

HTML 5

pawel@cs.uni.wroc.pl

www.itcourses.eu

WPROWADZENIE

DOM

- Document Object Model

Specyfikacja

- <https://www.w3.org/DOM/DOMTR>

Co to jest i do czego?

```
...<!DOCTYPE html> == $0
<!-- $Id: DOMTR.html,v 1.80 2012/05/02 23:14:54 plehegar Exp $-->
<html lang="en" xml:lang="en">
  <head>...</head>
  <body onload="init()">
    <p align="right" class="navbar">
      <a href="/">
        
      </a>
      <a href="/Architecture/" rel="in-domain">
        
      </a>
      <small>...</small>
    </p>
    <h1>Document Object Model (DOM) Technical Reports</h1>
    <p>
      "
      See also "
      <a href="/TR/">other W3C Technical Reports</a>
      "
      "
    </p>
    <div id="TableOfContents">...</div>
    <div id="dom4">...</div>
    <div id="dom3">...</div>
    <div id="dom2">...</div>
    <div id="dom1">...</div>
    <div id="others">...</div>
    <hr>
    <address>...</address>
    <p class="copyright">...</p>
  </body>
</html>
```

WPROWADZENIE

Nowe znaczniki (tzw. semantic tags) + JS API = HTML5

Standard W3C przy udziale Google, Microsoft, Apple, itd.

Interpretacja i wygląd znaczników zależy od przeglądarki

Aktualna dokumentacja:

- <http://www.w3.org/TR/html5/>

Working group editor's draft:

- <http://dev.w3.org/html5/>

WSPARCIE PRZEZ PRZEGLĄDARKI

Test na stronie: <http://html5test.com/>

	Chrome	Opera	Firefox	Edge	Safari
Upcoming					
Current	52 → 492		48 → 461	14 → 460	9.1 → 370
Older	51 → 492	37 → 489	47 → 456	13 → 433	9.0 → 360
	50 → 489	30 → 479	46 → 456	12 → 377	8.0 → 354
	45 → 482	12.10 → 309	45 → 456	Internet Explorer	
	40 → 456		40 → 429	11 → 312	
	30 → 394		35 → 397	10 → 265	
			30 → 378	9 → 113	

- na 555 możliwych do zdobycia punktów

Sporo przykładów i przegląd możliwości przeglądarek:

<http://html5demos.com/>

CO NOWEGO W HTML5?

Nowe znaczniki do budowy dokumentu

Nowe znaczniki formularzy (wraz z nowymi typami)

Renderowanie grafiki 2D, czyli słynny canvas

Obsługa audio i video

Mechanizm przeciągnij i upuść

Aplikacje w trybie offline

Przechowywanie danych po stronie klienta (web storage)

Dane strukturalne (MicroData, data-*)

CO NOWEGO W HTML5?

Poniższe nie jest częścią standardu, ale jest wspierane przez przeglądarki (i często kojarzone z HTML5)

- Obsługa geolokalizacji
- Procesy działające w tle
- WebSQL, Indexed Data API
- File API, File Writer
- Web Sockets

HTML5 VS. XHTML5

XHTML5 oznacza, że składnia jest zgodna z XML

Powinna zostać dołączona przestrzeń nazw XML-a

Serwer powinien wysłać typ `application/xhtml+xml` zamiast `text/html`

Przewaga HTML5

- Drobne błędy składniowe mogą nie wpłynąć na prezentację dokumentu (parser XML jest dużo bardziej rygorystyczny)

Przewaga XHTML5

- Można łączyć z innymi standardami XML jak MathML czy SVG

Do poczytania:

<https://blog.whatwg.org/xhtml5-in-a-nutshell>

STRUKTURA DOKUMENTU HTML5

```
<!DOCTYPE html>
<html>
<head>
<title>Title of the document</title>
<meta charset="UTF-8">
</head>

<body>
The content of the document.....
</body>

</html>
```


STRUKTURA DOKUMENTU XHTML5

```
<!DOCTYPE html>  
<html xmlns="http://www.w3.org/1999/xhtml">  
<head>  
<title>Title of the document</title>  
<meta charset="UTF-8"/>  
</head>  
  
<body>  
The content of the document.....  
</body>  
  
</html>
```

STRUKTURA DOKUMENTU

Article (blog post, user comment)

Aside

Audio, Video

Canvas

Summary i details

Embed

Figure, Figcation

Header, Footer

Address

Section

Mark

Meter

Output

Progress

Time

Wbr

Nav

DEMO

01-simplepage.html

- przegląd w różnych przeglądarkach

NOWE KONTROLKI FORMULARZY

color

datalist

date

datetime

datetime-local

email

month

number

range

search

tel

time

url

week

DEMO

02-newmarkup.html

- przegląd w różnych przeglądarkach

WALIDACJA FORMULARZY

Określana przez atrybuty kontrolek

Podpowiedź

- placeholder

Mamy do dyspozycji:

- required
- min, max
- pattern
- maxlength
- ...

Wyłączenie walidacji

- `<form novalidate ...>`
- `<input type="submit" formnovalidate="formnovalidate" ...>`

Domyślne zachowanie przeglądarki vs. pseudo klasy CSS3

DEMO

03-simplevalidation1.html

03-simplevalidation2.html

WALIDACJA FORMULARZY

Manualne sterowanie komunikatami błędów, czyli użycie „built-in validation rules”

- `valueMissing`
- `typeMismatch`
- `patternMismatch`
- `tooLong`
- `rangeUnderflow` i `rangeOverflow`
- `stepMismatch`
- `valid`

Metoda `checkValidity()` i właściwość `validity`

<https://developer.mozilla.org/en-US/docs/Web/API/ValidityState>

DEMO

04-bettervalidation1.html

04-bettervalidation2.html

GEOLOKALIZACJA

Główna metoda: `getCurrentPosition()`

Dwa parametry:

- `showPosition` – callback, do którego przekazywany jest obiekt lokalizacji o następujących właściwościach:
 - `coords.latitude`, `coords.longitude`, `coords.accuracy`
 - `coords.altitude`, `coords.altitudeAccuracy`
 - `coords.heading`
 - `coords.speed`
 - `Timestamp`
- `showError` – callback, do którego przekazywany jest obiekt błędu, jeśli wystąpił, o następujących polach:
 - `error.PERMISSION_DENIED`:
 - `error.POSITION_UNAVAILABLE`:
 - `error.TIMEOUT`:
 - `error.UNKNOWN_ERROR`:

GEOLOKALIZACJA

Są też

- metody `watchPosition()`
- `clearWatch()`

do śledzenia pozycji (sens przy smartfonie, tablecie, ewentualnie laptopie)

DEMO

05-geolocation.html

- odpalamy na serwerze

WEB WORKERS

Mechanizm uruchamiania procesów w tle

Mamy dwa rodzaje web worker'ów

- Dedicated – powiązany z konkretnym oknem
- Shared – wspólny dla całej domeny

Worker nie ma dostępu do

- DOM
- Okna (Window)
- Strony, z której został wywołany (host page)

Worker ma dostęp do

- Przeglądarki (appName, userAgent, ...)
- setTimeout, setInterval, clearTimeout, clearInterval
- XMLHttpRequest

WEB WORKERS

Czego potrzebujemy?

- skrypt procesu w osobnym pliku
- metoda `postmessage(dane)` do wysyłania komunikatów
 - Uwaga: przy wysyłaniu dane są kopiowane, więc wielkich danych może pojawić się kłopot
- zdarzenie `worker.onmessage` do obsługi wysłanych komunikatów

Zastosowania

- Obliczenia
- Generowanie grafiki
- Kryptografia (szyfrowanie, ...)

Wsparcie

- Chrome, FF, Opera, Safari (czyli brakuje IE)

DEMO

06-worker.html

07-jsonworker.html

08-queueworker.html

- działa pod Opera i Chrome, ale Chrome dopiero po opublikowaniu

WEB SOCKETS

Powstał przy okazji prac nad HTML5

Mechanizm do komunikacji dwukierunkowej w trybie full-duplex

Dobre źródło: <http://www.websocket.org/>

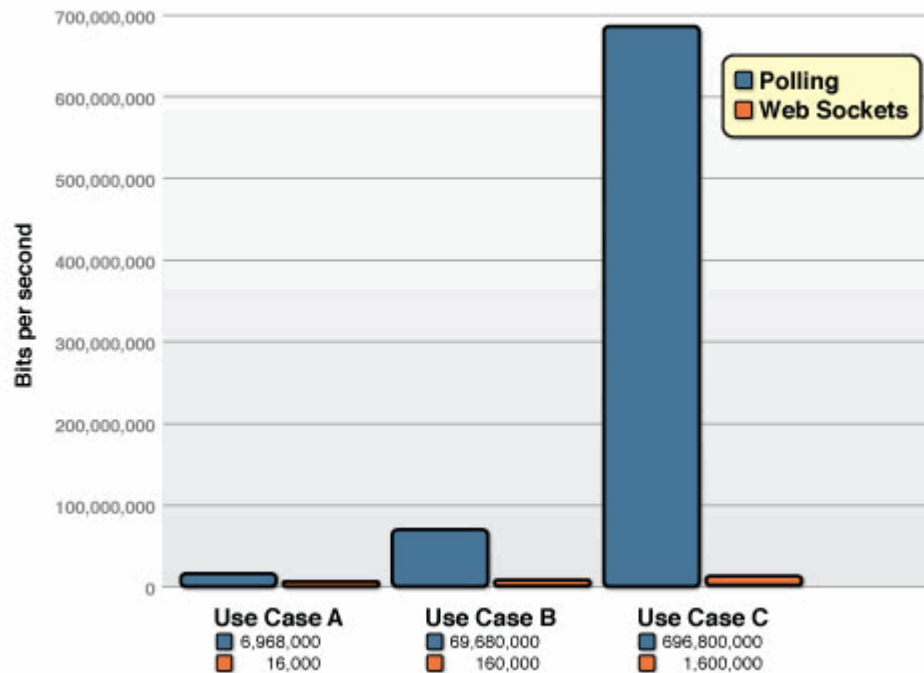
Web Sockets vs. HTTP

- HTTP działa w trybie half-duplex i wykorzystuje *polling* (request-response pattern)
- Wysyłane nagłówki
 - HTTP: 2KB
 - WS: 100KB
- Schematy
 - HTTP: http:// i https://, WebSocket: ws:// i wss://
- WebSocket duży lepszy przy aplikacjach typu real-time data

WEB SOCKETS

Porównanie ze strony websocket.org

- Case A: 1000, Case B: 10000, Case C: 100000 klientów otrzymujących 1 wiadomość na sekundę



WEB SOCKETS

Wsparcie w przeglądarkach

- Chrome, FF, Opera, Safari, IE

Niekiedy trzeba wsparcie włączyć

- FF – about:config i filtr network.websocket
- Opera – config#UserPrefs|EnableWebSockets
- Chrome – domyślnie jest wszystko włączone
 - Obecnie już wszędzie jest to włączane

Potrzebny jest także serwer

- <http://kaazing.com/products/kaazing-websocket-gateway>
- <http://code.google.com/p/phpwebsocket/>
- <http://nugget.codeplex.com/>
- Porównanie produktów
 - http://en.wikipedia.org/wiki/Comparison_of_WebSocket_implementations

WEB SOCKETS

Jak to działa?

- Po stronie klienta
 - Tworzymy obiekt `socket = new WebSocket(host)`
 - gdzie to np. `ws://localhost:8000/socket/server/startDaemon.php`
 - Podpinamy funkcje pod zdarzenia
 - `onopen`
 - `onmessage`
 - `onclose`
 - `Onerror`
 - Do wysyłania komunikatów mamy
 - `socket.send(data)`
- Po stronie serwera uruchamiamy usługę
- Fajna strona:
 - <http://net.tutsplus.com/tutorials/javascript-ajax/start-using-html5-websockets-today/>

WEBSTORAGE

Mamy do dyspozycji:

- localStorage - bezterminowe
- sessionStorage – per sesja (do zamknięcia okna przeglądarki)

Bezpieczeństwo: per sesja i per domena

Pojemność: 2-10MB, zwykle ok. 5MB

WebStorage vs. Cookies

- Tylko po stronie klienta
- Większa pojemność

W obu repozytoriach

- dane są pamiętane jako pary klucz/wartość
- wszystko jest typu string

Wsparcie we wszystkich przeglądarkach

- Chrome 4, FF 3.5, Opera 10.6, Safari 4, IE 8 (i wyższych)

WEBSTORAGE

API

- `storage.clear`
- `storage.getItem`
- `storage.key`
- `storage.length`
- `storage.remainingSpace`
- `storage.removeItem`
- `storage.setItem`

- `event` (tylko w IE 9)

WEBSTORAGE

Dostęp na różne sposoby:

- `localStorage.zmienna = "wartosc"`
- `localStorage["zmienna"] = "wartosc"`
- `localStorage.setItem("zmienna", "wartosc")`

- `alert(localStorage.zmienna)`
- `alert(localStorage["zmienna"])`
- `alert(localStorage.getItem("zmienna"))`

WEBSTORAGE

Limit miejsca (quota)

- Na każdej przeglądarce inny
- Dodanie nowej wartości warto opakować w try..catch..

```
try {  
    window.localStorage.setItem(key,value) ;  
}  
catch(e) {  
    // obsługa błędu  
}
```
- W przypadku przekroczenia limitu pojawi się błąd
 - DOM Exception: QUOTA_EXCEEDED_ERR (22)
- `storage.remainingSpace` działa tylko w IE 9

WEBSTORAGE

Można także śledzić zmiany:

- podpinamy zdarzenie:
`window.addEventListener("storage", handle_storage, false)`
- tworzymy funkcję obsługi
`function handle_storage(e) { }`
 - gdzie e ma pola
key, oldValue (wartość null dla setItem), newValue, url lub uri

Niektóre przeglądarki wspierają także Web SQL Database, co daje interfejs SQL, a nie tylko key/value.

Fajny artykuł o web storage: <http://diveintohtml5.info/storage.html>

DEMO

09-webstorage.html

Przeglądamy

- <http://wroclawski.eu/09-webstorage.html>
- <http://wroclawski.eu/storage/09-webstorage.html>
- <http://spacerownik.wroclawski.eu/09-webstorage.html>
- <http://itcourses.eu/w3/09-webstorage.html>

CANVAS

... czyli obrazki i rysowanie

Tworzymy płótno:

- `<canvas id="myCanvas" width="200" height="100"></canvas>`

a potem kod do rysowania:

- ```
<script type="text/javascript">
var c=document.getElementById("myCanvas");
var ctx=c.getContext("2d");
ctx.fillStyle="#FF0000";
ctx.fillRect(0,0,150,75);
</script>
```

# DEMO

Oglądamy stronę <http://www.html5canvastutorials.com/>

# DRAG & DROP

Miły dla oka efekt, dostępny dotychczas tylko przez spory zestaw skryptów JavaScript

## Jak to robimy w HTML?

- Ustawiamy content jako „przesuwalny”
  - `draggable=true`
  - dla `img` jest to domyślne
- Podpinamy funkcje pod zdarzenia
  - **dragstart**, `drag`, `dragenter`, `dragleave`, **dragover**, **drop**, `dragend`
- Tam, gdzie obiekt ma być „upuszczony”, trzeba go wstawić
  - Wykorzystujemy tzw. `datatransfer object`

## Dokumentacja

- [https://developer.mozilla.org/en-US/docs/Web/API/HTML\\_Drag\\_and\\_Drop\\_API](https://developer.mozilla.org/en-US/docs/Web/API/HTML_Drag_and_Drop_API)

## Bardzo dobry artykuł

- <http://www.html5rocks.com/en/tutorials/dnd/basics/>

# DEMO

10-drag&drop1.html

10-drag&drop2.html

# DANE STRUKTURALNE

Dane mogą być prezentowane w formacie przyjaznym dla

- Człowieka (HTML5+CSS3)
- Maszyny (XML, JSON)

MicroData czy data-\* umożliwiają połączenie obu światów

W ramach dokumentu HTML5 dodajemy informacje, które mogą zostać przetworzone przez maszyny

Zastosowanie

- MicroData – korzysta z tego np. wyszukiwarka Google
- data-\* - łatwe do przetwarzania w JavaScript

# DANE STRUKTURALNE

## Narzędzie do testowania

- <https://search.google.com/structured-data/testing-tool>

## Do przeczytania:

- <https://schema.org/>
- <https://developers.google.com/search/docs/guides/intro-structured-data>
- [https://en.wikipedia.org/wiki/Microdata\\_\(HTML\)](https://en.wikipedia.org/wiki/Microdata_(HTML))
- [https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/Using\\_data\\_attributes](https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/Using_data_attributes)
- <http://www.hongkiat.com/blog/better-ux-html-data-attributes/>
- <https://www.sitepoint.com/use-html5-data-attributes/>

# POZOSTAŁE MOŻLIWOŚCI

## Bardzo dużo nowych zdarzeń

- [http://www.w3schools.com/tags/ref\\_eventattributes.asp](http://www.w3schools.com/tags/ref_eventattributes.asp)

## Wsparcie dla grafiki wektorowej SVG

- ```
<svg xmlns="http://www.w3.org/2000/svg" version="1.1" height="190">  
  <polygon points="100,10 40,180 190,60 10,60 160,180"  
    style="fill:lime;stroke:purple;stroke-width:5;fill-rule:evenodd;">  
</svg>
```

Wsparcie dla tworzenia wzorów w MathML

Video & audio

Server-Sent Events

Aplikacje offline (app cache)

- Główny bohater: plik manifest

Nieźłe tutoriala

- <http://www.w3schools.com/html/>
- <http://www.tutorialspoint.com/html5/>