

Język XML w aplikacjach z bazami danych — możliwości zastosowania, pierwsze doświadczenia

Tomasz Traczyk
ttraczyk@ia.pw.edu.pl

Włodzimierz Macewicz
wmacewicz@ia.pw.edu.pl



Politechnika Warszawska
Wydział Elektroniki i Technik Informacyjnych

Język XML w aplikacjach z bazami danych

Wprowadzenie — cele języka XML

- Składniki XML
- XML a HTML
- Możliwości zastosowania XML w systemach informatycznych
- Pierwsze doświadczenia z XML
- Podsumowanie



Język XML w aplikacjach z bazami danych

2

Cele XML

Cele XML

- Możliwość tworzenia specjalizowanych języków do zapisu dokumentów
- Łatwe i precyzyjne definiowanie składni tych języków
- Wyświetlanie dokumentów przez uniwersalne i specjalizowane przeglądarki
- Działanie w sieci WWW
- Możliwość użycia różnego typu powiązań między dokumentami

Spodziewane zyski

- Łatwa wymiana danych i dokumentów między różnymi systemami informatycznymi
- różnymi społecznościami użytkowników
- Tworzenie i prezentacja dokumentów reprezentujących różne dziedziny za pomocą ujednoczonego oprogramowania
- Dystrybucja specjalistycznych dokumentów bezpośrednio w WWW
- Uproszczone i ujednoczone przetwarzanie i wyszukiwanie danych
- Łatwy podział dokumentów na części i współdzielenie części (także odległych)
- Wyparcie formatów „firmowych”, kłopotliwych w obróbce i konwersji



Język XML w aplikacjach z bazami danych

3

Co to jest XML

Co to jest XML

- *Extensible Markup Language*
- Metajęzyk
 - umożliwia definiowanie języków
 - służy do zapisu różnorodnych dokumentów
 - przeznaczony m.in. do użycia w WWW
- Znakowanie znaczeniowe
- Do formatowania wizualnego — *style-sheets*

Geneza XML

- Podzbiór SGML (*Standard Generalised Markup Language*)
- Uproszczony przez usunięcie trudnych i niekoniecznych konstrukcji

Co zapewni a wybrany podzbiór?

- Możliwość definiowania
 - struktury dokumentu
 - znaczników (*tags*)
- Łatwość interpretacji dokumentu
 - przez przeglądarki
 - nawet bez definicji struktury dokumentu
- Możliwość tworzenia połączeń (*links*)
 - między dokumentami
 - przez sieć



Język XML w aplikacjach z bazami danych

4

Wprowadzenie — cele języka XML

Składniki XML

- XML a HTML
- Możliwości zastosowania XML w systemach informatycznych
- Pierwsze doświadczenia z XML
- Podsumowanie



Język XML w aplikacjach z bazami danych

5

Składniki XML

- Dokument
 - prolog
 - element główny i elementy podrzędne
 - łączniki (*Xlinks*)
- DTD (*Document Type Definition*)
 - definicje elementów
 - określenie możliwego następowania elementów
 - definicje atrybutów elementów
 - definicje jednostek (*entities*)
- *Style-sheets*
 - określenie atrybutów wyświetlania dla elementów
 - możliwość różnej prezentacji tego samego dokumentu w zależności od środowiska
 - w przygotowaniu język XSL (*Extensible Style Language*)
 - składnia oparta na XML
 - idea działania wzięta z DSSSL



Język XML w aplikacjach z bazami danych

6

Dokument w XML

```
<?xml version="1.0" standalone="no" encoding="ISO-8859-2"?>
<!DOCTYPE notka SYSTEM "http://www.nul.nul/notka.dtd">

<!-- To jest przykład dokumentu w XML -->
<notka>
  <do> Czytelnicy </do>
  <od> &autor; </od>
  <temat typ="pilne"> Dokument typu Konspekt </temat>
  <par> Dokument typu Konspekt po przetworzeniu do HTML można obejrzeć
  w typowej przeglądarce, co pokazuje
  <figref refid="rys1"> rys. 1 </figref>.
</par>
  <figura id="rys1">
    <grafika adres="http://www.nul.nul/rys1.gif" typ="GIF" />
  <podpis> Konspekt w typowej przeglądarce </podpis>
</figura>
  <par> Natomiast wydrukowanie konspektu za pomocą narzędzia do
  tworzenia raportów przysparza sporo problemów.
</par>
  &dalej;
</notka>
```

DTD

```
<!DOCTYPE notka [
  <!ELEMENT notka (od, do, temat?, (par|figura)+) >
  <!ELEMENT par (#PCDATA|figref)+ >
  <!ELEMENT do (#PCDATA) >
  <!ELEMENT od (#PCDATA) >
  <!ELEMENT temat (#PCDATA) >
  <!ELEMENT figura (grafika,podpis?) >
  <!ELEMENT grafika EMPTY >
  <!ELEMENT podpis (#PCDATA) >
  <!ELEMENT figref (#PCDATA) >
  <!ATTLIST temat typ (pilne|normalne) "normalne" >
  <!ATTLIST figura id ID #REQUIRED >
  <!ATTLIST figref refid IDREF #IMPLIED >
  <!ATTLIST grafika adres %URL; #REQUIRED
  typ NOTATION (GIF|PNG|JPEG) "GIF" >
  <!NOTATION GIF SYSTEM "c:\windows\system\gif.dll" >

  <!ENTITY autor "Tomasz Traczyk" >
  <!ENTITY dalej SYSTEM "http://www.nul.nul/reszta.xml" >
]>
```

Poprawność dokumentu w XML

Poprawność w sensie 'well-formed'

- Warunki
 - niepuśta zawartość
 - wszystkie konstrukcje poprawne w sensie 'well-formed' (wg specyfikacji języka)
 - wszystkie konstrukcje rozpoczęte muszą być jawnie zakończone
 - każda konstrukcja zawarta w innej musi być w niej zawarta w całości
 - tylko jedna konstrukcja (element główny) nie zawarta w żadnej innej
 - brak odwołań do niestandardowych definicji nie zawartych w tym dokumencie
- Nie musi mieć DTD (dokument 'standalone')
- Mimo braku DTD może być wyświetlany przez przeglądarki
- Przeznaczenie: tylko wyświetlanie

Poprawność w sensie 'valid'

- Warunki
 - jest 'well-formed'
 - powiązany z istniejącym DTD
 - pełna zgodność zawartości z DTD
 - wszystkie konstrukcje poprawne w sensie 'valid' (wg specyfikacji języka)
- Przeznaczenie: przetwarzanie
- Do przetwarzania z reguły niezbędny jest dostęp do DTD

Wprowadzenie — cele języka XML

Składniki XML

XML a HTML

Możliwości zastosowania XML w systemach informatycznych

Pierwsze doświadczenia z XML

Podsumowanie

XML a HTML

XML

- Przeznaczenie
 - metajęzyk
 - służy do definiowania języków
- Realizacja
 - podzbiór SGML
 - brak predefiniowanych znaczników
- Składnia
 - element niepuśły musi być jawnie zakończony
 - element puśły musi być oznaczony />
 - wrażliwy na wielkość liter w nazwach elementów i atrybutów
 - wartości atrybutów muszą być ujęte w cudzysłów

HTML

- Przeznaczenie
 - gotowy język
 - służy do zapisu stron WWW
- Realizacja
 - konkretyzacja SGML
 - zestaw predefiniowanych znaczników
- Składnia
 - niektóre elementy niepuśłe nie muszą być zakończone jawnie
 - elementy puśłe nie muszą być wyróżnione
 - niewrażliwy na wielkość liter
 - wartości atrybutów mogą, ale nie muszą być ujęte w cudzysłów

Wprowadzenie — cele języka XML

Składniki XML

XML a HTML

Możliwości zastosowania XML w systemach informatycznych

Pierwsze doświadczenia z XML

Podsumowanie

Do czego może służyć XML

Dokumenty tekstowe

- Zalety
 - struktura dokumentu dowolnie złożona, ale ściśle zdefiniowana — przetwarzanie
 - znakovanie znaczeniowe, a nie typograficzne (wizualne) — przenośność
 - plik tekstowy — łatwość modyfikacji i przetwarzania
 - uniwersalny i ustandaryzowany język — możliwość użycia standardowych narzędzi
- Wady
 - większa długość plików
 - większy czas w czytływania i zapisu dokumentów

Specjalizowane struktury danych

- Przykładowe dziedziny
 - wzory matematyczne
 - wzory chemiczne
 - specyficzne opisy konstrukcji
 - GIS, CAD
 - wymiana danych handlowych (EDI)
- Istniejące języki „branżowe”
 - MathML
 - CML (*Chemical Markup Language*)
 - XDFL (*Extensible Forms Description Language*) — EDI



Do czego może służyć XML (2)

Wyszukiwanie danych w Internecie

- Standardowy opis stron dla przeglądarek
- Selektywne wyszukiwanie
- Propozycje
 - RDF (*Resource Description Framework*)
 - MCF (*Meta Content Framework*)
 - *Web Collections*

Wykorzystanie w programach powszechnego użytku

- Postać
 - czytelna
 - podatna na modyfikacje
- Możliwość obróbki różnymi narzędziami
- Brak problemów z konwersją
- Możliwość bezpośredniego (bez konwersji na HTML) umieszczenia w WWW
- Ujednoczenie formatów dla różnych składników pakietów biurowych
- Brak uzależnienia od jednego producenta



XML w systemach z bazami danych

Zewnętrzna reprezentacja danych

- Reprezentacja poza bazą danych
- Łatwy zapis danych
 - relacyjnych
 - obiektowych
- Kłopot tylko z danymi binarnymi
- Zastosowanie
 - przesyłanie danych
 - prezentacja w WWW

Wewnętrzna reprezentacja danych

- Reprezentacja w bazie danych
- Przydatna dla danych o strukturze
 - zmiennej
 - złożonej
- Zapis w formie CLOB zamiast BLOB
- Możliwości
 - przetwarzanie w bazie danych
 - wyszukiwanie kontekstowe
- Przykładowe zastosowania
 - złożone dokumenty tekstowe
 - arkusze kalkulacyjne



Problemy z XML w aplikacjach z bazami danych

Zapis dokumentu XML w bazie danych

- Problemy
 - złożoność struktury dokumentów
 - podatność struktury na modyfikacje
- Reprezentacja
 - relacyjna
 - skomplikowana
 - nienaturalna
 - trudna do przetwarzania
 - nieelastyczna
 - relacyjno-obiektowa
 - można zapisać złożone struktury
 - brak elastyczności zapisu
 - „czysto” obiektowa
 - naturalna reprezentacja złożoności
 - elastyczność — użycie dziedziczenia
 - mała popularność OODBMS

Rozwiązania w bazie relacyjnej

- Reprezentacja generyczna
 - elastyczna
 - nieczytelna
 - trudna do przetwarzania
- Częściowa strukturalizacja
 - zasada:
 - wyższe poziomy — struktury w bazie danych
 - niższe poziomy — CLOB
 - duże możliwości przetwarzania i wyszukiwania dla wyższych poziomów
 - wyszukiwanie kontekstowe dla niższych poziomów
 - elastyczność: możliwość rozbudowy w dół



Problemy z XML w aplikacjach z bazami danych (2)

Problemy konwersji XML ↔ baza danych

- Wczytywanie danych
 - brak typowych narzędzi
 - trzeba pisać specjalizowane programy
 - potrzebny parser z interfejsem do bazy danych
- Wyprowadzanie danych
 - niedostosowanie typowych narzędzi, np. do raportowania
 - stosunkowo łatwo przystosować narzędzia do prezentacji w WWW



Wprowadzenie — cele języka XML

Składniki XML

XML a HTML

Możliwości zastosowania XML w systemach informatycznych

Pierwsze doświadczenia z XML

Podsumowanie



Prezentacja problemu

Problem

- System wspomaganie pracy dziekanatu dużego wydziału wyższej uczelni
- Problem przechowywania i prezentacji konspektów wykładanych przedmiotów (ok. 800 sztuk)

Budowa konspektu

- Nagłówek
 - dane przedmiotu
 - słowa kluczowe
- Kilka sekcji o ustalonych tytułach i kolejności
- Sekcja jest tekstem z elementami formatowania
 - paragrafy
 - wymuszone złamanie linii
 - kursywa
 - listy zwykłe i numerowane
 - spis literatury

Potrzeby

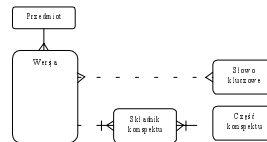
- Wyszukiwanie, np. wg
 - tytułu i identyfikatora przedmiotu,
 - słów kluczowych
- Wyprowadzanie w postaci
 - raportu (wydruku)
 - książki
 - stron WWW
- Wprowadzanie
 - przez typowy formularz
 - z plików tekstowych o ustalonej postaci



Rozwiązanie

Struktura danych

- Częściowa strukturalizacja
 - dane przedmiotu (z wersją)
 - słowa kluczowe
 - części konspektu
- Niższe poziomy struktury przechowywane tekstowo w XML



Aplikacja

- Import z plików w XML z pomocą specjalizowanego programu
 - szybkie działanie (język C)
 - konwersja z różnych stron kodowych
 - sprawdzanie poprawności danych (np. zgodności ze słownikiem)
- Wydruk raportu
 - Oracle Reports
 - uproszczony wygląd
 - konwersja XML → tekst funkcją PL/SQL
- Wyprowadzanie w HTML
 - Oracle Web Application Server
 - aplikacja wygenerowana przez WebServer Generator
 - pełne formatowanie
 - konwersja XML → HTML funkcją PL/SQL
- Wyprowadzanie w TeX
 - za pomocą specjalizowanego programu
 - w postaci specjalnie zdefiniowanych makr
 - specjalny styl w TeX formatuje wizualnie



Rozwiązanie (2)

```
<?xml version="1.0"? standalone="yes" encoding="iso-8859-2"?>
<!DOCTYPE eres_konspekty>

<przedmiot id="KBD2" wersja="1">
  <słowa_kluczowe> "bazy danych", "Oracle" </słowa_kluczowe>
  <konspekt>
    <czesc_konspektu id="Streszczenie">
      Monograficzny przedmiot poświęcony bazie danych i
      narzędziom Oracle.
    </czesc_konspektu>
    <czesc_konspektu id="Treść przedmiotu">
      <p> Omawiane są podstawowe zagadnienia związane z
      wykorzystaniem RDBMS Oracle7 i <i>Oracle8</i> oraz
      administrowaniem nimi.
      </p>
      <p> Przedstawiane są także narzędzia do budowy aplikacji:
      <ul>
        <li> Oracle Forms, </li>
        <li> Oracle Reports. </li>
      </ul>
      </p>
    </czesc_konspektu>
  </konspekt>
</przedmiot>
```



- Wprowadzenie — cele języka XML
- Składniki XML
- XML a HTML
- Możliwości zastosowania XML w systemach informatycznych
- Pierwsze doświadczenia z XML
- Podsumowanie



Podsumowanie

Stan prac nad XML

- Gotowa specyfikacja XML (W3C)
- Specyfikacje Xlink i XSL w przygotowaniu
- Opracowano kilka języków na bazie XML
- Pojawiają się
 - parsery
 - specjalizowane edytory
 - przeglądarki
- Deklarowane w sparcie ze strony „wielkich” firm

XML a bazy danych

- Możliwe zastosowania
 - liczne
 - ciekawe
 - obiecujące
- Problemy
 - z reprezentacją XML w bazach danych
 - z konwersją i budową aplikacji
- Brak dostosowanych narzędzi



Język XML w aplikacjach z bazami danych

- Wprowadzenie — cele języka XML
- Składniki XML
- XML a HTML
- Możliwości zastosowania XML w systemach informatycznych
- Pierwsze doświadczenia z XML
- Podsumowanie

