

WOLNE OPROGRAMOWANIE DO PROJEKTOWANIA BAZ DANYCH PRZESTRZENNYCH*

FREE SOFTWARE FOR DESIGNING SPATIAL DATABASES

Piotr Cichociński, Ewa Dębińska

Katedra Geomatyki, Wydział Geodezji Górniczej i Inżynierii Środowiska
Akademia Górniczo-Hutnicza im. Stanisława Staszica w Krakowie

Słowa kluczowe: wolne oprogramowanie, Ujednolicony Język Modelowania, bazy danych przestrzennych

Keywords: free software, Unified Modeling Language, spatial databases

Wstęp

Projektowanie baz danych przeprowadza się w trzech głównych etapach nazwanych projektowaniem pojęciowym, projektowaniem logicznym oraz projektowaniem fizycznym. Kluczowym zadaniem jest zbudowanie schematu pojęciowego, czyli precyzyjne zdefiniowanie obiektów zainteresowania i zidentyfikowanie związków między nimi. Obecnie najpopularniejszym sposobem tworzenia baz danych jest użycie narzędzi komputerowo wspomaganej inżynierii oprogramowania (ang. *Computer Aided Software Engineering – CASE*) (Dębińska, Cichociński, 2006).

Dane modeluje się z wykorzystaniem metodyki obiektowej (Zeiler, 1999), a do zapisu schematu pojęciowego bazy danych wykorzystuje się ujednolicony język modelowania (ang. *Unified Modeling Language – UML*), co pozwala na opisanie rzeczywistości w sposób bardziej naturalny, ułatwia śledzenie powiązań pomiędzy poszczególnymi obiektami oraz umożliwia automatyczne wygenerowanie bazy danych zgodnej z założoną specyfikacją.

Bazy danych przestrzennych są rodzajem baz danych uwzględniających przestrzenny aspekt danych, co oznacza, że baza danych oprócz atrybutów opisowych obiektów zawiera również zapis ich geometrii. Dostępna obecnie oferta oprogramowania pozwalającego na zapisanie, z wykorzystaniem narzędzi CASE, projektu bazy danych przestrzennych w języku UML jest stosunkowo uboga. Niewiele jest narzędzi CASE pozwalających na tworzenie diagramów UML, które umożliwiają projektowanie baz danych przestrzennych.

Jeszcze do niedawna przeprowadzenie kompletnego procesu projektowania baz danych przestrzennych z wykorzystaniem metod obiektowych umożliwiały jedynie aplikacje komer-

* Praca naukowa zrealizowana w ramach badań statutowych prowadzonych w roku 2011 w Katedrze Geomatyki Wydziału Geodezji Górniczej i Inżynierii Środowiska AGH w Krakowie.

cyjne (przykładem może być duet: program Microsoft Visio jako narzędzie do zapisu modeli i system ArcGIS firmy ESRI, w którym baza danych jest fizycznie implementowana). Jednak twórcy wolnego oprogramowania powoli zaczynają dostrzegać konieczność posiadania odpowiednich narzędzi również w tym zakresie geoinformatyki. Jest to szczególnie istotne także z tego powodu, że coraz więcej aktów prawnych, norm i standardów zawiera w swojej treści schematy zapisane z użyciem graficznych notacji obiektowych. Przykładem mogą być specyfikacje INSPIRE czy też szczegółowe zapisy instrukcji technicznej G-5. Zastosowanie odpowiednich narzędzi może stanowczo uprościć tworzenie baz danych i aplikacji zgodnych z tak zaprezentowanymi wymaganiami.

Normalizacja w zakresie informacji geograficznej

Podwaliny dla baz danych przestrzennych dały działania prowadzone na polu normalizacji w dziedzinie informacji geograficznej przez Międzynarodową Organizację Normalizacyjną (ang. *International Organization for Standardization* – ISO) – organizację pozarządową skupiającą krajowe instytucje normalizacyjne. Komitet techniczny ISO/TC 211 „Geographic information/Geomatics” od roku 1994 zmierza do ustanowienia zestruturyzowanego zbioru norm dla informacji dotyczących obiektów lub zjawisk związanych bezpośrednio bądź pośrednio z położeniem względem Ziemi. Normy te mają określać metody, narzędzia i usługi do zarządzania danymi (włączając w to definiowanie i opisywanie), pozyskiwanie, przetwarzanie, analizowanie, udostępnianie, prezentowanie i przesyłanie tych danych w postaci cyfrowej pomiędzy różnymi użytkownikami, systemami i lokalizacjami. Tym samym mają wspierać zrozumienie i wykorzystanie informacji geograficznej, zwiększać jej dostępność, integrację, wspólne wykorzystanie, pozwalając na współdziałanie systemów komputerowych wzbożonych o możliwości geoprzestrzenne i ułatwiając tworzenie infrastruktur informacji przestrzennej na poziomie lokalnym, regionalnym i globalnym.

Oczywiście działania w obszarze informacji geograficznej nie mogą się odbywać w oderwaniu od technologii informatycznych. Skorzystano z nich budując schematy dla informacji geograficznej.

W normie ISO 19101 określono ciąg działań prowadzący od modelu pojęciowego do schematu pojęciowego. Model pojęciowy jest niesformalizowanym, abstrakcyjnym opisem wybranego fragmentu świata rzeczywistego. Po zastosowaniu do zapisu modelu pojęciowego języka schematu pojęciowego, dostarczającego elementów semantycznych i syntaktycznych pozwalających na precyzyjną i jednoznaczną definicję, staje się on schematem pojęciowym. Ponieważ język schematu pojęciowego zapewnia jednolitość metody i formatu opisu informacji, możliwy jest odczyt i modyfikacja schematu pojęciowego zarówno przez użytkownika, jak i przez systemy komputerowe.

Specyficznym dla informacji geograficznej schematem jest schemat przestrzenny, służący do opisu charakterystyk przestrzennych obiektów geograficznych. Został on zdefiniowany w normie ISO 19107. Schemat przestrzenny obejmuje definicje obiektów/elementów geometrycznych i topologicznych w dwu- i trójwymiarowej przestrzeni, które pozwalają na określanie cech przestrzennych obiektów geograficznych. Każdy element geometryczny jest zbiorem punktów wraz z informacją o systemie odniesienia opartym na współrzędnych. Norma wymienia między innymi takie zaawansowane elementy geometryczne jak klotoida czy siatka nieregularnych trójkątów (ang. *Triangular Irregular Network* – TIN).

Do zapisu schematów w dziedzinie informacji geograficznej zaproponowano w specyfikacji technicznej ISO/TS 19103 stosowanie języka UML. Jest to przyjęty w świecie informatycznym formalizm służący do zapisywania modeli systemów informatycznych. Umożliwia on opisywanie takich elementów jak klasy, obiekty, czynności, przypadki użycia, interakcje, etc. W zakresie informacji geograficznej wykorzystywany jest głównie diagram klas.

W przypadku precyzyjnego opisu położenia i kształtu obiektów mamy do czynienia z określaniem położenia opartym na współrzędnych. Temu zagadnieniu poświęcona jest norma ISO 19111. Istotnym elementem opisywanym w tej normie jest kwestia odniesień przestrzennych, czyli określenia układu współrzędnych, w którym wyrażane są współrzędne określające kształt i położenie obiektów.

Jednak znajdujące się w normie ISO 19107 możliwości zapisu informacji geograficznej były zbyt rozbudowane, żeby można je było szybko zaimplementować. Dlatego w normie ISO 19125-1 zaproponowano wyróżnienie spośród wszystkich obiektów geometrycznych (czasami zbyt skomplikowanych) podzbioru tak zwanych prostych obiektów. Ograniczają się one do punktów, linii i obszarów (poligonów) położonych na płaszczyźnie i liniowej interpolacji kształtu pomiędzy wierzchołkami. Dodatkowo wszystkie obiekty odwołują się do jednego systemu odniesienia opartego na współrzędnych.

W drugiej części tej normy (ISO 19125-2) określono sposoby przechowywania, wyszukiwania, zapytań i aktualizacji kolekcji prostych obiektów geograficznych z poziomu języka SQL w relacyjnych bazach danych. W szczególności podano nazwy i definicje typów geometrycznych SQL. Założono, że: kolekcje obiektów będą przechowywane w postaci tabel zawierających kolumny z wartościami geometrycznymi, pojedynczy obiekt będzie wierszem tabeli, atrybuty nieprzestrzenne obiektów będą odwzorowywane na kolumny o typach opartych na standardowych typach danych SQL, a atrybuty przestrzenne obiektów będą odwzorowywane na kolumny, których typy danych bazują na koncepcji dodatkowych typów danych geometrycznych dla SQL. Tabelę, której wiersze reprezentują te obiekty, nazwano tabelą obiektów (ang. *feature table*). Przyjęto również założenie, że tabela ta powinna zawierać przynajmniej jedną kolumnę o wartościach geometrycznych.

Ponadto, w powyżej wymienionej normie zaproponowano dwa sposoby zapisu geometrii oparte na predefiniowanych typach danych SQL i na rozszerzeniach SQL o typy geometryczne. Rozwiązanie pierwsze polega na wprowadzeniu dodatkowej tabeli zawierającej znormalizowany schemat geometrii, to jest stałą liczbę kilku kolumn, w których zapisywane są naprzemiennie współrzędne X i Y kolejnych punktów charakterystycznych obiektów. W przypadku, gdy takich punktów jest więcej niż dostępnych kolumn, zapisywane są one w kolejnych wierszach i cała grupa wierszy oznaczana jest jako zawierająca punkty jednego obiektu. W praktyce jednak zwykle wykorzystywane jest rozwiązanie drugie, polegające na zapisie wszystkich współrzędnych obiektu w postaci binarnej w jednym polu.

Narzędzia do projektowania baz danych przestrzennych

Opracowanie formalnych podstaw dla baz danych przestrzennych spowodowało powstawanie rozszerzeń tradycyjnych relacyjnych baz danych umożliwiających zapis informacji geograficznej. W grupie wolnego oprogramowania można wśród nich wymienić PostGIS jako rozszerzenie bazy PostgreSQL i Spatialite będący rozszerzeniem SQLite.

Jednak wraz z rozwojem baz danych przestrzennych, w rodzinie wolnego oprogramowania nie szedł w parze rozwój narzędzi umożliwiających ich projektowanie.

Tymczasowym rozwiązaniem można nazwać tworzenie części opisowej bazy danych w narzędziach do projektowania „zwykłych” baz danych, a następnie dodawanie do już wygenerowanej bazy danych kolumn geometrycznych. Spełnia to ostatecznie swoje zadanie, czyli pozwala na utworzenie bazy danych przestrzennych – jednak model będący jej podstawą jest niekompletny.

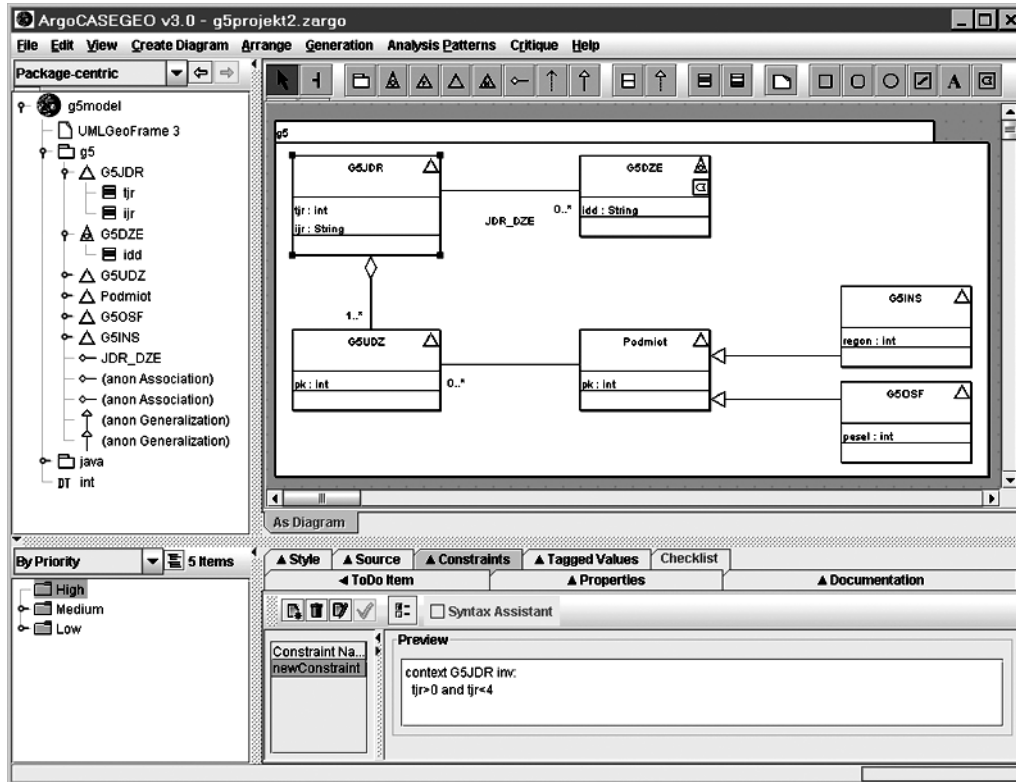
Dopiero w ostatnich latach pojawiły się dwa niekomercyjne narzędzia umożliwiające projektowanie baz danych przestrzennych. Są to programy ArgoCASEGeo i MOSKittGeo. Według zapewnień ich autorów, każdy z tych programów powinien umożliwić zrealizowanie ciągu działań prowadzących od utworzenia schematu pojęciowego do wygenerowania pustej bazy danych o strukturze zgodnej z projektem, gotowej do wypełnienia danymi. Próbie modelowania poddano wybrane obiekty ewidencji gruntów i budynków zdefiniowane w Instrukcji technicznej G-5. W aneksie nr 2 do tej instrukcji znajduje się zapisany w notacji OMT (Rumbaugh’a) *schemat podstawowych relacji pomiędzy obiektami w ewidencji*, który trzeba było przelożyć na język UML.

ArgoCASEGeo

Narzędzie to zostało opracowane na Wydziale Informatyki Uniwersytetu Vicoso (Brazylia) na bazie oprogramowania ArgoUML (Lisboa Filho i in., 2004), napisanej w języku Java aplikacji do tworzenia diagramów UML. ArgoCASEGeo umożliwia projektowanie baz danych przestrzennych w oparciu o model GeoFrame (Bedard, Larrivee, 2008). Istotą GeoFrame jest rozszerzenie standardowych symboli reprezentujących klasy obiektów w języku UML o piktogramy (rys. 1), pozwalające rozróżnić obiekty przestrzenne i nieprzestrzenne, a także szybko zidentyfikować rodzaj geometrii (punkt, linia, poligon).

Domyślnie wybranym rodzajem tworzonego diagramu jest GeoFrame, ale można również wybrać inne typowe diagramy UML. Od rodzaju diagramu zależna jest lista dostępnych na odpowiednim pasku narzędzi. Prace przy tworzeniu diagramu rozpoczęto od umieszczenia pakietu, wewnątrz którego wstawiono poszczególne klasy obiektów. Klasy obiektów mogą posiadać atrybuty i operacje. Dla każdego atrybutu można wybrać z listy wymagany typ danych. Jedyny problem, jaki napotkano na tym etapie działań, pojawił się podczas definicji atrybutu *Rodzaj jednostki rejestrowej* (TJR) klasy obiektów *Jednostka rejestrowa* (G5JDR). Atrybut ten może przyjmować trzy wartości: 1 – gruntowa, 2 – budynkowa, 3 – lokalowa. Ponieważ ArgoCASEGeo nie udostępnia typów wyliczeniowych, zdefiniowano odpowiednie ograniczenie wyrażone w języku OCL (ang. *Object Constraint Language*) (rys. 1).

Po zbudowaniu modelu można było przejść do etapu generowania na jego podstawie bazy danych. Program teoretycznie daje trzy możliwości: pliki Shape, model Terralib oraz skrypt SQL zgodny z bazą danych Oracle Spatial. W praktyce okazało się, że zadziałał tylko ostatni wariant. W związku z występowaniem w modelu związku generalizacji trzeba było wybrać jeden z trzech sposobów zaimplementowania go w wynikowych tabelach bazy danych: połączenia wszystkich klas związanych generalizacją w jedną tabelę, utworzenia tabel z klas podrzędnych i usunięcia klasy nadrzędnej oraz utworzenia tabel z wszystkich klas. Po sprawdzeniu wszystkich trzech wybrano sposób drugi. Pewną niedogodnością w tej sytuacji jest wymóg istnienia w każdej klasie obiektów atrybutu, pełniącego rolę klucza głównego (ang.

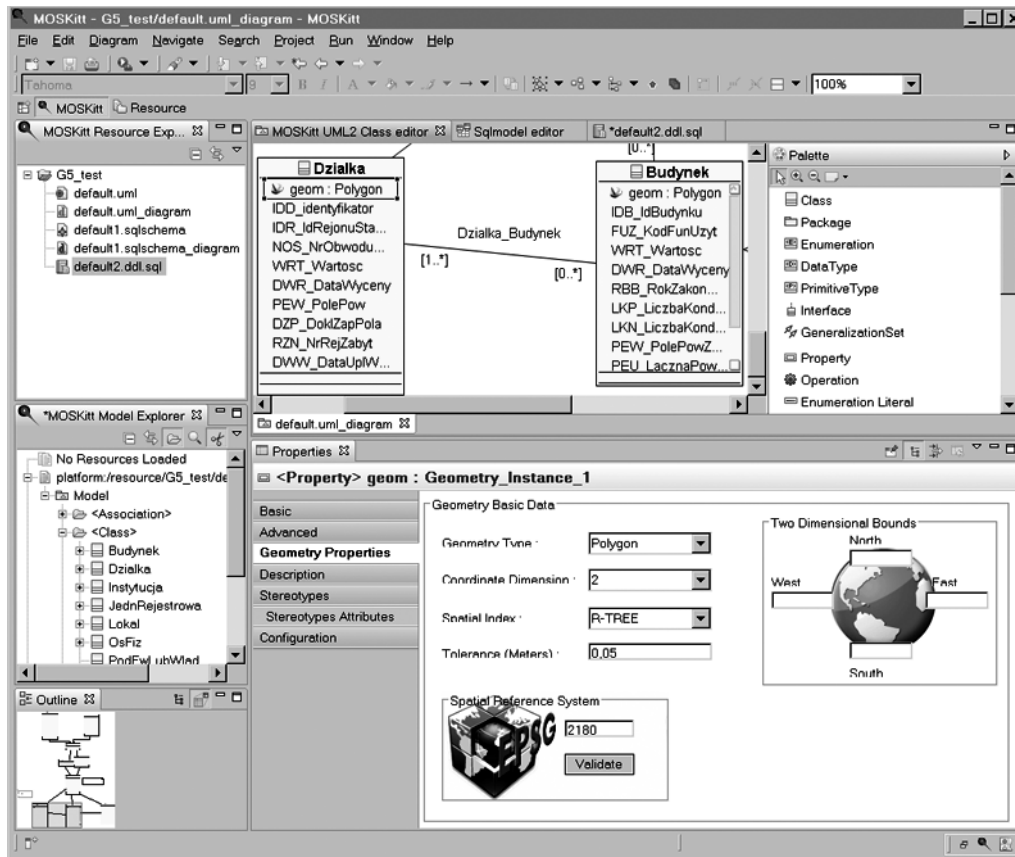


Rys. 1. Okno programu ArgoCASEGeo

Primary Key), który zupełnie niepotrzebnie przenosi się z klasy nadrzędnej do wynikowych tabel odpowiadających klasom podrzędnym. Uruchomienie otrzymanego skryptu SQL w środowisku bazy danych Oracle pozwoliło uzyskać pustą strukturę bazy danych gotową do wypełnienia danymi. Jedyne, aczkolwiek istotne zastrzeżenie, jakie można mieć do uzyskanego rezultatu, wynika z faktu nieuwzględnienia ograniczenia narzuconego w modelu na wartości atrybutu TJR.

MOSKitt Geo

Narzędzie to zostało opracowane w hiszpańskiej firmie Prodevelop. W praktyce jest rozszerzeniem pozwalającym na obsługę danych przestrzennych w programie MOSKitt służącym do projektowania baz danych. Stało się to możliwe dzięki wprowadzeniu typów danych geometrycznych: *Unknown Geometry*, *Point*, *Line*, *Polygon*, *MultiPoint*, *MultiLine*, *MultiPolygon*, *Geometry Collection*. Dla każdego definiowanego atrybutu o charakterze przestrzennym określa się liczbę współrzędnych (2, 3 lub 4), układ współrzędnych (poprzez identyfikator w bazie danych EPSG), zakres współrzędnych w postaci prostokąta ograniczającego oraz metodę indeksowania przestrzennego (rys. 2).



Rys. 2. Okno programu MOSKitt Geo

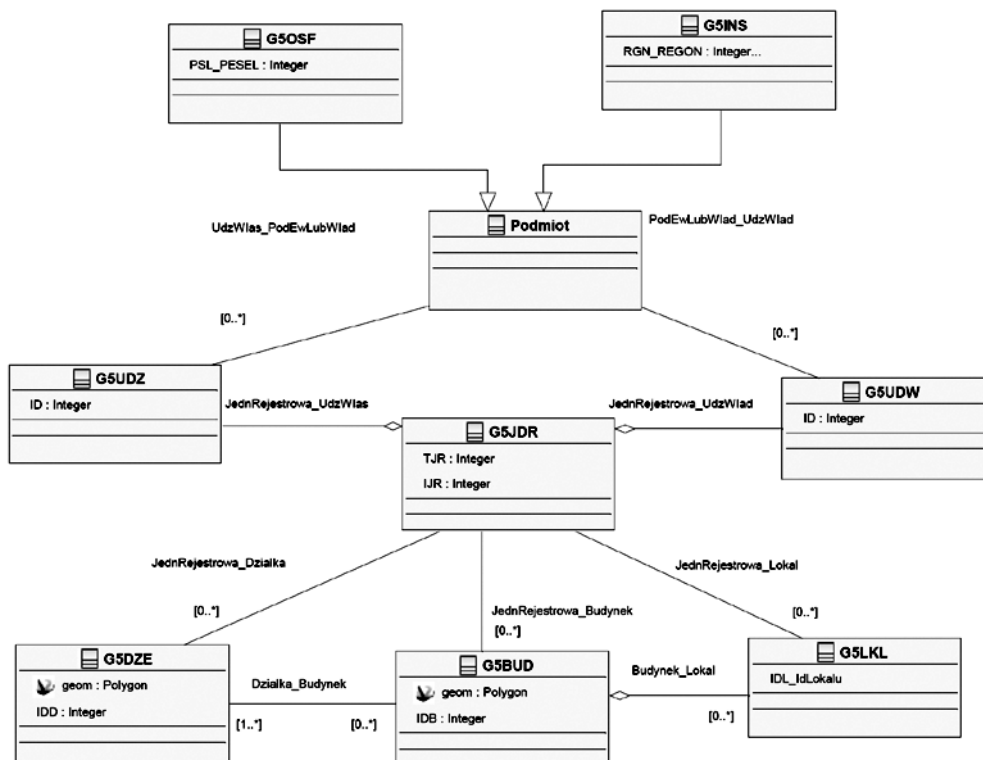
Podczas projektowania bazy danych przestrzennych w MOSKitt użytkownik ma do wyboru trzy sposoby indeksowania danych: B-Tree, R-Tree, GIST. Najstarszy z wymienionych jest indeks B-Tree (Bayer, McCreight, 1972). Organizacja danych w postaci B-Tree pozwala na wyszukiwanie oparte na dokładnym dopasowaniu, poszukiwaniu wzorca oraz sprawdzaniu zakresu, co sprowadza się do operatorów: < mniejszy, <= mniejszy równy, = równy, >= większy równy, > większy. Indeks jest dynamiczny, jego zawartość zmienia się wraz ze zmianami danych. Nie znajduje zastosowania do danych przestrzennych.

W przypadku indeksu R-Tree (Guttman, 1984) wykorzystywane są minimalne prostokąty ograniczające (ang. *Minimum Bounding Rectangle* – MBR). Prostokąty tworzone według ustalonego algorytmu oraz wszystkie obiekty z analizowanych danych przestrzennych dostają unikalne oznaczenia, które następnie organizuje się w drzewo R-Tree.

O ile indeks R-Tree przeznaczony jest dla danych przestrzennych, B-Tree dla nieprzestrzennych, to indeks GIST (Hellerstein i in., 1995) jest uniwersalny i można go stosować w obu przypadkach. Był tworzony z myślą o danych złożonych, takich jak dane przestrzenne, multimedia, narzędzia CAD, biblioteki dokumentów, system identyfikacji linii papilarnych. Pozwala na indeksowanie danych różnych typów, przez stosowanie różnych struktur drzewiastych.

Tworzenie schematu UML. Po zainstalowaniu w programie MOSKitt rozszerzenia dla danych przestrzennych (trzeba zwrócić uwagę, że poprawnie współpracuje ono w chwili obecnej wyłącznie z jedną ze starszych wersji programu MOSKitt – 1.3.0), można było przystąpić do tworzenia schematu UML. Ważne jest, aby już na etapie zakładania nowego projektu zaznaczyć opcję uwzględniającą przestrzenne typy danych (ang. *Add data types from Spatial Group*). W oknie edytora graficznego (rys. 2) umieszcza się kolejne klasy obiektów, które łączy się liniami reprezentującymi związki. Dla klas obiektów można określić ich atrybuty. Mogą one być jednego ze standardowych typów (liczby całkowite, rzeczywiste, teksty, etc.). Mogą to być również typy geometryczne. Pierwszy problem, który napotkano związany był z typami wyliczeniowymi. Program umożliwia wstawienie do schematu klasy wyliczeniowej (ang. *Enumeration*). Może ona następnie być wykorzystana jako typ dla atrybutów lub też można powiązać ją relacją z klasą obiektów, co jest równoważne z utworzeniem atrybutu takiego typu. Niestety ani jeden ani drugi zapis nie przenosi się do kolejnego etapu, czyli powstałego po transformacji modelu związków encji, a co za tym idzie nie jest również uwzględniony w finalnym kodzie zapisanym w języku SQL.

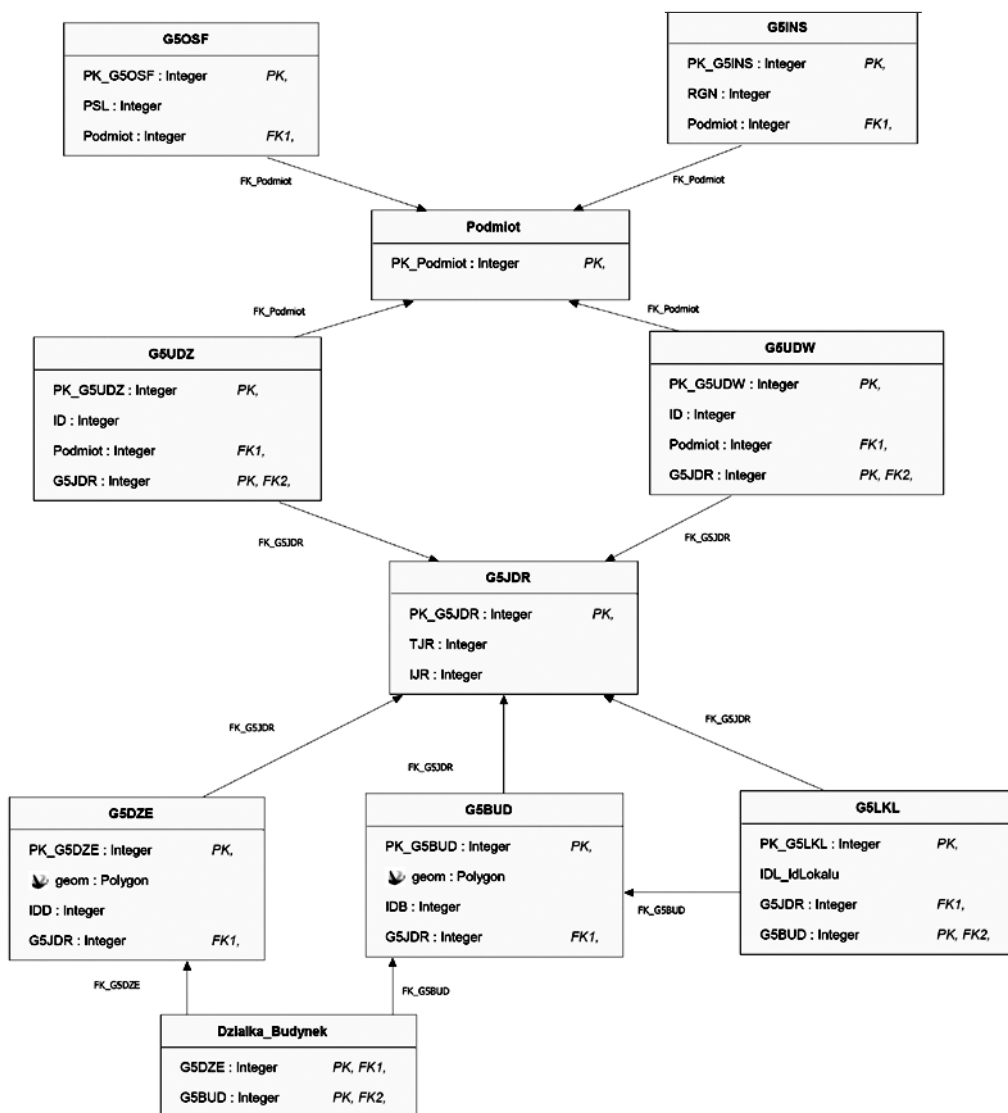
Drugi z dostrzeżonych problemów ma związek z klasami abstrakcyjnymi. Jako klasę abstrakcyjną postanowiono zdefiniować klasę *Podmiot*. W katalogu obiektów Instrukcji technicznej G-5 nie ma takiej klasy (jest tylko przedstawiona na schemacie), więc zastosowano klasę abstrakcyjną, aby w efekcie nie powstała odpowiadająca jej tabela. Rzeczywiście po-



Rys. 3. Utworzony w programie MOSKitt Geo diagram UML dla wybranych obiektów ewidencji gruntów i budynków

wstały po transformacji model związków encji nie zawiera takiej tabeli, ale niestety giną również wszelkie powiązania tej klasy abstrakcyjnej (w szczególności związki generalizacji z klasami podrzędnymi). Dlatego ostatecznie zamiast klasy abstrakcyjnej zastosowano klasę rzeczywistą (rys. 3).

Transformacja modelu do schematu bazy danych jest kolejnym etapem po utworzeniu i zapisaniu modelu (rys. 4). Relacje zostają odwzorowane jako pary *klucz obcy-klucz główny* w powiązanych tabelach. W ten sam sposób zapisywane są wszystkie relacje: generalizacja,



Rys. 4. Utworzony w programie MOSKitt Geo diagram związków encji dla wybranych obiektów ewidencji gruntów i budynków

kompozycja, etc. W tym miejscu można było zauważyć problem zarówno z typem wyliczeniowym, jak i z klasą abstrakcyjną.

Wygenerowanie kodu w języku SQL jest ostatnim etapem zgodnym z konkretnym, wybranym systemem zarządzania bazą danych. W tym przypadku pozostając dalej w grupie wolnego oprogramowania wybrano jako docelową bazę PostGIS. Z tym etapem działań może być związany napotkany wcześniej problem z typami wyliczeniowymi. Otóż plik SQL jest generowany dla systemu PostgreSQL (w oparciu o który działa nakładka PostGIS) w wersji 8.2. Natomiast możliwość definiowania typów wyliczeniowych pojawiła się dopiero w kolejnej wersji 8.3.

Pomijając opisane powyżej trudności, uruchomienie otrzymanego skryptu SQL w środowisku bazy danych PostgreSQL/PostGIS pozwoliło uzyskać pustą strukturę bazy danych gotową do wypełnienia danymi.

Podsumowanie

Dostępne od niedawna, w rodzinie wolnego oprogramowania, narzędzia umożliwiające modelowanie baz danych przestrzennych mogą stanowić alternatywę dla znanych komercyjnych rozwiązań. Ich największą zaletą jest uwzględnienie elementów opisujących geometryczne cechy obiektów w jednym spójnym modelu. Przetestowane narzędzia dysponują przyjaznym dla użytkownika interfejsem, a dodatkowo ArgoCASEGeo oferuje mechanizm sprawdzania na bieżąco poprawności tworzonego modelu i podpowiedzi w zakresie rozwiązywania napotkanych błędów.

Niestety, jak na razie narzędzia te nie są wolne od błędów, ograniczeń i niedoskonałości. Do niedogodności związanych z korzystaniem z tych programów można zaliczyć ubogą lub nieaktualną, nienadążającą za rozwojem opisywanego oprogramowania dokumentację, a dodatkowo, jak można podejrzewać z powodu małej jeszcze popularności tych narzędzi, nie funkcjonują typowe dla wolnego oprogramowania fora dyskusyjne. W obu programach trzeba pamiętać, że usunięcie graficznego elementu diagramu klawiszem *Delete* nie powoduje usunięcia elementu z modelu, a tylko z rysunku. Dany element można usunąć z poziomu eksploratora modelu lub poprzez menu kontekstowe *Usuń z modelu*. Obydwa programy mają pewne ograniczenia funkcjonalne, lecz można mieć nadzieję, że w kolejnych wersjach zostaną one usunięte.

Literatura

- Bayer R., McCreight E., 1972: Organization and Maintenance of Large Ordered Indexes. *Acta Informatica*, Vol. 1, Fasc. 3.
- Bedard Y., Larrivee S., 2008: Modeling with Pictogrammic Languages. [In:] Encyclopedia of GIS. Springer US.
- Dębińska E., Cichociński P., 2006: Zastosowanie narzędzi CASE do projektowania baz danych systemów informacji geograficznej. *Geodezja*, tom 12, zeszyt 2/1. Uczelniane Wydawnictwa Naukowo-Dydaktyczne AGH, Kraków.
- Gaździcki J., 2001: Leksykon geomatyczny. PTIP, Warszawa.
- Guttman A., 1984: R-Trees: A Dynamic Index Structure for Spatial Searching. [W:] Yormark B. (ed.), SIGMOD'84, Proceedings of Annual Meeting, Boston, Massachusetts, June 18-21, 1984. ACM Press.
- Hellerstein J.M., Naughton J.F., Pfeffer A., 1995: Generalized Search Trees for Database Systems. Proceedings of 21st International Conference on Very Large Data Bases, Zurich, September 1995.
- Instrukcja, 2003: Instrukcja Techniczna G-5 Ewidencja gruntów i budynków. Główny Urząd Geodezji i Kartografii, Warszawa.

- ISO/TS 19103:2005 Geographic information – Conceptual schema language.
- Lisboa Filho J., Rodrigues Junior M. F., Daltio J., Sodre V. F., 2004: ArgoCASEGEO – an open source CASE tool for Geographic Information Systems modelling using the UML-GeoFrame model. [In:] Proceedings of 7th International Conference on Information Systems Implementation and Modelling (ISIM '04), Roznov pod Radhosem, Czech Republic.
- PN-EN ISO 19101:2005 Informacja geograficzna – Model tworzenia norm.
- PN-EN ISO 19107:2005 Informacja geograficzna – Schemat przestrzenny.
- PN-EN ISO 19111:2007 Informacja geograficzna – Odniesienia przestrzenne za pomocą współrzędnych.
- PN-EN ISO 19125-1:2006 Informacja geograficzna – Środki dostępu do obiektów prostych – Część 1: Wspólna struktura.
- PN-EN ISO 19125-2:2006 Informacja geograficzna – Środki dostępu do obiektów prostych – Część 2: Opcja SQL.
- Zeiler M., 1999: Modeling our World. The ESRI Guide to Geodatabase Design. ESRI Press, Redlands.

Abstract

Designing databases is carried out in three main stages, known as the conceptual design, logical design and physical design. Currently, the most popular way to create databases is to use computer-aided engineering tools (Computer Aided Software Engineering – CASE).

The key task in creating the conceptual scheme is to precisely define the objects of interest and to identify the relationships between them. For recording the conceptual scheme of a database, a Unified Modelling Language is used (UML), which facilitates, among others, tracking the links between individual objects, and allows automatic generation of a database in accordance with the required specification. Spatial databases are a kind of subtype of databases with added spatial factor, which means that, in addition to the descriptive attributes of the objects, the database concerned also contains a record of their geometry. Formal grounds for such method of spatial data recording are specified in the ISO 19100 series standards and Open Geospatial Consortium (OGC) specifications.

Until recently, a complete design process of spatial databases using object methods was possible only with commercially available applications (for example: Microsoft Visio as a tool for writing models and ESRI ArcGIS system, in which the database is physically implemented). However, the free software authors begin to recognise the need of having appropriate tools, also in this area of geoinformatics. This is especially important also because of the fact that more and more laws, norms and standards include schemes written in UML. Examples include the INSPIRE specifications or detailed records of the G-5 technical instruction. Using the right tools can definitely simplify the creation of databases and applications meeting the requirements given.

At present, there are two programs ensuring the essential functionality: ArgoCASEGeo and MOSKitt Geo. The paper details, using those tools and sample schemes, a series of actions leading to create the concept scheme needed to generate an empty database with structure consistent with design, ready to be filled with data. Application of the software utilised is evaluated in terms of correct and user-friendly operation.

dr inż. Piotr Cichociński
Piotr.Cichocinski@agh.edu.pl

dr inż. Ewa Dębińska
Ewa.Debinska@agh.edu.pl