

Warszawa, 15 stycznia 2004

mgr inż. Mariusz Trzaska
Katedra Systemów Informacyjnych,
PJWSTK

Autoreferat planowanej rozprawy doktorskiej

Usability of Visual Information Retrieval Metaphores for Object-Oriented Databases”

(Użyteczność metafor wizyjnych dla wyszukiwania informacji w obiektowych bazach danych)

Obiektowe bazy danych wniosły nową jakość do sposobu przechowywania i przetwarzania danych. Zastosowany nowy model organizacji danych jest bliższy ludzkiemu sposobowi postrzegania świata, niż dotychczas stosowany model relacyjny. Dzięki wykorzystaniu takich koncepcji jak: klasa, obiekt, generalizacja/specjalizacja, asocjacja i metoda, cały proces projektowania struktury bazy danych stał się bardziej intuicyjny. W efekcie, w porównaniu do relacyjnej bazy danych, znacząco poprawiła się czytelność tworzonego schematu.

Ubočnym efektem pojawienia się obiektowych baz danych, jest mnogość niekompatybilnych standardów, rozwiązań implementacyjnych i koncepcji naukowych. Wytlumaczeniem takiego stanu rzeczy jest fakt, że obiektowe bazy danych to młoda dziedzina. Wydaje się, że z czasem sytuacja się unormuje i powstaną przejrzyste standardy dotyczące tej gałęzi informatyki.

Bazy danych są masowo wykorzystywane z wielu powodów. Zapewniają obsługę transakcji, bezpieczeństwo przechowywanych danych, dbają o ich spójność i integralność. Jednakże kluczową cechą systemu zarządzania dowolną bazą danych jest jej zdolność do udostępniania przechowywanych danych. Jest wiele sposobów realizacji tej cechy, od prostego filtrowania danych, poprzez tekstowe języki zapytań, na zaawansowanych graficznych interfejsach użytkownika kończąc.

Tekstowe języki zapytań obsługującego konkretną bazę danych są własnością o ogromnym znaczeniu. Wydaje się, że jedną z głównych przyczyn sukcesu relacyjnych baz danych był tekstowy język zapytań zwany *Structural Query Language* (SQL). Jest on rozwijany do dziś. Niestety, kolejne wersje wprowadzają coraz więcej skomplikowanych konstrukcji. Wystarczy przypomnieć, że pierwsza specyfikacja SQL zajmowała ok. 100 stron, zaś najnowsza liczy ich ponad 2000. Również najbardziej znany język zapytań dla obiektowych baz danych *Object Query Language* (OQL) wzbudza wiele kontrowersji. Zarzuca mu się m. in. brak precyzyjnej semantyki (spowodowanej niezbyt dokładnym modelem danych) co mocno utrudnia, czy wręcz uniemożliwia, realizację jego sprawnego interpretera i automatycznego optymalizatora zapytań.

Graficzne interfejsy użytkownika, ułatwiające wyszukiwanie informacji w bazie danych, stanowią szybko rozwijającą się gałąź informatyki. Najważniejszą przyczyną tego rozwoju jest to, że skomplikowana składnia i semantyka tekstowych języków zapytań jest trudna do opanowania i efektywnego używania przez przeciętnego użytkownika. Do tej pory powstało bardzo dużo różnych systemów, które można zakwalifikować do grupy graficznych

interfejsów użytkownika służących do odzyskiwania informacji z obiektowych baz danych. Mimo dużej ich różnorodności, można wyróżnić dwie zasadnicze grupy:

- graficzne języki zapytań (*graphical query languages*).
- przeglądarki (*browsers, graphical browsing interfaces*),

Graficzne języki zapytań, pod względem koncepcyjnym są bardzo podobne do swoich tekstowych odpowiedników. Różnią się tym, że zamiast pisać tekst kwerendy, „rysujemy” ją na ekranie. Istnieje wiele różnych systemów, działających w oparciu o ten paradygmat, poczynając od takich, które umożliwiają stworzenie tylko prostych zapytań, a kończąc na zaawansowanych aplikacjach, korzystających z trójwymiarowej wizualizacji i oferujących takie same możliwości jak ich tekstowe odpowiedniki. Często graficzne zapytanie jest tłumaczone na swoją tekstową postać i dopiero wtedy ewaluowane. Zależnie od konkretnego systemu, otrzymane dane mogą być wyświetlone w postaci tekstowej (np. tabele) lub graficznej. Praca z takimi systemami wymaga dokładnego rozpoznania przez użytkownika dokładnej składni graficznej oraz semantyki przypisanej do tej składni. Zazwyczaj, powszechny użytkownik, do którego te interfejsy są adresowane, nie spełnia tego wymagania.

Przeglądarki, należące do drugiej grupy omawianych systemów, można zdefiniować jako aplikacje umożliwiające ręczne „przemieszczanie się” pomiędzy obiektami, dokładnie tak samo, jak użytkownicy nawigują w sieci Web. Przechodząc (przy pomocy powiązań) od jednego obiektu do innych z nim powiązanych, użytkownik jest w stanie odnaleźć interesujące go informacje. Ten sposób pracy nie wymaga dokładnego określenia kryteriów wyszukiwania. W związku z tym jest szczególnie użyteczny gdy takie kryteria (w rozumieniu warunków logicznych) nie są łatwo definiowalne lub nie są znane przed podjęciem akcji wyszukiwania.

Innym kryterium podziału omawianych graficznych narzędzi jest grupa użytkowników do których skierowany jest konkretny system. Możemy wyróżnić dwie takie grupy osób:

- profesjonalni informatycy,
- użytkownicy bez wykształcenia informatycznego (bez znajomości pojęć baz danych).

Te grupy różnią się, w zakresie wymagań stawianych omawianym systemom. Różnice te dotyczą nie tylko sposobu pracy, skomplikowania interfejsu, ale, co przeważnie jest ze sobą związane, również możliwości udostępnianych przez dane narzędzie.

W rozprawie zostanie zbadana przydatność istniejących systemów korzystających z graficznych metafor dostępu do obiektowych baz danych. Badania te również uwzględnią niejednorodne oczekiwania poszczególnych rodzajów użytkowników. Zaproponowane będzie nowe podejście, rozszerzające metaforę przeglądania (*browsing*) poprzez jej powiązanie z metaforami graficznych języków zapytań. Proponowane rozwiązanie rozszerza i łączy w spójną całość kilka metafor, które wcześniej funkcjonowały niezależnie od siebie:

- Nawigacja intencyjna (*intentional navigation*), bazuje na przemieszczaniu się w grafie zbudowanym z grup obiektów (jako wierzchołków) i powiązań pomiędzy nimi (jako krawędzi). Każdy wierzchołek zawiera obiekty należące do pewnej klasy. Obiekty mogą zostać oznaczone (*marked*) na skutek kilku rodzajów działań, m. in. filtrowania, nawigacji wzdłuż wybranej asocjacji, pobrania z koszyka.
- Nawigacja ekstensyjna (*extensional navigation*) bazuje na podobnych założeniach, ale dotyczy działań na poszczególnych obiektach, zamiast na ich grupach.

- Koszyki jako trwałe, personalizowany skład wyszukanych obiektów. Umożliwiają zapamiętanie referencji do dowolnego obiektu (lub ich grupy), opisanie go i późniejsze użycie.

Wykonany prototyp udowadnia użyteczność tych pomysłów. Potwierdzają je badania przeprowadzone w rzeczywistym systemie (aplikacje zbudowane na prototypie ICONS). Aktualnie zostały podjęte prace nad rozszerzeniem funkcjonalności prototypu. Rozważane jest m. in. zrealizowanie wizualnych złączeń, projekcji obiektów, dodania informacji o usługach „handlowych” dostarczanych przez obiekty i zaimplementowaniu metafor w środowisku 3D. Innym kierunkiem badań jest rozszerzenie prototypu na sieć danych i usług oferowanych w ramach nowego kierunku w Internecie, określanego jako Web Services i Semantic Web.