

Wprowadzenie do technologii XML

Bartłomiej Świercz

Katedra Mikroelektroniki i Technik Informatycznych

Łódź, 6 października 2005 roku

- 1 Wprowadzenie
 - Informacje organizacyjne
 - Omówienie przedmiotu
- 2 XML
 - Wstęp
 - XML vs HTML
 - Struktura XML
- 3 DTD
 - Wstęp
 - Struktura

Informacje o przedmiocie

Dokładne informacje o przedmiocie „Rozproszone technologie obiektowe w środowisku WebServices” znajdują się na stronie:

<http://neo.dmcs.p.lodz.pl/webservices/>

Kontakt

Wykład prowadzi: mgr inż. **Bartłomiej Świercz**

Doktorant w Katedrze Mikroelektroniki i Technik Informatycznych

Email: **swierczu@dmcs.pl**

Telefon: 631-26-20

Warunki zaliczenia

Warunkiem zaliczenia wykładu jest pomyślne zrealizowanie projektu. Projekt będzie realizowany w grupach 2-3 osobowych. Każda grupa będzie prezentować projekt na dwóch ostatnich wykładach.

Program wykładu

Celem wykładu jest przedstawienie technologii XML i pokrewnych do niej technologii:

- XML: DTD, XML Schema, XSLT.
- Analiza leksykalna dokumentów XML.
- Protokoły XML-RPC i SOAP.
- Usługi internetowe budowane w oparciu o WebServices.

Ćwiczenia będą realizowane przy użyciu różnych technologii i języków programowania.

Laboratorium

Ocena z laboratorium jest średnią arytmetyczną ocen z poszczególnych ćwiczeń.

Czym jest XML?

Najmodniejszym tematem w branży informatycznej. W każdej gazecie informatycznej znajdziemy wzmiankę o XML. Google na zapytanie o XML zwraca 619,000,000 dokumentów, Amazon oferuje 1,039 książek o tej tematyce.

Czym jest XML?

Najmodniejszym tematem w branży informatycznej. W każdej gazecie informatycznej znajdziemy wzmiankę o XML. Google na zapytanie o XML zwraca **619,000,000** dokumentów, Amazon oferuje 1,039 książek o tej tematyce.

Czym jest XML?

Najmodniejszym tematem w branży informatycznej. W każdej gazecie informatycznej znajdziemy wzmiankę o XML. Google na zapytanie o XML zwraca 619,000,000 dokumentów, Amazon oferuje **1,039** książek o tej tematyce.

Czym jest XML?

XML (Extensible Markup Language) - wywodzi się od języka SGML i jest językiem znaczników służącym do **opisu danych**. Dane przechowywane są w postaci **tekstowej** w dokumencie o ściśle określonej strukturze. XML możemy uznać za:

- technologie opisu i przechowywania danych,
- rodzinę technologii do prezentacji i przetwarzania danych,
- podstawowy składnik nowoczesnych technologii rozproszonych.
- XML jest stosowany również do opisu swoich własnych rozszerzeń.

Czym jest XML?

XML (Extensible Markup Language) - wywodzi się od języka SGML i jest językiem znaczników służącym do **opisu danych**. Dane przechowywane są w postaci **tekstowej** w dokumencie o ściśle określonej strukturze. XML możemy uznać za:

- technologie opisu i przechowywania danych,
- rodzinę technologii do prezentacji i przetwarzania danych,
- podstawowy składnik nowoczesnych technologii rozproszonych.
- XML jest stosowany również do opisu swoich własnych rozszerzeń.

Czym jest XML?

XML (Extensible Markup Language) - wywodzi się od języka SGML i jest językiem znaczników służącym do **opisu danych**. Dane przechowywane są w postaci **tekstowej** w dokumencie o ściśle określonej strukturze. XML możemy uznać za:

- technologie opisu i przechowywania danych,
- rodzinę technologii do prezentacji i przetwarzania danych,
- podstawowy składnik nowoczesnych technologii rozproszonych.
- XML jest stosowany również do opisu swoich własnych rozszerzeń.

Czym jest XML?

XML (Extensible Markup Language) - wywodzi się od języka SGML i jest językiem znaczników służącym do **opisu danych**. Dane przechowywane są w postaci **tekstowej** w dokumencie o ściśle określonej strukturze. XML możemy uznać za:

- technologie opisu i przechowywania danych,
- rodzinę technologii do prezentacji i przetwarzania danych,
- podstawowy składnik nowoczesnych technologii rozproszonych.
- XML jest stosowany również do opisu swoich własnych rozszerzeń.

Czym jest XML?

XML (Extensible Markup Language) - wywodzi się od języka SGML i jest językiem znaczników służącym do **opisu danych**. Dane przechowywane są w postaci **tekstowej** w dokumencie o ściśle określonej strukturze. XML możemy uznać za:

- technologie opisu i przechowywania danych,
- rodzinę technologii do prezentacji i przetwarzania danych,
- podstawowy składnik nowoczesnych technologii rozproszonych.
- XML jest stosowany również do opisu swoich własnych rozszerzeń.

Gdzie stosujemy XML?

XML jest standardem przemysłowym i stosowany jest we wszystkich dziedzinach informatyki:

- prawo - LegalXML, matematyka - MatchML, chemia - ChemXL.
- dokumenty i grafika: SVG, OpenOffice.
- komunikacja: XML-RPC, SOAP.
- prezentacja, transformacja, walidacja: XSL-FO, XSLT, XML-Schema.

Gdzie stosujemy XML?

XML jest standardem przemysłowym i stosowany jest we wszystkich dziedzinach informatyki:

- prawo - LegalXML, matematyka - MatchML, chemia - ChemXL.
- dokumenty i grafika: SVG, OpenOffice.
- komunikacja: XML-RPC, SOAP.
- prezentacja, transformacja, walidacja: XSL-FO, XSLT, XML-Schema.

Gdzie stosujemy XML?

XML jest standardem przemysłowym i stosowany jest we wszystkich dziedzinach informatyki:

- prawo - LegalXML, matematyka - MatchML, chemia - ChemXL.
- dokumenty i grafika: SVG, OpenOffice.
- komunikacja: XML-RPC, SOAP.
- prezentacja, transformacja, walidacja: XSL-FO, XSLT, XML-Schema.

Gdzie stosujemy XML?

XML jest standardem przemysłowym i stosowany jest we wszystkich dziedzinach informatyki:

- prawo - LegalXML, matematyka - MatchML, chemia - ChemXL.
- dokumenty i grafika: SVG, OpenOffice.
- komunikacja: XML-RPC, SOAP.
- prezentacja, transformacja, walidacja: XSL-FO, XSLT, XML-Schema.

Gdzie stosujemy XML?

XML jest standardem przemysłowym i stosowany jest we wszystkich dziedzinach informatyki:

- prawo - LegalXML, matematyka - MatchML, chemia - ChemXL.
- dokumenty i grafika: SVG, OpenOffice.
- komunikacja: XML-RPC, SOAP.
- prezentacja, transformacja, walidacja: XSL-FO, XSLT, XML-Schema.

XML - samoopisujące się dane

- Znacznik opisuje znaczenie i funkcję elementu.
- Programista/aplikacja może przetwarzać dokument mając na uwadze typ elementu a nie sposób jego prezentacji.
- XML umożliwia bezpieczną wymianę danych pomiędzy różnymi systemami i prezentację wyników na różnych platformach: komputer osobisty, PDA, kasa fiskalna itd.

XML - samoopisujące się dane

- Znacznik opisuje znaczenie i funkcję elementu.
- Programista/aplikacja może przetwarzać dokument mając na uwadze typ elementu a nie sposób jego prezentacji.
- XML umożliwia bezpieczną wymianę danych pomiędzy różnymi systemami i prezentację wyników na różnych platformach: komputer osobisty, PDA, kasa fiskalna itd.

XML - samoopisujące się dane

- Znacznik opisuje znaczenie i funkcję elementu.
- Programista/aplikacja może przetwarzać dokument mając na uwadze typ elementu a nie sposób jego prezentacji.
- XML umożliwia bezpieczną wymianę danych pomiędzy różnymi systemami i prezentację wyników na różnych platformach: komputer osobisty, PDA, kasa fiskalna itd.

Przykład

Opis płyt dvd znajdujących się w wypożyczalni

```
<?xml version='1.0' encoding='iso-8859-2'?>
<katalog>
  <dvd id='1'>
    <tytuł>Rambo I</tytuł>
    <cena>5</cena>
    <sztuk>10</sztuk>
  </dvd>
  <dvd id='2'>
    <tytuł>Rambo II</tytuł>
    <cena>7</cena>
    <sztuk>12</sztuk>
  </dvd>
</katalog>
```

Czym nie jest XML?

- XML nie jest językiem programowania. Nie można przeprowadzić w nim obliczeń.
- Nie opisuje w jaki sposób dane mają być prezentowane. Jednak technologie rozwinięte wokół XML pozwalają na transformację i prezentację danych.

Czym nie jest XML?

- XML nie jest językiem programowania. Nie można przeprowadzić w nim obliczeń.
- Nie opisuje w jaki sposób dane mają być prezentowane. Jednak technologie rozwinięte wokół XML pozwalają na transformację i prezentację danych.

HTML ...

- Podobnie jak XML, HTML jest językiem znaczników.
- W odróżnieniu od XML, HTML opisuje sposób prezentacji, a nie dane.
- Przeglądarki HTML starają się czytać i interpretować nieprawidłowe dokumenty HTML (brak kończących znaczników itd.).
- HTML ma tylko jedno medium prezentacji. Jest nim standardowy ekran komputera.

HTML ...

- Podobnie jak XML, HTML jest językiem znaczników.
- W odróżnieniu od XML, HTML opisuje sposób prezentacji, a nie dane.
- Przeglądarki HTML starają się czytać i interpretować nieprawidłowe dokumenty HTML (brak kończących znaczników itd.).
- HTML ma tylko jedno medium prezentacji. Jest nim standardowy ekran komputera.

HTML ...

- Podobnie jak XML, HTML jest językiem znaczników.
- W odróżnieniu od XML, HTML opisuje sposób prezentacji, a nie dane.
- Przeglądarki HTML starają się czytać i interpretować nieprawidłowe dokumenty HTML (brak kończących znaczników itd.).
- HTML ma tylko jedno medium prezentacji. Jest nim standardowy ekran komputera.

HTML ...

- Podobnie jak XML, HTML jest językiem znaczników.
- W odróżnieniu od XML, HTML opisuje sposób prezentacji, a nie dane.
- Przeglądarki HTML starają się czytać i interpretować nieprawidłowe dokumenty HTML (brak kończących znaczników itd.).
- HTML ma tylko jedno medium prezentacji. Jest nim standardowy ekran komputera.

Problemy wynikające z obecnego stanu technologii HTML

- Jedna strona HTML jest wyświetlana w różny sposób przez różne przeglądarki.
- Ze względu na rozszerzenia implementowane przez firmy niezależnie od komisji standaryzacyjnej, konieczne jest przygotowanie wielu wersji tej samej strony dla najpopularniejszych przeglądarek.

Rozwiązanie ...

W celu uniezależnienia dokumentów HTML od konkretnego medium prezentacji powstała nowa wersja HTML, która jest podzbiorem języka XML. Technologia ta nazywa się XHTML.

Problemy wynikające z obecnego stanu technologii HTML

- Jedna strona HTML jest wyświetlana w różny sposób przez różne przeglądarki.
- Ze względu na rozszerzenia implementowane przez firmy niezależnie od komisji standaryzacyjnej, konieczne jest przygotowanie wielu wersji tej samej strony dla najpopularniejszych przeglądarek.

Rozwiązanie ...

W celu uniezależnienia dokumentów HTML od konkretnego medium prezentacji powstała nowa wersja HTML, która jest podzbiorem języka XML. Technologia ta nazywa się XHTML.

Problemy wynikające z obecnego stanu technologii HTML

- Jedna strona HTML jest wyświetlana w różny sposób przez różne przeglądarki.
- Ze względu na rozszerzenia implementowane przez firmy niezależnie od komisji standaryzacyjnej, konieczne jest przygotowanie wielu wersji tej samej strony dla najpopularniejszych przeglądarek.

Rozwiązanie ...

W celu uniezależnienia dokumentów HTML od konkretnego medium prezentacji powstała nowa wersja HTML, która jest podzbiorem języka XML. Technologia ta nazywa się XHTML.

Struktura dokumentu XML

Rekomendacja W3C znajduje się na stronie:

<http://www.w3.org/TR/REC-xml/>

Dokument XML składa się z trzech sekcji:

- Prolog.
- Treść.
- Epilog.

Jedynie sekcja treść jest obowiązkowa. Pozostałe dwie nie muszą występować. Dodatkowym elementem jest komentarz:

```
<!-- mój komentarz -->
```

Prolog

Prolog stanowi pierwszą sekcję dokumentu i powinien rozpoczynać się od deklaracji XML. Deklaracja XML wygląda następująco:

```
<?xml version='1.0'?'>
```

Deklaracja ta zawiera informacje o typie dokumentu (XML) i jego wersji (1.0). Dodatkowo można podać informację o kodowaniu i zaznaczyć czy dokument wymaga zewnętrznych dokumentów:

```
<?xml version='1.0' encoding='utf-8'  
standalone='yes'?'>
```

Treść

W treści dokumentu znajdują się właściwe dane. Treść dokumentu XML zawiera tylko jeden element objęty parą znaczników:

```
<dane> ...</dane>
```

W początkowej parze znaczników może być zagnieżdżona dowolna ilość elementów (znaczników) XML.

Epilog

Epilog jest najczęściej pomijany. Może zawierać instrukcje dotyczące przetwarzania dokumentu.

Element

Element jest podstawowym pojemnikiem do przechowywania danych. Definiuje typ i przeznaczenie przechowywanych danych.

```
<imię> Bartłomiej </imię>
```

Element może nie posiadać danych. Dozwolony jest brak zawartości pomiędzy znacznikami:

```
<imię> </imię>
```

Można użyć skróconego zapisu:

```
<imię/>
```

Element

Element jest podstawowym pojemnikiem do przechowywania danych. Definiuje typ i przeznaczenie przechowywanych danych.

```
<imię> Bartłomiej </imię>
```

Element może nie posiadać danych. Dozwolony jest brak zawartości pomiędzy znacznikami:

```
<imię> </imię>
```

Można użyć skróconego zapisu:

```
<imię/>
```

Element

Element jest podstawowym pojemnikiem do przechowywania danych. Definiuje typ i przeznaczenie przechowywanych danych.

```
<imię> Bartłomiej </imię>
```

Element może nie posiadać danych. Dozwolony jest brak zawartości pomiędzy znacznikami:

```
<imię> </imię>
```

Można użyć skróconego zapisu:

```
<imię/>
```

Element i atrybuty

Każdy element może posiadać dodatkowe atrybuty:

```
<dvd id='19922' typ='komedia'>Shrek</dvd>
```

Użycie atrybutu jest ściśle określone:

- Wszystkie wartości atrybutów muszą być ujęte w cudzysłowy lub apostrofy.
- Nie można użyć kilkakrotnie tej samej nazwy wewnątrz jednego atrybutu.
- W wartościach atrybutów nie mogą występować znaki < i &. Zamiast nich stosujemy < i %amp;.

Element i atrybuty

Każdy element może posiadać dodatkowe atrybuty:

```
<dvd id='19922' typ='komedia'>Shrek</dvd>
```

Użycie atrybutu jest ściśle określone:

- Wszystkie wartości atrybutów muszą być ujęte w cudzysłowy lub apostrofy.
- Nie można użyć kilkakrotnie tej samej nazwy wewnątrz jednego atrybutu.
- W wartościach atrybutów nie mogą występować znaki < i &. Zamiast nich stosujemy < i %amp;.

Element i atrybuty

Predefiniowane znaki w dokumentach XML:

Sekwencja:	Znak:
<	<
>	>
&	&
"	"
'	'

Dodatkowe informacje o składni XML

- XML jest wrażliwy na wielkość liter. Znacznik `<from>` i `<From>` to dwa różne znaczniki.
- XML uwzględnia białe znaki pomiędzy znacznikami.
- Jedyne poprawne zakończenie linii dokumentu XML to znak LF (`'\n'`).

Dodatkowe informacje o składni XML

- XML jest wrażliwy na wielkość liter. Znacznik `<from>` i `<From>` to dwa różne znaczniki.
- XML uwzględnia białe znaki pomiędzy znacznikami.
- Jedyne poprawne zakończenie linii dokumentu XML to znak LF (`'\n'`).

Dodatkowe informacje o składni XML

- XML jest wrażliwy na wielkość liter. Znacznik `<from>` i `<From>` to dwa różne znaczniki.
- XML uwzględnia białe znaki pomiędzy znacznikami.
- Jedyne poprawne zakończenie linii dokumentu XML to znak LF (`'\n'`).

Znaczniki

W XML znaczniki muszą być nazywane zgodnie z zasadami:

- Nazwa może zawierać litery, cyfry i inne znaki drukowalne.
- Nazwa musi zaczynać się od litery.
- Nazwa nie może zaczynać się od ciągu 'xml', 'XML', 'Xml'.
- Nazwa nie może zawierać znaków białych.

Definicja typu dokumentu

Aby dokument XML był użyteczny konieczne jest oprócz zdefiniowania składni zdefiniowanie znaczników i określenie sekwencji w jakich mogą występować.

Opis logiczny dokumentu realizuje się między innymi za pomocą definicji typu dokumentu **DTD** (Document Type Definition).

Definicja DTD

DTD jest dokładną specyfikacją tego, co może pojawić się w dokumencie XML i ściśle określa strukturę drzewa dokumentu (poziomy zagnieżdżeń). Dokument XML, który nie zawiera opisu DTD określa się mianem **niepoprawnego**.

DTD jest szkieletem dokumentu XML.

Podstawą DTD jest element

Podstawę DTD stanowi deklaracja **ELEMENT**, która ma postać:

```
<!ELEMENT nazwa_elementu ...>
```

Za nazwą znacznika można wymienić zawarte w nim elementy podrzędne.

Elementy podrzędne

Zasady dodawania elementów podrzędnych:

- , - Przecinek używany jest do tworzenia listy elementów podrzędnych, które muszą wystąpić w odpowiedniej kolejności.
- | - Znakiem 'pipe' rozdziela się elementy opcjonalne.
- ? - Pytajnik oznacza opcjonalny element podrzędny.
- * - Dowolna liczba wystąpień elementu podrzędnego.
- + - Co najmniej jedno wystąpienie elementu podrzędnego.
- (...) - Grupowanie elementów podrzędnych.

Przykład

```
<!ELEMENT kanapka (chleb, (dżem | nutella), chleb) >
```

```
<!ELEMENT kanapka (chleb, (dżem | nutella)?, chleb) >
```

```
<!ELEMENT napój (herbata, (kakao | sok)?) >
```

```
<!ELEMENT posiłek (kanapka+, napój) >
```

Przykład

```
<!ELEMENT kanapka (chleb, (dżem | nutella), chleb) >
```

```
<!ELEMENT kanapka (chleb, (dżem | nutella)?, chleb) >
```

```
<!ELEMENT napój (herbata, (kakao | sok)?) >
```

```
<!ELEMENT posiłek (kanapka+, napój) >
```

Przykład

```
<!ELEMENT kanapka (chleb, (dżem | nutella), chleb) >
```

```
<!ELEMENT kanapka (chleb, (dżem | nutella)?, chleb) >
```

```
<!ELEMENT napój (herbata, (kakao | sok)?) >
```

```
<!ELEMENT posiłek (kanapka+, napój) >
```

Przykład - wypożyczalnia DVD

Opis struktury drzewiastej bez uwzględnienia typów

```
<!ELEMENT katalog (dvd+) >  
<!ELEMENT katalog (tytuł, cena, sztuk) >
```

Typy elementów

Typy danych

- CDATA - napis.
- ID - nazwa unikatowa w dokumencie XML.
- IDREF - odwołanie do innego elementu po ID.
- ENTITY - nazwa zewnętrznej jednostki.
- NOTATION - zdefiniowana na zewnątrz notacja.

Kwantyfikatory typów danych

- #REQUIRED - atrybut wymagany.
- #IMPLIED - atrybut opcjonalny.
- #FIXED <wartość> - atrybut musi mieć podaną wartość.

Typy elementów

Typy danych

- CDATA - napis.
- ID - nazwa unikatowa w dokumencie XML.
- IDREF - odwołanie do innego elementu po ID.
- ENTITY - nazwa zewnętrznej jednostki.
- NOTATION - zdefiniowana na zewnątrz notacja.

Kwantyfikatory typów danych

- #REQUIRED - atrybut wymagany.
- #IMPLIED - atrybut opcjonalny.
- #FIXED <wartość> - atrybut musi mieć podaną wartość.

Przykład - wypożyczalnia DVD

Opis struktury drzewiastej z uwzględnieniem typów

```
<!ELEMENT katalog (dvd+) >  
<!ELEMENT dvd (tytuł, cena, sztuk) >  
<!ATTLIST dvd id ID #REQUIRED >  
<!ELEMENT tytuł (#PCDATA) >  
<!ELEMENT cena (#PCDATA) >  
<!ELEMENT sztuk (#PCDATA) >
```

Powiązanie DTD z dokumentem XML

Pełna definicja dokumentu XML z wypożyczalni DVD:

```
<!DOCTYPE katalog [  
  <!ELEMENT katalog (dvd+) >  
  <!ELEMENT dvd (tytuł, cena, sztuk) >  
  <!ATTLIST dvd id ID #REQUIRED >  
  <!ELEMENT tytuł (#PCDATA) >  
  <!ELEMENT cena (#PCDATA) >  
  <!ELEMENT sztuk (#PCDATA) >  
>
```

Dołączanie DTD do dokumentu XML

Opis struktury dokumentu XML w postaci definicji DTD można dołączyć do dokumentu na dwa sposoby:

- Poprzez umieszczenie go w prologu dokumentu:

```
<?xml version='1.0'?>  
<!DOCTYPE znacznik-root [  
...  

```

- Dołączenie zewnętrznego pliku z opisem DTD do dokumentu XML:

```
<?xml version='1.0'?>  
<!DOCTYPE znacznik-root SYSTEM 'nazwa_pliku'>
```

Dołączanie DTD do dokumentu XML

Opis struktury dokumentu XML w postaci definicji DTD można dołączyć do dokumentu na dwa sposoby:

- Poprzez umieszczenie go w prologu dokumentu:

```
<?xml version="1.0"?>  
<!DOCTYPE znacznik-root [  
...  

```

- Dołączenie zewnętrznego pliku z opisem DTD do dokumentu XML:

```
<?xml version="1.0"?>  
<!DOCTYPE znacznik-root SYSTEM "nazwa_pliku">
```