

Instrukcja 7

Multimedialne aspekty stron WWW.

Wszelkie multimedia na stronach WWW umieszczane są najczęściej w postaci obiektów, które do poprawnego działania wymagają odpowiednich wtyczek (które użytkownik strony musi przeważnie instalować ręcznie).

a) Najstarszym i nadal dostępnym sposobem umieszczania obiektów osadzonych (tak się nazywa m.in. elementy strony tworzone przy użyciu technologii Flash) jest dodanie znacznika <object>. Powala on na dodanie dowolnego obiektu osadzonego (do jakiego zalicza się kontrolki ActiveX, pliki PDF, aplety Java, pliki SWF, pliki audio, pliki video, itp.). Znacznik ten posiada swój odpowiednik domykający go. Pomiedzy <object></object> można umieszczać dowolny komunikat tekstowy, który ukaże się użytkownikowi w przypadku, gdy jego przeglądarka nie będzie w stanie obsłużyć danego formatu (np. brak wtyczki Adobe Flash, brak obsługi ActiveX itd.). Atrybuty znacznika object dostępne w specyfikacji HTML5:

- data – określa źródło obiektu, które ma zostać osadzone poprzez znacznik (przykładowo adres do pliku swf czy pdf)
- form – wskazuje na formularz, do którego przynależy znacznik; w specyfikacji HTML5 <object> może przynależeć (bądź może zostać zagnieżdżony) do standardowych formularzy HTML (**NA CHWILĘ OBECNĄ PRZEGLĄDARKI IGNORUJĄ TEN ATRYBUT**)
- height – określa wysokość obiektu
- width – określa szerokość obiektu
- name – pozwala na przypisanie nazwy dla danego obiektu; nazwa nie ma żadnego znaczenia dla niego samego, natomiast dla projektanta strony jest substytutem atrybutu id (pozwala na odwoływanie się do niego poprzez skrypty JavaScript oraz CSS)
- type – informuje przeglądarkę o typie multimedialnym wstawianego obiektu; jeżeli projektant pominie atrybut wtedy przeglądarka będzie sama próbowała wydedukować typ obiektu
- usemap – atrybut działa identycznie jak w przypadku znaczniki

Przykład wstawionej animacji Flash:

```
<object width="400" height="400" data="http://www.w3schools.com/html/helloworld.swf"
name="hello_object" type="application/x-shockwave-flash">Twoja przeglądarka nie może
załadować obiektu!</object>
```

Dla obiektów osadzonych istnieje możliwość ustawiania parametrów. Parametry to nic innego jak zmienne, które poprzez znaczniki przesyła się do elementu źródłowego. Przykładowo dla animacji flash można przesyłać parametr autoodtwarzania, wartości początkowej dla pola tekstowego czy innych. Właściwości znacznika <param>:

- name – nazwa parametru (zmiennej), której wartość ma zostać zmieniona po załadowaniu się obiektu
- value – wartość dla przesyłanego parametru

Przykład

```
<object width="400" height="400" data="http://www.w3schools.com/html/helloworld.swf"
name="hello_object" type="application/x-shockwave-flash">
<param name='quality' value='medium' />
<param name='bgcolor' value='#00FF00' />
<param name='FlashVars' value='imie=Piotr&nazwisko=Piotrowski' />
Twoja przeglądarka nie może załadować obiektu!</object>
```

INFORMACJA: Podane wyżej parametry zostaną co prawda przesłane do obiektu helloworld.swf lecz nie zostaną w żaden sposób wykorzystane – nie posiada on bowiem takowych zmiennych (przykład ma tylko ukazać jak dodawać znacznik <param>)

b) HTML5 wprowadza dodatkowy „nowy” znacznik pozwalający na umieszczanie treści osadzonych <embed>. Tak naprawdę znacznik ten przeglądarki obsługują od kilku lat chociaż nie należał on do standardu HTML 4 (powodował błąd walidacji). W przeciwieństwie do <object> nie pozwala on na przekazywanie parametrów do obiektu oraz na dodawanie alternatywnego tekstu w przypadku niemożliwości załadowania wtyczki (nie posiada znacznika zamykającego). Atrybuty znacznika <embed>:

- height – określa wysokość obiektu
- width – określa szerokość obiektu
- type – działa tak jak atrybut w <object>
- src – zawiera adres pliku i/lub obiektu, który ma zostać dołączony do dokumentu HTML

Przykład użycia – dołączenie dźwięku w tle strony WWW:

```
<embed src="http://www.w3schools.com/html/horse.mp3" style='position: absolute; top: 0px; height: 0px; visibility: hidden;'/>
```

INFORMACJA: Powyższa składnia spowoduje uruchomienie dźwięku bez możliwości jego wyłączenia. Proszę pamiętać iż muzyka w tle może być bardzo źle odbierana przez odwiedzających stronę. Najlepszym rozwiązaniem jest zawsze umożliwienie włączenia/wyłączenia jej na żądanie. W tym celu proszę zapoznać się z punktem dotyczącym odtwarzacza muzyki.

1. Odtwarzacz muzyki.

Już od HTML 4 istniała możliwość ustawiania w tle muzyki przy przeglądaniu strony. Niestety wszelkie odtwarzacze muzyki mogły być jedynie tworzone poprzez zewnętrzne aplikacje, jak Java (aplety) lub Flash (osadzone obiekty).

Wraz z debiutem wersji 5 dodano, pozostawiając poprzednie opcje, zupełnie nowe metody dodania dźwięku na naszej stronie WWW. Odtąd bez dodatkowych wtyczek bądź osadzonych obiektów można dodać odtwarzacz muzyki. Przynajmniej w teorii, gdyż w praktyce problemem staje się wciąż niejednorodny format plików muzycznych obsługiwany przez poszczególne przeglądarki oraz wersja, od której wspierają one znaczniki audio (najbardziej opornymi przeglądarkami w tym temacie są Internet Explorer oraz Opera).

Najstarszym sposobem jest wstawienie pliku muzycznego w znacznik <object> lub <embed> (patrz poprzedni przykład):

```
<object height="50" width="100" data="http://www.w3schools.com/html/horse.mp3">Przełęczarka nie pozwala na odtwarzanie plików MP3</object>
```

HTML5 wnosi dodatkowy, nowy znacznik odtwarzania dźwięków <audio>. Został on zaprojektowany w celu eliminacji potrzeby pisania zewnętrznych wtyczek oraz odtwarzaczy muzyki. Problemem jest niespójność obsługi standardów audio przez poszczególne przeglądarki – niektóre przeglądarki obsługują tylko format mp3, inne tylko wav, jeszcze inne ogg. Na chwilę obecną tylko 2 najpopularniejsze, Firefox oraz Chrome, są w stanie odtworzyć wszystkie trzy formaty.

INFORMACJA: Ze względu na rozmiar plików nie zalecane jest stosować formatu wave. O wiele lepszym rozwiązaniem jest równolegle umieszczać pliki w formacie mp3 oraz ogg. Niestety

nakłada to na projektanta dodatkowy obowiązek konwertowania danego pliku na dwa formaty (nie dzieje się to automatycznie). Powstało kilka stron oferujących konwersję plików poprzez strony WWW (przykładem mogą być chociażby <http://audio.online-convert.com/> oraz <http://media.io/pl/>). Inną możliwością jest konwersja plików poprzez takie programy jak Windows Media Player, VLC Media Player czy też poprzez osobną aplikację Pazera Free Audio Extractor 2.0.

Znacznik <audio> posiada kilka swoich własnych atrybutów:

- autoplay – jeżeli zostanie ustawiony to utwór będzie odtwarzany zaraz po załadowaniu go do pamięci podręcznej przeglądarki; może przyjąć tylko wartość 'autoplay'
- controls – dodaje pasek nawigacji wskazanego utworu (przycisk uruchomienia/zatrzymania, suwak postępu, czas odtwarzania oraz poziom głośności/wyciszenia); może przyjąć tylko wartość 'controls'
- loop – jeżeli zostanie ustawiony to utwór będzie odtwarzany w pętli; może przyjąć tylko wartość 'loop'
- muted – domyślnie wycisza dźwięk utworu; może przyjąć tylko wartość 'muted'
- preload – określa, czy utwór ma być ładowany (pobierany z serwera) przez przeglądarkę wraz z pozostałą jej zawartością (ustawienie domyślne, 'auto'), czy mają być pobierane tylko metadane (długość utworu, nazwa itp., 'metadata') bądź w ogóle ma być nie pobierany wraz ze stroną (będzie dosłany później, np. przy rozpoczęciu odtwarzania, 'none')
- src – określa źródło dźwięku (ścieżka do pliku), który ma być odtwarzany; przeważnie nie podaje się go ze względu na brak możliwości dodania kilku plików (w różnych formatach) i wykorzystuje się do tego dodatkowy znacznik <source>

Znacznik <source> posiada następujące atrybuty:

- src – patrz znacznik <audio>
- type – określa typ osadzonego pliku dźwiękowego; możliwe rodzaje to: 'audio/ogg' dla plików ogg, 'audio/mpeg' dla mp3 lub 'audio/wav' dla plików wav (IANA MIME oficjalnie nie posiada tego typu!)
- media – pozwala na określenie typu urządzenia oraz jego parametrów, dla których ma być odtworzony plik audio; w przypadku niezgodności zadeklarowanego typu z urządzeniem wyświetlającym stronę plik audio nie będzie pobierany; niestety na tę chwilę żadna z przeglądarek nie jest w stanie poprawnie interpretować tego atrybutu!

Przykłady dodawania dźwięku poprzez znacznik <audio>:

<!-- to rozwiązanie jest mało uniwersalne-->

```
<audio src="http://www.w3schools.com/html/horse.ogg" controls='controls'>
```

Aktualnie używana przeglądarka nie wspiera znacznika <audio>

```
</audio>
```

<!-- to rozwiązanie jest uniwersalne tylko w przypadku przeglądarek wspierających HTML5-->

```
<audio controls='controls'>
```

```
  <source src="http://www.w3schools.com/html/horse.ogg" type="audio/ogg">
```

```
  <source src="http://www.w3schools.com/html/horse.mp3" type="audio/mpeg">
```

Aktualnie używana przeglądarka nie wspiera znacznika <audio>

```
</audio>
```

<!--to rozwiązanie powinno zadziałać na niemal wszystkich przeglądarkach; niestety nie będzie wspierane w przypadku starszych przeglądarek bez dodatkowych, zewnętrznych wtyczek-->

```
<audio controls='controls'>
```

```
  <source src="http://www.w3schools.com/html/horse.mp3" type="audio/mpeg">
```

```
  <source src="http://www.w3schools.com/html/horse.ogg" type="audio/ogg">
```

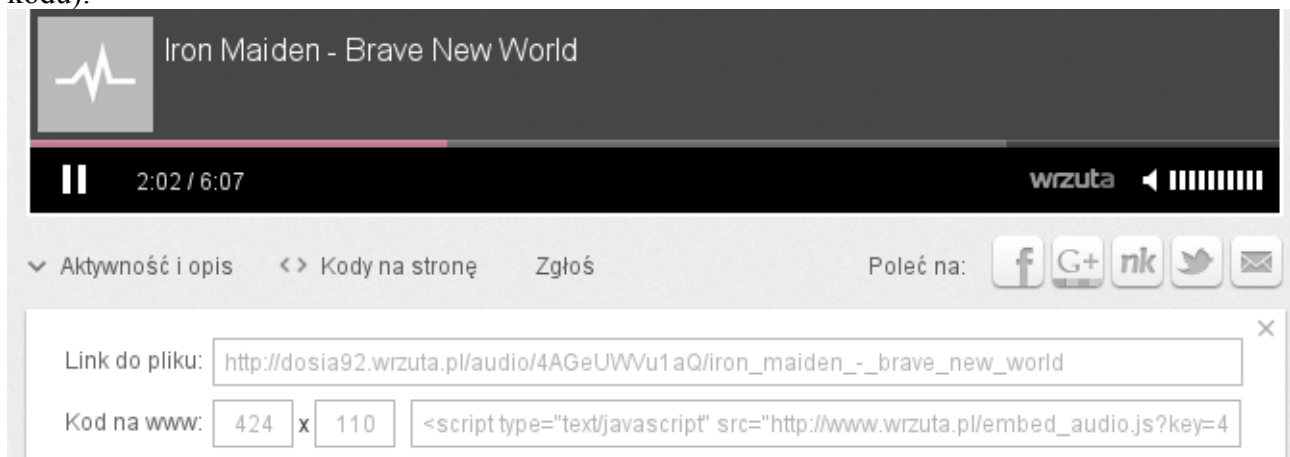
```
  <embed src="http://www.w3schools.com/html/horse.mp3">
```

</audio>

Biorąc pod uwagę wymienione wyżej problemy odtwarzanie muzyki na stronach WWW stanowi niemałe wyzwanie dla ich projektantów. Jeżeli mamy świadomość, że naszą stroną będą odwiedzać osoby, które mogą nie posiadać przeglądarek spełniających wymagania odtwarzania dźwięków, a nie chcemy też dodawać bądź tworzyć zewnętrznych odtwarzaczy muzyki (np. poprzez wtyczkę Flash) możemy zawsze umieścić najprostszego odsyłacza do danego pliku. Przeglądarka sama zadecyduje co zrobić z plikiem – odtworzyć go bezpośrednio w przeglądarce bądź w zewnętrznym programie (w najgorszym wypadku plik zostanie zapisany na dysku użytkownika):

```
<a href="http://www.w3schools.com/html/horse.mp3">Play the sound</a>
```

Innym rozwiązaniem jest dodanie na stronie muzyki z portalu np. wrzuta.pl. Po wstawieniu utworu można zrobić do niego odniesienie na swojej stronie (poprzez pobranie i wstawienie odpowiedniego kodu):



Z powyższego menu (dostępnego po wybraniu „Kody na stronę”) należy skopiować zawartość pola „Kod na www”.

```
<script type="text/javascript" src="http://www.wrzuta.pl/embed_audio.js?
key=4AGeUWVu1aQ&login=dosia92&width=424&height=110"></script>
```

2. Odtwarzacz wideo

Niestety, podobnie jak w przypadku plików audio, tak i w przypadku video istnieją bardzo duże problemy by zapewnić każdemu odwiedzającemu naszą witrynę poprawną obsługę odtwarzania filmów. Nie istnieje bowiem jednolity standard kodowania oraz umieszczania takich materiałów w obrębie dokumentu HTML.

Najstarszym i do niedawna najpowszechniejszym rozwiązaniem odtwarzania filmów na stronach WWW było wyświetlanie ich poprzez zewnętrzny odtwarzacz, najczęściej tworzony w technologii Flash. Dodanie materiału, z punktu widzenia projektanta strony, jest bardzo proste i wygląda następująco:

```
<object data="http://www.w3schools.com/html/intro.swf" height="200" width="200">Nie
posiadasz wtyczki Flash! Pobierz ją ze strony <a href="http://www.adobe.com">producenta</a>!
</object>
```

Od kilku lat widać jednak trend powolnego wycofywania technologii Flash przez kolejnych producentów zarówno systemów operacyjnych jak i samych urządzeń multimedialnych. Na chwile obecną już niemal tylko komputery stacjonarne/laptopy PC są w stanie obsługiwać najnowsze wydania wtyczki firmy Adobe. Natomiast tablety, telefony multimedialne, systemu z rodziny OS X

oraz Linux są pozbawione nowszych wersji Flash (na systemach Linux/OS X poprawne działanie wtyczek gwarantuje przeglądarka Google Chrome oraz aplikacja pipelight).

INFORMACJA: Podobnie do projektów Flash można dodawać także projekty pisane w Silverlight (odpowiednik Flash od firmy Microsoft). Są one jednak poprawnie obsługiwane jedynie w systemie Microsoft Windows. Systemy OS X/Linux mogą od pewnego czasu korzystać z dodatkowej aplikacji Pipelight by otwierać tego typu zawartość. Przykład umieszczenia Silverlight na stronie WWW:

```
<object width="300" height="300"
  data="data:application/x-silverlight-2"
  type="application/x-silverlight-2" >
  <param name="source" value="SilverlightApplication1.xap"/>
</object>
```

Innym rozwiązaniem jest użycia znacznika <embed>:

```
<embed src="http://www.w3schools.com/html/intro.swf" height="200" width="200"/>
```

Niestety pozostaje problem obsługi wtyczki Flash.

HTML5 przynosi zupełnie nowe możliwości dodawania plików wideo. Problem, podobnie jak w przypadku plików muzycznych, stanowi niejednorodność obsługiwanych standardów przez poszczególne przeglądarki.

Dodanie pliku wideo odbywa się poprzez znacznik <video>. Jego atrybuty są niemal takie same jak znacznika <audio>. Różniące się to:

- poster – pozwala na ustawienie obrazu, który będzie wyświetlał się jako domyślny widok przestrzeni odtwarzacza nim użytkownik go uruchomi; należy podać adres do pliku graficznego (tak jak np. w znacznikach)

Dodawanie kilku plików w różnych formatach, podobnie jak w przypadku <audio> odbywa się poprzez znacznik <source> (mający te same właściwości). Typy plików dla atrybutu type:

- video/mp4 dla plików mp4
- video/ogg dla plików ogg
- video/webm dla plików WebM (nie jest oficjalnie ujęty przez IANA MINE)

Przykłady umieszczania filmów wideo:

```
<!-- najprostsze i niestety najbardziej zawodne umieszczenie pliku wideo-->
<video src="http://www.w3schools.com/html/movie.ogg" controls='controls'>
  Aktualnie używana przeglądarka nie wspiera znacznika video!
</video>
```

```
<!-- rozwiązanie które gwarantuje odtworzenie filmu na wszystkich przeglądarkach wspierających
standard HTML5-->
```

```
<video width="320" height="240" controls='controls'>
  <source src="http://www.w3schools.com/html/movie.mp4" type="video/mp4">
  <source src="http://www.w3schools.com/html/movie.ogg" type="video/ogg">
```

Aktualnie używana przeglądarka nie wspiera znacznika video!

```
</video>
```

```
<!-- rozwiązanie najlepsze – gwarantuje wyświetlenie filmu niemal na każdej przeglądarce (wyjątek stanowią systemy bez wtyczki Flash -->
<video width="320" height="240" controls='controls'>
  <source src="http://www.w3schools.com/html/movie.mp4" type="video/mp4">
  <source src="http://www.w3schools.com/html/movie.ogg" type="video/ogg">
  <object data="http://www.w3schools.com/html/movie.mp4" width="320" height="240">
    <embed src="http://www.w3schools.com/html/movie.swf" width="320" height="240">
  </object>
</video>
```

INFORMACJA: Podobnie do zawartości <audio>, tak i <video> musi posiadać pliki w różnych formatach. I tym razem to projektant musi zadbać o konwersję materiałów pomiędzy poszczególnymi formatami. Konwersji można dokonać poprzez odpowiednie strony internetowe:
- <http://video.online-convert.com/>
- <http://www.clipconverter.cc/>
lub poprzez odpowiednie programy, jak Windows Media Player, VLC Media Player czy też Any Video Converter.

W przypadku filmów projektant ma jeszcze dwa inne wyjścia.

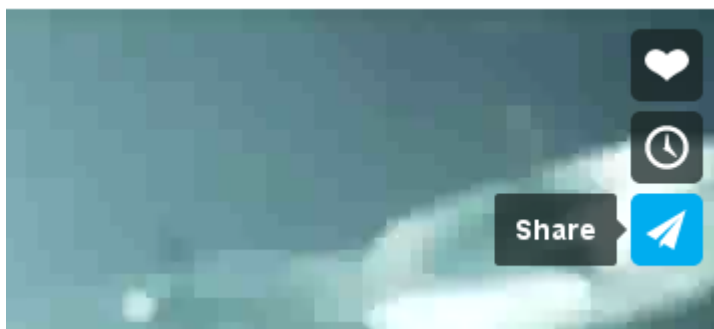
Pierwszym z nich jest dodanie odsyłacza do materiału filmowego. Wtedy jeżeli przeglądarka nie będzie w stanie go wyświetlić otworzy program systemowy powiązany z danym rozszerzeniem pliku bądź po prostu zapisze go na dysk.

Przykład:

```
<a href="http://www.w3schools.com/html/intro.swf">Zobacz film</a>
```

Drugim, o wiele sprytniejszym, a zarazem mniej pracochłonnym, jest wstawienie filmu np. na serwis www.vimeo.com, www.youtube.com bądź dailymotion.com. Następnie wystarczy wstawić swój filmik na stronę poprzez wklejenie odpowiedniego fragmentu kodu.

Wstawianie filmu z serwisu vimeo.com:



Odtwarzając dowolny film (czyjś bądź własny) umieszczamy kursor myszy w prawym górnym rogu filmu. Pojawią się trzy przyciski. Nas interesuje ten z papierowym samolotem – zaznaczony na powyższym zrzucie.

Embed – Hide options

```
<iframe src="//player.vimeo.com/video/89606656" width="500" height="281" frameborder="0" webkitallowfullscreen mozallowfullscreen allowfullscreen></iframe> <p><a href="http://vimeo.com/89606656">Arch Enemy - War Eternal</a> from <a href="http://vimeo.com/user23125610">Jennifer santis</a> on <a href="https://vimeo.com">Vimeo</a>.</p>
```

Size: × pixels

Color: or

Intro: Portrait Title Byline

Special stuff:

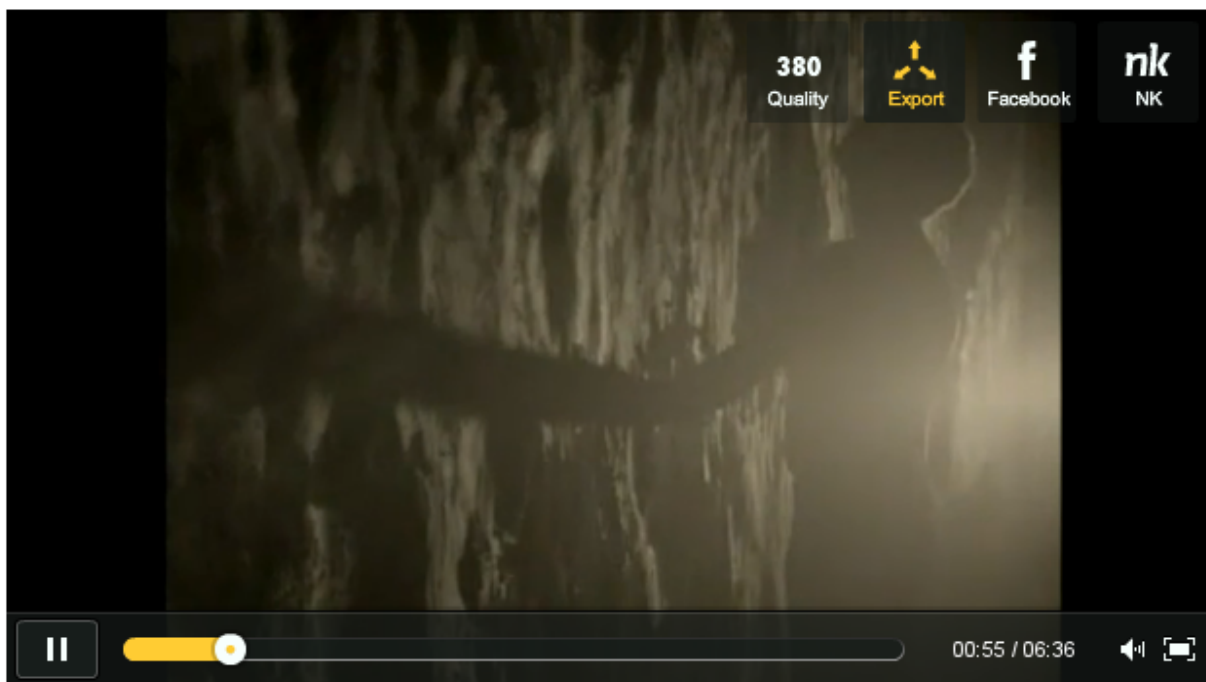
- Autoplay this video.
- Loop this video.
- Show text link underneath this video.
- Show video description below video.

Wyświetli się prosty edytor eksportu odtwarzacza. Można zmieniać jego rozmiar, napisy początkowe, kolory itp. Gdy zakończymy zmiany wystarczy skopiować zaznaczony kod w sekcji Embed i umieścić go na swojej stronie:

```
<iframe src="//player.vimeo.com/video/89606656" width="500" height="281" frameborder="0" webkitallowfullscreen mozallowfullscreen allowfullscreen></iframe> <p><a href="http://vimeo.com/89606656">Arch Enemy - War Eternal</a> from <a href="http://vimeo.com/user23125610">Jennifer santis</a> on <a href="https://vimeo.com">Vimeo</a>.</p>
```

Oczywiście kod ten można zmodyfikować wedle własnego uznania.

Wstawianie filmu z serwisu dailymotion.com:



Metallica The Unforgiven 2

Serwis posiada dwie metody pobierania kodu odtwarzacza. Pierwsza z nich prezentuje się podobnie jak w przypadku vimeo – najeżdżamy na górną część filmu. Pojawiają się 4 ikony, z których należy wybrać „Export”, a następnie „Embed”. Potrzebny kod zostanie skopiowany do schowka – wystarczy wkleić do w kod strony.

Drugim sposobem jest wybranie zakładki Eksport znajdującej się pod filmem:

Info **Eksport** Dodaj do

Permalink

Krótki URL

Pobierz kod

```
<iframe frameborder="0" width="480" height="270" src="//www.dailymotion.com/embed/video/xq989n" allowfullscreen></iframe><br /><a href="http://www.dailymotion.com/video/xq989n_metallica-the-unforgiven-2_music" target="_blank">Metallica The Unforgiven 2</a><i> przez <a href="http://www.dailymotion.com/zozzoni" target="_blank">zozzoni</a></i>
```

Tryb autoodtwarzania

Zaczyna się o

Rozmiar playera

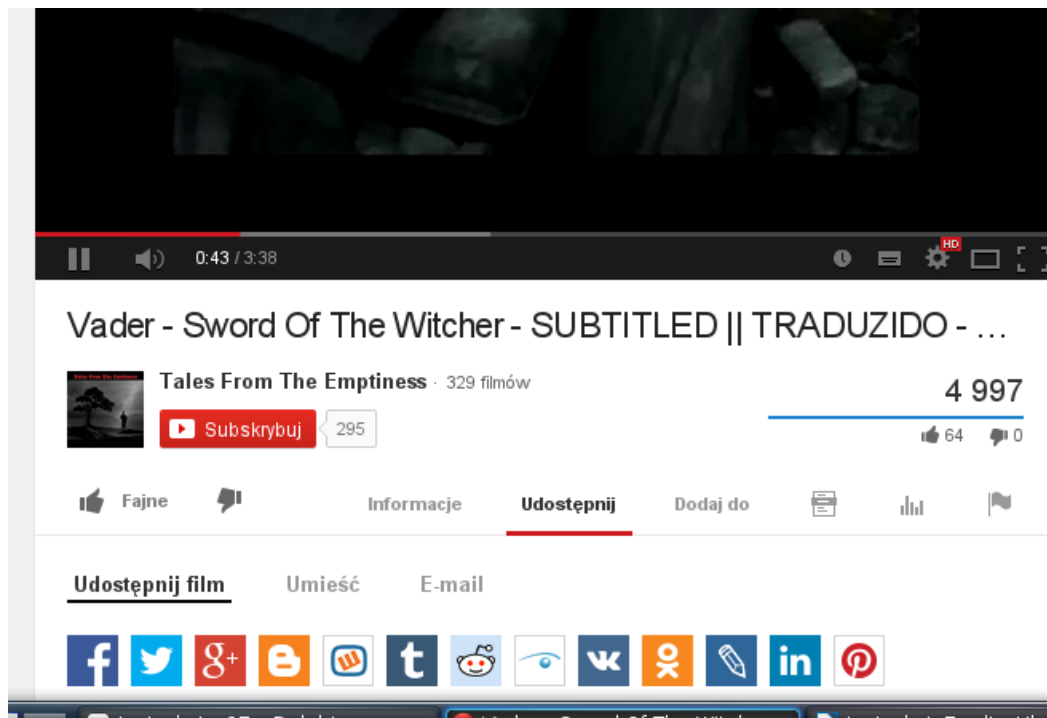
- średnie (480x270)
- Małe (320x180)
- średnie (480x270)
- Duże (560x315)
- Standardowa szerokość

[Więcej opcji & podgląd »](#)

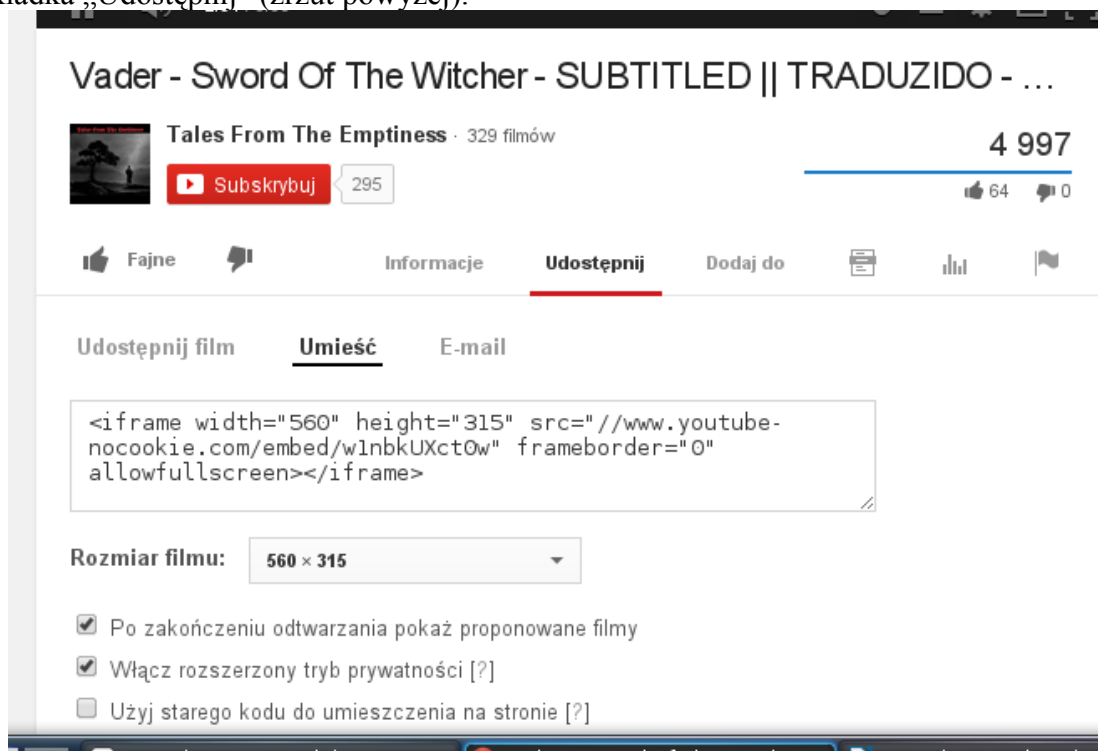
Tak jak w przypadku vimeo możemy zmienić opcje filmu (więcej opcji po kliknięciu „Więcej opcji & podgląd”). Po wszystkim wystarczy skopiować zawartość pola „Pobierz kod”.

```
<iframe frameborder="0" width="480" height="270"
src="//www.dailymotion.com/embed/video/xq989n" allowfullscreen></iframe><br /><a
href="http://www.dailymotion.com/video/xq989n_metallica-the-unforgiven-2_music"
target="_blank">Metallica The Unforgiven 2</a> <i> przez <a
href="http://www.dailymotion.com/zozzoni" target="_blank">zozzoni</a></i>
```

Wstawianie filmu z serwisu youtube.com:



Wstawianie odtwarzacza jest tak samo proste jak w dailymotion. Pod oglądanym filmem znajduje się zakładka „Udostępnij” (zrzut powyżej).



Po jej aktywowaniu wystarczy wybrać opcję „Umieść”. Otrzymamy kod pływającej ramki, który

wystarczy skopiować i wkleić na naszą stronę. Ciekawą opcją jest zaznaczenie opcji „Włącz rozszerzony tryb prywatności”. Dzięki niemu przeglądający filmik na naszej stronie pozostanie „anonimowy” dla serwisu youtube (nie będą zapisywane żadne ciasteczka oraz preferencje z tego odtwarzacza).

```
<iframe width="560" height="315" src="//www.youtube-nocookie.com/embed/w1nbkUXct0w"
frameborder="0" allowfullscreen></iframe>
```

3. Podstawy tworzenia własnego odtwarzacza muzyki/filmów – podstawowe funkcje JavaScript.

JavaScript daje możliwość tworzenia od podstaw własnego odtwarzacza muzyki. Do użytku projektanta oddane odpowiednie funkcje (metody), właściwości oraz zdarzenia. Niestety nie wszystkie są jeszcze poprawnie obsługiwane przez przeglądarki (sporo z nich w ogóle nie posiada zaimplementowanej obsługi). Tak czy inaczej już na chwilę obecną można tworzyć własne listy utworów/filmów, tworzyć własne paski nawigacji po utworach itp.

Lista działających metod:

String canPlayType(typ_formatu) – funkcja pozwala na szybkie sprawdzenie czy przeglądarka, na której została wywołana, obsługuje wskazany format audio bądź wideo.

Możliwe wartości parametru *typ_formatu*:

- video/ogg
- video/mp4
- video/webm
- audio/mpeg
- audio/ogg
- audio/mp4
- video/ogg; codecs="theora, vorbis"
- video/mp4; codecs="avc1.4D401E, mp4a.40.2"
- video/webm; codecs="vp8.0, vorbis"
- audio/ogg; codecs="vorbis"
- audio/mp4; codecs="mp4a.40.5"

Wartości, które może zwrócić funkcja:

- probably – prawdopodobnie obsługuje („najpewniejsza” forma odpowiedzi)
- maybe – chyba obsługuje (sama przeglądarka jest niepewna swoich możliwości...)
- brak wartości (zwrócona wartość "") - nie obsługuje

load() - funkcja ładuje ponownie dane źródło; powodem przeładowania może być ponowna chęć odtworzenia muzyki/filmu bądź odtworzenie nowego pliku (zmiana źródła nie spowoduje automatycznie ponownego odtworzenia filmu).

play() - funkcja rozpoczyna/wznawia odtwarzanie zawartości pliku od ostatniego punktu zatrzymania

pause() - zatrzymuje odtwarzanie zawartości pliku

Lista dostępnych właściwości multimediiów:

- autoplay – określa czy dane źródło ma być automatycznie odtwarzane czy nie; przyjmuje wartości *true* lub *false*
- buffered – przechowuje zakres czasowy zbuforowanego fragmentu pliku audio/wideo. Jeżeli łączna długość źródła jest duża i użytkownik przełączał się pomiędzy poszczególnymi fragmentami

pliku to właściwość będzie przechowywać je wszystkie jako oddzielne części. Właściwość posiada następujące parametry:

- length – zawiera informację ile zakresów jest aktualnie przechowywanych w zmiennej buffered
- start(indeks) – zwraca jako float (zmiennoprzecinkowa wartość) czas (w sekundach) rozpoczęcia zakresu zbuforowanego fragmentu audio/wideo (numer zakresu określa wartość indeks = 0,1,2...,n)
- end(indeks) – zwraca jako float czas zakończenia zakresu zbuforowanego fragmentu audio/wideo

- controls – przechowuje informację czy do znacznika multimedialnego jest dołączony pasek nawigacyjny; możliwe wartości to true lub false.

- currentSrc – przechowuje adres pełnej ścieżki do aktualnie odtwarzanego źródła

- currentTime – przechowuje czas (w sekundach) aktualnej pozycji odtwarzania (np. 30 sekunda piosenki). Wartość można zmieniać poprzez podanie innej niż aktualna

- defaultPlaybackRate – poprzez tę właściwość można zmieniać szybkość odtwarzania audio/wideo; możliwe wartości to:

- 1.0 – odtwarzanie z normalną prędkością

- 0.5 – odtwarzanie z połową prędkości

- 2.0 – odtwarzanie z podwójną prędkością

- -0.5 – odtwarzanie z połową prędkości, wstecz

- -1.0 – odtwarzanie z normalną prędkością, wstecz

- duration – zwraca pełną długość utworu umieszczonego w znaczniku multimedialnym

- ended – zwraca wartość true lub false w zależności, czy odtwarzanie utworu zakończyło się lub nie

- loop – określa czy utwór ma być odtwarzany w pętli (true) lub nie (false)

- mediaGroup – pozwala na przypisanie utworów do pewnego zbioru (z odpowiednią nazwą); pomocne przy tworzeniu listy odtwarzania

- muted – określa, czy dźwięk w ma być wyciszony (true) bądź odtwarzany normalnie (false)

- networkState – określa stan załadowania pliku audio/wideo. Możliwe wartości:

- 0 (NETWORK_EMPTY) – źródło nie zostało jeszcze zainicjowane (nie rozpoczęło się jego pobieranie)

- 1 (NETWORK_IDLE) – źródło jest aktywne lecz nie używa sieci (zostało pobrane/odtwarzany jest zasób lokalny)

- 2 (NETWORK_LOADING) – przeglądarka pobiera plik

- 3 (NETWORK_NO_SOURCE) - nie znaleziono źródła

- paused – określa, czy odtwarzany materiał jest wstrzymany (true) czy odtwarzany (false)

- playbackRate – patrz defaultPlaybackRate (lecz tylko względem aktualnego źródła)

- played – podobnie jak przy buffered lecz określa zakresy, w których użytkownik REALNIE przesłuchał/obejrzał materiał (a nie tylko go zbuforował)

- preload – patrz opis preload przy znaczniku audio

- readyState – przechowuje informację o gotowości materiału do odtworzenia; możliwe wartości:

- 0 (HAVE_NOTHING) – brak informacji o gotowości źródła (zapewne jego brak)

- 1 (HAVE_METADATA) – metadane (długość, nazwa, zbuforowanie itp.) o źródle są dostępne

- 2 (HAVE_CURRENT_DATA) – dane do aktualnie wyświetlanej ramki są dostępne lecz dla następnej już nie (są np. w trakcie pobierania)

- 3 (HAVE_FUTURE_DATA) – dane aktualnej oraz co najmniej następnej ramki są już zbuforowane przez przeglądarkę

- 4 (HAVE_ENOUGH_DATA) – przeglądarka posiada wystarczającą ilość danych ze źródła by rozpocząć odtwarzanie

- seekable – określa zakres części filmu, do którego użytkownik może się przełączyć (czyli

fragment, który został już zbuforowany do pamięci podręcznej użytkownika).

- src – tak jak w przypadku atrybutu src
- textTracks – daje dostęp do ścieżek tekstowych dostępnych dla danego filmu; ścieżki te mogą być np. napisami do filmu (tłumaczenie) lub stanowić komentarz do fragmentu obrazu.
- volume – pozwala na regulację głośności filmu; przyjmuje liczby z zakresu 0.0 – 1.0 (gdzie 0.0 to 0%, a 1.0 to 100% głośności)

Zdarzenia multimedialne działają tak samo jak pozostałe zdarzenia poszczególnych znaczników HTML (click, load, unload itp.). Oznacza to, iż jeżeli np. opisywane jest zdarzenie **play** to jego odpowiednikiem w znaczniku HTML będzie **onplay** (analogicznie do **click** → **onclick**)

Lista dostępnych zdarzeń:

- abort - zachodzi gdy ładowanie pliku zostało przerwane (błąd połączenia z siecią itp.)
- canplay – występuje gdy zasób jest możliwy do odtworzenia (został pobrany do pamięci podręcznej)
- canplaythrough – zostaje wywołane gdy przeglądarka ma wystarczającą ilość danych by odtworzyć film/muzykę
- durationchange – wywołuje się gdy zostanie zmieniony czas trwania nagrania
- emptied – zostaje wywołane gdy bieżąca lista jest pusta
- ended – występuje gdy aktualnie odgrywana lista zakończyła się
- error - wywoła się w przypadku błędów przy ładowaniu pliku
- loadeddata – wywołuje się gdy przeglądarka załaduje aktualną ramkę filmu
- loadedmetadata – wywołuje się gdy przeglądarka pobierze metadane utworu (tytuł, długość, napisy itp.)
- loadstart – wywołuje się gdy przeglądarka zaczyna pobierać plik z multimediami
- pause - zachodzi gdy użytkownik wstrzyma odtwarzanie
- play - zachodzi gdy użytkownik zacznie odtwarzanie
- playing – zachodzi gdy zasób jest gotowy do otworzenia po zatrzymaniu odtwarzania
- progress – występuje gdy przeglądarka pobiera plik multimedialny
- ratechange – zachodzi gdy użytkownik zmienia prędkość odtwarzania
- seeked – zachodzi po ręcznej zmianie punktu odtwarzania pliku (zmiana na pasku postępu)
- seeking – występuje w trakcie przenoszenia punktu odtwarzania pliku
- stalled – zachodzi gdy dane multimedialne dla zasobu nie są dostępne
- suspend – występuje gdy przeglądarka zaprzestała pobierania danych filmu/muzyki
- timeupdate – zdarzenie wywoła się przy zmianie (dodaniu/odjęciu) zasobów odtwarzania w liście
- volumechange – występuje przy zmianie poziomu głośności
- waiting – zdarzenie występuje gdy odtwarzanie zatrzymuje się w oczekiwaniu na zbuforowanie kolejnej części materiału