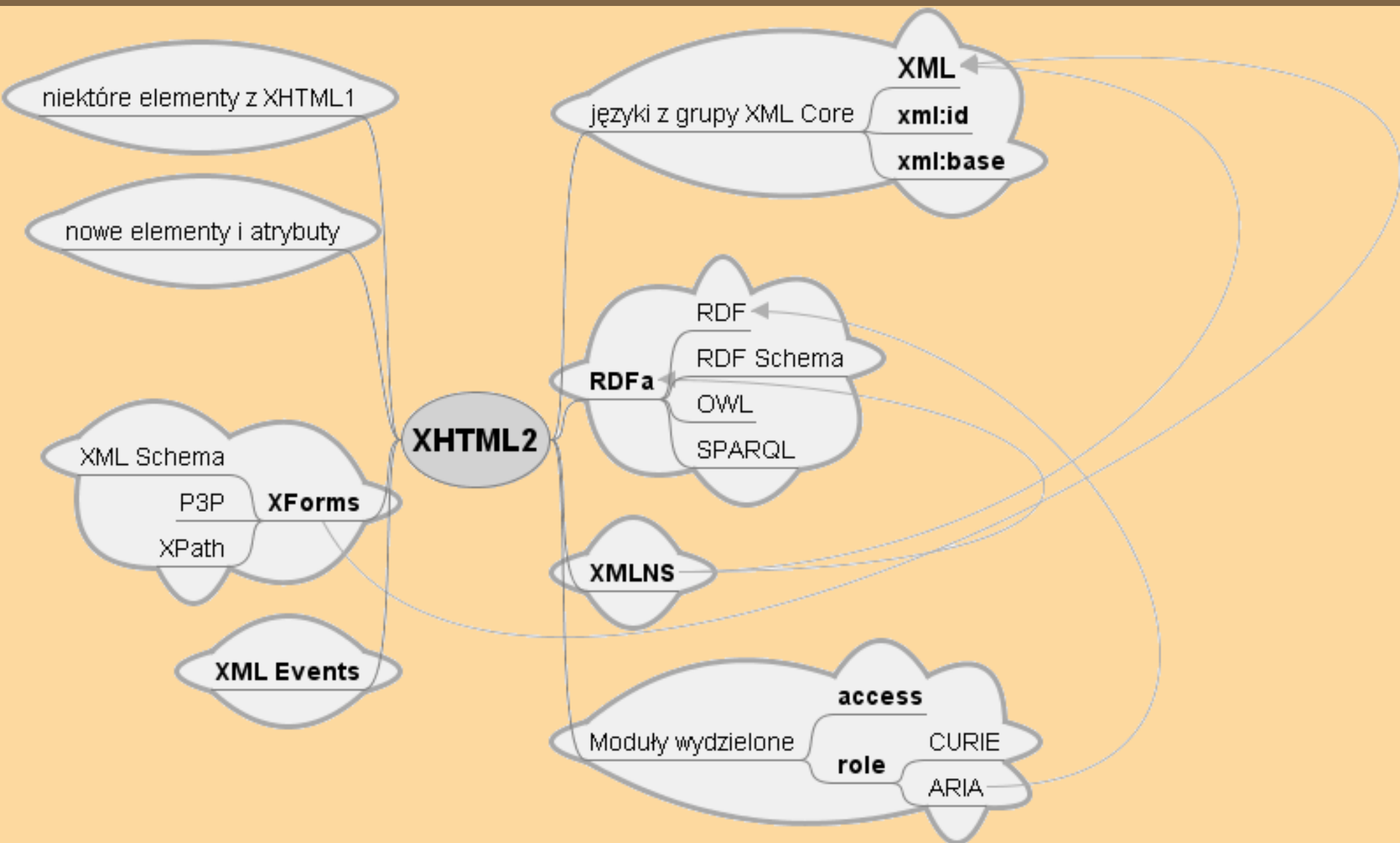


XHTML2 i HTML5

Przyszłość WWW oparta o nowe standardy sieciowe — HTML5 i XHTML2

mgr inż. Dominik Tomaszuk
Uniwersytet w Białymstoku

XHTML2



XML

- XHTML2 to **aplikacja XML-a**, która nie daje innych możliwości parsowania
- Struktura XHTML2 jest zapisana w **RELAX NG** i **XML Schema**
- XHTML2 silnie wykorzystuje **technologie XML**

xml:id i xml:base

- **xml:id** to predefiniowany atrybut o typie identyfikatora (m.in. musi być unikalny) – odpowiednik ID z DTD
- **xml:base** to predefiniowany atrybut określający ścieżkę bazową – może być używany w hiperłączach prostych oraz zawartości wstawianej, odpowiednik starego znacznika `<base/>`

RDFa

- RDFa czyli **RDF w atrybutach**
- Celem RDF jest umożliwienie maszynowego przetwarzania abstrakcyjnych opisów zasobów w sposób automatyczny
- Opis zasobu za pomocą trzech elementów: **podmiotu, predykatu i obiektu**
- **RDFa** to jedna z notacji RDF

RDFa

- RDF Schema i OWL są językami reprezentacji danych
- RDF Schema i OWL są oparte o RDF oraz mogą być zapisane w składni RDFa
- Dostęp do RDF może odbywać się przez język zapytań SPARQL
- Nowe atrybuty: **about**, rel/rev, href/src/**resource**, **property**, content, **datatype**, **typeof**

XMLNS

- XMLNS to **przestrzenie nazw w XML**
- XMLNS pozwalają korzystać z kilku aplikacji XML w obrębie jednego dokumentu
- XMLNS w **RDFa** wykorzystywane są do definiowania słowników, np. **Dublin Core, FOAF i DOAP**

```
<div xmlns:dc="http://purl.org/dc/elements/1.1/"  
  about="#prezentacja">  
  <span property="dc:creator">Dominik Tomaszuk</span>  
</div>
```

access i role

- **Element access** to rozszerzenie koncepcji atrybutu `accesskey` i lepsze rozwiązania dla dostępności WAI
- **Atrybut role** to klasyfikacja roli elementów w XHTML-u
- Taksonomia w rolach jest zdefiniowana w RDF i OWL

```
<access targetrole="navigation" key="M"/>
```

```
<div role="navigation">Menu...</div>
```

access i role

- **Atrybut role** wykorzystuje koncepcje specyfikacji **ARIA** w celu dostępności dynamicznej zawartości dla osób niepełnosprawnych.
- Access i role korzystają z **CURIE**
- CURIE to rozszerzenie wartości atrybutów o **XMLNS**
- CURIE wykorzystywane są również w **RDFa** i **SPARQL**

XForms

- XForms to nie tylko formularze, XForms to **model przetwarzania XML i język opisu interfejsu użytkownika**
- XForms wykorzystuje wzorzec **MVC** (model – widok – kontroler)
- XForms to walidacja formularzy dzięki typom danych **XML Schema** (w tym przez wyrażenia regularne)
- Prywatność dzięki **P3P**, dostęp dzięki **XPath**

XForms

```
<model>
<!-- model -->
<instance>
<data xmlns=""><q/></data>
</instance>
<submission action="s.php" method="get"
id="s"/>
</model>
<!-- widok -->
<input ref="q"><label>Szukaj</label></input>
<submit
submission="s"><label>Znajdź</label></submit>
```

XML Events

- XML Events służy do **obsługi zdarzeń** i współpracy ze skryptami, np w JavaScript
- Każda zmiana zdarzenia wiąże się tylko ze zmianą skryptu

```
<ev:listener observer="pomoc"  
handler="#pokaz" event="DOMActivate" />  
<handler id="pokaz" type="text/javascript">  
    pokaz();  
</handler>  
<trigger id="pomoc">Pomoc</trigger>
```

XML Events

- Jedno zdarzenie nie musi być związane z jednym językiem skryptowym

```
<input type="submit">
```

```
  <handler ev:event="DOMActivate"  
type="text/javascript"> validate();  
</handler>
```

```
  <handler ev:event="DOMActivate"  
type="text/python"> validate </handler>
```

```
  <handler ev:event="DOMFocusIn" type="text/  
javascript"> help(); </handler>
```

```
</input>
```

Nowe elementy

- Sekcje i nagłówki (działy)
- Separatory
- Nowe wiersze

```
<section>
```

```
  <h>XHTML2</h> ,
```

```
  <l>XHTML2</l> <l>służy do tworzenia stron  
internetowych.</l>
```

```
<separator />
```

```
</section>
```

Nowe elementy

- Blok kodu
- Listy nawigacyjne

```
<nł>
```

```
  <label>Menu</label>
```

```
  <li href="xhtml2.html">XHTML2</li>
```

```
  <li href="html5.html">HTML5</li>
```

```
</nł>
```

Nowe atrybuty

- Wyglądy
- Edycje
- Hipertekst dla każdego elementu

```
<p layout="relevant"  
href="http://www.uwb.edu.pl/" hreflang="pl"  
hrefmedia="screen">
```

Uniwersytet

w

Białymstoku

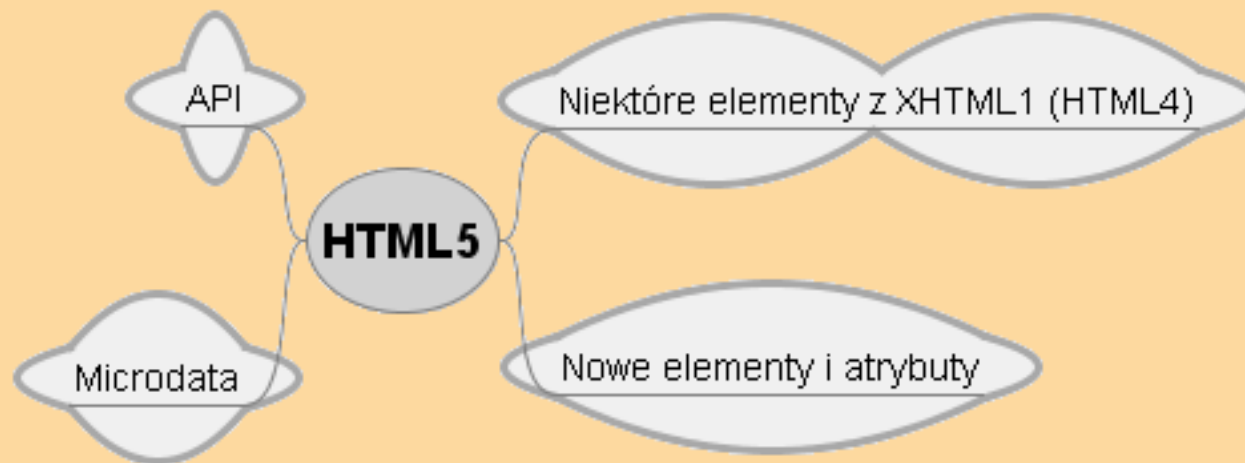
```
</p>
```

Nowe atrybuty

- Nowy dwukierunkowy tekst (wraz z notacją rubinową) to ulepszenie internacjonalizacji
- Osadzanie dla każdego elementu

```
<p src="budynek" type="image/png, image/gif"
xml:lang="pl" dir="ltr">Budynek
Uniwersytetu.</p>
```

HTML5



HTML5 – dwa języki

- Specyfikacja określa dwa sposoby parsowania:
 - pseudo-SGML – **HTML5**
 - XML – **XHTML5**
- Obsługa `xml:base` i `xml:lang` – tylko XHTML5
- Obsługa XMLNS – tylko XHTML5
- Obsługa `document.write()` i `document.writeln()` – tylko HTML5

HTML5 – dwa języki

- Typy MIME: text/html – HTML5, application/xml lub application/xhtml+xml – XHTML5
- Brak RCDATA w XHTML5 (elementy title i textarea)
- DOCTYPE - w HTML5 wymagany, w XHTML5 opcjonalny
- Element noscript – tylko w HTML5
- W XHTML5 domyślne kodowanie to UTF-8 lub UTF-16

Microdata

- Microdata to uboższy **RDFa dla HTML5**
- Microdata służy do **opisywania treści dla odczytu maszynowego**
- Microdata to grupy par **nazwa-wartość**
- Dwa atrybuty **item** i **itemprop**

```
<section item="org.example.uczelnia">  
  <h1 itemprop="nazwa">Uniwersytet w  
  Białymstoku</h1>  
</section>
```

Microdata

- Microdata to predefiniowane słowniki: **vCard**, **vEvent** oraz wartości do określania **licencji**
- Microdata pozwala na konwersje do **JSON**, **RDF**, vCard, iCalendar, Atom

```
<div item="vcard">  
  <span itemprop="fn">Dominik Tomaszuk</span>  
  <p itemprop="org" item>  
    <span itemprop="organization-name">Uniwersytet w  
    Białymstoku</span>  
  </p>  
</div>
```

Nowe API

- API do elementu **canvas**,
- API odtwarzania wideo i audio elementy **video** i **audio**),
- API do **trwałego składowania** (klucz – wartość i bazy danych SQL),
- API do **trybu offline** aplikacjach webowych,
- API do **rejestrowania** przez aplikacje webowe **protokołów i mediów**,

Nowe API

- API do **edycji** (atrybut contenteditable),
- API do **drag & drop** (atrybut draggable),
- API do **gniazdek webowych**,
- API do obsługi **historii**,
- API do **wiadomości** „cross-document”,
- API do **wysyłania zdarzeń** (element eventsource).

Nowe API

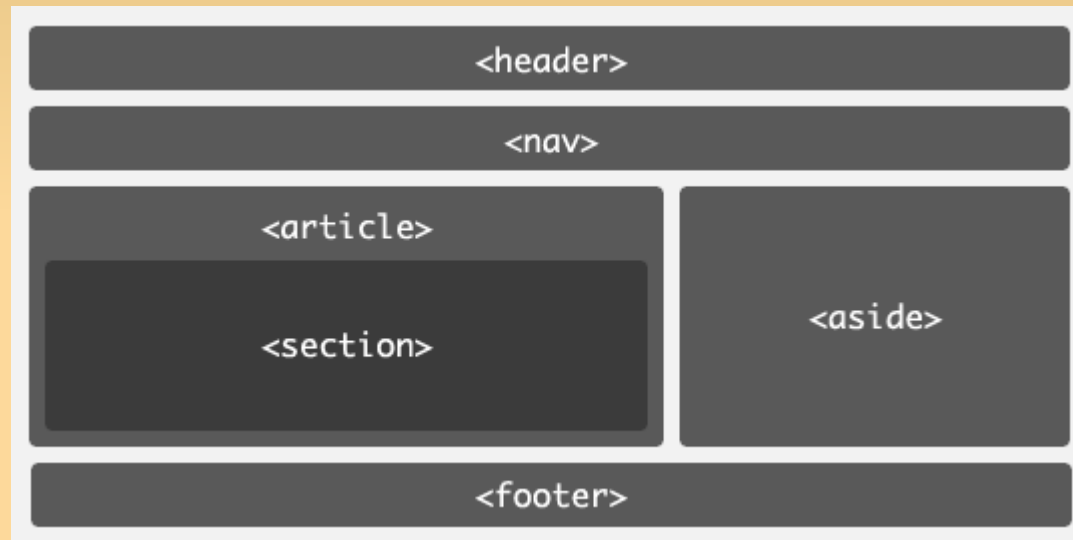
```
/* odtwarzanie wideo */  
var v = document.getElementsByTagName("video")[0];  
v.play();  
/* trwałe składowanie klucz-wartość */  
sessionStorage.imie = "Dominik";  
alert( "imie = " + sessionStorage.imie );  
/* tryb offline w aplikacjach webowych */  
alert(navigator.onLine ? "Jesteś online" :  
"Jesteś offline");
```

Rozszerzenia API

- **getElementsByClassName()** – aby wybrać elementy poprzez ich nazwę klasy
- **innerHTML** – aby serializować dokumenty
- **activeElement** oraz **hasFocus** – aby określić który element jest w danym momencie aktywny
- **getSelection()** – zwraca obiekt reprezentujący aktualny wybór
- **designMode** i **execCommand()** – często stosowane do edycji dokumentów

Nowe elementy

- section
- article
- aside
- header
- footer
- nav



Nowe elementy

- dialog
- figure
- audio i video
- embed
- meter
- time

```
<video autoplay>
```

```
  <source src="film.ogg" type="video/ogg">
```

```
</video>
```

Nowe elementy

- canvas
- details
- datalist
- eventsource
- keygen
- output

```
<input list="jezyki">  
<datalist id="jezyki">  
  <option value="HTML5">  
</datalist>
```

Nowe atrybuty

- @ping
- @charset,
- @autofocus
- @form
- @autocomplete, @min, @max, @pattern,
@step
- @async

```
<meta charset="utf-8">
```

Nowe atrybuty

- @manifest
- @contenteditable
- @contextmenu,
- @draggable
- @hidden
- @data-*
- @role, @aria-* (role i ARIA)

Nowe atrybuty

```
<label> ID:
```

```
  <input pattern="[0-9][A-Z]{3}" name="id"  
  autocomplete="off">
```

```
</label>
```

```
<div contenteditable="true">Witam!</div>
```

```
<span data-wzrost="cm">180</span>
```

```
<script src="skrypt.js" async></script>
```

Podobieństwa

- Usunięcie elementów prezentacji
- Listy nawigacyjne
- Bardziej dopasowane formularze (więcej kontrolek, walidacja)
- Bardziej semantyczny język znaczników
- Wprowadzenie elementów do notacji rubinowej
- Wsparcie dla ARIA i role

Główna różnica

HTML5 to ewolucja,
XHTML2 to rewolucja

Koniec

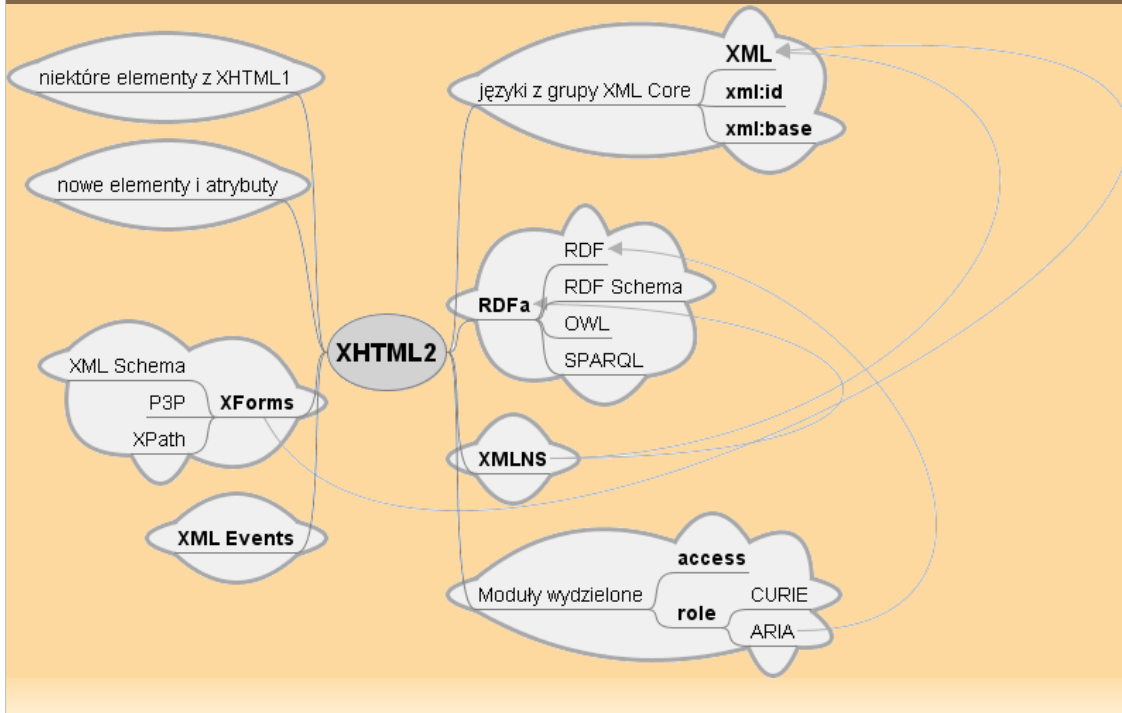
Dziękuję i proszę o pytania

XHTML2 i HTML5

Przyszłość WWW oparta o nowe standardy sieciowe — HTML5 i XHTML2

mgr inż. Dominik Tomaszuk
Uniwersytet w Białymstoku

XHTML2



XML

- XHTML2 to **aplikacja XML-a**, która nie daje innych możliwości parsowania
- Struktura XHTML2 jest zapisana w **RELAX NG** i **XML Schema**
- XHTML2 silnie wykorzystuje **technologie XML**

xml:id i xml:base

- **xml:id** to predefiniowany atrybut o typie identyfikatora (m.in. musi być unikalny) – odpowiednik ID z DTD
- **xml:base** to predefiniowany atrybut określający ścieżkę bazową – może być używany w hiperłączach prostych oraz zawartości wstawianej, odpowiednik starego znacznika `<base/>`

RDFa

- RDFa czyli **RDF w atrybutach**
- Celem RDF jest umożliwienie maszynowego przetwarzania abstrakcyjnych opisów zasobów w sposób automatyczny
- Opis zasobu za pomocą trzech elementów: **podmiotu, predykatu i obiektu**
- **RDFa** to jedna z notacji RDF

RDFa

- RDF Schema i OWL są językami reprezentacji danych
- RDF Schema i OWL są oparte o RDF oraz mogą być zapisane w składni RDFa
- Dostęp do RDF może odbywać się przez język zapytań SPARQL
- Nowe atrybuty: **about**, rel/rev, href/src/**resource**, **property**, content, **datatype**, **typeof**

XMLNS

- XMLNS to **przestrzeń nazw w XML**
- XMLNS pozwalają korzystać z kilku aplikacji XML w obrębie jednego dokumentu
- XMLNS w **RDFa** wykorzystywane są do definiowania słowników, np. **Dublin Core, FOAF i DOAP**

```
<div xmlns:dc="http://purl.org/dc/elements/1.1/"  
  about="#prezentacja">  
  <span property="dc:creator">Dominik Tomaszuk</span>  
</div>
```

access i role

- **Element access** to rozszerzenie koncepcji atrybutu accesskey i lepsze rozwiązania dla dostępności WAI
- **Atrybut role** to klasyfikacja roli elementów w XHTML-u
- Taksonomia w rolach jest zdefiniowana w RDF i OWL

```
<access targetrole="navigation" key="M"/>
```

```
<div role="navigation">Menu...</div>
```

access i role

- **Atrybut role** wykorzystuje koncepcje specyfikacji **ARIA** w celu dostępności dynamicznej zawartości dla osób niepełnosprawnych.
- Access i role korzystają z **CURIE**
- CURIE to rozszerzenie wartości atrybutów o **XMLNS**
- CURIE wykorzystywane są również w **RDFa** i **SPARQL**

XForms

- XForms to nie tylko formularze, XForms to **model przetwarzania XML i język opisu interfejsu użytkownika**
- XForms wykorzystuje wzorzec **MVC** (model – widok – kontroler)
- XForms to walidacja formularzy dzięki typom danych **XML Schema** (w tym przez wyrażenia regularne)
- Prywatność dzięki **P3P**, dostęp dzięki **XPath**

XForms

```
<model>
  <!-- model -->
  <instance>
    <data xmlns=""><q/></data>
  </instance>
  <submission action="s.php" method="get"
  id="s"/>
</model>
<!-- widok -->
<input ref="q"><label>Szukaj</label></input>
<submit
  submission="s"><label>Znajdź</label></submit>
```

XML Events

- XML Events służy do **obsługi zdarzeń** i współpracy ze skryptami, np w JavaScript
- Każda zmiana zdarzenia wiąże się tylko ze zmianą skryptu

```
<ev:listener observer="pomoc"
handler="#pokaz" event="DOMActivate"/>
<handler id="pokaz" type="text/javascript">
    pokaz();
</handler>
<trigger id="pomoc">Pomoc</trigger>
```

XML Events

- Jedno zdarzenie nie musi być związane z jednym językiem skryptowym

```
<input type="submit">  
  <handler ev:event="DOMActivate"  
type="text/javascript"> validate();  
</handler>  
  <handler ev:event="DOMActivate"  
type="text/python"> validate </handler>  
  <handler ev:event="DOMFocusIn" type="text/  
javascript"> help(); </handler>  
</input>
```

Nowe elementy

- Sekcje i nagłówki (działy)
- Separatory
- Nowe wiersze

```
<section>
```

```
  <h>XHTML2</h> ,
```

```
  <l>XHTML2</l> <l>służy do tworzenia stron  
internetowych.</l>
```

```
<separator />
```

```
</section>
```

Nowe elementy

- Blok kodu
- Listy nawigacyjne

```
<nl>  
  <label>Menu</label>  
  <li href="xhtml2.html">XHTML2</li>  
  <li href="html5.html">HTML5</li>  
</nl>
```

Nowe atrybuty

- Wyglądy
- Edycje
- Hipertekst dla każdego elementu

```
<p layout="relevant"  
href="http://www.uwb.edu.pl/" hreflang="pl"  
hrefmedia="screen">
```

Uniwersytet

w

Białymstoku

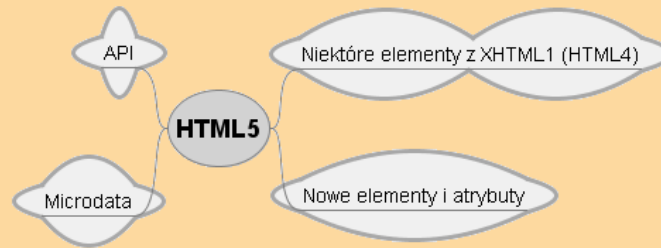
```
</p>
```

Nowe atrybuty

- Nowy dwukierunkowy tekst (wraz z notacją rubinową) to ulepszenie internacjonalizacji
- Osadzanie dla każdego elementu

```
<p src="budynek" type="image/png, image/gif"
xml:lang="pl" dir="ltr">Budynek
Uniwersytetu.</p>
```

HTML5



HTML5 – dwa języki

- Specyfikacja określa dwa sposoby parsowania:
 - pseudo-SGML – **HTML5**
 - XML – **XHTML5**
- Obsługa `xml:base` i `xml:lang` – tylko XHTML5
- Obsługa XMLNS – tylko XHTML5
- Obsługa `document.write()` i `document.writeln()` – tylko HTML5

HTML5 – dwa języki

- Typy MIME: text/html – HTML5, application/xml lub application/xhtml+xml – XHTML5
- Brak RCDATA w XHTML5 (elementy title i textarea)
- DOCTYPE - w HTML5 wymagany, w XHTML5 opcjonalny
- Element noscript – tylko w HTML5
- W XHTML5 domyślne kodowanie to UTF-8 lub UTF-16

Microdata

- Microdata to uboższy **RDFa dla HTML5**
- Microdata służy do **opisywania treści dla odczytu maszynowego**
- Microdata to grupy par **nazwa-wartość**
- Dwa atrybuty **item** i **itemprop**

```
<section item="org.example.uczelnia">  
  <h1 itemprop="nazwa">Uniwersytet w  
  Białymstoku</h1>  
</section>
```

Microdata

- Microdata to predefiniowane słowniki: **vCard**, **vEvent** oraz wartości do określania **licencji**
- Microdata pozwala na konwersje do **JSON**, **RDF**, vCard, iCalendar, Atom

```
<div item="vcard">  
  <span itemprop="fn">Dominik Tomaszuk</span>  
  <p itemprop="org" item>  
    <span itemprop="organization-name">Uniwersytet w  
    Białymstoku</span>  
  </p>  
</div>
```

Nowe API

- API do elementu **canvas**,
- API odtwarzania wideo i audio elementy **video** i **audio**),
- API do **trwałego składowania** (klucz – wartość i bazy danych SQL),
- API do **trybu offline** aplikacjach webowych,
- API do **rejestrowania** przez aplikacje webowe **protokołów i mediów**,

Nowe API

- API do **edycji** (atrybut contenteditable),
- API do **drag & drop** (atrybut draggable),
- API do **gniazdek webowych**,
- API do obsługi **historii**,
- API do **wiadomości** „cross-document”,
- API do **wysyłania zdarzeń** (element eventsource).

Nowe API

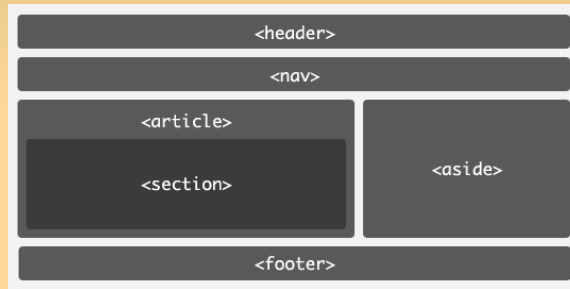
```
/* odtwarzanie wideo */  
var v = document.getElementsByTagName("video")[0];  
v.play();  
/* trwałe składowanie klucz-wartość */  
sessionStorage.imie = "Dominik";  
alert( "imie = " + sessionStorage.imie );  
/* tryb offline w aplikacjach webowych */  
alert(navigator.onLine ? "Jesteś online" :  
"Jesteś offline");
```

Rozszerzenia API

- **getElementsByClassName()** – aby wybrać elementy poprzez ich nazwę klasy
- **innerHTML** – aby serializować dokumenty
- **activeElement** oraz **hasFocus** – aby określić który element jest w danym momencie aktywny
- **getSelection()** – zwraca obiekt reprezentujący aktualny wybór
- **designMode** i **execCommand()** – często stosowane do edycji dokumentów

Nowe elementy

- section
- article
- aside
- header
- footer
- nav



Nowe elementy

- dialog
- figure
- audio i video
- embed
- meter
- time

```
<video autoplay>  
  <source src="film.ogg" type="video/ogg">  
</video>
```

Nowe elementy

- canvas
- details
- datalist
- eventsources
- keygen
- output

```
<input list="jezyki">  
<datalist id="jezyki">  
  <option value="HTML5">  
</datalist>
```

Nowe atrybuty

- @ping
- @charset,
- @autofocus
- @form
- @autocomplete, @min, @max, @pattern, @step
- @async

```
<meta charset="utf-8">
```

Nowe atrybuty

- @manifest
- @contenteditable
- @contextmenu,
- @draggable
- @hidden
- @data-*
- @role, @aria-* (role i ARIA)

Nowe atrybuty

```
<label> ID:
```

```
  <input pattern="[0-9][A-Z]{3}" name="id"
  autocomplete="off">
```

```
</label>
```

```
<div contenteditable="true">Witam!</div>
```

```
<span data-wzrost="cm">180</span>
```

```
<script src="skrypt.js" async></script>
```

Podobieństwa

- Usunięcie elementów prezentacji
- Listy nawigacyjne
- Bardziej dopasowane formularze (więcej kontrolek, walidacja)
- Bardziej semantyczny język znaczników
- Wprowadzenie elementów do notacji rubinowej
- Wsparcie dla ARIA i role

Główna różnica

HTML5 to ewolucja,
XHTML2 to rewolucja

Koniec

Dziękuję i proszę o pytania