

Prezentacja i transformacja

Bartłomiej Świercz

Katedra Mikroelektroniki i Technik Informatycznych

Łódź, 21 października 2005 roku

1 Prezentacja

- Przykładowa aplikacja
- CSS - Cascading Style Sheets
- CSS2 a XML
- CSS2 a XSLT

2 XPath

- Wstęp
- Składnia i terminologia XPath

3 XSLT

- Wstęp
- Struktura XSLT

Przykładowa aplikacja

Podczas wykładów przykładowe aplikacje będą dotyczyły obsługi wypożyczalni płyt DVD. Dane o dostępnych płytach będą zapisane w pliku XML, którego definicja DTD ma postać:

Plik: wypożyczalnia.dtd

```
<!ELEMENT katalog (dvd+) >  
<!ELEMENT dvd (tytul, aktorzy, sztuk) >  
<!ATTLIST dvd id CDATA #REQUIRED >  
<!ELEMENT tytul (#PCDATA) >  
<!ELEMENT aktorzy (aktor+) >  
<!ELEMENT aktor (#PCDATA) >  
<!ELEMENT sztuk (#PCDATA) >
```

Przykładowa aplikacja

Plik: wypożyczalnia.xml

```
<?xml version="1.0" encoding="iso-8859-2"?>
<!DOCTYPE katalog SYSTEM "wypożyczalnia.dtd">
<katalog>
  <dvd id="1">
    <tytul>Rambo I</tytul>
    <aktorzy>
      <aktor>Sylvester Stallone</aktor>
    </aktorzy>
    <sztuk>2</sztuk>
  </dvd>
  ...
</katalog>
```

Prezentacja za pomocą CSS

Co to jest CSS?

- CSS jest skrótem od Cascading Style Sheetsh.
- Style określają w jaki sposób mają być wyświetlone poszczególne elementy dokumentu.
- Style zostały dodane do HTML 4.0 w celu rozwiązania wielu problemów.
- Zewnętrzne style oszczędzają pracę programisty/grafika.

Prezentacja za pomocą CSS

Co to jest CSS?

- CSS jest skrótem od Cascading Style Sheetsh.
- Style określają w jaki sposób mają być wyświetlone poszczególne elementy dokumentu.
- Style zostały dodane do HTML 4.0 w celu rozwiązania wielu problemów.
- Zewnętrzne style oszczędzają pracę programisty/grafika.

Cascading Style Sheets, Level 2

Fragment dokumentu Technical Report CSS2 Specification:

*This specification defines Cascading Style Sheets, level 2 (CSS2). CSS2 is a style sheet language that allows authors and users to attach **style** (e.g., fonts, spacing, and aural cues) to structured documents (e.g., HTML documents and **XML** applications). By **separating** the **presentation** style of documents from the **content** of documents, CSS2 simplifies Web authoring and site maintenance.*

Style CSS2 a dokumenty XML

Style mogą być użyte z każdym dokumentem strukturalnym jakim jest XML. Powoduje to, że **XML jest bardziej zależny od stylów niż HTML!**

Włączanie CSS do dokumentu XML

Aby skojarzyć arkusz styli CSS z dokumentem XML należy do nagłówka (prologu) dokumentu dołączyć deklarację:

```
<?xml:stylesheet type="text/css" href="style.css"?>
```

Element blokowy czy rozwijany?

W arkuszu CSS należy zdecydować, które elementy są blokowe (zakończone znakiem nowej linii), a które rozwijane w miejscu wystąpienia.

```
katalog, dvd, tytuł, \  
aktorzy, sztuk {display: block}  
aktor {display: inline}
```

Element blokowy czy rozwijany?

W arkuszu CSS należy zdecydować, które elementy są blokowe (zakończone znakiem nowej linii), a które rozwijane w miejscu wystąpienia.

```
katalog, dvd, tytuł, \  
aktorzy, sztuk {display: block}  
aktor {display: inline}
```

Przykład arkusza stylu CSS

Arkusz CSS:

```
katalog {background-color: #ffffff; width=100%;}  
dvd {background-color: #aaaaaa; display: block;  
margin-bottom: 30pt;}  
tytul {display: block; color: #0000ff; font-size:  
20pt;}  
aktorzy, sztuk {display: block; color: #000000;  
font-size: 14pt;}  
aktor {display: inline; font-style: italic;}
```

Cele projektowe stylu CSS2

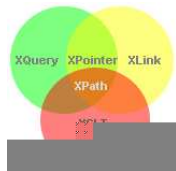
- Przyszła i wsteczna kompatybilność.
- Przeznaczone do dokumentów strukturalnych.
- Niezależność od platformy i urządzenia.
- Proste i łatwe nanoszenie zmian.
- Minimalizacja obciążenia łącza sieciowego.

Style CSS2 a transformacje XSLT

Style CSS2	Transformacja XSLT
Brak możliwości zmodyfikowania struktury dokumentu.	Zmiana struktury dokumentu.
Element może być skojarzony z pojedynczym obiektem wizualnym.	Element może być dowolnie grupowany.
Prezentacja za pomocą jednego modelu wizualnego.	Transformacja do dowolnego modelu wizualnego.

Technologia XPath

XPath jest technologią opartą o XML służącą do wyszukiwania informacji w dokumentach XML. XPath służy do “poruszania” się po elementach i atrybutach dokumentu.



XPath jest główną częścią standardu XSLT W3C. Technologie XQuery i XPointer zostały również zaprojektowane jako wyrażenia XPath. **Zrozumienie XPath jest niezbędne do zrozumienia zaawansowanych technologii XML.**

Czym jest XPath?

- XPath jest składnią służącą do definiowania części dokumentu XML.
- XPath używa opisu “ścieżek” do nawigowania po dokumencie XML.
- XPath zawiera wbudowaną bibliotekę standardowych funkcji.
- XPath jest głównym elementem XSLT.
- XPath jest standardem W3C.

XPath używa opisu “ścieżek” . . .

XPath używa wyrażeń w postaci “ścieżek” do wybierania węzła lub grupy węzłów w dokumencie XML. Wyrażenia te (“ścieżki”) wyglądają jak tradycyjne odwołania do plików w systemie komputerowym.

Standardowe funkcje XPath

XPath zawiera ponad sto wbudowanych funkcji. Funkcje te służą do manipulacji ciągami znaków, wartości numerycznych, porównywania daty i czasu, operacji na węzłach dokumentu oraz dostarcza funkcji do obsługi wartości logicznych.

Terminologia XPath

XPath rozróżnia siedem typów węzłów:

- Element.
- Atrybut.
- Tekst.
- Przestrzeń nazw.
- Instrukcję przetwarzania.
- Komentarz.
- Węzeł dokumentu.

Przykład

Plik: wypożyczalnia.xml

```
<?xml version="1.0" encoding="iso-8859-2"?>
<!DOCTYPE katalog SYSTEM "wypożyczalnia.dtd">
<katalog>
  <dvd id="1">
    <tytul>Rambo I</tytul>
    <aktorzy>
      <aktor>Sylvester Stallone</aktor>
    </aktorzy>
    <sztuk>2</sztuk>
  </dvd>
  ...
</katalog>
```

Przykład

Typy węzłów:

- `<katalog>` – węzeł dokumentu (root).
- `<tytul>` – węzeł element.
- `id` – węzeł atrybut.

Relacje pomiędzy węzłami

Rodzic – każdy element i atrybut ma jednego przodka.

Dziecko – każdy element ma zero lub więcej dzieci.

Rodzeństwo – węzły które mają tego samego rodzica.

Przodek – rodzic, rodzic rodzica, itd.

Potomek – dziecko, dziecko dziecka, itd.

Wybieranie węzła (węzłów)

XPath używa pojęcia ścieżki do określenia węzła lub grupy węzłów zgodnie z poniższymi regułami:

Wyrażenie	Znaczenie
nazwa węzła	Wybierz wszystkie dzieci węzła.
/	Wybierz węzeł root.
//	Wybierz węzły od aktualnego położenia pasujące do wyrażenia, nie ważne gdzie się znajdują.
.	Aktualny węzeł.
..	Wybierz rodzica aktualnego węzła.
@	Wybierz atrybut.

Wybieranie węzła - przykłady

Wyrażenie	Znaczenie
katalog	Wybierz wszystkie węzły dzieci węzła katalog.
/katalog	Wybierz węzeł root katalog.
katalog/dvd	Wybierz wszystkie węzły dvd, które są dziećmi węzła katalog
//dvd	Wybierz wszystkie węzły dvd, nie ważne gdzie się znajdują.
//@id	Wybierz wszystkie atrybuty id.

Szczegółowe określanie węzła

XPath posiada dodatkową konstrukcję pozwalającą na określenie konkretnego węzła lub jego wartości:

- `//katalog/dvd[1]`
- `//katalog/dvd[last()]`
- `//katalog/dvd[position()<3]`
- `//dvd[@id]`
- `//dvd[@id="1"]`
- `//dvd[sztuk>3]`

Oznaczanie nieznanych węzłów

Symbol	Znaczenie
*	Pasuje do każdego węzła.
*@	Pasuje do każdego atrybutu.
node()	Pasuje do węzła dowolnego typu.

Dodatkowo można użyć operatora “|” w celu wybrania kilku ścieżek (AND).

Przykład: XPath i Python

Skrypt Pythona odnajdujący zadany węzeł:

```
import libxml2
doc = libxml2.parseFile ("wypożyczalnia.xml")
path = doc.xpathEval ("//dvd")
for element in path:
    print element.content
```

XPath Axes

Axes	Znaczenie
ancestor	Wybierz wszystkich przodków aktualnego węzła.
ancestor-or-self	Wybierz wszystkich przodków oraz aktualny węzeł.
attribute	Wybierz wszystkie atrybuty aktualnego węzła.
child	Wybierz wszystkie dzieci aktualnego węzła.
descendant	Wybierz wszystkich potomków aktualnego węzła.
descendant-or-self	Wybierz wszystkich potomków aktualnego węzła i aktualny węzeł.

XPath Axes

Axes	Znaczenie
following	Wybierz wszystko co jest w dokumencie z znacznikiem zamknięcia aktualnego węzła.
following-sibling	Wybierz rodzeństwo za aktualnym węzłem.
namespace	Wybierz wszystkie przestrzenie nazw aktualnego węzła.
parent	Wybierz przodka aktualnego węzła.
preceding	Wybierz wszystko co znajduje się w dokumencie przed aktualnym węzłem.
preceding-sibling	Wybierz rodzeństwo przed aktualnym węzłem.
self	Wybierz siebie.

Wstęp

Początkowo W3C chciało zaprojektować język XSL (EXtensible Stylesheet Language) ponieważ istniało zapotrzebowanie na XML'owy odpowiednik arkusza styli. W ramach XSL powstały technologie:

- XSLT – język do transformacji dokumentów XML.
- XPath – język do nawigacji po dokumentach XML.
- XSL-FO – język do formatowania dokumentów XML.

Czym jest XSLT?

- XSLT oznacza XSL Transformations.
- XSLT jest najważniejszą częścią XSL.
- XSLT służy do transformacji dokumentu XML w dowolny inny dokument.
- XSLT używa XPath do nawigacji po dokumentach XML.
- XSLT jest rekomendowany przez W3C.

Najczęściej mówi się, że XSLT przekształca drzewo **źródłowe** XML w drzewo **wyników** XML.

Czym jest XSLT?

- XSLT oznacza XSL Transformations.
- XSLT jest najważniejszą częścią XSL.
- XSLT służy do transformacji dokumentu XML w dowolny inny dokument.
- XSLT używa XPath do nawigacji po dokumentach XML.
- XSLT jest rekomendowany przez W3C.

Najczęściej mówi się, że XSLT przekształca drzewo **źródłowe** XML w drzewo **wyników** XML.

Preambuła

Każdy dokument stylu XSLT musi zawierać w nagłówku:

```
<xsl:stylesheet version="1.0"  
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
```

lub

```
<xsl:transform version="1.0"  
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
```

Obydwie deklaracje **przestrzeni nazw** są synonimami.

Włączanie arkusza stylu XSL do XML

Aby dołączyć arkusz stylu XSL (transformację XSLT) należy w nagłówku dokumentu XML dopisać:

```
<?xml-stylesheet type="text/xsl"  
href="wypożyczalnia.xsl"?>
```

Struktura XSLT na przykładzie

Plik wypożyczalnia.xsl:

```
<?xml version="1.0" encoding="iso-8859-2"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/">
  <html>
    <body>
      <h1>wypożyczalnia DVD:</h1>
      <h2>Tytuł: </h2>
      <h3>Aktor: </h3>
    </body>
  </html>
</xsl:template>
</xsl:stylesheet>
```

Struktura XSLT na przykładzie

Plik wypożyczalnia.xsl:

```
<xsl:template match="/katalog/dvd">
  <html>
    <body>
      <h1>wypożyczalnia DVD:</h1>
      <h2>Tytuł: <xsl:value-of select="tytul"/></h2>
      <h3>Aktor: <xsl:value-of select="aktorzy"/></h3>

      </body>
    </html>
  </xsl:template>
```

Struktura XSLT na przykładzie

Plik wypożyczalnia.xsl:

```
<xsl:template match="/">
  <html>
    <body>
      <h1>wypożyczalnia DVD:</h1>
      <xsl:for-each select="katalog/dvd">
        <h2>Tytuł: <xsl:value-of select="tytul"/>h2>
        <h3>Aktor: <xsl:value-of
          select="aktorzy"/></h3>
      </xsl:for-each>
    </body>
  </html>
</xsl:template>
```

Dodatkowe instrukcje XSLT

Oprócz pokazanych na przykładzie instrukcji możliwe jest umieszczenie w arkuszu XSLT dodatkowych instrukcji sterujących:

```
<xsl:sort select="tytuł"/>
```

```
<xsl:if test="sztuk > 10">  
</xsl:if>
```

```
<xsl:choose>  
  <xsl:when test="sztuk > 10">  
    ...  
  </xsl:when>  
  <xsl:otherwise>  
    ...  
  </xsl:otherwise>  
</xsl:choose>
```

Dodatkowe instrukcje XSLT

Oprócz pokazanych na przykładzie instrukcji możliwe jest umieszczenie w arkuszu XSLT dodatkowych instrukcji sterujących:

```
<xsl:sort select="tytul"/>
```

```
<xsl:if test="sztuk > 10">  
</xsl:if>
```

```
<xsl:choose>  
  <xsl:when test="sztuk > 10">  
    ...  
  </xsl:when>  
  <xsl:otherwise>  
    ...  
  </xsl:otherwise>  
</xsl:choose>
```

Dodatkowe instrukcje XSLT

Oprócz pokazanych na przykładzie instrukcji możliwe jest umieszczenie w arkuszu XSLT dodatkowych instrukcji sterujących:

```
<xsl:sort select="tytul"/>
```

```
<xsl:if test="sztuk > 10">  
</xsl:if>
```

```
<xsl:choose>  
  <xsl:when test="sztuk > 10">  
    ...  
  </xsl:when>  
  <xsl:otherwise>  
    ...  
  </xsl:otherwise>  
</xsl:choose>
```