



# HTML5 versus Flash – możliwości tworzenia interaktywnych multimediiów dla m-learningu

dr inż. JOLANTA BRZOSTEK-PAWŁOWSKA

Instytut Maszyn Matematycznych, Warszawa

Poziom rozwoju standardu HTML5 [4] oraz technologii tworzących środowisko rozwoju i udostępniania aplikacji i treści w tym standardzie jest na tyle zaawansowany, że warto bliżej rozpoznać tę technologię i zacząć stosować w nowych projektach. Większość przeglądarek WWW, tzw. Internetu stacjonarnego (komputery stacjonarne, laptopy) w znacznym stopniu wspiera już HTML5, słabiej radzą sobie z pełnym wykorzystaniem HTML5 przeglądarki mobilne ze względu na ograniczenia sprzętu mobilnego. HTML5 uznawany jest jako idealny dla urządzeń mobilnych, ponieważ może zapewniać dostępność ze wszystkich platform mobilnych. A brak tej dostępności jest zmorą wszystkich deweloperów tworzących rozwiązania na sprzęt mobilny.

Konieczność tworzenia kolejnych wersji produktu dla różnych środowisk mobilnych jest czynnikiem pogarszającym wskaźniki efektywności produkcji. HTML5 wykonuje samodzielnie funkcje, dotychczas wspomagane różnymi wtyczkami do przeglądarek („playery”, „viewery”). Dokładniej mówiąc – samodzielnie wraz z przyjaciółmi, jakimi są kaskadowe arkusze stylów CSS level 3 (CSS 3) i Javascript (JS). W HTML5 nie istnieje problem konieczności instalowania wtyczek lub ich niedostępności, czego przykładem może być technologia Flash, niedopuszczona przez Steve Jobsa (Thoughts on Flash, [2]) na urządzenia iPhone i iPad firmy Apple, jako nie otwarty standard. Apple, Google, nawet Microsoft ze swoją technologią Silverlight (konkurencyjną do Flash), idą wyraźnie w kierunku HTML5, CSS3 i JavaScript. Apple i Google stworzyły nawet strony dedykowane promocji HTML5 <http://www.apple.com/html5/> i <http://www.html5rocks.com/>. Na tej ostatniej stronie o nazwie „HTML5 Rocks” Google prezentuje rozwiązania, które mogą być wykonane przy pomocy HTML5 (oraz CSS3 i JS) takie, jak: interaktywne prezentacje, przewodniki, „sandboxy” (generowanie gradientów i innych wizualnych obiektów i efektów) i inne nowości HTML5.

Wydaje się, że HTML5 może stać się alternatywą dla Flash, zwłaszcza w tworzeniu aplikacji i treści na urządzenia mobilne. Ma poparcie nie tylko Apple, Google, Microsoft, ale również wspierają go platformy Mozilla, Opera i BlackBerry. Przeczy tej tezie jednak dominująca na profesjonalnych forach dyskusyjnych opinia, że ogłaszanie śmierci Flash jest przedwczesne. Zwłaszcza w zastosowaniach edukacyjnych, gdzie ok. 80% interaktywnych treści edukacyjnych jest wykonana we Flashu. Flash nie wymaga znajomości programowania, co jest istotne dla rzeszy dydaktyków medialnych tworzących interaktywne szkolenia e-learningowe z wykorzystaniem narzędzi autorskich opartych na Flash [1]. HTML5 wymaga znajomości programowania w JScript, modelu DOM współpracy strony z przeglądarką oraz skomplikowanego języka zapisu stylów CSS.

Rzeczywiście, HTML5 nie jest narzędziem do bezpośredniego używania przez dydaktyków. W obszarze edukacji internetowej nie stanie się raczej technologią popularną, mimo że rozwiązuje w dużym stopniu problemy cross-platformowości mobilnych treści. Dotychczasowa praktyka wskazuje, że kłopoty z niedostępnością Flasha na iDevices (iPhone, iPod, iPad) rozwiązywane są

często przez przepisywanie treści dla tych urządzeń z wykorzystaniem technologii Apple XCode.

## Potrzeby i oczekiwania m-learningu

Dziś można już powiedzieć, że oczekiwania użytkowników urządzeń mobilnych w zakresie wsparcia szkoleniowego i wsparcia wiedzą są inne, niż oczekują tego na swoich komputerach stacjonarnych (i laptopach). Również producencki kierunek „lilipucenia” treści szkoleń dostępnych w Internecie stacjonarnym poprzez udostępnianie ich na urządzeniach mobilnych jest przeszłością. Z urządzenia mobilnego nikt nie zechce uczestniczyć w całym procesie szkoleniowym „krok po kroku”, który trwa długo i dostarcza wiele treści. Z urządzenia mobilnego sięga się po krótkie kawałki wiedzy, dostosowane do kontekstu sytuacyjnego i lokalizacyjnego. Dla użytkowników urządzeń mobilnych potrzebne są „just-for-me bite-size learning modules”.

Proces uczenia się mobilnego użytkownika „krok po kroku” wyparty został przez proces uczenia ciągłego, formalnego i nieformalnego, gdziekolwiek on jest. Zwłaszcza, gdy jest na stanowisku pracy, nieodzowne staje się wsparcie (*performance support*) kontekstową wiedzą, na żądanie użytkownika, gdy tylko taka potrzeba powstanie. Trzeba jednak zaznaczyć, że rozwój wsparcia pracownika w organizacjach zaznacza się odchodzeniem od tworzenia i dostarczania li tylko treści, a raczej ukierunkowuje się na tworzenie całych wydajnościowych ekosystemów organizujących udogodnienia zwiększające kreatywność, innowacyjność, współpracę i produktywność pracowników.

Ekosystemy stają się coraz bardziej inteligentne, potrafią zbierać dane o użytkowniku i coraz bardziej celnie – w czasie, miejscu i w potrzebie – „wystrzelują” do użytkownika porcje informacji. Jednym z kanałów dostępu do ekosystemów są urządzenia mobilne. Z badań opisanych w raporcie *Mobile Learning: Landscape and Trends* [3] wynika, że prognozy zastosowania m-learningu w organizacjach wskazują na pewną stagnację (ok. połowy badanych organizacji nie przewiduje zmian w rozszerzaniu zastosowań m-learningu), wskazując jednocześnie jako 4 główne (opinia ok. 50% badanych) bariery w zakresie rozwoju zastosowań m-learningu – kłopoty w integracji z istniejącymi systemami (i ich zasobami), niski poziom bezpieczeństwa mobilnego kanału, brak standardów i konieczność tworzenia wielu wersji dla różnych platform mobilnych (co łączy się z kosztami). Około połowa badanych organizacji oczekuje możliwości automatycznej adaptacji treści na różne urządzenia mobilne oraz jednokrotnego tworzenia treści udostępnianej później na różnych urządzeniach mobilnych). Wiadąc, że jedną z głównych bolączek m-learningu jest brak uniwersalizmu (*cross-platforms*) tworzonych zasobów.

## Wideo w m-learningu

Potrzeba łatwych (szybkich i niedrogich) technik tworzenia treści na urządzenia mobilne spowodowała wzrost atrakcyjności i popularności technik audio i video podcastów jako nośników treści informacyjnych. Z drugiej strony, ta atrakcyjność powodowana jest przez:



- wszechobecność wsparcia techniki videocastów realizowanego bezpośrednio przez system operacyjny (iOS Apple) lub łatwo instalowalne aplikacje natywne (*playery*), jak np. w systemie Android,
- raz stworzony podcast może być dostępny z wielu, a nawet wszystkich mobilnych urządzeń.

Innym czynnikiem tej popularności jest ogólny kierunek potrzeby wizualizacji informacji. Elliot Masie, w raporcie z lutego 2012 r. [4] dotyczącym kierunków w e-learningu, odnotowuje dramatyczny wzrost zapotrzebowania na treści w formie wideo, raportowany przez liderów e-learningu z całego świata oraz zauważalne ukierunkowanie się pracowników na wsparcie krótkimi wideo z YouTube. Zauważany jest wzrost roli wideo w webinarach, chatach i telekonferencjach oraz wsparcia pracowników przez coachów na stanowiskach pracy. Stąd widać, że zarówno producenci, jak i odbiorcy preferują wideo jak nośnik treści.

Atrakcyjność technik wideo dla producentów/dostawców treści jest zwiększana dużym stopniem ich kompatybilności (*cross platforms*) dla platform mobilnych. Systemy iOS, Android, Windows Phone, BlackBerry akceptują format wideo mp4. Również dostępne odtwarzacze wideo dla tych platform dobrze sobie radzą z różnymi formatami. Dostosowanie wideo do rozdzielczości i wielkości ekranu może być osiągnięte przez konwersję wideo (z zachowaniem proporcji szerokości i wysokości) lub dodawaniem przez odtwarzacze czarnych pasów poziomych i/lub pionowych do wyświetlanego obrazu. Warto zwrócić uwagę na fakt, że nie zawsze większy ekran smartfona ma większą rozdzielczość (np. Apple iPhone 4 s – przekątna 3,5 cala, rozdzielczość 640×960 px, Samsung Galaxy II S T-mobile – przekątna 4,52 cala, rozdzielczość 480×800 px). Można przyjąć, że rozdzielczość 640×480 px, którą posiadają urządzenia Apple, jest i będzie dostępna na innych urządzeniach mobilnych. Przyjęcie mp4 i 640×480 px zapewni kompatybilność wideo, niekoniecznie zaś jakość obrazu.

## Interaktywne wideo i technologie ich produkcji

Więszym problemem jest uzyskanie na mobilnych platformach kompatybilności wideo, „zawierających” (w uproszczeniu, w rzeczywistości interaktywne elementy są nanoszone na wideo) interaktywne elementy, takie jak linki w obrębie wideo, linki na ze-

wnątrz wideo, przyciski, ankiety, testy. Technika interaktywnych wideo jest atrakcyjna nie tylko dla reklamodawców, którzy linki w wideo mogą wykorzystywać w celu wykonania skoków z wideo, np. do stron WWW z prezentacjami produktów. Interaktywne wideo to jedna z atrakcyjniejszych obecnie technik szkoleniowych przekazywania treści do i pozyskiwania danych od odbiorcy, która angażuje użytkownika oraz pozwala zindywidualizować przekazywane mu treści. Odbiorca szkolenia w formie interaktywnego wideo (również vodcastu) może sam wybrać fragment najbardziej go interesujący lub fragment pomocy (znajdującej się wewnątrz nagrania lub na zewnątrz) uzupełniającej informacje bieżąco wyświetlane.

Taka możliwość indywidualizacji szkolenia zwiększa jego efektywność. Odbiorca szybciej uzyskuje wiedzę (z krótkiego wideo), bardziej adekwatną do bieżącej potrzeby (sam decyduje o wyborze odpowiedniej sekwencji z informacją). Może też przekazywać odpowiedzi na pytania stanowiące ankietę sondującą opinie lub testy mierzące poziom nabytej wiedzy. Rys. 1 przedstawia fragment wideo uczącego rozwiązywania zadania ze stereometrii, z powoływaniem się na podstawy teoretyczne wyświetlane po naciśnięciu linków opatrzonych znakiem „?”. Jeden powoduje pokazane na rysunku wyświetlenie dodatkowej pomocy zawartej w wideo, inny może powodować skok do zewnętrznego URL i wyświetlenie pomocy zapisanej na zewnętrznej stronie WWW.

Problem z kompatybilnością interaktywnych mobilnych wideo wynika z różnych technologii ich wykonywania. Inne są natywne technologie stosowane przez Google dla wideo dla systemu Android (Google SDK, XML plus natywny ładowalny odtwarzacz), inne przez Microsoft dla Windows Phone 7 (media player na bazie Silverlight), inne przez Apple dla iOS. Apple proponuje zaawansowaną technologię Live Stream wbudowanego w strumień, według protokołu Apple **HTTP Live Stream**, pliku metadanych wskazującego na dodatkowe pliki ID3 z tagami ID3 wskazującymi na URL zasobów do wyświetlenia dodatkowych informacji. Protokół **HTTP Live Stream realizowany jest przez wbudowany w iOS „odtwarzacz” – klasę MPMoviePlayerController**, zdefiniowaną w środowisku iOS SDK – Xcode [2]. Przykład interaktywnego wideo wykonanego w technologii Live Stream przedstawia rys. 2.

Obliczanie wysokości  $h$   
Z prostokątnego trójkąta  $\Delta SS'C$  i funkcji trygonometrycznych kąta  $\beta$  możemy obliczyć 1)  $h$  oraz 2)  $a$ .

1) Obliczamy  $h$ :  
Ponieważ kąt  $\beta = 45^\circ$ , a krawędź boczna  $b = 10$ , otrzymujemy:  
 $\frac{h}{10} = \sin 45^\circ$  czyli:  $\frac{h}{10} = \frac{\sqrt{2}}{2}$   
z powyższej proporcji otrzymujemy:  
 $2h = 10\sqrt{2}$   
Tak więc wysokość ostrosłupa wynosi  $h = 5\sqrt{2}$

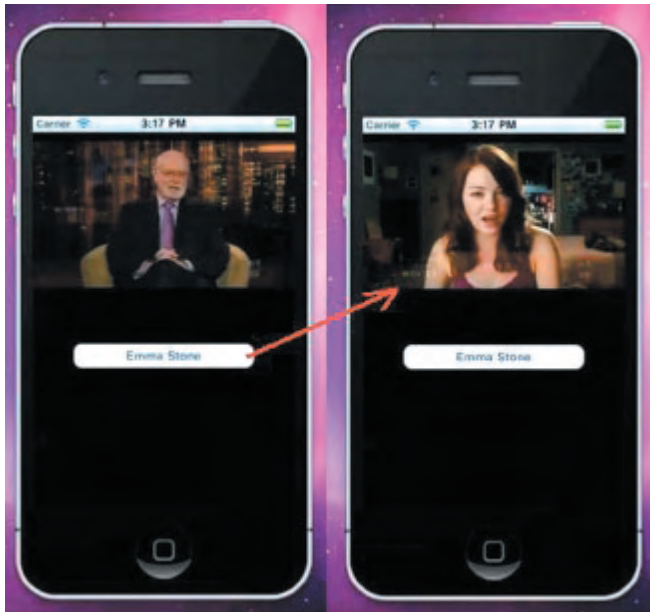
Rys. 1 a. Przykład interaktywnego wideo  
Fig. 1 a. An example of interactive video

Obliczanie wysokości  $h$   
Z prostokątnego trójkąta  $\Delta SS'C$  i funkcji trygonometrycznych kąta  $\beta$  możemy obliczyć 1)  $h$  oraz 2)  $a$ .

1) Obliczamy  $h$ :  
Ponieważ kąt  $\beta = 45^\circ$ , a krawędź boczna  $b = 10$ , otrzymujemy:  
 $\frac{h}{10} = \sin 45^\circ$  czyli:  $\frac{h}{10} = \frac{\sqrt{2}}{2}$   
z powyższej proporcji otrzymujemy:  
 $2h = 10\sqrt{2}$   
Tak więc wysokość ostrosłupa wynosi  $h = 5\sqrt{2}$

Wartości funkcji sinus dla wybranych kątów z I kwadrantu					
$\alpha$	$0^\circ$	$30^\circ$	$45^\circ$	$60^\circ$	$90^\circ$
$\sin$	0	$\frac{1}{2}$	$\frac{\sqrt{2}}{2}$	$\frac{\sqrt{3}}{2}$	1

Rys. 1 b. Przykład dodatkowej informacji w wideo wyświetlanej po naciśnięciu na link „?” wskazany czerwoną strzałką (źródło własne)  
Fig. 1 b. An example of additional information in the video when you press the button “?” indicated with a red arrow



Rys. 2. Interaktywne wideo na iPhone wykonane w technologii Apple Live Stream (naciśnięcie przycisku Emma Stone powoduje prezentację aktorki Emmy Stone) [8]

Fig. 2. Interactive video on the iPhone made in the Apple Live Stream technology (pressing the Emma Stone button causes the presentation of Actress Emma Stone)

Jeszcze inne technologie tworzenia interaktywnych wideo oferują producenci budujący swoje narzędzia w oparciu o technologię Flash poprzez wbudowywanie w stronę html i tworzenie dodatkowych plików.swf, sterujących nanoszeniem interaktywnych obiektów na mp4. Pliki sterujące.swf interpretowane są przez Adobe Flash Player, wtyczkę do przeglądarek WWW.

Firma Adobe zaprzestała rozwoju wtyczek Flash Player do przeglądarek mobilnych (komunikat z jesieni 2011 r.), kierując prace rozwojowe na technologie związane z HTML5, może zgodnie z sugestią Steve'a Jobsa: „*Nowe, otwarte standardy, stworzone w „mobilnej erze” – np. HTML5 – zwyciężą na urządzeniach mobilnych (jak i na komputerach osobistych). Być może Adobe powinno w większym stopniu skupić się na tworzeniu świetnych narzędzi standardu HTML5 patrząc w przyszłość, a w mniejszym – na krytykowaniu Apple za pozostawienie przeszłości tam, gdzie jej miejsce.*”

W związku z tym technologie interaktywnych wideo oparte na Flash dla platform mobilnych mogą się dziś wydać mało perspektywiczne. Można powiedzieć – szkoda, ponieważ praktycznie dla wszystkich platform mobilnych, z wyjątkiem iOS, Flash był dostępny. Czy obecnie opracowywane przez Adobe konwertery Flasha na HTML5 będą w 100% przenosić interaktywność? Dla dydaktyków medialnych byłoby to idealne wyjście z kłopotu stwarzanego przez firmę Apple na jej sprzecz. „Tony” istniejącego interaktywnego contentu edukacyjnego utworzonego we Flashu mogłyby stać się dostępne ze wszystkich urządzeń mobilnych. Zaangażowane dotychczas pieniądze w tworzenie tego contentu oraz doświadczenie we Flashu dydaktyków na całym świecie stanowią silne bariery powstrzymujące szybkie zniknięcie z rynku technologii Flash (i rezygnowanie z przychodów z jej sprzedaży). Wobec silnych oporów użytkowników Flash i inwestorów contentu wytworzonego we Flash przed rezygnacją z tej technologii, warto poznać obecne i przyszłe technologie wspomagające udostępnianie tego contentu na urządzeniach mobilnych, w tym z grupy iDevice.

## Technologie wsparcia Flash

Silny opór użytkowników przed zakończeniem rozwoju Flash i skala kłopotów z uzyskaniem cechy *cross-platforms* mobilnego contentu, wynikających z konieczności dostosowania do rozmiaru i rozdzielczości ekranu oraz problem braku implementacji Flash w niektórych urządzeniach mobilnych, generuje kreatywność twórców innowacyjnych technologii umożliwiających zarówno dostosowywanie interaktywnych treści Flash (animacje, wideo) do rozmiarów i rozdzielczości ekranu urządzenia, jak również pokonania braku implementacji „Flash playera” dla danego urządzenia. Najnowsze (2. połowa 2011 r.) rozwiązania bazują na architekturze przeglądarki „w chmurze”. Wirtualna przeglądarka „w chmurze” składa się z dwóch części – serwera Web z internetową aplikacją, zachowująca się jak „tradycyjna” przeglądarka nieprzycięta do możliwości mobilnego sprzętu, oraz z natywnej aplikacji instalowalnej w urządzeniu mobilnym. Serwer komunikuje się zarówno z natywną aplikacją, jak i pobiera stronę WWW z Flashem ze źródłowego serwera. Pobieranie strony z Flash rozpoczyna natywna aplikacja, która przekazuje do serwera dane o stronie oraz o parametrach technicznych urządzenia. Serwer odtwarza bez ograniczeń Flasha na swojej platformie i do aplikacji natywnej śle już właściwie przetworzony strumień danych, dostosowany do ekranu.

Przykładami takich innowacyjnych wirtualnych platform pośredniczących, rozlokowanych „w chmurze” są Cloud Browse (firmy AlwaysOn Technologies, Inc.) oraz iSwifter (najnowszy start-up YouWeb Incubator). Kierunek rozwoju technologii strumieniujących na żądanie multimedia (audio, wideo) rozwija się silnie – przykładami mogą być iCloud (firmy Apple) i Netflix (nowa firma na giełdzie NASDAQ). A skoro iDevices otrzymują strumień od iSwiftera, to być może w przyszłości będzie mógł być on kierowany za pomocą technologii AirPlay (firmy Apple) do odbiornika AirPlay, jakim jest Apple TV. Historia rozwoju technologii zanotowałyby pewnie ironicznie dziwny case – udostępniania rozwiązań Flash przez internetową telewizję *on-demand* Apple TV.

Podobne technologie przeglądarek „w chmurze” rozwijają się również dla rozwiązań HTML5, dla platform nie implementujących tego standardu. Pomijany w artykule jest oddzielny problem dostosowania platform LMS (*Learning Management System*) do urządzeń mobilnych, który musi być rozwiązywany przez oddzielnie stworzony system lub tylko zmianę interfejsu z użytkownikiem (dla interfejsu w HTML5 być może tylko przez zmianę stylu określonego w CSS3).

## HTML5 – krótka charakterystyka

HTML5 – od niedawna oficjalnie HTML – jest rozwinięciem języka HTML oraz jego odmiany XHTML, jednak jego specyfikacja w wielu miejscach została dopracowana. W odróżnieniu od planowanego, lecz porzuconego w 2009 roku standardu XHTML 2.0, HTML5 zachowuje kompatybilność wsteczną. Rozwijany od początków XXI wieku, formalnie dalej w opracowaniu, uznawany jest – słusznie lub nie – za pogromcę Flasha i idealnie bez barier, dopasowanego do świata różnorodnych urządzeń mobilnych.

HTML5 wprowadza wiele nowości, m.in. kilkadziesiąt nowych tagów związanych z:

- semantyczną warstwą (nowe tagi lepiej organizujące zawartość strony i w związku z tym lepiej wyszukiwaną przez wyszukiwarki internetowe),
- tworzeniem formularzy (autokontrola zawartości pól),
- aspektami multimedialnymi.

Jednak prawdopodobnie najlepszym elementem HTML5 jest zdolność *cache'owania* danych w pamięci podręcznej aplikacji. Jest to przydatne do przechowywania skryptów JavaScript, CSS oraz obrazów w pamięci urządzenia. Dzięki temu, nawet w przy-





padku, kiedy użytkownik zostanie przełączony do trybu offline, większość z wymaganych elementów interfejsu będzie nadal dostępna.

Kolejnym atutem HTML5 jest uniwersalność. Nie tylko dla wielu platform mobilnych, ale również dla wielu różnych urządzeń w obrębie tej samej platformy. Utrzymanie semantyczności kodu oraz stworzenie dobrze wyglądającej strony bez użycia JavaScriptu i kaskadowych arkuszy daje może nie gwarancję, ale duże szanse zachowania dobrego wyglądu strony HTML5 bez względu na urządzenie mobilne, które ją pobrało.

## Multimedia w HTML5

Ponieważ artykuł skupia się głównie na wizualnych, interaktywnych możliwościach przekazu informacji, w tym treści szkoleniowych na urządzenia mobilne, warto przedstawić główne tagi HTML5 związane z multimediami.

### Wideo i audio

HTML5 wprowadza możliwość osadzania plików multimedialnych za pomocą samego HTML. Nie będą już do tego wymagane technologie takie jak Flash ani inne wtyczki (*plug-in*). Przykład kodu HTML5 z tagiem `video` i z tagiem `audio`, dalej omówionymi, podano na rys. 3.

```
<video controls=" controls" width=" 640" height=" 480" poster="
GrafikaNaStarcie.jpg">
  <source src=" MojeVideo.mp4" type=" video/mp4"/>
  <source src=" MojeVideo.webm" type=" video/webm"/>
  <source src=" MojeVideo.ogv" type=" video/ogg"/>
  <object type=" application/x-shockwave-flash" width=" 640"
height=" 480" data=" player.swf?file=MojeVideo.mp4">
  <param name=" movie" value=" player.swf?file=MojeVideo.
mp4"/>
  <a href=" MojeVideo.mp4">Pobierz plik</a>
</object>
</video>

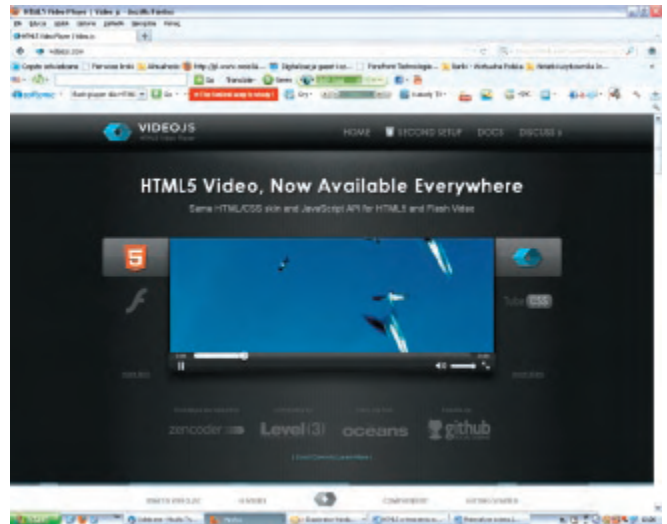
<audio controls=" controls">
  <source src=" Wyklad.ogg" type=" audio/ogg"/>
  <source src=" Wyklad.mp3" type=" audio/mpeg"/>
  <object type=" application/x-shockwave-flash" data=" player.
swf?soundFile=Wyklad.mp3">
  <param name=" movie" value=" player.swf?soundFile=Wyklad.
mp3"/>
  <a href=" Wyklad.mp3">Pobierz plik</a>
</object>
</audio>
```

**Rys. 3. Przykład kodu HTML5 z wideo i audio akceptowanego przez większość przeglądarek WWW z pozycji 2–8 tabeli**  
**Fig. 3. Example HTML5 video and audio code that are accepted by most Web browsers, items 2–8 of Table**

Atrybuty tagu `video` pozwalają określić rozmiary wideo na ekranie (*width*, *high*), dodanie przez przeglądarkę na wideo sterujących kontrolki o wyglądzie zależnym od przeglądarki (*controls*), zawartość pierwszej klatki (*poster*), typ MIME (*type*) i nazwę pliku źródłowego wideo (*source*), odtwarzacz Flash i plik mp4 dla niego (*object*).

Inne atrybuty tagu `video` umożliwiają:

- `autoplay` (): automatyczny start po załadowaniu strony,
- `loop` (): odtwarzanie przez przeglądarkę wideo w pętli,
- `preload` ('*auto*', '*metadata*', '*none*'): wartości tego atrybutu pozwalają określić, czy całe wideo ma być ładowane (*auto*) pod-



**Rys. 4. Przykład wideo wbudowanego w stronę HTML5 [11]**  
**Fig. 4. Example of embedded video in HTML5 page**

czas ładowania strony, czy tylko jego metadane (*metadata*) lub, że w ogóle w czasie ładowania strony nie będą ładowane ani wideo, ani jego metadane (*none*); większość przeglądarek przez domniemanie przyjmuje wartość *metadata*.

Podobne atrybuty są dla tagu `audio`.

Przykład „natywnego” wideo mającego wsparcie w samej przeglądarce przedstawia rys. 4.

Nadal jednak nie jest zaaprobowany przez rynek dostawców standard kodowania multimedialnych w HTML5 (kodeki wideo i audio). Grupa Robocza opracowująca standard HTML5 już w 2009 r. zrezygnowała z wysiłku ustalenia grupy jednolitych kodeków ze względu na niemożność uzgodnienia jej z największymi producentami rynkowymi. Trudności w ujednoczeniu kodeków wynikają z silnych obaw producentów, co do możliwości obudzenia się tzw. śpiących patentów (lub patentu) po przyjęciu jakiegoś kodeka jako obowiązującego w HTML5 i po zrealizowaniu w oparciu o niego produktów. Obecną sytuację wsparcia przez przeglądarki WWW HTML5 i Flash oraz ich udział w rynku podaje tabela. Zestawione w niej zostały wspierane formaty wideo, z pominięciem formatu Ogg Theora z kontenera Ogg, wspieranego przez Google dla rozwiązań mobilnych oraz kontrowersyjnego formatu

Udział w rynku przeglądarek WWW i wsparcie przez nie HTML5  
 The market share of web browsers support the HTML5

Przeglądarka WWW	Udział w rynku	Wsparcie HTML5	Wsparcie Flash
1. Internet Explorer 6/7/8	31%	Nie	Tak
2. Internet Explorer 9	25%	Tak(MP4)	Nie
3. Firefox	25%	Tak(WebM)	Tak
4. Chrome	21%	Tak (MP4+WebM)	Tak
5. Internet Explorer 9	8%	Tak (MP4)	Tak
6. iOS	4%	Tak (MP4)	Nie
7. Safari	3%	Tak (MP4)	Tak
8. Android	2%	Tak (MP4)	Być może
9. Opera	2%	Tak (WebM)	Tak
10. BlackBerry	1%	Nie	Nie



H.264 wspieranego przez Microsoft. Format H.264 nie jest w pełni bezpłatny i nie znajduje poparcia Google. W tabeli pominięte zostały formaty, które mogą być odtwarzane przez przeglądarki po zainstalowaniu wtyczek (np. WebM w Internet 9, lub H.264 w Chrome).

Rozwiązaniem problemu braku jednolitych kodeków dla wideo i audio jest przygotowywanie plików wideo i audio w formatach wspieranych przez przeglądarki. Trzeba skonstatować, że HTML5 nie ułatwia producentom procesu przygotowania wideo i audio. Natomiast upraszcza użytkownikom końcowym korzystanie ze stron utworzonych w HTML5 zawierających wideo lub audio, ponieważ HTML5, bez względu na urządzenie dostępne jest dla nich transparentny pod warunkiem, że zostaną uprzednio zrobione wersje plików wideo/audio i podane jako wartości atrybutów *source* (rys. 3) tagu *video* i/lub *audio*.

### Grafika rastrowa

Tag *canvas* definiuje kontener grafiki – jest to prostokątny obszar grafiki rastrowej, w którym można rysować za pomocą JavaScript.

Funkcjonalność oferowana przez *canvas*:

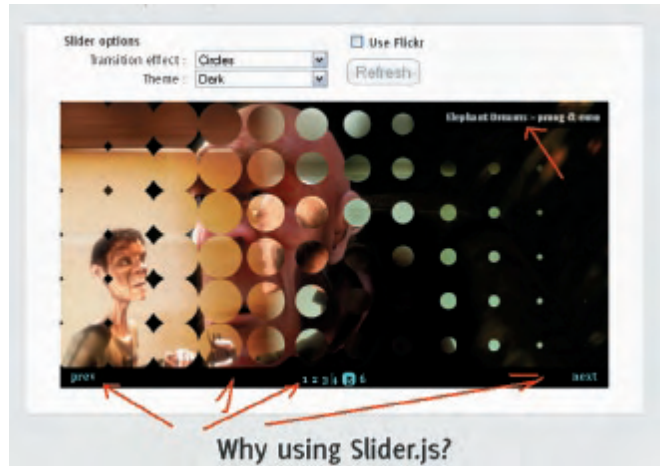
- rysowanie prostokątów i ścieżek (linie, łuki, krzywe Beziera lub krzywe drugiego stopnia),
- wypełnianie figur kolorem (przezroczystość), wzorem lub gradientem (liniowym lub radialnym),
- rysowanie napisów (lider),
- przekształcenia (przesuwanie, skalowanie, obracanie, przekrzywienie itp.),
- osadzanie obrazów rastrowych (PNG, JPEG, GIF),
- cieniowanie,
- style kompozycji określające, jak nowe treści są rysowane na istniejących składnikach *canvas*.

Typowe zastosowania *canvas*:

- ◆ grafy i wykresy,
- ◆ gry,
- ◆ aplikacje.

Tag *canvas* wprowadza wiele możliwości, wymaga jednak programowania w JavaScript i dla edukatorów, aby mogli bezpośrednio wykorzystywać ten silnik animacji jest trudny, użytkownik musi operować na zbyt niskim poziomie technicznym, aby osiągnąć efekty. Powstające biblioteki JavaScript, np. LiquidCanvas czy Rgraph, w ramach modelu DOM ułatwiają pracę, dają już gotowe funkcje rysowania, animowania grafiki i inne funkcje. Przykładem jeszcze jednej biblioteki jest *slider.js*. Jest to biblioteka jQuery, wykorzystująca style CSS Transition i tag *canvas*, przy pomocy której budowane są bardzo popularne obecnie „slajdery” – silniki do przeglądania obrazów/slajdów i nawigacji po nich. Przykład „slajdera” przedstawia rys. 6. Slidery służące do prezentowania kolejnych obrazów, treści mają dostępne elementy nawigacji, takie jak wybór pozycji lub przyciski *next/prev*, przycisk *pause* oraz bardzo pożyteczny *hot spot* – link do realizacji skoku na zewnątrz „slajdera”. Ten prosty stosunkowo silnik może być bardzo użyteczny dla celów przekazywania treści szkoleniowych. Na przykład kolejne odsłony (slajdy) „slajdera” prezentują podstawowe treści, zaś ich rozszerzenie lub pomoce wykonywane mogą być poprzez skoki na zewnątrz. Przykłady „slajderów” i w różny sposób zrealizowanych dla nich elementów interaktywnych przedstawiają rys. 7 i 8.

„Slajdery” są analogiem interaktywnych wideo, np. nagranych z prezentacji Power Point i opatrzonych napisami i interaktywnymi elementami Flasha lub XML z JavaScriptem (w zależności od użytych narzędzi produkcyjnych). Mają tę przewagę, że są transplatformowe, trans-przeglądarkowe.



Rys. 6. Przykład „slajdera” w momencie wykonywania przejścia między slajdami (według CSS Transitions). Strzałki oznaczone przez „1” wskazują elementy nawigacji, strzałka na górze wskazuje link skoku do zewnętrznego URL [12]

Fig. 6. Slider in the moment of the transition between slides (by CSS Transitions). Arrows marked by “1” indicates the navigation items, the arrow on the top indicates the link of the jump to an external URL



Rys. 7. Slider z naniesionym tekstem na grafikę, z przyciskami nawigacji i pauzy [12]

Fig. 7. Slider with text to graphics, the navigation buttons and press pause



Rys. 8. Przykład „slajdera” z zaznaczonymi elementami interaktywnymi, w tym menu (z lewej strony) [12]

Fig. 8. Slider with marked interactive elements such as menu (left side)



Canvas może wprowadzić prawdziwą rewolucję w świecie technologii WWW. Pozwala tworzyć interaktywne aplikacje, które mogą z powodzeniem konkurować z odpowiednikami desktopowymi.

## Inne innowacyjne elementy HTML5

Oprócz tagów video, audio i canvas wspierających interaktywne multimedia istnieją w HTML5 i związanej z nim specyfikacji CSS3 oraz w rekomendowanych w specyfikacji HTML5 technologiach, inne innowacyjne elementy, które umożliwią opracowywanie m.in. nowoczesnych narzędzi autorskich dla tworzenia treści w pełni wykorzystujących możliwości HTML5.

### HTML5 i rekomendowane w nim technologie:

- Web Storage – możliwość zapamiętywania przez przeglądarkę dużej ilości danych w czasie sesji lub między sesjami,
- Indexed Databases – zdefiniowany interfejs API (w modelu JScript DOM) do baz danych, zapewnia obsługę baz danych przez przeglądarkę,
- Drag&Drop – specyfikacja HTML5 definiuje mechanizm przesuwania i upuszczania elementów na stronie w wyniku zdarzeń na obiekcie (kliknięcie, najechanie myszy),
- Web Workers – specyfikacja HTML5 definiuje interfejs programistyczny dla przeglądarki umożliwiającej wielowątkowe wykonywanie skryptów JScript (np. naliczania w tle statystyk z testów),
- Cors (*Cross-origin resource sharing*) – umożliwia dostęp z przeglądarki do zasobów zlokalizowanych w różnych URL, nie tylko z lokalizacji strony,
- Data URI – umożliwia (w kodzie html strony lub css) przekazanie do przeglądarki krótkich zawartości binarnych bezpośrednio na stronie (szybciej), a nie poprzez żądanie protokołu http zawierające adres zasobu binarnego (dla wielu krótkich plików lepiej, w celu przyspieszenia, ich zawartość podawać w kodzie strony).

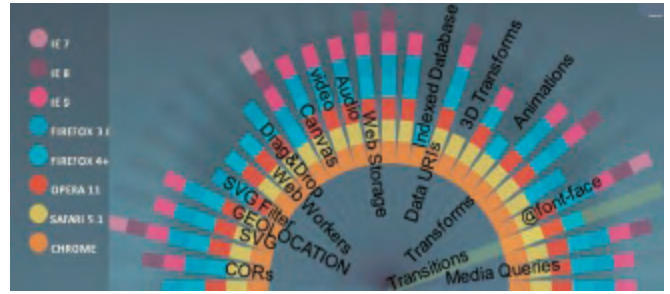
### CSS3

- Media Queries – umożliwia odpytanie przeglądarki o parametry urządzenia, na którym działa, i w wyniku odpowiedzi dostosowanie stylu przekazywanych na stronie treści;
- Transition – określa sposób przejścia między kolejnymi grafikami (slajdami);
- Transform (2D i 3D) – pozwala określić sposób przekształcania obiektu graficznego w dwóch lub trzech wymiarach;
- @font-face – umożliwia zdefiniowanie lokalizacji czcionki do ściągnięcia w sytuacji jej nieobecności (np. symboli matematycznych).

Warto zwrócić uwagę na te elementy HTML5 (Indexed Databases, Web Storage, Web Workers), które mogą mieć zastosowanie w pracy z elektronicznym kursem w trybie offline, bez połączenia z Internetem lub na element Cors, umożliwiający np. pobieranie różnych elementów elektronicznego kursu z repozytoriów w różnych lokalizacjach oraz na elementy (Media Queries, @font-face) dające możliwości dostosowania, niezauważalnego dla użytkownika, stylu stron (treści) do charakterystyki urządzenia.

Rysunek 9 przedstawia stopień wsparcia HTML5, z zaznaczeniem elementów szczególnie użytecznych do tworzenia interaktywnych treści, ich dystrybucji oraz dostępności i atrakcyjności dla użytkownika. Zwraca uwagę niski stopień implementacji innowacyjnych elementów HTML5 w przeglądarkach IE w wersjach poniżej 9. Dopiero od wersji 9 IE MS oficjalnie wspiera HTML5.

Wszystkie zaprezentowane elementy mają względną wadę – są trudne w bezpośrednim użyciu dla twórców treści szkolenio-



Rys. 9. Stopień wsparcia przez przeglądarki WWW elementów HTML5 użytecznych dla tworzenia i udostępniania interaktywnych treści szkoleniowych (źródło własne, na podstawie [10])

Fig. 9. The support degree by the web browser of HTML5 elements useful for creating and sharing interactive training content

wych, którzy na ogół nie są programistami. Dydaktycy medialni i autorzy treści muszą poczekać na narzędzia autorskie oparte na standardzie HTML5, produkujące treści zapisane w formach (np. obiekty SCORM, video, audio) dostosowanych zarówno do różnorodnych urządzeń mobilnych i ich systemów operacyjnych, jak i komputerów stacjonarnych z systemami Windows i OS. Zaczynają już takie środowiska edycyjne powstawać. Na przykład Adobe opracowuje technologię Edge tworzenia interaktywnych treści w HTML5 na urządzenia mobilne, zaprzestając rozwoju od twarzaczy Flasha dla tych urządzeń. Innym przykładem jest seria produktów Claro, zgodnych ze standardem e-learningu SCORM 2004 i AICC, umożliwiających tworzenie treści interaktywnych, opartych na możliwościach HTML5 dla komputerów stacjonarnych i urządzeń mobilnych.

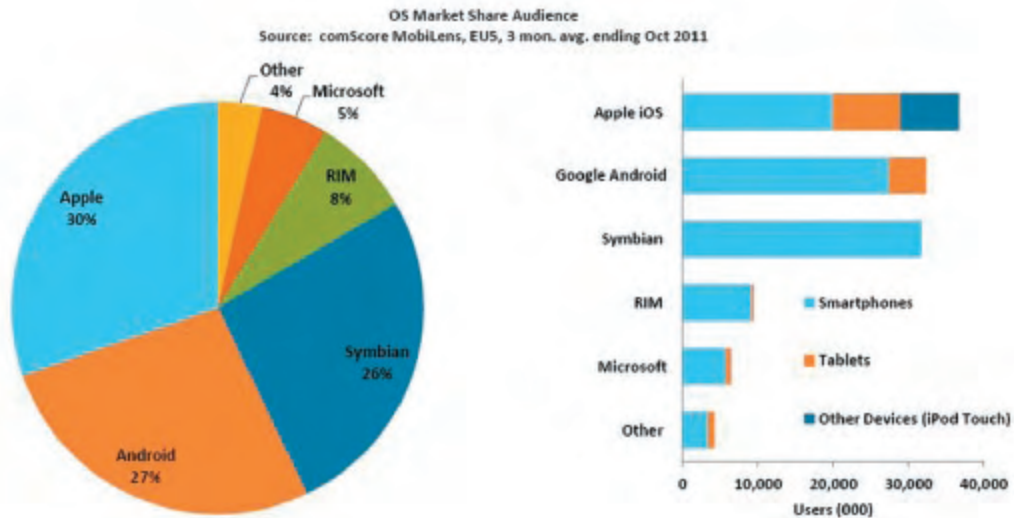
### Podsumowanie

W dyskusji, czy HTML5 jest zabójcą Flasha i czy rozpoczął się początek końca Flasha wraz z początkiem HTML5, można powiedzieć, że tak – rozpoczął się bardzo powolny schyłek Flasha, który potrwa wiele lat dopóty, dopóki:

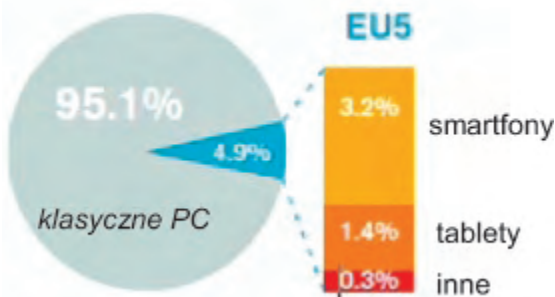
- nie wejdą na rynek i nie upowszechnią się narzędzia tworzące interaktywne, multimedialne treści w HTML5, dziś na rynku są dopiero pierwsze początki,
- nie wejdzie na rynek produkcji kontentu pokolenie ludzi nie mających doświadczeń we Flash, zdobywających na starcie doświadczenie w HTML5,
- nie zostaną „przestawieni” na tory nowej technologii HTML5 i nowych narzędzi ludzie, którzy latami pracowali we Flash,
- nie zostaną wycofane, z powodu procesu starzenia się technologicznego, zasoby z udziałem Flash,
- Adobe nie wypuści narzędzi opartych na HTML5 (nowe wersje Edge?), umożliwiających tworzenie nowych treści w HTML5 na komputery stacjonarne oraz konwertujące istniejące treści oparte na Flash na standard HTML5.

HTML5 jest transparentny dla większości użytkowników końcowych. Odbiorcy treści szkoleniowych – pracownicy, studenci, uczniowie będą mogli bezproblemowo pobierać ze swoich urządzeń mobilnych treści zawierające interaktywne video, czy to w wyniku zastosowania wbudowanego w stronę HTML5 video, czy też w wyniku metody bardziej obciążającej producentów treści, polegającej na przygotowaniu różnych wersji video, aby przeglądarka zdecydowała, którą wersję załadować z serwera. Trudno jednoznacznie stwierdzić, do jakich formatów może ograniczyć się przygotowywanie wersji video. Uwzględniając konieczność instalacji wtyczek do przeglądarek można ograniczyć się do dwóch – w formacie H.264 i OGG Theora, ponieważ te dwa





Rys. 10. Udział w konsumenckim rynku europejskim urządzeń mobilnych i tabletów [5]  
Fig. 10. Participation in the European market mobile devices



Rys. 11. Udział w ruchu w Internecie klasycznych komputerów PC, smartfonów i tabletów [5]  
Fig. 11. Participation in the Internet traffic of the classic PCs, mobile devices and tablet PCs

standardy akceptują prawie wszystkie przeglądarki. Lepszym wyborem, bo bez konieczności wtyczek, w ślad za informacjami zebranymi w tabeli, może być para MP4 i WebM.

W zależności od potrzeby użytkownika, treści będą mogły być dostępne przez przeglądarkę online lub offline bez potrzeby stałego dostępu do Internetu, co jest wygodne zwłaszcza w podróży. Możliwość pracy w trybie offline bez Internetu poprzez przeglądarkę wyprze zapewne natywne odtwarzacze SCORM borykające się z odtwarzaniem wideo niedopasowanych do danego urządzenia mobilnego, zawartych w obiektach szkoleniowych zgodnych ze standardem SCORM.

Raporty z badań przeprowadzonych w okresie sierpień-październik 2011 r., przedstawione na konferencji [5], dotyczące europejskiego (badania przeprowadzono w 5 krajach europejskich) rynku konsumenckiego urządzeń mobilnych wskazują na zwiększanie się w ogromnym tempie używania smartfonów i w zależności od pory dnia, nawet wypieranie klasycznych komputerów PC przez urządzenia mobilne, w tym tablety, nieuznawane za urządzenia mobilnych użytkowników ze względu na realizowany dostęp do Internetu głównie z hot spotów, a nie poprzez 3G. Zanotowano eksplozję na poziomie 330,2% systemu Android mającego w końcu roku 2011 27% udziału w rynku europejskim. Nadal jednak największy udział (30%) w tym rynku ma ekosystem iDevice urządzeń Apple (rys. 10), który wykazuje nadal ekspansję wzrostu na poziomie 54,3%.

Warto zauważyć, że ruch w Internecie w większości powodują użytkownicy z komputerów PC (rys. 11).

Wspomniany raport wykazuje, że ponad 70% użytkowników smartfonów sięga za ich pomocą po treści informacyjne, w tym ponad 60% z nich specjalnie przygotowane na urządzenia mobilne. Łatwo przewidzieć, że ta tendencja wzrostowa będzie kontynuowana, do czego w coraz większym stopniu przyczyniać się będzie wsparcie HTML5 przez Apple, Google i Microsoft i upowszechnianie narzędzi generujących treści w HTML5 dla urządzeń mobilnych.

Penetracja rynku przez urządzenia mobilne w 5 krajach europejskich przy średnim poziomie 42% jest nawet wyższa od poziomu 39% w Stanach Zjednoczonych i cały czas ma kierunek wzrostowy. Specjaliści zaangażowani w produkcję kontentu, zwłaszcza kontentu szkoleniowego, powinni mieć na uwadze, że na fali tego głównego nurtu wzrostu ilości i używalności urządzeń mobilnych dla celów informacyjnych będzie wzrastał apetyt odbiorców na treści krótkie, dokładnie sprofilowane pod potrzeby i niezwłocznie dostarczane użytkownikowi (*just-in-time, just-enough, and just-for-me bite-size learning module*). Uwzględniając w dodatku coraz większe zapotrzebowanie na wizualizację informacji można wskazać na profilowane pod parametry urządzenia (przez przeglądarkę HTML5) wideo w formatach nie wymagających instalowania kodeka, jako jedną z najbardziej popularnych form przekazywania treści w przyszłości.

## Literatura

- [1] Georgetes Ch.: How to Cheat in Adobe Flash CS5: The Art of Design and Animation. Focal Press, 2010.
- [2] Mark D., Nuttinh J., La Marche J.: Beginning iOS 5 Development: Exploring the iOS SDK. Apress, 2011.
- [3] Clark N. Quinn, Ph. D.: M-Learning: Landscape and Trends. The Elearning Guild Research, 2011.
- [4] Learning TRENDS by Elliott Masie, February 1, 2012, #707 – Updates on Learning, Business & Technology, <http://www.masie.com>
- [5] Whitepaper&Slides, "Connected Europe: How Smartphones and Tablets are Shifting Media Consumption" Conference, January 23, 2012, by Digital Life Design (DLD).
- [6] Netografia:
- [7] [www.w3.org/TR/html5/](http://www.w3.org/TR/html5/)
- [8] <http://www.apple.com/hotnews/thoughts-on-flash/>
- [9] <http://vimeo.com/16455248>
- [10] <http://vandelaydesign.com/blog/web-development/jquery-image-galleries/>
- [11] <http://html5readiness.com/>
- [12] <http://videojs.com/>
- [13] <http://sliderjs.org/>