Narzędzia informatyczne w językoznawstwie

HTML i XHTML

Marcin Junczys-Dowmunt
junczys@amu.edu.pl

Zakład Logiki Stosowanej
http://www.logic.amu.edu.pl

17. października 2007

Po co językoznawcom (X)HTML?

- Prawie cała widzialna część internetu opiera się na HTML i jego odmianach

Po co językoznawcom (X)HTML?

- Prawie cała widzialna część internetu opiera się na HTML i jego odmianach
- Wniosek z tego: prawie wszystkie dane lingwistyczne pochodzące z internetu będą osadzone w HTML

Po co językoznawcom (X)HTML?

- Prawie cała widzialna część internetu opiera się na HTML i jego odmianach
- Wniosek z tego: prawie wszystkie dane lingwistyczne pochodzące z internetu będą osadzone w HTML
- Jeśli chcemy w jakikolwiek sposób przetwarzać informacje z internetu, nie obejdzie się bez podstawowych znajomości HTML (dlatego HTML dla językoznawców)

Po co językoznawcom (X)HTML?

- Prawie cała widzialna część internetu opiera się na HTML i jego odmianach
- Wniosek z tego: prawie wszystkie dane lingwistyczne pochodzące z internetu będą osadzone w HTML
- Jeśli chcemy w jakikolwiek sposób przetwarzać informacje z internetu, nie obejdzie się bez podstawowych znajomości HTML (dlatego HTML dla językoznawców)
- Jeśli mamy zamiar umieszczać własne treści w internecie, to lepiej korzystać z XHTML niż HTML.
- Będziemy wtedy tworzyć strony bardziej przystosowane do przetwarzania automatycznego i świecić dobrym przykładem (dlatego XHTML)

Ale co to właściwie HTML?

Wikipedia (EN)

- **HTML** (ang. *HyperText Markup Language*, pl. *hipertekstowy język znaczników*) jest dominującym językiem dla stron internetowych

Ale co to właściwie HTML?

Wikipedia (EN)

- **HTML** (ang. *HyperText Markup Language*, pl. *hipertekstowy język znaczników*) jest dominującym językiem dla stron internetowych
- Służy do opisu struktury informacji tekstowych w dokumencie
  - oznacza wybrane części tekstu jako nagłówki, akapity, listy itp.
  - wzbogaca tekst o formularze, obrazki i inne obiekty
  - kojarzy ze sobą dokumenty powiązane tematycznie (odsyłacze)

Ale co to właściwie HTML?

Wikipedia (EN)

- **HTML** (ang. *HyperText Markup Language*, pl. *hipertekstowy język znaczników*) jest dominującym językiem dla stron internetowych
- Służy do opisu struktury informacji tekstowych w dokumencie
  - oznacza wybrane części tekstu jako nagłówki, akapity, listy itp.
  - wzbogaca tekst o formularze, obrazki i inne obiekty
  - kojarzy ze sobą dokumenty powiązane tematycznie (odsyłacze)
- HTML jest zapisywany jako zwykły tekst za pomocą znaczników otoczonych ostrymi nawiasami

Ale co to właściwie HTML?

Wikipedia (EN)

- **HTML** (ang. *HyperText Markup Language*, pl. *hipertekstowy język znaczników*) jest dominującym językiem dla stron internetowych
- Służy do opisu struktury informacji tekstowych w dokumencie
  - oznacza wybrane części tekstu jako nagłówki, akapity, listy itp.
  - wzbogaca tekst o formularze, obrazki i inne obiekty
  - kojarzy ze sobą dokumenty powiązane tematycznie (odsyłacze)
- HTML jest zapisywany jako zwykły tekst za pomocą znaczników otoczonych ostrymi nawiasami
- Nazwa HTML jest nieraz stosowana jako hiperonim dla wszystkich innych pokrewnych formalizmów, w tym XHTML

Historia HTML

1989 Tim Berners-Lee (CERN) rozwija pierwszy internetowy system hipertekstowy

1990 Powstaje W3C (World Wide Web Consortium)

1993 Specyfikacja SGML

1995 Pierwsza oficjalna wersja: HTML 2

01 1997 HTML 3.2 próba uwzględnienia konsekwencji *wojny przeglądarek*

12 1997 HTML 4.0 pierwsze czystki

1999 HTML 4.01 jak na razie ostatnia wersja HTML

2000 XHTML 1.0 czyli uzgodnienie HTML 4.01 z XML

2001 XHTML 1.1 ostatnia oficjalna wersja

Struktura dokumentu HTML (1)

Znaczniki HTML składają się rożnych rodzajów jednostek, najważniejsze to:

Struktura dokumentu HTML (1)

Znaczniki HTML składają się rożnych rodzajów jednostek, najważniejsze to:

- elementy (główne znaczniki)

Struktura dokumentu HTML (1)

Znaczniki HTML składają się rożnych rodzajów jednostek, najważniejsze to:

- elementy (główne znaczniki)
- atrybuty (metadane dot. znaczników)

Struktura dokumentu HTML (1)

Znaczniki HTML składają się rożnych rodzajów jednostek, najważniejsze to:

- elementy (główne znaczniki)
- atrybuty (metadane dot. znaczników)
- dane tekstowe (tekst na stronie)

Struktura dokumentu HTML (1)

Znaczniki HTML składają się rożnych rodzajów jednostek, najważniejsze to:

- elementy (główne znaczniki)
- atrybuty (metadane dot. znaczników)
- dane tekstowe (tekst na stronie)
- encje (znaki szczególne, np. jawne spacje)

Struktura dokumentu HTML (2)

Elementy zwykle składają się z trzech częsci:

- znacznik początkowy (w postaci `<znacznik>`)
- zawartości elementu (tekst lub inne elementy)
- znacznik końcowy (w postaci `</znacznik>`)

Struktura dokumentu HTML (2)

Elementy zwykle składają się z trzech częsci:

- znacznik początkowy (w postaci `<znacznik>`)
- zawartości elementu (tekst lub inne elementy)
- znacznik końcowy (w postaci `</znacznik>`)

Niektóre elementy można opisać dokładniej za pomocą atrybutów

- atrybuty umieszczamy w znaczniku początkowym (np. `<znacznik atrybut1="wartość1" atrybut2="wartość2" ... atrybutN="wartośćN">`)
- znaczniki końcowe raczej nie mogą zwierać atrybutów

Struktura dokumentu HTML (3)

Ogólna struktura dokumentu HTML:

- element główny każdego dokumentu HTML to `html`
- element główny zawiera dwa kolejne elementy:
  - `head` (nagłówek dokumentu)
  - `body` (treść dokumentu)
- na początku dokumentu *powinna* się znaleźć informacja o typie dokumentu

```
<!DOCTYPE html PUBLIC ... >
<html>
  <head> ... </head>
  <body> ... </body>
</html>
```

Mały przykład dokumentu HTML

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html401/loose.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type"
    content="text/html; charset=UTF-8">
    <title>Tytuł dokumentu</title>
  </head>
  <body background="white" text="black">
    <h1>To jest dokument <font color="red">HTML</font></h1>
    <p>A tutaj mamy jakiś przykładowy akapit, który służy
    jedynie celom poznawczym. <br>
    <a href="http://www.ij.amu.edu.pl">link do strony
    instytutu</a>
    <p> To drugi akapit, który zawiera kilka tzw.
    <i>encji</i>: <br> \&amp; \  \&Ouml; &szlig;
  </body>
</html>
```

XHMTL

- XHMTL to eXtensible HyperText Markup Language

[1] Każdy standardowy parser XML poradzi sobie z XHTML ale niekoniecznie z HTML

[2] To akurat zależy niestety w znacznym stopniu od przeglądarki

XHMTL

- XHMTL to eXtensible HyperText Markup Language
- Jest rozwinięciem standardu HTML 4.01
- Można powiedzieć, że jest przecięciem HTML 4.01 i XML (oba języki są podzbiorami SGML)

[1] Każdy standardowy parser XML poradzi sobie z XHTML ale niekoniecznie z HTML

[2] To akurat zależy niestety w znacznym stopniu od przeglądarki

XHMTL

- XHMTL to eXtensible HyperText Markup Language
- Jest rozwinięciem standardu HTML 4.01
- Można powiedzieć, że jest przecięciem HTML 4.01 i XML (oba języki są podzbiorami SGML)
- XHTML jest lepszy pod względem automatycznego przetwarzania[1] (bardziej rygorystyczna składnia)

[1] Każdy standardowy parser XML poradzi sobie z XHTML ale niekoniecznie z HTML

[2] To akurat zależy niestety w znacznym stopniu od przeglądarki

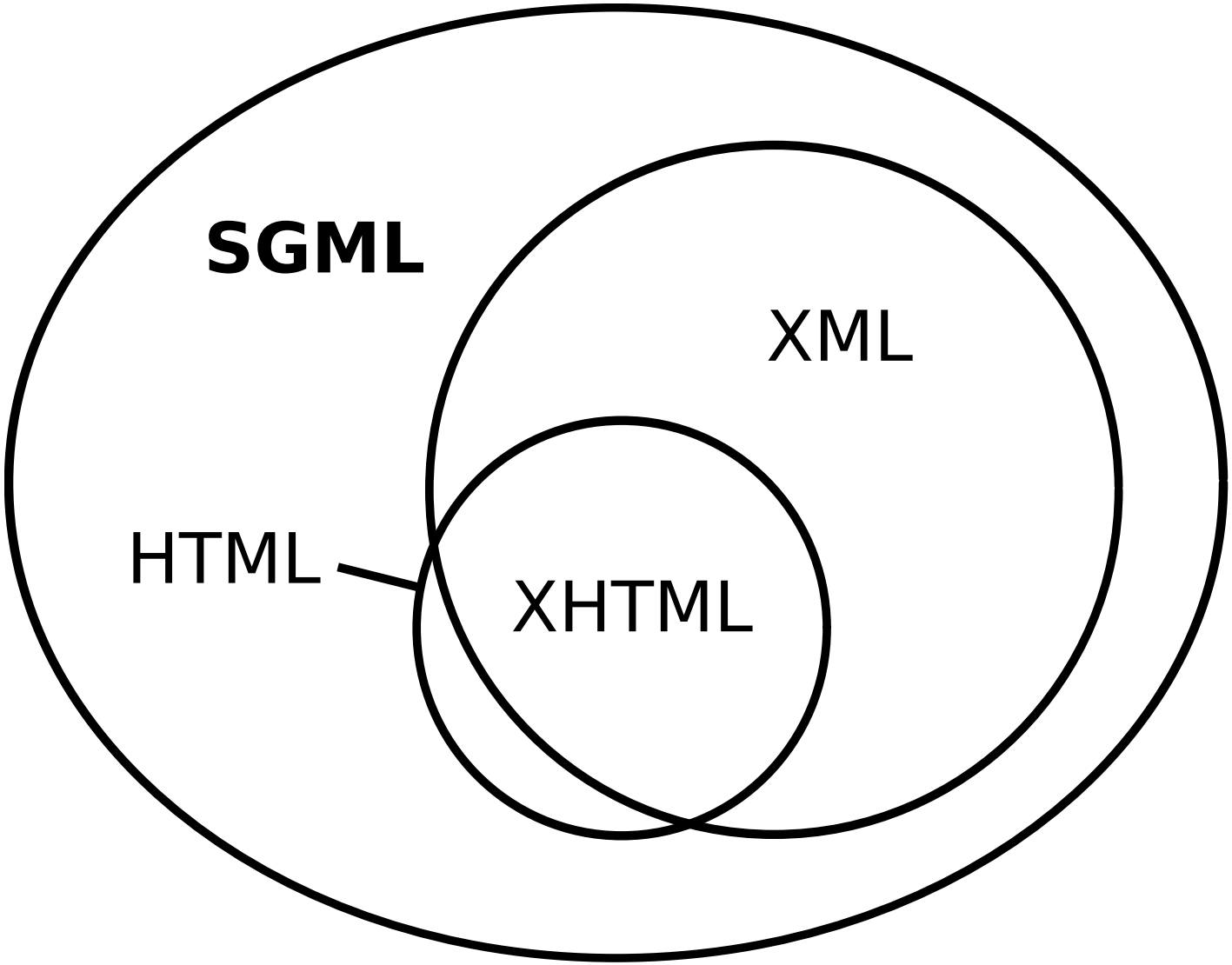
XHMTL

- XHMTL to eXtensible HyperText Markup Language
- Jest rozwinięciem standardu HTML 4.01
- Można powiedzieć, że jest przecięciem HTML 4.01 i XML (oba języki są podzbiorami SGML)
- XHTML jest lepszy pod względem automatycznego przetwarzania[1] (bardziej rygorystyczna składnia)
- Pozwala na korzystanie z różnych rozszerzeń XML, np. MathML, SVG itp. [2]

[1] Każdy standardowy parser XML poradzi sobie z XHTML ale niekoniecznie z HTML

[2] To akurat zależy niestety w znacznym stopniu od przeglądarki

HTML, XHTML i XML



Aspekt przetwarzania XHTML

- Przeglądarki przetwarzające HTML wewnętrznie korygują błedy i niedociągniecia

Aspekt przetwarzania XHTML

- Przeglądarki przetwarzające HTML wewnętrznie korygują błedy i niedociągniecia
- XHTML wymaga pełnej zgodności ze specyfikacją XML, inaczej nie jest możliwe wyświetlenie dokumentu

Aspekt przetwarzania XHTML

- Przeglądarki przetwarzające HTML wewnętrznie korygują błędy i niedociągniecia
- XHTML wymaga pełnej zgodności ze specyfikacją XML, inaczej nie jest możliwe wyświetlenie dokumentu
- Pozorna niedogodność jest tak naprawdę zaletą: wymusza większą staranność przy tworzeniu stron internetowych

Aspekt przetwarzania XHTML

- Przeglądarki przetwarzające HTML wewnętrznie korygują błędy i niedociągniecia
- XHTML wymaga pełnej zgodności ze specyfikacją XML, inaczej nie jest możliwe wyświetlenie dokumentu
- Pozorna niedogodność jest tak naprawdę zaletą: wymusza większą staranność przy tworzeniu stron internetowych
- Możliwość walidacji stron

XHTML w porównaniu do HTML (1)

- Każdemu znacznikowi otwierającemu odpowiada znacznik zamykający (np. `<li> ... </li>`)

XHTML w porównaniu do HTML (1)

- Każdemu znacznikowi otwierającemu odpowiada znacznik zamykający (np. `<li> ... </li>`)
- Puste elementy są także zamykane (np. zamiast `<br>` stosujemy `<br />`)

XHTML w porównaniu do HTML (1)

- Każdemu znacznikowi otwierającemu odpowiada znacznik zamykający (np. `<li> ... </li>`)
- Puste elementy są także zamykane (np. zamiast `<br>` stosujemy `<br />`)
- Poprawne zagnieżdżanie (np. zamiast `<p>tekst <em>wyróżniony</p></em>` - `<p>tekst <em>wyróżniony</em></p>`)

XHTML w porównaniu do HTML (1)

- Każdemu znacznikowi otwierającemu odpowiada znacznik zamykający (np. `<li> ... </li>`)
- Puste elementy są także zamykane (np. zamiast `<br>` stosujemy `<br />`)
- Poprawne zagnieżdżanie (np. zamiast `<p>tekst <em>wyróżniony</p></em>` - `<p>tekst <em>wyróżniony</em></p>`)
- Nazwy elementów i atrybutów pisane małymi literami

XHTML w porównaniu do HTML (1)

- Każdemu znacznikowi otwierającemu odpowiada znacznik zamykający (np. `<li> ... </li>`)
- Puste elementy są także zamykane (np. zamiast `<br>` stosujemy `<br />`)
- Poprawne zagnieżdżanie (np. zamiast `<p>tekst <em>wyróżniony</p></em>` - `<p>tekst <em>wyróżniony</em></p>`)
- Nazwy elementów i atrybutów pisane małymi literami
- Wartości artrybutów w cudzysłowie (np. `<td rowspan="3">`)

XHTML w porównaniu do HTML (1)

- Każdemu znacznikowi otwierającemu odpowiada znacznik zamykający (np. `<li> ... </li>`)
- Puste elementy są także zamykane (np. zamiast `<br>` stosujemy `<br />`)
- Poprawne zagnieżdżanie (np. zamiast `<p>tekst <em>wyróżniony</p></em>` - `<p>tekst <em>wyróżniony</em></p>`)
- Nazwy elementów i atrybutów pisane małymi literami
- Wartości artrybutów w cudzysłowie (np. `<td rowspan="3">`)
- Niedozwolona minimalizacja elementów (np. zamiast `<textarea readonly>` - `<textarea readonly="readonly">`)

XHTML w porównaniu do HTML (2)

- Główny element `hmtl` musi zawierać atrybut `xmlns` (np. `<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pl">`)

XHTML w porównaniu do HTML (2)

- Główny element `hmtl` musi zawierać atrybut `xmlns` (np. `<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pl">`)
- Dokument rozpoczyna się od (opcjonalnej) deklaracji XML, np. `<?xml version="1.0" encoding="iso-8859-2"?>`

XHTML w porównaniu do HTML (2)

- Główny element `hmtl` musi zawierać atrybut `xmlns` (np. `<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pl">`)
- Dokument rozpoczyna się od (opcjonalnej) deklaracji XML, np. `<?xml version="1.0" encoding="iso-8859-2"?>`
- Należy zastsować odpowiednią definicję typu dokumentu (np. dla XHMTL 1.0 `<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">`)

Przykład dokumentu XHTML 1.0

```
<?xml version="1.0" encoding="ISO-8859-2"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pl">
  <head>
    <title>Przykład dokumentu zgodnego z XHTML 1.0 Strict</title>
    <link rel="stylesheet" type="text/css" href="style.css" />
  </head>
  <body>
    <p>To jest przykład.</p>
  </body>
</html>
```

Walidacja dokumentów (X)HTML

Każdy dokument XHTML musi spełniać dwa rodzaje poprawności:

Walidacja dokumentów (X)HTML

Każdy dokument XHTML musi spełniać dwa rodzaje poprawności:

- **Poprawność składniową** (*well-formedness*) - zgodność z podstawowymi regułami składni XML

Walidacja dokumentów (X)HTML

Każdy dokument XHTML musi spełniać dwa rodzaje poprawności:

- **Poprawność składniową** (*well-formedness*) - zgodność z podstawowymi regułami składni XML
- **Poprawność strukturalną** (*validity*) - zgodność z daną definicją typu dokumentu (DTD)

Walidacja dokumentów (X)HTML

Każdy dokument XHTML musi spełniać dwa rodzaje poprawności:

- **Poprawność składniową** (*well-formedness*) - zgodność z podstawowymi regułami składni XML
- **Poprawność strukturalną** (*validity*) - zgodność z daną definicją typu dokumentu (DTD)

Poprawność składniowa jest sprawdzana przez dowolny parser XML, w tym np. przeglądarka Firefox. Jęsli występuje bład, to parser ma obowiązek nie wyświetlić dokumentu.

Walidacja dokumentów (X)HTML

Każdy dokument XHTML musi spełniać dwa rodzaje poprawności:

- **Poprawność składniową** (*well-formedness*) - zgodność z podstawowymi regułami składni XML
- **Poprawność strukturalną** (*validity*) - zgodność z daną definicją typu dokumentu (DTD)

Poprawność składniowa jest sprawdzana przez dowolny parser XML, w tym np. przeglądarka Firefox. Jęsli występuje bład, to parser ma obowiązek nie wyświetlić dokumentu.

Poprawność strukturalna jest sprawdzana przez tzw. *walidatory*. Walidator porównuje dokument z podaną definicją typu dokumentu.

Semantic (X)HTML - Idealizacja

- Semantic (X)HTML to nie tyle standard co pewien styl tworzenia stron
- Dążymy do seperacji treści od formatu

Semantic (X)HTML - Idealizacja

- Semantic (X)HTML to nie tyle standard co pewien styl tworzenia stron
- Dążymy do seperacji treści od formatu
- Rezygnujemy z elementów lub atrybutów służących tylko do formatowania, np. `<font>`, `<marque>` itp.

Semantic (X)HTML - Idealizacja

- Semantic (X)HTML to nie tyle standard co pewien styl tworzenia stron
- Dążymy do seperacji treści od formatu
- Rezygnujemy z elementów lub atrybutów służących tylko do formatowania, np. `<font>`, `<marque>` itp.
- Przykład: Różnica między `<i>` a `<em>`
- Nadajemy dokumentom strukturę logiczną
- Formatowanie odbywa się na innym poziomie, np. CSS (Cascading Style Sheets)

Przydatne linki

Specyfikacja HTML 4.01 http://www.w3.org/TR/html4/

Specyfikacja XHTML 1.0 http://www.w3.org/TR/xhtml1/

Specyfikacja XHTML 1.1 http://www.w3.org/TR/xhtml11/

Walidator W3C http://validator.w3.org

Kurs HTML http://webmaster.helion.pl/kurshtml/

Skrót HTML http://www.w3schools.com/html/html_quick.asp

Kurs XHTML http://kurs.browsehappy.pl