPARQL JSON

Rysunek 7. Odpowiedź na zadane wcześniej pytanie

Źródło: opracowanie własne.

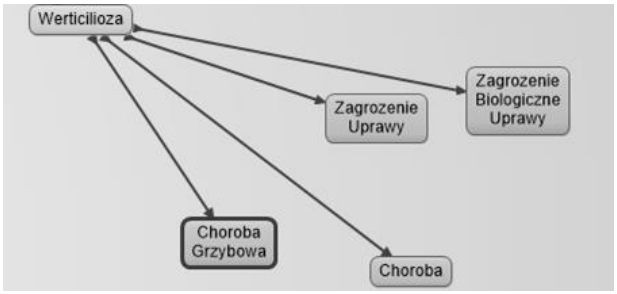
Odpowiedzią jest zawartość bazy danych wyświetlona w formie tabelarycznej (rys. 7). Zawarty w tabeli 2 przykład ilustruje proste zapytanie stworzone w zaimplementowanym w AllegroGraph języku SPARQL, w wyniku którego wyświetlana jest pełna zawartość (zarówno nazwa relacji jak i jej argumenty są zmiennymi) bazy danych.

Innym sposobem generowania zapytań w języku SPARQL jest edytor graficzny Gruff [Gruff] (rys. 8).



Rysunek 8. Ekran programu Gruff wyświetlającego odpowiedź w formie graficznej na pytanie z tabeli 2

Źródło: opracowanie własne.



Rysunek 9. Odpowiedź na pytanie, czym jest *werticilioza* wyrażona graficznie

Źródło: opracowanie własne.

Wskazując w nim konkretne węzły grafu (obiekty) można zadawać pytania o ścieżki predykatów (relacji) je łączących. Zadając predykaty możemy pytać o obiekty, które są przez nie łączone (rys. 9).

PODSUMOWANIE

Semantyczne systemy reprezentacji wiedzy są podstawą powstającego już Web 3.0 i są niezbędne we wszystkich technologiach należących do kategorii *Big Data* – charakterystycznej dla rozwoju Internetu Rzeczy (*Internet of Things*). Otwiera to nowy rozdział w rozwoju informatyki ukierunkowany na wbudowywanie w otaczającą nas rzeczywistość fizyczną pól sensorowych i efektorowych generujących peta-bajtowe strumienie danych, które gromadzone i przetwarzane w semantycznych bazach wiedzy, utworzą wirtualną reprezentację tej rzeczywistości zgodnie z metaforą „Speaking Things Approach".

Dostępne środowiska *softwareowe*, w tym *open source*, umożliwiają w miarę efektywne (chociaż nie do końca zautomatyzowane) budowanie semantycznych baz wiedzy opartych o metodologię ontologicznej inżynierii wiedzy.

BIBLIOGRAFIA

Allegrograph, http://franz.com.

Dobrowolski D., 2014, *Zarządzanie wiedzą produkcyjną w procesach naturalnych produkcji rolniczej z wykorzystywaniem sieci semantycznych*, Politechnika Świętokrzyska (przygotowywana do druku), Kielce.

Drupal, http://www.drupal.org.

FluentEditor; http://www.cognitum.eu.

Gruff, http://franz.com.

O'Brien S., 2003, *Controlling Controlled English – An Analysis of Several Controlled Language Rule Sets.* Dublin, EAMT-CLAW.

Protege; http://protege.stanford.edu.

Sowa J.F., 2000, *Knowledge Representation: Logical, Philosophical, and Computational Foundations.* Pacific Grove, CA, Brooks Cole Publishing Co.

Watson M., 2011, *Practical Semantic Web and Linked Data Applications.* Raleight, N.C. USA: Mark Watson.

Streszczenie

Ontologiczna inżynieria wiedzy jest dobrą podstawą metodologiczną, a ontologie dziedzin przedmiotowych ważnym elementem konstrukcyjnym semantycznych systemów reprezentacji wiedzy. W artykule omówiono budowanie ontologii w oparciu o edytor ontologii FluentEditor i język CNL (Controlled Natural Language). Przykładową ontologię dotyczącą fragmentu procesu produkcji rolniczej wykorzystano do budowy semantycznej bazy wiedzy. W tym celu wykorzystano projekt architektury opartej o strukturalno-funkcjonalną kompozycję systemów AllegroGraph, Drupal i MongoDB.

Słowa kluczowe: bazy wiedzy, sieć semantyczna, ontologia, Fluent Editor, Protege, Drupal, AllegroGraph, MongoDB

Ontological Knowledge Engineering

Summary

Ontological knowledge engineering is a good methodological background of semantic knowledge representation in systems design, and ontologies are important components of it. The paper presents process building ontology using the ontology editor FluentEditor and CNL (Controlled Natural Language). A sample ontology of agricultural production process was used to build an example of semantic knowledge base. For this purpose we applied software systems like AllegroGraph, Drupal and MongoDB.

Keywords: knowledge, semantic web, ontology, Fluent Editor, Protege, Drupal, AllegroGraph, MongoDB

JEL: D80, C88, L86