

Programowanie komunikacji człowiek-komputer

dr inż. Joanna Ochelska-Mierzejewska

XPointer

XPointer

- Standard definiuje adresowanie dokumentów XML-owych i ich fragmentów, zgodnie ze składnią URI
 - `http://www.sejm.gov.pl/ustawa.xml#def-podatnik`
- Rekomendacje W3C z 2003 r.
 - XPointer Framework (`http://www.w3.org/TR/xpfr-framework/`)
 - XPointer element() Scheme (`http://www.w3.org/TR/xpfr-element/`)
 - XPointer xmlns() Scheme (`http://www.w3.org/TR/xpfr-xmlns/`)
 - XPointer xpointer() Scheme (`http://www.w3.org/TR/xpfr-xpointer/`, Working Draft)

Schematy xpointer, xmlns i element

xpointer

• ścieżki XPath:
`http://www.sejm.gov.pl/ustawa.xml#xpointer(/art[5]/par[2])`

xmlns

• Obsługa przestrzeni nazw
• Do wykorzystania w dalszej części ścieżki
`ustawa.xml#xmlns(pr=http://www.sejm.gov.pl/prawo)xpointer(/art[5]/par[2])`

Schematy xpointer, xmlns i element

element

Przykłady:

- element z atrybutem ID o podanej wartości:
`dokument.xml#element(def-podatnik)`
- element o podanej pozycji (ewentualnie względem elementu z atrybutem ID o podanej wartości):
`dokument.xml#element(/1/4/3)`
`dokument.xml#element(def-podatnik/2/3)`
- składnia skrócona:
`dokument.xml#def-podatnik`
`dokument.xml#/1/2/3`
`dokument.xml#def-podatnik/2/3`

XInclude

7

XInclude – włączanie zawartości dokumentów

- Cel: włączanie zawartości jednego dokumentu XML do innego
- Rekomendacja W3C XML Inclusions (XInclude) 1.0 z 2006 r.
- Możliwości
 - włączanie całych dokumentów (nazwa pliku lub URL)
 - włączanie fragmentów (wskazanych przez XPath)
 - określanie zawartości używanej w razie błędu

<http://www.w3.org/TR/xinclude/>

8

XInclude – przykład

```

<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<wynik>
  <xi:include
    xmlns:xi="http://www.w3.org/2001/XInclude"
    href="salatka.xml#pointer(/przepis/titul)" ?>
  <xi:fallback>
    <bład>Brak przepisu</bład>
  </xi:fallback>
</xi:include>
</wynik>

```

```

Wynik przetworzenia
<wynik>
  <tytul>Salatka z ogórków</tytul>
</wynik>

```

9

Typowe zastosowania XML

10

Popularne zastosowania XML

- zapis tekstu
 - XML DocBook
 - TEI – Text Encoding Initiative
 - ODF (OpenDocument)
 - Office Open XML
- zapis metainformacji (metadanych)
 - Dublin Core
 - RDF – Resource Description Framework
 - Topic Maps
- zapis informacji naukowych
 - MathML – Mathematical Markup Language
- multimedia
 - SVG – Scalable Vector Graphics

11

Zapis tekstu – XML DocBook

12

Zapis tekstu – DocBook

- Przeznaczony do kodowania dokumentacji oprogramowania
 - struktura książki i artykułu
 - dodatkowe struktury specyficzne dla dokumentacji oprogramowania
- łatwy do dostosowania do własnych potrzeb
 - silnie sparymetryzowane DTD (możliwość włączania poszczególnych elementów struktury)
 - większość zmian można uzyskać bez modyfikowania DTD
- Prace od 1991 r., obecna wersja: 5.0, zarządzana przez OASIS
- Przyjęty z otwartymi ramionami przez środowisko open-source i używany do tworzenia dokumentacji wielu projektów (KDE, GNOME, dokumentacja jądra Linuksa, ...)
- Z racji wieku standardu istnieje wiele narzędzi do jego obsługi (edytory WYSIWYG, przekształcenia XSLT, ...)

<http://www.docbook.org>

13

DocBook – przykład

Element główny <book> (książka), oprócz tego rozdziały, sekcje, streszczenia, wycieczki historyczne...

```
<book xmlns:id="katalog" version="5.0"
  xmlns="http://docbook.org/ns/docbook">
  <title>Świat Zofii</title>
  <chapter xml:id="rozdzial1">
    <title>Ogród Eden</title>
    <para>Komu 3000 lat nie mówia nic, niech w ciemności niewiedzy
      żyje z dnia na dzień.
    </para>
    <para>--- <emphasis>Goethe</emphasis></para>
  </chapter>
  <chapter xml:id="rozdzial2">
    <title>Cylinder</title>
    <para>Zdziwienie jest tą przyczyną, dla której ludzie zaczęli
      filozofować.
    </para>
  </chapter>
</book>
```

14

Zapis tekstu – Text Encoding Initiative

TEI

15

Zapis tekstu – TEI

- TEI (ang. *Text Encoding Initiative*) – standard zapisu danych humanistycznych i lingwistycznych
 - prace od 1987 r.
 - początkowo wersje SGML-owe, obecnie XML-owa
 - obecna wersja: P5 z 2007 r.
 - schemat zawiera moduły (zestawy znaczników) włączone w zależności od potrzeb

16

TEI – przykład

```
<TEI xmlns="http://www.tei-c.org/ns/1.0">
  <teiHeader> ... </teiHeader>
  <text>
    <body>
      <schemaSpec ident="odd1" start="TEI">
        <moduleRef key="header"/>
        <moduleRef key="core"/>
        <moduleRef key="tei"/>
        <moduleRef key="textstructure"/>
      </schemaSpec>
    </body>
    ...
  </text>
</TEI>
```

17

TEI – przykład minimalnego nagłówka

```
<teiHeader>
  <fileDesc>
    <titleStmt>
      <title>Opowiadania "Złoty żuk" i "Wahadło", wersja elektroniczna
      </title>
      <author>Poe, Edgar Allan</author>
      <respStmt>
        <resp>wyboru dokonał</resp>
        <name>James D. Benson</name>
      </respStmt>
    </titleStmt>
    <publicationStmt>
      <distributor>Oxford Text Archive</distributor>
    </publicationStmt>
    <sourceDesc>
      <bibl>The complete writings of Edgar Allan Poe, collected and edited by
        Phillip S. Foner (New York, Citadel Press, 1945)
      </bibl>
    </sourceDesc>
  </fileDesc>
</teiHeader>
```

18

Zapis tekstu – OpenDocument

ODF

19

Zapis tekstu - OpenDocument

- **OpenDocument** (ODF, OASIS Open Document Format for Office Applications, ISO/IEC 26300) to otwarty standard formatu plików pakietów biurowych
 - Zakres: dokumenty tekstowe, arkusze kalkulacyjne, wykresy i prezentacje multimedialne
 - Historia
 - prace od 2002 r.; pierwsi twórcy formatu bazowali na XML-owym formacie pakietu OpenOffice.org, obecnie jest rozwijana przez OASIS (Organization for the Advancement of Structured Information Standards)
 - standard ISO od 2006 r.
 - Cel: stworzenie otwartej alternatywy dla formatów pakietu Microsoft Office

20

ODF? Kto tego używa?

- Programy obsługujące OpenDocument
 - OpenOffice.org,
 - KOffice,
 - Google Docs,
 - LotusNotes R8...
- Prezentowany jako alternatywa dla formatów uchodzących za otwarte (np. Office Open XML), lecz nie powstałych na drodze porozumienia pomiędzy dostawcami oprogramowania biurowego oraz użytkownikami (w tym rządami państw)
- Unia Europejska i MSWiA rekomendują OpenDocument jako bazę dla standardowych formatów plików i wymiany dokumentów

<http://docs.oasis-open.org/office/v1.1/OS/OpenDocument-v1.1.pdf>

21

ODF od środka

- Dwa sposoby zapisu dokumentu
 - w pojedynczym dokumencie XML-owym (<office:document>)
 - w pakiecie zawierającym pliki
 - content.xml: treść dokumentu (<office:document-content>)
 - styles.xml: style (<office:document-styles>)
 - meta.xml: metadane (<office:document-meta>)
 - settings.xml: dane aplikacji (<office:document-settings>)

22

Zestaw znaczników ODF

- office – dla elementów ogólnych, nie należących do innych przestrzeni nazw
- meta – dla metadanych
- config – dla ustawień aplikacji
- text – dla elementów opisujących fragmenty tekstu (także np. zawartość komórki arkusza kalkulacyjnego)
- table – dla elementów tabelarycznych
- drawing
- presentation
- form
- script
- style
- ...

23

ODF – przykład

Przykład metadanych

```
<office:meta>
  <dc:title>Dlaczego dzieciolki nie cierpią na wstrząs mózgu?</dc:title>
  <dc:creator>Philip May</dc:creator>
  <dc:date>1999-10-19T11:16:17</dc:date>
  <meta:editing-duration>PT5H10M10S</meta:editing-duration>
  <meta:keyword>Dzieciolki</meta:keyword>
  <meta:keyword>Wstrząśnienie mózgu</meta:keyword>
  <meta:user-defined meta:name="źródło">
    <meta:value-type="string">IgNoble 2006</meta:value>
  </meta:user-defined>
</office:meta>
```

24

ODF – przykład

Przykład zawartości dokumentu

```
<office:document-content xmlns:office="..." .. office:version="1.0">
  <office:automatic-styles>
    <style:style style:name="P" style:family="paragraph">
      <style:paragraph-properties fo:text-align="center">
        <style:text-properties style:name="Arial" fo:font-size="18pt"
          fo:font-weight="bold"/>
      </style:style>
    </office:automatic-styles>
    <office:body>
      <office:text>
        <text:p text:style-name="P">Format OpenDocument jest standardem
          <text:a xlink:type="simple" xlink:href="http://oasis-open.org">OASIS</text:a>
        </text:p>
      </office:text>
    </office:body>
```

25

Zapis tekstu – Office Open XML

26

Zapis tekstu – Office Open XML

- Office Open XML (OOXML, OpenXML, MSOOXML) to alternatywa do OpenDocument – specyfikacja dokumentów elektronicznych mająca zapewnić wsteczną zgodność z istniejącymi dokumentami w formatach Microsoft Office oraz pełną obsługę funkcji oferowanych przez ten pakiet
- Historia
 - 2006: specyfikacja zatwierdzona jako standard ECMA International
 - 2007: porażka w głosowaniu nad szybką ścieżką zatwierdzenia OOXML jako standardu ISO
 - 2008: standard jednak przyjęty

<http://www.ecma-international.org/publications/standards/Ecma-376.htm>

27

OOXML w pigułce

- Struktura pliku
 - dokument OOXML to pakiet (skompresowany ZIP)
 - wewnątrz: pliki XML-owe zawierające dane, metadane, ... zgodne z językami znacznikovymi
 - WordprocessingML (word/document.xml, word/styles.xml...)
 - SpreadsheetML
 - PresentationML
 - ...
 - kompatybilne z binarnymi formatami MS Office w zakresie reprezentowalności ich możliwości
 - niekompatybilne z XML-owymi formatami Office XP i 2003

28

WordProcessingML

- Typowa struktura dokumentu
 - <w:document> – element główny
 - <w:body> – kolekcja akapitów i własności sekcji
 - <w:p> – akapit (kolekcja fragmentów)
 - <w:t> – fragment tekstu o określonych własnościach (ciąg, ang. run)
 - <w:t> – czysty tekst (ang. text range)

```
<w:document xmlns:w="...">
  <w:body>
    <w:p>
      <w:t>
        <w:t>Hello, world.</w:t>
      </w:t>
    </w:p>
  </w:body>
</w:document>
```

```
<w:p>
  <w:wp>
    <w:js w:val="content"/>
    <w:rPr>
      <w:color/>
    </w:rPr>
    <w:t>
      <w:ff>
        <w:lb/>
      </w:ff>
      <w:tnaprawiac/>
    </w:t>
    <w:st/>
  </w:wp>
```

29

OOXML kontra OpenDocument: wojna standardów

30

OOXML kontra OpenDocument: wojna standardów

- Fanacy ODF
 - Wasza specyfikacja ma 6000 stron! Ciężko będzie to wdrożyć w całości...
- Wielbiciele OOXML
 - Ale tam są przykłady i dokładne opisy, a w ODF-ie tylko spis znaczników
 - Poza tym musimy zachować kompatybilność z dokumentami Office, bo jest ich dużo, czego nie można powiedzieć o ODF...
- Fanacy ODF
 - W ODF-ie można reprezentować to samo, co w OOXML-u
 - Więcej i lepiej, bo wasza specyfikacja zawiera błędy. Dlaczego rok 1900 jest przestępny?

31

OOXML kontra OpenDocument: wojna standardów

- Wielbiciele OOXML
 - A ODF jest prostaki w porównaniu z OOXML-em! Używanie jednego typu tabeli dla wszystkich rodzajów dokumentów...
- Fanatycy ODF
 - Nie jest prostaki, tylko prosty, i elegancki w swej prostocie
 - Gdybyście nam pomogli, też byście tak mieli
 - MS był członkiem OASIS, kiedy powstał ODF. Dlaczego robiliście swoje pod stołem?
- Wielbiciele OOXML
 - Ba chcieliście dać przewagę OpenOffice. Nie dalibyśmy rady tego wdrożyć — w 2005 r.
 - OpenOffice już używał ODF-u, a my dopiero zaczynaliśmy

32

OOXML kontra OpenDocument: wojna standardów

- Fanatycy ODF
 - Eeee lam, przecież pracowaliśmy nad tym już od 2002 r.
 - Poza tym byliśmy pierwsi, a skoro jest już jeden standard, to dlaczego go nie wykarzydocić?
 - Wy w ogóle nie lubicie standardów, bo ODF używa MathML, SVG, a wy wszystko po swojemu, nawet kody języków macie inne niż cały świat...
 - A co z prawami autorskimi i patentowymi?
- Wielbiciele OOXML
 - Grrrr!!!

33

Zapis metainformacji (metadanych) – Dublin Core

34

Zapis metainformacji – Dublin Core

- Standard reprezentacji metadanych do opisu zasobów (ISO 15836)
- Utrzymywany przez DCM (Dublin Core Metadata Initiative) – organizację promującą stosowanie standardów metadanych i rozwój specjalizowanych słowników pojęć do opisu zasobów

<http://dublincore.org>

35

Dublin Core

- Standard definiuje 15 elementów
 - <dc:title> – tytuł/nazwa opisywanego zasobu
 - <dc:creator> – twórca zasobu
 - <dc:subject> – temat poruszanego zagadnienia
 - <dc:description> – opis
 - <dc:publisher> – wydawca (podmiot odpowiedzialny za udostępnienie)
 - <dc:contributor> – współtwórca
 - <dc:date> – data wydarzenia z życia zasobu
 - <dc:type> – typ/rodzaj zawartości zasobu
 - <dc:format> – format (fizyczny lub cyfrowy)
 - <dc:identifier> – identyfikator zasobu
 - <dc:source> – źródło pochodzenia informacji zawartych w zasobie
 - <dc:language> – język zawartości
 - <dc:relation> – odniesienie do powiązanych zasobów
 - <dc:coverage> – dziedzina/obszar objęty zawartością zasobu
 - <dc:rights> – informacja o prawach do zasobu

36

Dublin Core – przykład

```

RDF
<rdf:Description>
  <dc:title>Wartość energetyczna napojów</dc:title>
  <dc:publisher>Jurgen Lager</dc:publisher>
  <dc:description>Czy lepiej dostać w głowę pustą czy pełną butelkę piwa?
</dc:description>
  <dc:date>2009-06-01</dc:date>
  <dc:language>pl</dc:language>
</rdf:Description>

HTML
<meta name="DC.title" lang="pl" content="Wartość energetyczna napojów">
<meta name="DC.creator" content="Jan Byczewski">
<meta name="DC.description" lang="pl" content="Czy lepiej dostać w głowę pełną czy pustą butelkę piwa?">

```

37

Zapis metainformacji (metadanych) – Resource Description Framework (RDF)

38

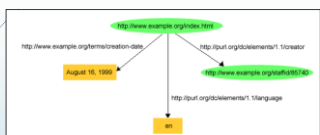
Resource Description Framework (RDF)

- RDF – konkurencyjna (W3C, rekomendacja w 1999 r.) metoda definiowania wiedzy poprzez opis zasobów
- Reprezentacja wiedzy w RDF
 - zdania logiczne w postaci trójki (ang. *triple*) „podmiot-relacja-przedmiot” (np. „<Stanisław Lem>-<jest-autorem>-<Solaris>”)
 - podmiot i przedmiot są zasobami
 - relacja (własność, ang. *property*) może być zasobem
 - skoro własność jest zasobem, można ją opisać inną własnością, czego wynikiem może być zawiązany metagraf (węzły = zasoby, kłki = własności)
 - rodzaje własności są nieograniczone
- Specyfikacja RDF definiuje sposób serializacji grafu do XML-a (RDF/XML)
- Zasoby identyfikowane są (oczywiście) poprzez URI

<http://www.w3.org/RDF>
<http://www.w3.org/1999/02/rdf-primer/>

39

RDF – graf i jego serializacja



Notacja R3

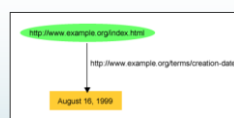
```
<http://www.example.org/index.html> <http://purl.org/dc/elements/1.1/creator>
<http://www.example.org/staff/85740> .
<http://www.example.org/index.html>
<http://www.example.org/terms/creation-date> "August 16, 1999" .
<http://www.example.org/index.html>
<http://purl.org/dc/elements/1.1/language> "en" .
```

Jeszcze prościej

```
ex:index.html dc:creator ex:staff:85740 .
ex:index.html ex:terms:creation-date "August 16, 1999" .
ex:index.html dc:language "en" .
```

40

RDF/XML



Trójki

```
ex:index.html ex:terms:creation-date "August 16, 1999" .
```

RDF/XML

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns"
xmlns:ex:terms="http://www.example.org/terms/"
>
<rdf:Description rdf:about="http://www.example.org/index.html">
  <ex:terms:creation-date>August 16, 1999</ex:terms:creation-date>
</rdf:Description>
</rdf:RDF>
```

41

RDQL (RDF Data Query Language)

- RDQL – język zapytań wzorowany na SQL

Zapytanie

```
SELECT ?x, ?fname
WHERE { ?x, <http://www.w3.org/2001/vcard-rdf/3.0FN>, ?fname }
```

Wynik

x	fname
<http://somewhere/JohnSmith/>	"John Smith"
<http://somewhere/RebeccaSmith/>	"Becky Smith"
<http://somewhere/SarahJones/>	"Sarah Jones"
<http://somewhere/MattJones/>	"Matt Jones"

42

Zapis metainformacji (metadanych) – Topic Maps

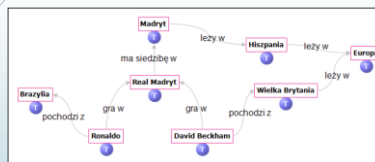
43

Zarządzanie wiedzą

- Co do tej pory mogło kojarzyć nam się z zarządzaniem wiedzą?
 - moduł synonimów w wyszukiwarce (szukam „błękitnej bluzeczki”, znajduje się „niebieski sweterek”)?
 - model podobieństw (szukam Toyoty Yaris, znajduje się Opel Corsa)?
- Co można rozumieć jako zarządzanie wiedzą?
 - dobrze praktyki zarządzania przedsiębiorstwem?
 - rozwiązania technologiczne usprawniające pracę (CMS, portal korporacyjny, ...)?
 - różne obszary zainteresowań sztucznej inteligencji (automatyczne wnioskowanie, uczenie maszynowe, systemy eksperckie)?

44

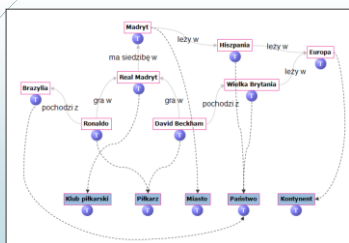
Intuicyjny model wiedzy – siatka pojęć



- Niektóre problemy
 - plaski model: pojęcie „Europa” jest słabo odróżnialne od pojęcia „Ronaldo”
 - siatka może się rozrastać w niekontrolowany sposób!

45

Klasy pojęć



46

Klasy + relacje = schemat mapy wiedzy

- Relacje między klasami są abstrakcją relacji pomiędzy pojęciami



- klasy + relacje = schemat mapy wiedzy (ontologia)

47

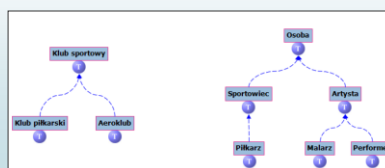
Schemat mapy wiedzy

- Po co tworzyć schematy?
 - aby wyrazić strukturę informacji i współdzielić jej rozumienie pomiędzy ludźmi lub automatami (→ łatwe zbieranie danych, tworzenie podsumowań itp.)
 - aby mieć możliwość wielokrotnego wykorzystania spójnych „paczek wiedzy”
 - aby dokonać analizy wiedzy danej dziedziny w interesującym nas aspekcie
- Uwaga
 - nie ma jednego właściwego schematu dla danej dziedziny wiedzy!
 - to sposób wykorzystania wiedzy wpływa na schemat i stopień jego szczegółowości

48

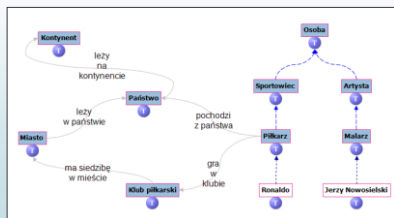
Hierarchia klas

- Klasy możemy hierarchizować



49

Mapa wiedzy = schemat mapy wiedzy + instance



50

Korzyści z użycia mapy wiedzy

- Użyteczność mapy wiedzy
 - wyszukiwanie
 - konfrontacja zapytania z modelem wiedzy, możliwość uszczegóławiania zapytań na podstawie modelu wiedzy
 - zawężanie zakresu poszukiwań na podstawie pojęć wybranych z mapy
 - ułatwiona klasyfikacja: dokonywanie dokumentów do mapy wiedzy na podstawie przekrojów mapy
 - unikana nawigacja: dostęp do dokumentów poprzez sieć pojęć

51

XML Topic Maps

- ISO 13250:2003 – standard reprezentacji wymiany wiedzy
- Pomysł
 - utworzenie nad warstwą zasobów warstwy abstrakcyjnych **pojęć** (tematów, ang. topics) z możliwością tworzenia **powiązań** (ang. associations) między nimi
 - powiązanie obu warstw poprzez **wystąpienia** (ang. occurrences) pojęć w zasobach
- Najpopularniejsza notacja: XML Topic Maps (XTM) 2.0 z 2006 r.

52

XML Topics Maps – wybrane elementy

- <topicMap> – korzeń dokumentu z definicją mapy pojęć
- <topic> – nazwa i lista wystąpień pojęcia
- <instanceOf> – informacja o powiązaniu pojęcia z klasą (pojęciem nadrzędnym); występuje w treści <topic>
- <topicRef> – odwołanie do już zdefiniowanego pojęcia (np. w celu określenia klasy)
- <occurrence> – informacja o wystąpieniu pojęcia
- <resourceRef> – odwołanie do zasobu (za pomocą URI)
- <association> – powiązanie między pojęciami
- ...

<http://www.topicmaps.org/xtm>

53

XML Topic Maps – przykład

```
<?xml version="1.0" encoding="UTF-8"?>
<topicMap>
  <topic id="kompozytor">
    <baseName>kompozytor</baseName>
  </topic>
  <topic id="chopin">
    <instanceOf>kompozytor</instanceOf>
    <baseName> Fryderyk Chopin </baseName>
    <occurrence>resourceRef xlink:href="http://www.example.org/chopin.htm"</occurrence>
  </topic>
  <topic id="polska">
    <instanceOf>kraje</instanceOf>
  </topic>
  <association>
    <topicRef xlink:href="#kompozytor"></topicRef>
    <member roleSpec="kompozytor">
      <topicRef xlink:href="#chopin"></topicRef>
    </member>
    <member roleSpec="polska">
      <topicRef xlink:href="#polska"></topicRef>
    </member>
  </association>
</topicMap>
```

54

Zapis zapis informacji naukowych – MathML – Mathematical Markup Language

55

Zapis informacji naukowych – MathML

- MathML (*Mathematical Markup Language*) to język służący do zapisu wzorów i symboli matematycznych
- Aktualna wersja – 3.0
- Dwa dialekty MathML-a (można łączyć)
 - składnia prezentacyjna (ang. *presentational markup*) – 30 elementów, 50 atrybutów
 - składnia semantyczna (ang. *content markup*) – ok. 100 elementów, 12 atrybutów

<http://www.w3.org/Math/>
<http://www.dessci.com/en/reference/mathml/default.htm>

56

Math – przykład

$$(a+b)^2$$

Składnia prezentacyjna

```
<mrow>
  <msup>
    <mfenced>
      <mrow>
        <mi>a</mi>
        <mo>+</mo>
        <mi>b</mi>
      </mrow>
    </mfenced>
    <mn>2</mn>
  </msup>
</mrow>
```

Składnia semantyczna

```
<mrow>
  <apply>
    <power/>
    <apply>
      <plus/>
      <ci>a</ci>
      <ci>b</ci>
    </apply>
    <cn>2</cn>
  </apply>
</mrow>
```

57

Multimedia – SVG – Scalable Vector Graphics

58

Multimedia – SVG

- Pozwala opisywać dwuwymiarową grafikę
 - kształty wektorowe (złożone z linii prostych i krzywych)
 - obrazy rastrowe
 - tekst (który jest wciągany tekstem, nie zbiorem kształtów)
 - grupowanie, transformacje, filtry, ...
- Aktualna wersja: 1.1 z 2003 r.

```
<svg>
  <circle style="fill: blue; stroke: red" cx="200" cy="200"
    r="100"/>
</svg>
```

<http://www.w3.org/Graphics/SVG/>