STUDIA OECONOMICA POSNANIENSIA 2013, vol. 1, no. 5 (254)

Szymon Łazaruk
Uniwersytet Ekonomiczny w Poznaniu, Wydział Informatyki i Gospodarki Elektronicznej, Katedra Informatyki Ekonomicznej
s.lazaruk@kie.ue.poznan.pl

WYKORZYSTANIE TECHNOLOGII SEMANTYCZNYCH W PROCESIE INTEGRACJI DANYCH NA POTRZEBY JEDNOSTEK SAMORZĄDU TERYTORIALNEGO

Streszczenie: Integracja danych pochodzących z różnych źródeł stanowi wyzwanie, nad którego rozwiązaniem pracują naukowcy z różnych ośrodków badawczych. Jednym z największych problemów procesu integracji jest ustalenie, które elementy danego schematu informacyjnego odpowiadają swoim znaczeniem elementom innego schematu. Wysoki poziom skomplikowania i długi czas tworzenia rozwiązań integrujących opartych na języku SQL pociąga za sobą wysokie koszty. Zmiana zbioru integrowanych źródeł często wiąże się z ponoszeniem kolejnych kosztów przebudowy logiki systemu. W celu rozwiązania tego problemu, w ostatnich latach coraz powszechniejsze staje się wykorzystanie technologii semantycznego Internetu. Proponowana w niniejszym artykule metoda integracji heterogenicznych baz danych, wykorzystująca technologie semantyczne, została wdrożona w ramach projektu „Platforma integracyjna jako metodyka tworzenia rozwiązań dla instytucji samorządowych", zrealizowanego przez zespół Katedry Informatyki Ekonomicznej Uniwersytetu Ekonomicznego w Poznaniu. U podstaw projektu leżało przekonanie, że współdzielenie zasobów (w tym danych) pozwoli jednostkom samorządowym na zmniejszenie kosztów tworzenia, wdrażania i utrzymania systemów przeznaczonych do świadczenia usług drogą elektroniczną.

Słowa kluczowe: technologie semantycznego Internetu, integracja danych, heterogeniczność źródeł danych.

Klasyfikacja JEL: C81, C88, D02, D83, M15, M21, H79, H83, O38.

Wstęp

Skuteczność budowania społeczeństwa informacyjnego w dużej mierze zależy od działań jednostek samorządu terytorialnego (JST). To na nich zazwyczaj spoczywa ciężar udostępniania obywatelom usług drogą elektroniczną. Niestety, możliwości zapewnienia odpowiedniego poziomu usług przez samorządy lokalne są często

w znacznym stopniu ograniczone. O ile samorządy wielkomiejskie, w ramach wykonywania swoich zadań, powszechnie udostępniają obywatelom narzędzia umożliwiające zarządzanie sprawami realizowanymi przez urząd, o tyle są to często autorskie koncepcje, a ich wielość skutkuje zmniejszeniem poziomu interoperacyjności [Abramowicz i in. 2008b]. Nawet w ramach pojedynczych jednostek samorządu terytorialnego (podobnie jak u podmiotów gospodarczych) często funkcjonuje wiele systemów informatycznych. Dane przechowywane są zatem w różnych, wzajemnie odizolowanych, heterogenicznych bazach. Prowadzi to do sytuacji, w której w ramach jednego podmiotu dane są rozproszone, a jednocześnie nie można ich zastąpić wspólnym magazynem danych, bo przynajmniej część baz musi pozostać w pełni operacyjna wobec wykorzystujących je aplikacji. W efekcie, przy budowaniu nowych systemów, unikając integracji już posiadanych danych, doprowadza się do ich powielenia, redundancji i ryzykuje się ich niespójność.

Tradycyjne podejścia do integracji danych nie dostarczają efektywnych, nieskomplikowanych i łatwych w implementacji sposobów na rozwiązanie przedstawionego problemu. W rezultacie JST są zmuszone ponosić wysokie koszty wdrożeń rozwiązań integrujących.

Artykuł jest poświęcony zagadnieniu integracji heterogenicznych baz danych. Przedstawiono w nim podstawowe problemy procesu integracji oraz zaproponowano wykorzystanie technologii semantycznych do opisu źródeł danych, w celu uproszczenia całego procesu integracji i w rezultacie obniżenia kosztów tworzenia nowych rozwiązań. W dalszej części artykułu, zamieszczono przykład zastosowania proponowanej metody oraz przedstawiono praktyczne korzyści wynikające z wdrożonego mechanizmu.

1. Tradycyjne podejścia do integracji danych

Istotnym aspektem związanym z tworzeniem systemów informatycznych jest zapewnienie ich interoperacyjności. Pojęcie interoperacyjności nie jest jednoznaczne i istnieje wiele jego definicji. Większość z nich powstała na potrzeby prowadzonych badań, wdrażanych projektów lub opracowywanych strategii. Najczęściej jednak wykorzystuje się definicje zaproponowane przez IEEE (Institute of Electrical and Electronics Engineers) oraz Komisję Europejską. Pozostałe definicje w dużym stopniu bazują na tych dwóch podstawowych.

Definicja 1 (IEEE). Interoperacyjność oznacza zdolność dwóch lub większej liczby systemów informatycznych lub ich komponentów do wymiany informacji i do jej użycia [IEEE 1990].

Definicja 2 (Komisja Europejska). Interoperacyjność oznacza zdolność systemów ICT i procesów biznesowych przez nie wspieranych do wymiany danych

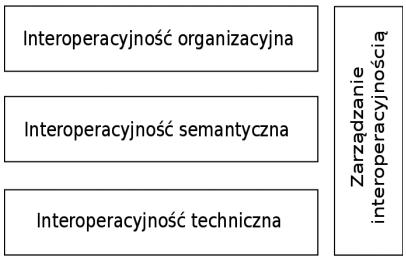
i do wykorzystywania współdzielonej między nimi informacji i wiedzy [European Commission 2004].

Pomimo różnic pomiędzy tymi definicjami można stwierdzić, że obydwie podkreślają zdolność systemów (lub komponentów, procesów biznesowych) do wymiany danych lub informacji, a następnie do ich wykorzystania, co jest sednem pojęcia interoperacyjności. Natomiast zagwarantowanie owej zdolności stanowi problem i wyzwanie dla wielu prowadzonych projektów. Zapewnienie interoperacyjności staje się kwestią pewnych porozumień pomiędzy podmiotami opierającymi swoje działanie na wykorzystaniu informacji. Postanowienia płynące z porozumień powinny mieć wpływ na systemy informatyczne w całym cyklu ich życia, a szczególnie w fazie analizy i projektowania, kiedy to decyduje się o przyszłym kształcie rozwiązania i ewentualnie o wykorzystaniu narzędzi integrujących.

W 2004 roku Komisja Europejska przedstawiła pierwszą wersję Europejskich Ram Interoperacyjności (European Interoperability Framework, EIF). Równocześnie powstało wiele krajowych ram interoperacyjności. Rozpoczęto liczne badania i projekty (zwłaszcza na gruncie europejskim) poświęcone temu zagadnieniu. Jednakże większość z nich opiera się na referencyjnym modelu interoperacyjności wprowadzonym w EIF. Zgodnie z nim, rozróżnia się się trzy podstawowe poziomy:

– interoperacyjność techniczną – odnoszącą się do aspektów technicznych łączenia systemów komputerowych i usług,
– interoperacyjność semantyczną – zakładającą możliwość poprawnego zrozumienia znaczenia informacji wymienianej przez różnorodne rozwiązania informatyczne,
– interoperacyjność organizacyjną – dotyczącą tworzenia procesów biznesowych i ustanawiania zasad współpracy pomiędzy organizacjami.

Dodatkowo wyróżnia się także warstwę „zarządzania" (ang. *governance*) interoperacyjności dotyczącą politycznych, prawnych, kulturowych i systemowych uwarunkowań, które mają wpływ na rozwój i wykorzystanie interoperacyjnych rozwiązań informatycznych.



Rysunek 1. Podstawowe warstwy interoperacyjności

Źródło: Tambouris i in. [2007]

Pojęcie integracji źródeł danych odnosi się do procesu integracji więcej niż jednego źródła (np. bazy danych) w jedną, ogólniejszą formę [Jarke i in. 2003]. Często pojęcie to jest używane jako określenie części ogólniejszego procesu, na przykład ma to miejsce w hurtowniach danych, kiedy po integracji źródeł danych zwykle następuje ich agregacja i analiza. Pojęcie integracji źródeł danych jest związane z pojęciem integracji danych i jest jedną z kluczowych kwestii dotyczących zapewnienia interoperacyjności systemów informacyjnych.

1.1. Problemy integracji

U podstaw badania metod i systemów integrujących leżą zjawiska autonomiczności źródeł danych oraz ich heterogeniczność.

Pojęcie autonomii definiujemy jako możliwość samostanowienia danego bytu o sobie. W odniesieniu do źródeł danych, autonomię należy rozumieć jako możliwość stanowienia podmiotu będącego właścicielem źródła danych o wszelkich aspektach jego istnienia. W szczególności wyróżniamy [Sheth 1999]:

- autonomię projektowania warstwy danych; dotyczy ona:
  - rodzaju reprezentacji danych,
  - schematu źródła danych,
  - rodzaju przechowywanej informacji (np. dziedziny, której dotyczy),
  - nazewnictwa poszczególnych elementów schematu informacyjnego,
  - ograniczeń narzuconych na dane,
  - danych samych w sobie,
- autonomię wykonania oznaczającą, że podmiot zewnętrzny nie może wymusić na źródle danych wykonania żadnych operacji (chyba że podmiot zarządzający źródłem zezwolił na taką możliwość),
- autonomię komunikacji odnoszącą się do zdolności decydowania o zaistnieniu komunikacji lub jej braku.

Aspekt autonomii źródła danych nabiera na znaczeniu, gdy przynajmniej jedno ze źródeł jest zarządzane przez podmiot niezainteresowany przeprowadzeniem procesu integracji, czego implikacją jest brak możliwości dostosowania schematu informacyjnego źródła w celu uproszczenia procesu integracji.

Zjawiskiem zajmującym szczególne miejsce w problemie integracji danych jest heterogeniczność ich źródeł [Hull 1997]. W szerszym ujęciu heterogeniczność systemów informacyjnych w znacznym stopniu utrudnia ich interoperacyjność [Aparício, Farias i dos Santos 2005]. W swojej pracy A.P. Sheth [1999] wyróżnił cztery poziomy heterogeniczności:

1) systemową – odnosi się do niekompatybilności sprzętowej i programowej systemów,
2) syntaktyczną – spowodowaną wykorzystaniem różnych języków zapytań i sposobów reprezentacji danych,

3) strukturalną – jest powodowana zastosowaniem różnych modeli i schematów źródeł danych,
4) semantyczną – przejawia się w wieloznaczności terminów.

C. Batini, M. Lenzerini i S.B. Navathe [1986], R.M. Colomb [1997] oraz W. Kim i J. Seo [1991] pominęli pierwszy poziom, wyróżniając tylko kolejne trzy.

Chen Hian Goh [Goh i in. 1999] wymienia trzy główne przyczyny występowania heterogeniczności semantycznej:

– konflikty zagmatwane (ang. *confounding conflicts*) – występują, gdy pewne informacje stwarzają pozory niesienia tego samego znaczenia, lecz różnią się w rzeczywistości (np. z powodu innego kontekstu czasowego),
– konflikty skalowania (ang. *scaling conflicts*) – występują, gdy stosowane są różne konteksty referencyjne dla określenia pewnej wartości (np. różne waluty),
– konflikty nazewnictwa (ang. *naming conflicts*) – występują, gdy zastosowane schematy nazewnictwa różnią się znacząco; najczęściej są spotykane w postaci synonimów i homonimów.

Najpoważniejszym problemem będącym przedmiotem badań nad integracją danych jest obecnie heterogeniczność semantyczna. W celu zapewnienia semantycznej interoperacyjności w heterogenicznych systemach informacyjnych, znaczenie informacji wymienianej pomiędzy tymi systemami powinno być jednakowe.

1.2. Pojęcie systemu integrującego

Systemy integrujące definiuje się formalnie jako trójkę [Lenzerini 2002]:

$$\langle G, S, M \rangle, \tag{1}$$

gdzie:

G – jest *globalnym schematem*, wyrażonym w postaci języka L_G nad alfabetem A_G; alfabet zawiera po jednym symbolu dla każdego elementu G (np. dla każdej relacji, w przypadku gdy G jest modelem relacyjnym, lub klasy, gdy G jest zorientowany obiektowo),

S – jest zbiorem *schematów (heterogenicznych) źródeł*, wyrażonym w postaci języka L_S nad alfabetem A_S; alfabet zawiera po jednym symbolu dla każdego elementu źródeł S,

M – jest *odwzorowaniem*, które „tłumaczy" zapytania pomiędzy G i S. Odwzorowanie M składa się z powiązań pomiędzy zapytaniami do G i zapytaniami do S; gdy użytkownik zadaje zapytanie do systemu integracyjnego, tak naprawdę kieruje zapytanie do schematu globalnego G, a wspomniane powiązania należące do M zapewniają połączenie pomiędzy elementami globalnego schematu i elementami schematu należącego do odpowiedniego źródła z S.

W literaturze przedstawiane są dwa podstawowe podejścia stosowane do określania odwzorowań M: *metoda globalnego widoku* (ang. *global-as-view*, GAV) [Halevy 2001] oraz *metoda lokalnych widoków* (ang. *local-as-view*, LAV) [Ullman

l997]. Sposoby te rzutują na dalsze tłumaczenie zapytań do poszczególnych źródeł danych.

Metoda globalnego widoku (GAV)
W tej metodzie, zwanej także „zapytanio-centryczną", globalny schemat jest modelowany jako zbiór widoków na *S*. Odwzorowanie *M* przydziela każdemu elementowi widoku *G* konkretne zapytanie do *S*. W podejściu tym przetwarzanie zapytań jest trywialne, ponieważ wszystkie powiązania pomiędzy *G* i *S* są dokładnie określone. Główna trudność wykorzystania metody GAV leży w procesie tworzenia mediatora, odpowiedzialnego za wybór konkretnego planu wykonania ekstrakcji oraz zajmującego się integracją wyekstrahowanych danych. W przypadku dodania nowego źródła do zbioru *S* mediator musi zostać zaktualizowany, co może być procesem pracochłonnym i w rezultacie kosztownym. Dlatego metoda globalnego widoku jest optymalna, gdy *S* nie ewoluuje znacząco w czasie.

Metoda lokalnych widoków (LAV)
W tej metodzie, zwanej także „źródło-centryczną", punktem wyjściowym jest przyjęcie globalnego schematu *G*. Następnie poszczególne źródła są modelowane jako zbiór widoków na ten schemat. Odwzorowanie *M* przydziela każdemu elementowi *S* konkretne zapytanie do *G*. W metodzie tej, w przeciwieństwie do metody GAV, powiązania pomiędzy *G* i *S* nie są już tak precyzyjnie określone. Trudność polega tu na określeniu, jak wyekstrahować pożądane elementy źródła, i jest przeniesiona na mechanizm wykonywania zapytań. Problem tłumaczenia zapytań w metodzie LAV należy do problemów NP-trudnych. Należy podkreślić, że w metodzie LAV dodanie nowego źródła do *S* nie powoduje konieczności zmiany schematu globalnego. Metoda ta cechuje się zatem wysoką modularnością i wielokrotnością użycia (zmiana źródła wymaga jedynie zmiany jego definicji).

2. Semantyka a integracja danych

Jednym z największych problemów w integracji danych pochodzących z różnych źródeł jest ustalenie, które z elementów danego schematu odpowiadają swoim znaczeniem danym elementom innego schematu [Lenzerini 2002]. Problem ten nie znika nawet wówczas, gdy oba (wszystkie) źródła danych wykorzystują ten sam model, na przykład relacyjne bazy danych, zarządzane przez system zarządzania bazą danych tego samego dostawcy.

W ramach pojedynczej bazy danych, jej twórcy za pomocą odpowiedniej składni kluczy obcych mogą, na poziomie strukturalnym, powiązać ze sobą dane zawarte w różnych tabelach. Jedna lub więcej kolumn z tabeli może być oznaczona jako klucz obcy, a jego (połączone – w przypadku kilku kolumn) wartości odpowiadają wartościom takiej samej liczby kolumn dla jednego lub więcej rekordów w innej tabeli.

Aplikacje SQL (*Structured Query Language*) łączą informacje zawarte w tych tabelach na podstawie relacji pomiędzy wartościami w odpowiednich kolumnach obu tabel. Niestety, składnia SQL nie ma operatora odpowiadającego żądaniu operacji *połącz dwie tabele na podstawie klucza obcego*, zatem tworząc aplikację, programista jest zmuszony określić, jakie relacje tego typu występują w bazie, i przy pisaniu każdego z zapytań w sposób jawny (z wykorzystaniem *join*) odpowiednio połączyć tabele. W małych bazach danych jest to oczywiście zadanie trywialne, jednak bazy wykorzystywane w przedsiębiorstwach zawierają setki, tysiące tabel, często również z setkami kolumn. Powoduje to, że zapamiętanie wszystkich relacji jest niemożliwe, a praca programisty jest bardzo utrudniona, gdyż musi on wielokrotnie sprawdzać w dokumentacji bazy związki pomiędzy kolumnami tabel. Oczywistym rozwiązaniem wydaje się „uwidocznienie" tych relacji, tak by można było je w prosty sposób odczytywać. Na poziomie składni języka przy budowaniu zapytania SQL istnienie relacji klucza obcego jest całkowicie bezużyteczne. Język SQL nie odpowie nam na pytanie, *czy dane dwie kolumny mają ze sobą jakiś związek*, i – o ile typ danych będzie się zgadzać – bez problemu połączymy kolumnę oznaczającą na przykład *numer* buta osoby z *numerem* budynku w adresie jakiegoś sklepu.

Jak widać, ta kwestia jest problemem już w przypadku jednej bazy danych, gdzie zazwyczaj stosuje się w miarę spójną terminologię. Natomiast w sytuacji integracji baz o różnych przeznaczeniach, sytuacja komplikuje się jeszcze bardziej.

W sytuacji idealnej, programista powinien mieć do dyspozycji narzędzie pozwalające łatwo identyfikować te powiązania pomiędzy tabelami, które są istotne w jego aplikacji. Przy tym ta identyfikacja nie musiałaby opierać się na analizie syntaktycznej nazw tabel/kolumn.

Pojawia się tu miejsce na zastosowanie technologii semantycznego Internetu[1]. Z ich wykorzystaniem można utworzyć pewne dodatkowe metadane, nieoparte na strukturze bazy, a umożliwiające wykonywanie operacji *join* bazujących na ujawnionych kluczach obcych. Co więcej, można by dodatkowo określić relacje, nieokreślone w sposób jawny w schemacie SQL bazy danych, w sytuacji, gdy mimo braku klucza obcego znaczenie kolumn dwóch tabel jest ściśle powiązane.

Niestety, utworzenie takich metadanych na dużą skalę dla istniejących baz wydaje się niemożliwe. Już samo ich budowanie w sposób spójny dla nowo tworzonych baz jest mało prawdopodobne. Pozostawieni zatem bez dodatkowych metadanych zawierających wszystkie te relacje, mamy jedynie do dyspozycji metadane strukturalne SQL – relacje klucza obcego. Nie dostarczają nam one co prawda informacji o *znaczeniu* danych, ale mówią, że *jakiekolwiek znaczenie ma dana kolumna, pokrywa się ono ze znaczeniem innej danej kolumny*.

Podobny poziom semantyki może być uzyskany dzięki zastosowaniu rozróżnialnego typu danych (ang. *distinct type*) [Tuzinkiewicz i Fedyczak 2006]. Typ ten jest

[1] Semantic Web stack, http://www.w3.org/2006/Talks/1023-sb-W3CTechSemWeb/SemWebStack-tbl-2006a.png [dostęp: 1.12.2012)].

oparty na innym typie danych, ma jednak własny zestaw operacji. Oznacza to, że można utworzyć na przykład typ danych *numer budynku* oparty na typie INTEGER, który nie będzie (domyślnie) miał żadnego zestawu operacji, a wartości tego typu nie będą mogły być użyte jako typ INTEGER. Należałoby określić operacje dozwolone na tym typie danych, a jego zastosowanie pozwoliłoby w łatwy sposób określić, które kolumny mają to samo znaczenie.

W wypadku wielu baz danych sytuacja się komplikuje. Większość silników bazodanowych nie wspiera „międzybazowych" kluczy obcych, a te, które wspierają, ograniczają tę funkcjonalność do baz uruchomionych na jednym serwerze. W rezultacie, relacje klucza obcego bardzo rzadko mogą być użyte do łączenia danych z dwóch lub więcej źródeł.

W każdym z takich źródeł możemy zdefiniować typy danych rozróżnialnych dla tych samych pojęć. Nazwa typu rozróżnialnego w każdej bazie może być inna, co jest pewnym utrudnieniem. Jednak przy relatywnie małej ilości dodanych semantycznych metadanych, można wykonać duży krok w kierunku unifikacji pojęciowej. Jeśli dostarczymy tych dodatkowych metadanych w postaci grafów RDF, używając RDFS i OWL[2], uzyskamy sposób zapisu relacji pomiędzy typami rozróżnialnymi oraz kolumnami przez nie zdefiniowanymi. Dzięki zastosowaniu automatycznego wnioskowania do całych kolekcji takich metadanych, możliwe byłoby wyznaczanie kolejnych zależności, nawet jeśli nie zostały one pierwotnie jawnie zdefiniowane.

2.1. Technologie semantyczne wykorzystywane do integracji danych

Interesującym kierunkiem badań nad integracją danych w ostatnich latach jest wykorzystanie technologii semantycznych – w szczególności chodzi tu o strukturę opisu zasobów (ang. *Resource Description Framework*, RDF) [W3C 2004b], język związanych z nim reprezentacji wiedzy RDF Schema (RDFS) [W3C 2004a] oraz protokół i język zapytań SPARQL (ang. *SPARQL Protocol and RDF Query Language*) [W3C 2008a, 2008b].

Struktura RDF jest językiem metadanych dla zasobów globalnej sieci. Podstawowym założeniem przy jego tworzeniu była zdolność do maszynowego przetwarzania opisów zasobów. Jest on wykorzystywany jako język modelowania informacji oraz reprezentowania informacji o dowolnych obiektach. Opierając się na koncepcji zdań, RDF składa się z trzech elementów: podmiotu, orzeczenia (predykatu) i dopełnienia (obiektu). Formalnie RDF jest strukturą opartą na grafie, w którym węzły reprezentują obiekty (będące zarówno podmiotami, jak i dopełnieniami relacji), a łuki reprezentują relacje. Ze względu na swoją ogólność RDF nadaje się do reprezentowania informacji o dowolnych obiektach. Ponieważ RDF nie narzuca żadnych ograniczeń na strukturę, w celu określenia specyficznych ograniczeń wykorzystuje

[2] Skróty wyjaśniono w dalszej części tekstu.

się dwie taksonomie: RDF Schema oraz OWL. Pierwsza z nich, RDFS, umożliwia definiowanie pojęć (zwanych klasami) oraz predykatów.

Elementy trójki RDF mogą być podane wprost w dokumencie za pomocą literału, jednak dalsza analiza takiego dokumentu musiałaby się opierać na syntaktyce wprowadzonego ciągu znaków. Dlatego zarówno w RDF, jak i w RDFS wykorzystuje się URI (and. *Unified Resource Identyfier*), czyli uniwersalne identyfikatory zasobów. Dzięki temu różne dokumenty mogą się odnosić do pojęć już wcześniej zdefiniowanych. W ten sposób zapewniamy pewien poziom interoperacyjności semantycznej i co za tym idzie w znacznym stopniu upraszczamy proces integracji danych określonych takimi pojęciami [Yung 2007].

Protokołem i językiem zapytań opracowanym specjalnie dla RDF jest SPARQL. Język zapytań SPARQL jest zbudowany na zasadzie wyszukiwania wzorców, przy czym zapytania wyglądają i zachowują się jak w RDF. Dzięki temu SPARQL pozwala między innymi na wykorzystywanie niejawnych (w zapytaniu) połączeń pomiędzy obiektami oraz na odpytywanie wielu zróżnicowanych źródeł danych w ramach jednego zapytania.

2.2. Adaptery danych

Jak przedstawiono wcześniej, integracja danych z różnych źródeł stanowi wyzwanie. Przeprowadzenie procesu integracji na poziomie języka SQL pomiędzy różnymi systemami zarządzania bazą danych nie jest możliwe, a budowanie aplikacji integrujących źródła danych i wykorzystujących język SQL może być bardzo skomplikowane, chociażby z tego powodu, że semantyka danych nie jest w żaden sposób odwzorowana w schematach źródeł. Język SPARQL jest pozbawiony ograniczeń SQL pod względem liczby źródeł danych oraz możliwości wykorzystania odwołań do zdefiniowanych uprzednio pojęć, jednak przeznaczony jest do odpytywania źródeł RDF. Jak stwierdzono uprzednio, RDF ma wysoki stopień ogólności. Pozwala on między innymi na reprezentację danych zawartych w modelu relacyjnym i przechowywanych w bazie danych [Berners-Lee 1998].

Przeniesienie danych przechowywanych w relacyjnych bazach do RDF jest trudne do wyobrażenia, chociażby ze względu na konieczność utrzymania tych źródeł w pełni operacyjnych wobec wykorzystujących je aplikacji. Powielenie danych i utrzymywanie równolegle dwóch magazynów jest również złym rozwiązaniem, przede wszystkim ze względu na problemy z synchronizacją danych.

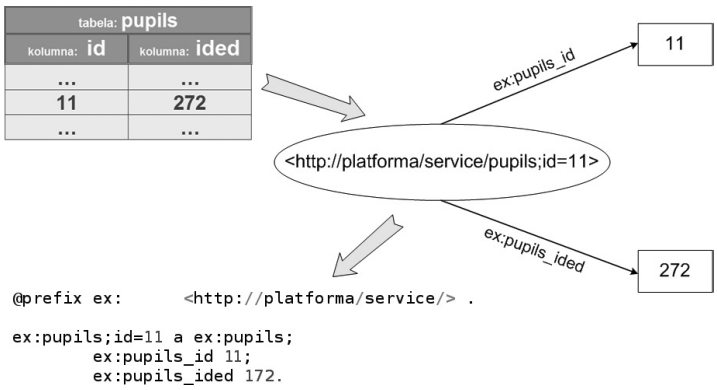
Rozwiązaniem tego problemu jest wykorzystanie adapterów danych, gdzie przez pojęcie adaptera autor niniejszego artykułu rozumie „narzędzie mogące – na żądanie – dokonywać ekstrakcji danych z istniejących ustrukturyzowanych źródeł danych oraz reprezentować je jako RDF" [Sauermann i Schwarz 2005].

2.3. Odwzorowanie relacyjnej bazy danych do RDF

Wykorzystanie adapterów danych daje możliwość wykorzystania infrastruktury RDF, RDFS oraz SPARQL w celu odpytywania rozproszonych baz danych, niezależnie od ich heterogeniczności. W celu zastosowania tej metody integracji danych, konieczne jest opracowanie zestawu reguł określających zasady odwzorowania pojęć biznesowych na elementy schematów informacyjnych. Utworzone odwzorowanie musi umożliwiać transformację identyfikatorów z powrotem do pierwotnych nazw tabel i kolumn w bazie. Dodatkowo, klucze podstawowe jednoznacznie identyfikujące rekordy w tabelach muszą być włączone w URI, by identyfikować węzły w grafie RDF. Wyróżnia się dwa podstawowe podejścia do tego zadania: automatyczne oraz konfigurowalne.

W 1998 roku Tim Berners-Lee określił ogólne zasady reprezentowania danych z modelu relacyjnego za pomocą RDF. Zgodnie z tymi zasadami:

- każda tabela jest reprezentowana przez klasę RDF,
- każdy wiersz jest instancją tej klasy,
- każda kolumna tabeli jest właściwością,
- każde pole tabeli jest wartością [Berners-Lee 1998].



Rysunek 2. Odwzorowanie modelu relacyjnego w RDF

Źródło: Opracowano na podstawie [Berners-Lee 1998]

Automatyczne tworzenie odwzorowań (według zasad zgodnych z powyższą koncepcją, a jedynie opartych na różnym zapisie) zostało zaimplementowane w wielu narzędziach. Główną zaletą, wynikającą z takiego podejścia, jest zwolnienie użytkownika z konieczności ingerowania w proces. Z drugiej strony, wykorzystując taki mechanizm, rezygnujemy z korzyści, które dałoby nam używanie wspólnego słownika pojęciowego, lub nawet połączenie elementów schematów źródeł danych z terminami występującymi w istniejących ontologiach dziedzinowych. Przykładowymi narzędziami implementującymi omawiany mechanizm są Virtuoso RDF

View [Blakeley 2007] i SquirrelRDF [Seaborne, Steer i Williams 2007], które używają unikatowego identyfikatora wiersza (klucza podstawowego) jako obiektu RDF, oraz D2RQ [Bizer i Cyganiak 2007; Bizer i Seaborne 2004], który dodatkowo umożliwia definiowanie odwzorowań przez użytkowników. Wykorzystanie automatycznego tworzenia odwzorowań zapewnia tym narzędziom szybkość i prostotę działania. Czasami wykorzystanie tego mechanizmu stanowi jedynie punkt wyjściowy do uszczegóławiania odwzorowań. Innym wariantem automatycznego tworzenia odwzorowań jest wykorzystanie istniejącej ontologii do stworzenia podstawowych odwzorowań [Hu i Qu 2007], które po sprawdzeniu pod kątem spójności są rozbudowywane. Wykorzystana ontologia podnosi jakość tak zbudowanych odwzorowań.

Drugą grupę podejść do tworzenia odwzorowań możemy podzielić na dwie części. W pierwszej grupie znajdą się takie narzędzia, jak wspomniane D2RQ [Bizer i Cyganiak 2007], które wykorzystują automatyczny mechanizm, a następnie pozwalają użytkownikowi połączyć część elementów schematu relacyjnej bazy danych z wybranymi przez siebie zasobami określonymi przez URI. Działania te mają charakter selektywny i są wtórne wobec automatycznego tworzenia odwzorowania. W drugiej grupie znalazły się rozwiązania polegające na włączeniu ontologii dziedzinowej, która zazwyczaj w relacyjnym modelu danych pozostaje niejawna. Wykorzystanie semantyki pozyskanej z ontologii, zamodelowanej w sposób jawny w repozytoriach RDF, pozwala aplikacjom na wykorzystanie tych „dodatkowych informacji" do wykonywania zapytań łączących ze sobą pojęcia powiązane w ontologii, na przykład biomedycznej [Sahoo i in. 2008]. Dodatkowo, jak wykazał Byrne [2008], wykorzystanie odwzorowań wyprowadzonych z ontologii dziedzinowej pozwala na znaczną redukcję trójek przechowujących redundantną lub niepotrzebną wiedzę. Ontologia może być zaczerpnięta z publicznie dostępnych zasobów lub wyprowadzona z lokalnych ontologii, będących wynikiem automatycznego procesu opisanego wcześniej.

Jedną z konsekwencji wyboru podejścia do tworzenia odwzorowań jest konieczność (lub jej brak) jawnego umieszczania w treści zapytania SPARQL zależności klucza obcego w celu powiązania zasobów. Ponieważ przedmiotem relacji klucza obcego jest zasób sam w sobie, a nie dane wykorzystane do jego identyfikacji, a węzeł w grafie RDF może służyć jako przedmiot i podmiot relacji, zapytanie SPARQL niesie ze sobą wystarczająco dużo informacji, by wyrazić relację klucza obcego. W związku z tym taka relacja może być przez to zapytanie wykorzystana niejawnie (bez konieczności jej jawnego wskazania).

3. Przykład zastosowania

Proponowana w niniejszym artykule metoda integracji heterogenicznych baz danych, wykorzystująca technologie semantyczne, została wdrożona w ramach projektu „Platforma integracyjna jako metodyka tworzenia rozwiązań dla instytucji

samorządowych", zrealizowanego przez zespół Katedry Informatyki Ekonomicznej Uniwersytetu Ekonomicznego w Poznaniu. U podstaw projektu leżało przekonanie, że współdzielenie zasobów (w tym danych) pozwoli jednostkom samorządowym na zmniejszenie kosztów tworzenia, wdrażania i utrzymania systemów przeznaczonych do świadczenia usług drogą elektroniczną [Abramowicz i in. 2008b]. Platforma, której koncepcja została szerzej opisana w osobnym artykule, miała umożliwiać działanie prostych usług, dostosowanych do potrzeb poszczególnych samorządów. Zaproponowano również metodykę tworzenia tych usług [Abramowicz i in. 2008a]. Ich opracowanie powinno być możliwe proste, oparte na nieskomplikowanych i powszechnie wykorzystywanych standardach, co w rezultacie ma skutkować relatywnie niskim kosztem tworzenia. Przy realizacji projektu dodatkowo opracowano i wdrożono Usługę, co miało na celu z jednej strony przetestowanie działania platformy, a z drugiej zademonstrowanie proponowanych w metodzie mechanizmów, w tym mechanizmu integracji heterogenicznych baz danych. Konkretne rozwiązanie integrujące jest wynikiem analizy kontekstu wdrożenia, w tym technicznych możliwości jednostki, w której wdrożenie miało nastąpić.

Ponieważ projekt był przeznaczony dla jednostek administracji samorządowej, Usługa musiała być związana z zadaniami publicznymi znajdującymi się gestii tychże jednostek. Jednym z obszarów, za który odpowiedzialny jest samorząd gminy, jest edukacja publiczna [Ustawa z dnia 8 marca 1990 r. o samorządzie gminnym; Ustawa z dnia 5 czerwca 1998 r. o samorządzie powiatowym]. W ramach prac nad Usługą nawiązano współpracę z Gimnazjum nr 2 im. Jana Kochanowskiego w Murowanej Goślinie. Współpraca ze szkołą pozwoliła na uszczegółowienie potrzeb biznesowych i technicznych. Jedną z potrzeb było dostarczenie narzędzia usprawniającego komunikację pomiędzy nauczycielami a rodzicami uczniów. Obecnie rodzice mają coraz mniej czasu, aby aktywnie uczestniczyć w życiu szkoły i na bieżąco śledzić osiągnięcia swoich dzieci. W rezultacie rodzice często zbyt późno dowiadują się o pewnych zdarzeniach. Ogranicza się także wpływ nauczycieli na postępowanie uczniów i proces wychowawczy.

3.1. Charakterystyka źródeł danych

Usługa opracowana na potrzeby gimnazjum w celu dostarczenia wszystkich funkcjonalności musiała korzystać zarówno z własnych danych, jak i z danych elektronicznego dziennika. Konieczne było zatem opracowanie mechanizmu integrującego. Jak przedstawiono wcześniej, problem integracji dotyczy źródeł różniących się pod względem schematu informacyjnego. Zróżnicowanie to może dotyczyć użytego modelu lub schematu źródła danych, gdzie poprzez model należy rozumieć formalizm, w którym tworzy się schemat. W przypadku Usługi mamy do czynienia z dwoma źródłami danych opartymi na modelu relacyjnym. O ile baza danych Usługi była tworzona w ramach projektu, o tyle wdrożony w Gimnazjum eDziennik jest produktem komercyjnym. Dostawca eDziennika ma wszelkie prawa w zakresie

autonomii projektowania warstwy danych swojej aplikacji, co oznacza, że nie była możliwa w żadnym zakresie modyfikacja modelu i schematu danych, sposobu ich reprezentacji ani istniejących zależności i ograniczeń narzuconych na dane. Z tego wynikała również heterogeniczność integrowanych baz danych. Ponieważ utworzona Usługa miała być stosowana wraz z elektronicznymi dziennikami innych dostawców, jej schemat oraz wykorzystane nazewnictwo nie mogło być dostosowane tylko do jednego produktu. Jedną z kluczowych przesłanek towarzyszących autorowi przy tworzeniu architektury danych Usługi było opracowanie metodyki tworzenia rozwiązań integrujących autonomiczne, rozproszone i heterogeniczne źródła danych, przy jednoczesnym nacisku na łatwość programowania aplikacji.

4. Zastosowane rozwiązanie integrujące

Najistotniejszą przesłanką wyboru rozwiązania integrującego źródła danych usługi końcowej było dostarczenie twórcom usług metody integracji, która:

- umożliwia integrację autonomicznych i rozproszonych źródeł danych,
- umożliwia integrację heterogenicznych źródeł danych, w szczególności:
  - możliwa jest integracja baz danych o różnych schematach informacyjnych,
  - możliwa jest integracja baz danych różnych producentów,
  - dodatkowo wybrane narzędzie pozwala na integrację baz danych oraz innych źródeł danych, np. repozytoriów RDF czy serwerów LDAP i IMAP,
- nie ogranicza liczby źródeł danych,
- w sytuacji gdy dodane zostaje nowe źródło danych, nie wymusza zmiany istniejącego kodu (w szczególności dotyczy to zmiany zapytań, których realizacja nie wymaga dostępu do nowego źródła),
- umożliwia budowę i realizację zapytań w oderwaniu od technologii, w której zostały utworzone źródła danych; zapytania mają być zatem budowane na wyższym poziomie niż na przykład zapytania SQL do bazy danych.

4.1. Charakterystyka rozwiązania

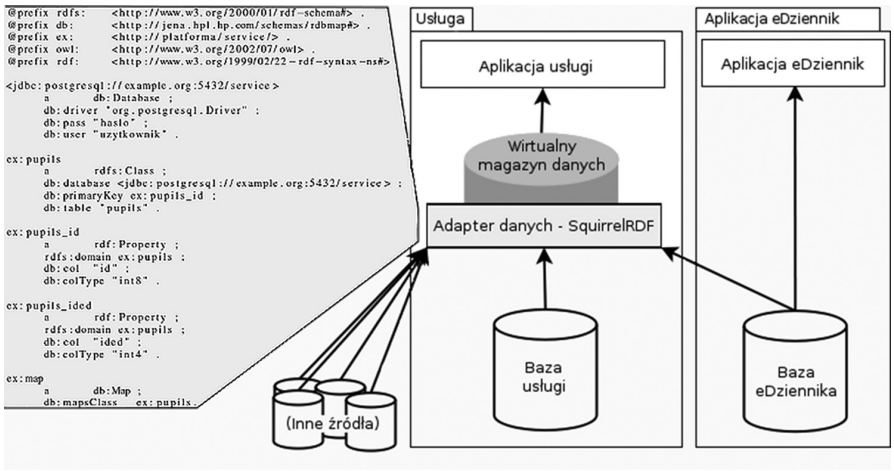
Proponowana metoda wykorzystuje opisane wcześniej technologie semantyczne:

- RDF – w celu utworzenia semantycznego opisu źródeł danych, odwzorowania pojęć biznesowych na elementy schematów informacyjnych źródeł danych,
- SPARQL – jako notację semantycznego języka zapytań, umożliwiając formułowanie zapytań rozciągniętych ponad wszystkimi źródłami danych.

Wykorzystane zostało również narzędzie SquirrelRDF[3], wchodzące w skład środowiska JENA[4] i oparte na silniku zapytań ARQ[5]. Szczególnie istotne jest wykorzy-

[3] Więcej na stronie http://jena.sourceforge.net/SquirrelRDF/ [dostęp: 1.12.2012].
[4] Więcej na stronie http://jena.sourceforge.net/index.html [dostęp: 1.12.2012].
[5] Więcej na stronie http://jena.sourceforge.net/ARQ/ [dostęp: 1.12.2012].

stanie wspomnianego odwzorowania pomiędzy pojęciami zawartymi w pliku RDF a elementami schematów informacyjnych. Pozwala to z jednej strony na budowanie zapytań w oderwaniu od schematu (a nawet modelu) źródła danych, a z drugiej na pozostawienie źródeł danych w nienaruszonej formie, w pełni operacyjnych wobec wykorzystujących je pierwotnie aplikacji. Narzędzie SquirrelRDF jest w tym procesie odpowiedzialne za tłumaczenie scentralizowanego zapytania SPARQL na rozproszone zapytania SQL oraz za konsolidację odpowiedzi do formy przewidzianej w zapytaniu.



Rysunek 3. Uproszczony schemat architektury Usługi wraz z plikiem konfiguracyjnym SquirrelRDF

Narzędzie SquirrelRDF używa metadanych, pochodzących ze sterownika JDBC, do konfiguracji, tj. identyfikacji tabel, kolumn i kluczy w bazie danych. Taka konfiguracja może być wykonana *ad hoc*, jednak w praktyce dużo wygodniej jest utworzyć plik konfiguracyjny i dostosować go do swoich potrzeb. Efektem procesu konfiguracji jest plik RDF (zaleca się stosowanie plików w formacie Turtle lub N3). Ów plik konfiguracyjny jest w rzeczywistości swego rodzaju mapą, zbudowaną na zasadzie RDFSchema, wzbogaconą o wyrażenia deklarujące klasy do odwzorowania oraz szczegóły połączeń z bazami danych. Nie jest konieczne opisywanie wszystkich elementów schematów integrowanych baz, wystarczy, aby w pliku znalazły się odwzorowania do interesujących nas elementów. Kluczowe dla zastosowania SquirrelRDF w celu integracji wielu źródeł danych jest to, że bazy danych są powiązane z klasami RDF wewnątrz pliku konfiguracyjnego, a nie z plikiem jako takim. Oznacza to, że jeden plik może zawierać szczegóły dotyczące elementów wielu baz, a w rezultacie, może być wykorzystany do realizacji scentralizowanych zapytań do rozproszonych źródeł. Rysunek 3 przedstawia uproszczony schemat architektury Usługi oraz część pliku konfiguracyjnego utworzonego w notacji N3 [Berners-Lee 2004].

4.2. Podsumowanie realizacji projektu

W projekcie osiągnięto między innymi następujące cele:
- Utworzono projekt platformy integracyjnej umożliwiającej świadczenie usług samorządu lokalnego poprzez kanał mobilny. Została opracowana architektura platformy, a następnie dokonano jej wdrożenia wraz z testową usługą demonstrującą między innymi mechanizmy integracji heterogenicznych baz danych z wykorzystaniem semantycznych języków reprezentacji wiedzy.
- Zaproponowano metodykę tworzenia usług mobilnych z wykorzystaniem opracowanej platformy usług. Metodyka ta została opisana w raporcie z wykonania projektu [Abramowicz i in. 2008b].

Podsumowanie

Zastosowanie zaprezentowanego podejścia do integracji danych pozwala na uproszczenie i przyspieszenie budowy systemów integrujących. W rezultacie oznacza to niższy koszt tworzenia, wdrażania i późniejszego utrzymywania opracowanych rozwiązań, co z kolei rzutuje na możliwości jednostek samorządowych do wprowadzania nowych usług elektronicznych dla obywateli.

Przedstawione rozwiązanie integrujące jest oparte na tzw. technologiach semantycznych. Są one otwartymi standardami z publicznie dostępnymi specyfikacjami, których użycie nie wiąże się z ponoszeniem żadnych opłat. Wykorzystanie RDF i budowanych za jego pomocą ontologii staje się coraz powszechniejsze, a liczba badań prowadzonych w ostatnich latach, związanych z ich wykorzystaniem, pozwala sądzić, że powszechność rozwiązań na nich opartych będzie dalej rosnąć. Wysoki poziom skomplikowania i długi czas tworzenia rozwiązań integrujących wykorzystujących SQL skutkuje wysokimi kosztami. Zmiana zbioru integrowanych źródeł często wiąże się z ponoszeniem kolejnych kosztów przebudowy logiki systemu. Jak wykazano, wykorzystanie technologii semantycznych pozwala na unifikację stosowanych pojęć i oderwanie się od syntaktycznej analizy powiązań elementów schematów źródeł. Pozwala to przenieść cały proces integracji na wyższy poziom. Programista zostaje zwolniony z konieczności posiadania szczegółowej wiedzy odnośnie do budowy każdego ze źródeł. Posługiwanie się tylko pojęciami biznesowymi (reprezentowanymi przez URI) jest wystarczające do budowania scentralizowanego zapytania SPARQL do rozproszonych i heterogenicznych źródeł. Dzięki wykorzystaniu opisanych poprzednio odwzorowań, programista musi jedynie wskazać, integracją których pojęć biznesowych, spośród reprezentowanych w systemie, jest zainteresowany. Skomplikowany proces przekształcania scentralizowanych zapytań SPARQL na zestawy zapytań SQL, w tym opracowywania planów ekstrakcji z poszczególnych źródeł, oraz konsolidacja uzyskanych wyników leżą już po stronie wykorzystanego narzędzia. To, wraz z prostotą składni SPARQL i jednocześnie

dużą ekspresywnością RDF, sprawia, że cały proces integracji jest dużo prostszy oraz wymaga od programisty znacznie mniej wysiłku i specjalistycznej wiedzy niż rozwiązania tradycyjne.

Opisane wdrożenie pokazuje praktyczne aspekty wykorzystania technologii semantycznych do integracji danych. Jakkolwiek konkretne rozwiązanie było dostosowane do wymagań związanych z rozwojem aplikacji dla szkoły i wynikało z charakterystyki projektu, to metoda integracji danych wykorzystująca technologie semantyczne ma charakter ogólny i nie jest w żaden sposób ograniczona liczbą i charakterem integrowanych źródeł.

Bibliografia

Abramowicz, W., Bassara, A., Łazaruk, S., Wiśniewski, M., Żebrowski, P., 2008a, *Platforma dla usług mobilnego samorządu*, w: *Systemy wspomagania organizacji 2008*, Naukowe Towarzystwo Informatyki Ekonomicznej, Katedra Informatyki Akademii Ekonomicznej w Katowicach, Ustroń, Polska.

Abramowicz, W., Bassara, A., Filipowska, A., Łazaruk, S., Wiśniewski, M., Żebrowski, P., 2008b, *Platforma integracyjna jako metodyka tworzenia rozwiązań dla instytucji samorządowych*, *Raport z wykonania projektu*, Uniwersytet Ekonomiczny w Poznaniu.

Aparício, A.S., Farias, O.L.M. i dos Santos, N., 2005, *Applying Ontologies in the Integration of Heterogeneous Relational Databases*, w: *AOW '05: Proceedings of the 2005 Australasian Ontology Workshop*, Australian Computer Society, Inc., Darlinghurst, Australia, s. 11–16.

Batini, C., Lenzerini, M., Navathe, S.B., 1986, *A Comparative Analysis of Methodologies for Database Schema Integration*, ACM Computing Surveys, vol. 18, no. 4, s. 323–364.

Berners-Lee, T., 1998, *Relational Databases on the Semantic Web*, Tech. Rep., W3C.

Berners-Lee, T., 2004, *Notation 3 – An Readable Language for Data on the Web*, W3C, http://www.w3.org/TR/rdf-schema/, n3 specification [dostęp: 1.12.2012].

Bizer, C., Cyganiak, R., 2007, *D2RQ – Lessons Learned*, w: *Position Paper for the W3C Workshop on RDF Access to Relational Databases*, http://www.w3.org/2007/03/RdfRDB/papers/d2rq-positionpaper/ [dostęp: 1.12.2012].

Bizer, C., Seaborne, A., 2004, *D2RQ – Treating Non-RDF Databases as Virtual RDF Graphs*, w: *ISWC 2004 (posters)*, http://sites.wiwiss.fu-berlin.de/suhl/bizer/pub/Bizer-D2RQ-ISWC2004-Poster.pdf [dostęp: 1.12.2012].

Blakeley, C., 2007, *RDF Views of SQL Data* (*Declarative SQL Schema to RDF Mapping*), wyd. v1.1, http://virtuoso.openlinksw.com/Whitepapers/pdf/Virtuoso_SQL_to_RDF_Mapping.pdf [dostęp: 1.12.2012].

Byrne, K., 2008, *Having Triplets – Holding Cultural Data as RDF*, w: Larson, M., Fernie, K., Oomen, J., Cigarran, J. (eds.), *Proceedings of the ECDL 2008 Workshop on Information Access to Cultural Heritage*, Aarhus, Denmark, September 18, 2008.

Colomb, R.M., 1997, *Impact of Semantic Heterogeneity on Federating Databases*, The Computer Journal, vol. 40, no. 5, s. 235–244.

European Commission, 2004, *European Interoperability Framework for Pan-European eGovernment Services Version 1.0*, Tech. Rep., European Commission.

Goh, C.H., Bressan, S., Madnick, S., Siegel, M., 1999, *Context Interchange: New Features and Formalisms for the Intelligent Integration of Information*, ACM Transactions on Information Systems, vol. 17, no. 3, s. 270–293.

Halevy, A.Y., 2001, *Answering Queries Using Views: A Survey*, The VLDB Journal, vol. 10, no. 4, s. 270–294.

Hu, W., Qu, Y., 2007, *Discovering Simple Mappings Between Relational Database Schemas and Ontologies*, w: *Proceedings of 6th International Semantic Web Conference* (*ISWC 2007*), *2nd Asian Semantic Web Conference* (*ASWC 2007*), Busan, Korea, s. 225–238.

Hull, R., 1997, *Managing Semantic Heterogeneity in Databases: A Theoretical Prospective*, w: *PODS '97: Proceedings of the Sixteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, ACM, New York, s. 51–61.

IEEE, 1990, *IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries*, Tech. Rep., Institute of Electrical and Electronics Engineers (IEEE).

Jarke, M., Lenzerini, M., Vassiliou, Y., Vassiliadis, P., 2003, *Fundamentals of Data Warehouses*, Springer Verlag.

Kim, W., Seo, J., 1991, *Classifying Schematic and Data Heterogeneity in Multidatabase Systems*, Computer, vol. 24, no. 12, s. 12–18.

Lenzerini, M., 2002, *Data Integration: A Theoretical Perspective*, w: *PODS '02: Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, ACM, New York, s. 233–246.

Sahoo, S.S., Bodenreider, O., Rutter, J.L., Skinner, K.J., Sheth, A.P., 2008, *An Ontology-driven Semantic Mash-up of Gene and Biological Pathway Information: Application to the Domain of Nicotine Dependence*, Journal of Biomedical Informatics, vol. 41, iss. 5, s. 752–765.

Sauermann, L., Schwarz, S. 2005, *Gnowsis adapter framework: Treating structured data sources as virtual rdf graphs*. w: Gil, Y., Motta, E., Benjamins, V.R., Musen, M.A. (eds.), *Proceedings of the 4th International Semantic Web Conference* (*ISWC 2005*), no. 3729, LNCS, 1016 ff., Springer, Galway, Ireland.

Seaborne, A., Steer, D., Williams, S., 2007, *SQL-RDF*, w: *W3C Workshop on RDF Access to Relational Databases*, Cambridge, USA.

Sheth, A.P., 1999, *Changing Focus on Interoperability in Information Systems: From System, Syntax, Structure to Semantics*, w: Goodchild, M.F., Kottman, C., Egenhofer, M.J. (eds.), *Interoperating Geographic Information Systems*, Kluwer Academic Publishers, Norwell, MA, http://lsdis.cs.uga.edu/library/download/S98-changing.pdf [dostęp: 1.12.2001].

Tambouris, E., Tarabanis, K., Peristeras, V., Liotas, N., 2007, *Study on Interoperability at Local and Regional Level*, http://www.epractice.eu/files/media/media1309.pdf [dostęp: 1.12.2012]

Tuzinkiewicz, L., Fedyczak, J., 2006, *Rozwój standardu SQL i jego implementacji*, w: Kozielski, S., Małysiak, B., Kasprowski, P., Mrozek, D. (red.), *Bazy danych: Struktury, algorytmy, metody*, Wydawnictwa Komunikacji i Łączności, Warszawa, s. 205-213.

Ullman, J.D., 1997, *Information Integration Using Logical Views*, w: *ICDT '97: Proceedings of the 6th International Conference on Database Theory*, Springer-Verlag, London, s. 19–40.

Ustawa z dnia 8 marca 1990 r. o samorządzie gminnym, Dz.U. nr 16, poz. 95.
Ustawa z dnia 5 czerwca 1998 r. o samorządzie powiatowym, Dz.U. nr 91 poz. 578.
W3C, 2004a, *RDF Vocabulary Description Language 1.0: RDF Schema*, W3C, http://www.w3.org/TR/rdf-schema/, w3C Recommendation [dostęp: 1.12.2012].
W3C, 2004b, *RDF/XML Syntax Specification (Revised)*, W3C, http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/, w3C Recommendation [dostęp: 1.12.20012].
W3C, 2008a, *SPARQL Protocol for RDF*, W3C, http://www.w3.org/TR/rdf-sparql-protocol/, w3C Recommendation [dostęp: 1.12.20012].
W3C, 2008b, *SPARQL Query Language for RDF*, W3C, http://www.w3.org/TR/rdf-sparql-query/, w3C Recommendation [dostęp: 1.12.20012].

APPLICATION OF SEMANTIC TECHNOLOGIES FOR THE NEEDS OF INTEGRATION OF HETEROGENEOUS DATABASES FOR LOCAL GOVERNMENT

Abstract: Integrating data from different sources is a challenge. One of the biggest problems of this process is to determine which elements of one information scheme correspond to relevant elements of another schema. The high level of complexity and time needed for development of solutions that integrate SQL-based result in high costs. Changing a set of integrated sources is often associated with incurring the costs of system-logic reconstruction. In order to solve this problem, in recent years the use of Semantic Web technologies has become increasingly popular. The method of integration of heterogeneous databases using semantic technologies, proposed in this paper, has been implemented within the project "Integration platform as a methodology for developing solutions for local government", realized by the Department of Information Systems at Poznań University of Economics. At the core of the project lays the conviction that sharing resources (including data) will allow local government units to reduce the costs of creating, implementing and maintaining systems for the provision of electronic services.