STUDIA INFORMATICA 2003
Volume 24 Number 4 (56)

Michał ŚWIDERSKI
Politechnika Śląska, Instytut Informatyki

JĘZYK XML, TECHNOLOGIE Z NIM ZWIĄZANE ORAZ KIERUNKI ICH ROZWOJU*)

Streszczenie. W artykule przedstawiono pochodzenie i składnię języka XML oraz najpopularniejsze obecnie metody przetwarzania, walidacji oraz transformacji dokumentów XML. Następnie zaprezentowano zastosowania języka XML w komunikacji, składowaniu danych, udostępnianiu funkcjonalności w sieci, budowaniu Semantic Web oraz tworzeniu systemów Web GIS.

Słowa kluczowe: XML, DTD, XML Schema, XSLT, SOAP, Usługi Web

THE XML, TECHNOLOGIES ASSOCIATED WITH XML AND DIRECTIONS OF THEIR DEVELOPMENT

Summary. This article describes origin and syntax of the XML, as well as the most common and up to date methods of parsing, validating and transforming XML documents. The usage of the XML in communication, data storage, providing functionality on the web, building Semantic Web and Web GIS systems is presented subsequently

Keywords: XML, DTD, XML Schema, XSLT, SOAP, Web Services

1. Opis języka XML

Język XML (eXtensible Markup Language), podobnie jak język HTML, wywodzi się z języka SGML (Standard Generalized Markup Language). W3C (World Wide Web Consortium) rozpoczęło prace nad XML w 1996 r., kiedy okazało się, że język SGML jest zbyt skomplikowany, by aplikacje internetowe mogły z niego korzystać, a język HTML

*) Praca finansowana z funduszu Badań Własnych Instytutu Informatyki w roku 2003.

oferuje niewystarczające mechanizmy łączenia dokumentów w Internecie, nie oddziela danych od sposobu ich reprezentacji oraz nie narzuca stałej struktury dokumentu, przez co słabo nadaje się do przechowywania danych [1].

Wobec tych niedogodności zaprojektowano język XML, który charakteryzują następujące zalety:

- Język XML oparto na standardzie Unicode, dzięki czemu dokumenty mogą być tworzone w dowolnym języku.
- Poprzez zastosowanie schematów XML może być strukturalizowany, tak by umożliwić walidację składni i zawartości dokumentu XML. Cecha ta umożliwia definiowanie własnych standardów formatów dokumentów.
- Dokumenty XML mogą zawierać części wielu innych dokumentów.
- XML jest prosty w użyciu, zrozumiały dla człowieka, wspierany przez dużą ilość narzędzi i wykorzystuje infrastrukturę przygotowaną dla języka HTML.

Wadą języka XML jest duża ilość znaczników wewnątrz dokumentu, przez co format XML jest dużo obszerniejszy niż formaty binarne, a co za tym idzie - wymaga szerszego pasma przesyłu w sieci, więcej pamięci dyskowej i operacyjnej. Dodatkowo analiza składniowa dokumentu XML jest wolniejsza niż interpretacja formatu binarnego i wymaga więcej pamięci operacyjnej [2].

1.1. Struktura dokumentu XML

Wszystkie dokumenty XML składają się z hierarchicznie uporządkowanych elementów. Elementy umieszczane są między znacznikiem początkowym i końcowym i mogą zawierać inne elementy oraz tekst. Znacznik otwierający składa się z nazwy znacznika i zbioru atrybutów. Znacznik zmykający zawiera jedynie nazwę elementu poprzedzoną prawym ukośnikiem. Dokument XML można zaczynać od deklaracji XML, gdzie określona jest obowiązkowo wersja XML (jak dotąd istnieje tylko wersja 1.0) oraz opcjonalnie sposób kodowania znaków oraz samodzielność dokumentu [2]. Przykładowy dokument XML przedstawiono na rys. 1.

```
01: <?xml version="1.0" encoding="UTF-8"?>
02: <?xml-stylesheet type="text/xsl" href="show_book.xsl"?>
03: <!DOCTYPE catalog SYSTEM "catalog.dtd">
04: <!--catalog last updated 2003-01-14-->
05: <catalog xmlns="http://www.example.com/catalog/">
06:     <book id="bk101">
07:         <author>Johnny Malony</author>
08:         <title>XML Developer's Guide</title>
09:         <genre>Computer</genre>
10:     </book>
11: </catalog>
```

Rys. 1. Przykład dokumentu XML
Fig. 1. The example of XML document

Pierwszy wiersz dokumentu jest deklaracją XML. Drugi zawiera instrukcję przetwarzania, służącą do powiązania dokumentu z arkuszem stylów. Trzeci wiersz określa element *catalog* jako korzeń dokumentu oraz specyfikuje zewnętrzny plik DTD, który zostanie użyty do walidacji dokumentu. Wiersz czwarty zawiera komentarz. Kolejne wiersze opisują elementy składające się na zawartość dokumentu. Nazwy elementów są jednoznacznie rozpoznawane dzięki umieszczeniu ich w przestrzeni nazw określonej za pomocą unikalnego adresu URI określonego za pomocą atrybutu *xmlns*.

2. Technologie związane z XML

Popularność i szeroki wachlarz zastosowań języka XML wynika w głównej mierze z technologii związanych z tym językiem, które umożliwiają parsowanie dokumentów, narzucanie im żądanej struktury, przekształcanie jednej struktury dokumentu w inną oraz prezentację danych w żądanym formacie. Jeśli dodamy do tego wielorakie zastosowania XML w komunikacji sieciowej oraz składowaniu danych, wyłoni się cała wszechstronność tego języka.

2.1. Analizatory składniowe

Dokument XML analizowany jest zazwyczaj w całości lub element po elemencie. Pierwsze podejście realizowane jest za pomocą analizatorów składniowych współpracujących z modelem DOM (Document Object Model), gdzie cały dokument jest wczytywany na raz i tworzona jest jego drzewiasta reprezentacja w pamięci. Wadą tego podejścia jest fakt, że cały dokument jest umieszczany w pamięci, co w przypadku wielkich dokumentów może okazać się niemożliwe.

Nowszą metodą analizy dokumentów XML jest standard SAX (Simple API for XML). Podejście to polega na czytaniu dokumentu znacznik po znaczniku i przesyłaniu rozpoznanych informacji, jako zdarzenia, do aplikacji użytkownika, która zyskuje pełną kontrolę nad dokumentem. Rozwiązanie to usuwa ograniczenie na wielkość dokumentu [3].

2.2. Walidacja dokumentów XML

Zdecydowana większość obecnie stosowanych analizatorów składniowych oferuje także możliwość walidacji dokumentu XML. Walidujące analizatory składniowe korzystają obecnie z dwóch najpopularniejszych standardów schematów dokumentu: DTD (Document Type Definition) oraz XML Schema. Obydwie metody porównują dokument z jego schematem i wyszczególniają listę miejsc, w których dokument różni się od zadanego schematu. Aplikacja może w tej sytuacji zadecydować czy odrzucić dokument, odrzucić nieprawidłowy element czy też przystąpić do naprawiania dokumentu.

2.2.1. DTD

Walidacja DTD weszła w skład rekomendacji W3C dla języka XML w wersji 1.0. Przykładowy schemat DTD przedstawiono na rys. 2.

```
01: <!DOCTYPE PGROUP [
02: <!ELEMENT PGROUP          (PERSONA+, GRPDESCR) >
03: <!ELEMENT PERSONA         (#PCDATA)   >
04: <!ELEMENT GRPDESCR        (#PCDATA)   >
05: ]>
```

Rys. 2. Przykład definicji DTD
Fig. 2. The example of DTD

Pierwsza linia określa *PGROUP* jako typ dokumentu, który jest także nazwą korzenia dokumentu. Znacznik *<!ELEMENT>* jest użyty do deklaracji elementu w dokumencie. Element dokumentu *<PGROUP>* musi zawierać elementy *<PERSONA>* oraz *<GRPDESCR>*. Znak "+" określa, że element *<PGROUP>* może zawierać jeden lub więcej elementów *<PERSONA>* [4].

Wadą DTD jest trudność w jej czytaniu i pisaniu, ponieważ nie wykorzystuje składni XML. Ponadto DTD nie zapewnia wsparcia dla typów danych oraz przestrzeni nazw.

2.2.2. XML Schema

Ograniczenia DTD zostały usunięte w nowszym standardzie zapisu schematu, zwanym XML Schema, który przeznaczony jest szczególnie do opisu złożonych struktur i typów danych. XML Schema jest konstruowany jako dokument XML, umożliwia także narzucenie

określonego typu danych na zawartości elementu lub atrybutu. XML Schema przedstawiony na rys. 3. odpowiada definicji DTD z rys. 2.

```
01: <?xml version="1.0"?>
02: <Schema name="schema_sample_1"
03:       xmlns="urn:schemas-microsoft-com:xml-data"
04:       xmlns:dt="urn:schemas-microsoft-com:datatypes">
05:     <ElementType name="PERSONA" content="textOnly" model="closed"/>
06:     <ElementType name="GRPDESCR" content="textOnly" model="closed"/>
07:     <ElementType name="PGROUP" content="eltOnly" model="closed">
08:         <element type="PERSONA" minOccurs="1" maxOccurs="*"/>
09:         <element type="GRPDESCR" minOccurs="1" maxOccurs="1"/>
10:     </ElementType>
11: </Schema>
```

Rys. 3. Przykład XML Schema
Fig. 3. The example XML Schema

2.3. Transformacje XSLT

Transformacje XSLT (Extensible Stylesheet Language Transformations), będące podzbiorem XSL (Extensible Stylesheet Language), umożliwiają reprezentację danych zawartych w dokumencie XML w żądany sposób, co jest szeroko stosowane w prezentacji XML jako HTML lub XHTML, a także w systemach CMS (Content Management Systems).

Technicznie XSLT jest dokumentem XML przeznaczonym do określania zasad, według których wejściowy dokument XML zostanie transformowany w inny dokument. XSLT zawiera zbiór reguł, które składają się z szablonu oraz wzorca. Procesor XSLT porównuje wszystkie węzły z dokumentu wejściowego z szablonami zawartymi w regułach. Gdy węzeł pasuje do szablonu, procesor zapisuje wzorzec z reguły do drzewa wyjściowego. Po zakończeniu analizy dokumentu wynik może zostać zapisany jako dokument XML, czysty tekst, HTML lub w innym wymaganym formacie.

Transformacje XSLT wykorzystują specyfikację XPath do jednoznacznego określania ścieżek wewnątrz dokumentu XML, XPointer do wskazania na odpowiedni fragment dokumentu XML oraz XLink do złożonego łączenia danych [2].

2.4. Komunikacja

Standard XML coraz szerzej wykorzystywany jest w komunikacji w sieci, a w szczególności w sieci Internet. Dokumenty XML o ściśle określonej strukturze służą do implementacji protokołów komunikacyjnych oraz udostępniania funkcjonalności.

2.4.1. SOAP

SOAP (Simple Object Access Protocol) jest prostym, rozszerzalnym protokołem bazującym na XML, przeznaczonym do wymiany informacji w zdecentralizowanym, rozproszonym środowisku. Standard SOAP definiuje zrąb struktury komunikatu, model przetwarzania komunikatów, zestaw zasad kodowania przy zapisie danych oraz dwa modele zdalnego wywoływania procedur: za pomocą wywołań podobnych do RPC oraz za pomocą przesyłu dokumentu XML.

2.4.2. WSDL

WSDL (Web Services Description Language) jest wykorzystywany do opisu serwisów Web, tak by oprogramowanie zdalne mogło automatycznie korzystać z udostępnianej przez nie funkcjonalności. WSDL jest dokumentem XML, który opisuje zestaw komunikatów SOAP oraz sposób ich wymiany z serwisem Web.

2.4.3. UDDI

UDDI (Universal Description, Discovery and Integration) określa standardowy sposób publikacji i wyszukiwania informacji o serwisach Web. Schematy XML Schema związane z UDDI definiują cztery zestawy informacji, które umożliwiają wykorzystanie opublikowanych serwisów Web: informacje biznesowe, ogólne informacje o serwisie Web, informacje potrzebne do połączenia z serwisem Web oraz specyfikację serwisu Web.

3. Kierunki rozwoju technologii związanych z XML

Ekspansja technologii związanych z XML w kierunku nowych dziedzin i zastosowań generuje specyficzne problemy, którym próbuje się zaradzić zarówno na poziomie programowym, jak i sprzętowym. Ważniejsze kierunki rozwoju i nowo powstałe problemy zostaną opisane poniżej.

3.1. Składowanie dokumentów XML w bazach danych

W związku z rosnącą popularnością XML wyłoniła się potrzeba przechowywania dokumentów XML w bazach danych. Jednak zastosowanie XML w tradycyjnych relacyjnych bazach danych napotkało na kilka problemów. Dokument XML może przechowywać wiele różnych struktur pochodzących z różnych przestrzeni nazw, co może sprawiać trudności w relacyjnej bazie danych, w których struktura przechowywanych danych powinna być jednolita. Kolejnym mankamentem jest brak walidacji dokumentu przez bazę danych, co

może skutkować wprowadzeniem do niej niepoprawnych informacji. Ostatecznie napotykamy na złożony problem modelowania relacji dla hierarchicznych struktur, jakimi są dokumenty XML [5].

Obecnie proponuje się dwa podejścia w rozwiązywaniu tych problemów. Starszym podejściem jest próba dostosowania istniejących relacyjnych baz danych do współpracy z XML poprzez dodanie pośredniej warstwy, która umożliwia zadawanie zapytań oraz prezentację wyników w formacie XML. Każda licząca się firma, tworząca oprogramowanie bazodanowe, wypracowała własne rozwiązania języka zapytań oraz warstwy pośredniej, jednak szczególną popularność zyskuje język zapytań XQuery, który jest implementowany zgodnie z rekomendacjami W3C [6].

Nowsze podejście bazuje na spostrzeżeniu, że lepiej jest przechowywać dokumenty XML w postaci XML, co usuwa konieczność każdorazowego tłumaczenia XML na relacyjny format danych lub w kierunku przeciwnym. Powstaje coraz więcej baz danych przechowujących bezpośrednio dokumenty XML (Native XML Databases), jednak wszystkie rozwiązania borykają się z problemem wydajności, modelowania relacji, utrzymywania spójności danych oraz standaryzacji rozwiązań [5].

3.2. Semantic WEB

Jednym z przyszłościowych zastosowań XML jest użycie go jako języka metadanych w projekcie Semantic Web, który jest wizją udostępnienia danych i funkcjonalności w sieci w dobrze zdefiniowany sposób, tak by aplikacje mogły posługiwać się zasobami w sieci jako składnicą danych i zestawem funkcjonalności gotowych do wykorzystania. Dzisiejsze formaty danych, oparte na XML, są dedykowane dla pewnej grupy zastosowań i przez to nieużyteczne dla innych. Projekt Semantic Web ma zdefiniować ogólne standardy struktur i połączeń danych w sieci [7].

3.3. Web GIS

Web GIS jest zbiorczym określeniem rozwiązań mających na celu udostępnianie informacji geograficznych w sieci w sposób wizualny i skalarny. Obecnie coraz większą popularność zyskuje standard GML (Geography Markup Language), który bazuje na standardach schematów XML opracowanych przez OGC (OpenGIS Consortium). Celem GML jest udostępnienie użytkownikom internetowym geograficznych informacji przy użyciu standardowej przeglądarki internetowej. GML, podobnie jak sam XML, nie określa sposobu reprezentacji danych, które opisuje, dlatego może być tłumaczony na grafikę rastrową lub wektorową, tekst, mowę lub dźwięk [8].

3.4. Wydajność

Naturalnym problemem pojawiającym się przy przetwarzaniu XML jest wydajność. Dokumenty XML wymagają analizy składniowej, walidacji, transformacji, czasem XML Routing (przesłania dokumentu na podstawie jego zawartości do odpowiedniego adresata). Efektem jest około dziesięciokrotne zwolnienie pracy serwerów aplikacji [9]. By poradzić sobie z tym problemem, rozwija się obecnie sprzęt sieciowy, tak by przejmował część zadań wykonywanych normalnie przez serwery aplikacji. Routery i akceleratory XML potrafią obecnie wykonać samodzielnie analizę składniową, walidację, transformację oraz XML Routing dokumentów XML [10].

3.5. Bezpieczeństwo

Korzystanie z komunikacji za pomocą XML stawia nowe wymagania systemom zabezpieczeń serwerów internetowych. Przychodzące z zewnątrz dokumenty XML powinny być zbadane pod względem zawartości i całe lub ich części powinny być przesyłane do osób z odpowiednimi uprawnieniami. Dodatkowo komunikaty SOAP przesyłane są jawnym tekstem poprzez port nr 80, przez co traktowane są podobnie jak dokumenty HTML, jednak po stronie serwera internetowego mogą wywoływać potencjalnie niebezpieczne procedury. By zabezpieczyć się przed tego typu atakami, wprowadzane są obecnie tzw. firewalle XML (XML firewall), które na bieżąco dokonują parsowania, koniecznych tranformacji i routowania oraz badają uprawnienia komunikatów SOAP do wykonywania żądanych procedur, z uwzględnieniem autoryzacji za pomocą technologii .NET Passport i SAML (Security Assertion Markup Language) [11].

4. Podsumowanie

Język XML, dzięki swojej prostocie, jasno określonej strukturze dokumentu oraz wielości technologii z nim związanych stał się *de facto* standardem dla wymiany danych w sieci Internet. Patrząc na kierunki rozwoju i ekspansję technologii bazujących na XML można spodziewać się zagospodarowywania przez nie coraz to nowych dziedzin związanych z przesyłaniem, przetwarzaniem i składowaniem danych.

LITERATURA

1. Sturm J.: Developing XML Solutions. Microsoft Press, Washington 2000.

2. Arciniegas F.: C++ XML. Mikom, Warszawa 2002.
3. Harold E., Scott W.: XML in a Nutshell. O'Reilly, Cambridge 2002.
4. XML Core. MSDN Library, January 2002.
5. Floyd M.: XML Exposed. PC Magazine, czerwiec 2002.
6. Funderburk J. E., Malaika S., Reinwald B.: XML programming with SQL/XML and XQuery. IBM Systems Journals, 2002, Vol. 41, No 4.
7. http://www.semanticweb.org
8. Geography Markup Language (GML) 2.0. http://www.opengis.net
9. Greenfield D.: XML: Racing ahead? Network Magazine, styczeń 2003.
10. Hicks M.: Appliances accelerate XML data traffic. Eweek, sierpień 2002.
11. Taft D. K.: XML firewalls aid services. Eweek, sierpień 2002.

Recenzent: Dr hab. inż. Franciszek Marecki

Wpłynęło do Redakcji 1 kwietnia 2003 r.

Abstract

The first section of this article describes the origin of XML, designed as a subset of SGML, and the basic structure of XML document. The sample XML code is presented (see Fig.1).

Then, in the second section, we move to technologies associated with XML that parse, validate and transform XML documents. Two basic concepts of parsing are presented i.e.: DOM and SAX, showing their similarities and differences. DTD and XML Schema specifications represent two most common methods of XML validation (see Fig.2 and 3.). XML transformations are shown with use of XSLT. At the end of this section we find out how XML facilitates communication over the internet with use of SOAP, Web Services, WSDL and UDDI.

In the third section, the author describes applications of XML in relational and native XML databases, vision of Semantic Web and GML specification. The problems of performance and security are also mentioned.

Adres

Michał ŚWIDERSKI: Politechnika Śląska, Instytut Informatyki, ul. Akademicka 16, 44-101 Gliwice, Polska, mswid@star.iinf.polsl.gliwice.pl .