

UPB ETTI

Master IISC

WEB INTERACTIV

Masterand FLOROIU ANDREI

CUPRINS

- 1. Adobe Flash – pg. 3**
 - 1.1. Formatul fişierelor – pg. 4**

- 2. Microsoft Silverlight – pg. 6**
 - 2.1. Versiuni – pg. 7**
 - 2.2. Moonlight – pg. 9**

- 3. Java FX – pg. 11**

- 4. HTML5 – pg. 13**

- 5. Bibliografie – pg. 14**

1. Adobe Flash

Adobe Flash (cunoscut inițial ca Macromedia Flash) este o platformă multimedia folosită pentru a adăuga conținut video, animații și interactivitate paginilor web. Flash este des folosit pentru reclame și jocuri. Recent, a fost propus ca o platformă pentru RIA – *Rich Internet Application*.



Fig. 1.1 – Flash logo

Flash manipulează vectori și grafică raster pentru a anima text, desene și imagini statice. Suportă streaming bidirecțional de conținut audio și video, și poate captura acțiunile utilizatorului folosind dispozitive de intrare ca mouse-ul, tastatură, microfon și cameră. Flash folosește limbajul de programare orientat pe obiecte numit *ActionScript*.

Conținutul Flash poate fi redat pe o mare gamă de sisteme de calcul și dispozitive ce folosesc *Adobe Flash Player*, ce este disponibil gratis pentru majoritatea browserelor web, sau *Flash Lite*, în cazul dispozitivelor mobile.

Flash a fost introdus în 1996 de către *Macromedia*, și este momentan dezvoltat de către *Adobe*. Precursorul Flashului a fost *SmartSketch*, o aplicație pentru desenare folosită pe calculatoarele de tip *pen* ce foloseau ca sistem de operare *PenPoint OS*. A fost apoi portat către *Microsoft Windows* și *Mac OS*. Din moment ce internetul devenea din ce în ce mai popular, *SmartSketch* a fost relansat sub denumirea de *FutureSplash*, un utilitar de creare a animațiilor web ce era în competiție cu *Macromedia ShockWave*. În 1995, *SmartSketch* a fost îmbunătățit aducându-se facilități precum animații cadru cu cadru și a fost relansat ca *FutureSplash Animator*. Produsul a fost oferit către *Microsoft* și *Adobe* și a fost folosit de către *Microsoft* pentru a crea *MSN*. În 1996, *FutureSplash* a fost achiziționat de către *Macromedia* și lansat ca *Flash*, prin compactarea denumirii *FutureSplash*. Limbajul de programare în care este scris Flash-ul este C++.

Adobe Labs (cunoscut înainte ca *Macromedia Labs*) reprezintă o sursă de știri și versiuni *pre-release* ale produselor și tehnologiilor dezvoltate de *Adobe*. Majoritatea inovațiilor, cum ar fi *Flash 9*, *Flex 3* și *ActionScript 3.0* au fost toate discutate sau lansate ca versiuni demonstrative pe situl lor.

Un segment de piață asupra căruia *Adobe* se concentrează îl reprezintă *RIA*. *Adobe Integrated Runtime (AIR)* reprezintă un mediu de rulare și dezvoltare a aplicațiilor dezvoltate folosind Flash, și poate fi folosit pentru a rula aplicațiile Flash similar rulării unei aplicații desktop convenționale. Instalarea platformei *AIR* a depășit 100 milioane instalări în februarie 2009. Probabil ca acest obiectiv nu ar fi fost atins atât de rapid dacă *AIR*-ul nu ar fi fost instalat automat cu *Adobe Acrobat Reader*. Mulți utilizatori nefiind conștienți de faptul că *AIR* rezidă pe calculatorul lor.

Două componente adiționale au fost gândite și propuse de *Adobe* pentru o implementare la scară largă în viitoarele versiuni Flash: prima este o opțiune de a difuza conținutul unei reclame înainte de a difuza conținutul video principal; și a doua o reprezintă o funcționalitate *DRM (Digital*

Rights Management). Astfel, Adobe oferă posibilitatea companiilor de a lega o reclamă de conținut și se asigura că amandouă sunt difuzate complet și neschimbate. Starea curentă a acestor două proiecte este neclară.

1.1. Formatul fișierelor

Fișierele Flash sunt stocate în formatul *SWF*, căruia i se spune în mod tradițional: „*ShockWave Flash*”. De obicei, un fișier cu extensia *.swf* poate fi folosit sub forma unui plugin a unei pagini web, sau redat folosind *Flash Player*, sau inclus într-un film de tip *Projector*, executabil pentru *Microsoft Windows*. Fișierele video *Flash* au extensia *.flv* și conținutul lor este redat fie folosind fișierul *.swf* în care sunt încorporate sau folosind un codec pentru un *player* obișnuit de conținut multimedia audio și video, ca *VLC*, *Windows Media Player*, *QuickTime*, *Winamp*, etc.

Folosirea graficii vectoriale împreună cu cod de program permite fișierelor Flash să fie mai mici - și astfel, consumul de bandă să fie mai mic – decât hărțile de biți sau clipurile video corespunzătoare. Pentru conținut într-un singur format (cum ar fi text, video sau audio), alternativele ar putea oferi o performanță mai bună și un consum de procesor mai redus decât varianta Flash, de exemplu atunci când se folosește transparență sau actualizări de imagini, efecte de imagini.

În plus față de motorul de randare vectorială, *Flash Player* include o mașină virtuală denumită *ActionScript Virtual Machine (AVM)* pentru interactivitate prin scripting la rularea aplicației, pentru suport video, audio bazat pe standardul MP3, și grafică bitmap. Începând cu *Flash Player* versiunea , oferă două codecuri: *On2 Technologies VP6* și *Sorenson Spark*, și suport la *run-time* pentru JPEG, JPEG progresiv, PNG și GIF. Pentru următoarea versiune, Flash va folosi un compilator de tipul *just-in-time* pentru motorul *ActionScript*.

Până la implementarea *HTML5*, redarea conținutului video pe o pagină web necesită pluginuri pentru browser, ce sunt implementați în mod unic de către vendori terți. Toate pluginurile video sunt gratis și independente de platformă, inclusiv *Flash Video* de la Adobe, ce a fost introdus inițial în versiunea 6 a *Flash Player*. *Flash Video* a fost o alegere foarte populară pentru majoritatea siturilor cu conținut video datorită răspândirii acestuia pe calculatoarele utilizatorilor și ușurința implementării. În 2010, compania Apple a criticat public compania Adobe pentru faptul ca pluginul Flash nu profită de accelerarea hardware afirmând că acesta este motivul principal pentru care Flash nu este implementat pe foarte multe dispozitive mobile. La scurt timp după critica adusă de Apple, Adobe a lansat versiunea 10.1 pentru Flash, versiune ce folosește accelerarea hardware pentru redarea conținutului atât video, cat și animații, text, efecte, etc, chiar și pe calculatoarele *Mac*.

Flash Audio este des codat folosind standardul MP3 sau AAC (*Advanced Audio Coding*), totuși nu suportă *ADPCM*, *Nellymoser* și *Speex*. Flash permite rate de eșantionare de 11, 22, 44.1 kHz. Nu suportă 48 kHz, ce este standard de eșantionare pentru TV și DVD.

În august 2007, Adobe a pe blogul său că *Flash Player 9 Update 3* va suporta și câteva părți din standardul internațional *MPEG-4*. În mod deosebit, suport pentru conținut video compresat folosind *H.264 (MPEG-4 Part 10)*, compresie audio folosind *AAC (MPEG-4 Part 3)*, *F4V*, *MP4 (MPEG-4 Part 14)*, *M4V*, *3GP* și *MOV*.

La lansarea următoarei mari versiuni de *Adobe Flash Player* se așteaptă și o versiune pe 64 biți pentru *Windows*, *Mac* și *Linux*. În momentul de față utilizatorii cu sisteme de operare pe 64 biți, sunt

sfătuiți să folosească browsere pe 32 biți pentru a avea disponibil pluginul pe 32 biți. Implementările cu surse deschise precum *Gnash* are și o versiune pe 64 biți.

Adobe a tot furnizat versiuni experimentale pe 64 biți pentru *Linux x86-64*. Proiectul a fost închis temporar în iunie 2010. Varianta pe 32 de biți este încă distribuită pentru toate versiunile de Linux, dar versiunea pe 64 biți poate fi instalată manual folosind surse de pachete software speciale.

2. Microsoft Silverlight

Microsoft Silverlight reprezintă o platformă pentru aplicații web ce oferă funcționalități asemănătoare celor oferite de către *Adobe Flash*, integrând multimedia, grafică, animații și interactivitate într-un singur mediu de rulare. Inițial a fost lansat ca un plugin de streaming video, versiunile următoare aducând în plus mai multă interactivitate, mai multe facilități și suport pentru limbajele *CLI (Common Language Infrastructure)* și instrumente de dezvoltare. Ultima versiune este numărul 4 lansată în aprilie 2010.

Este compatibil cu multiple browsere folosite pe sistemele de operare *Microsoft Windows* și *Mac OS X*. Pentru dispozitivele mobile, se așteaptă suport pentru *Windows Phone 7* și *Symbian Series 60* din 2010.



Fig 2.1 – Logo Microsoft Silverlight

Silverlight redă grafică memorată similar *Windows Presentation Foundation*, și integrează multimedia și interactivitate într-un singur mediu de rulare. În aplicațiile *Silverlight*, interfețele utilizator sunt declarate în *Extensible Application Markup Language (XAML)* și programate folosind un subset din platforma *.NET*. *XAML* poate fi folosit pentru marcarea graficii vectoriale și a animațiilor. De asemenea, *Silverlight* poate fi folosit pentru a construi aplicații de tip *sidebar* pentru *Windows Vista* și *Windows 7*.

Silverlight suportă conținut media de tipul *Windows Media Video (WMV)*, *Windows Media Audio (WMA)*, *MP3* pe toate browserele suportate fără a fi necesare *Windows Media Player*, pluginul *ActiveX* sau alte pluginuri. Suportă, de asemenea, și standardul video *VC-1*, deși doar în formatul *Advanced Systems Format (ASF)*. Începând cu versiunea 3, *Silverlight* suportă redarea conținutului video compresat folosind *H.264*. *Silverlight* permite încărcarea dinamică a conținutului *XML (eXtensible Markup Language)* ce poate fi manipulat folosind interfața *DOM (Document Object Model)*, o tehnică asemănătoare cu tehnicile *Ajax* convenționale. *Silverlight* expune un obiect *Downloader* ce poate fi folosit pentru a descărca conținut, de exemplu scripturi, conținut media, date, etc. Începând cu versiunea 2, logica aplicației poate fi scrisă folosind orice limbaj *.NET*, și câteva derivate a câtorva limbaje de programare cunoscute ca *IronRuby* sau *IronPython*.

2.1. Versiuni

Silverlight 1.0, ce a fost dezvoltat sub numele de cod *Windows Presentation Foundation/Everywhere (WPF/E)* și lansat în 2007, este alcătuit din nucleul a platformei de prezentare, ce este responsabilă pentru interfața grafică, interactivitate și input de la utilizator, obiecte de interfață grafică de bază, grafică și animație, redare conținut multimedia, *Digital rights management (DRM)*, și integrare *DOM*. Are următoarele componente:

- *Input* – se ocupă de dispozitivele de intrare: tastatură, mouse, stylus, etc.
- *UI core* – se ocupă de randarea imaginilor bitmap (incluzând imagini raster compresate folosind *JPEG*), grafică vectorială, text și animații
- *Media* – redare *MP3*, *WMA*, *WMV7*, *WMV8* și *WMV9/VC-1*
- *XAML* – permite construirea interfeței utilizator folosind *XML*.

O aplicație *Silverlight* prin invocarea unui control *Silverlight* din pagina *HTML*, ce încarcă apoi fișierul *XAML*. Fișierul *XAML* conține obiectul *Canvas*, ce acționează ca un loc în care rezidă restul elementelor. *Silverlight* oferă diferite primitive geometrice ca linii, elipse și alte forme, până la elemente de genul text, imagine, și media, etc. Elementele sunt poziționate corespunzător pentru a se crea interfața dorită. Orice formă geometrică arbitrară poate fi creată și folosită. Aceste elemente pot fi animate folosind *declanșatori de evenimente*; anumite animații sunt predefinite, altele pot fi create ca și compuși din efecte predefinite. Evenimentele ca mișcări ale tastaturii și mouseului pot declanșa și ele evenimente ce pot fi tratate de scripturi.

Manipularea programatică a interfeței utilizator este atinsă folosind limbajele de scripting pentru a modifica *DOM-ul* obiectului *Canvas*. Pentru a facilita acest lucru, *Silverlight* expune *DOM API (Application Programming Interface)*, accesibil din orice limbaj de scripting suportat de către *Silverlight*, care în versiunea 1.0 este limitat la *JavaScript* ce rulează doar în browser.

Totuși, nu există controale *UI* construite în *Silverlight*. Peste *Canvas-ul Silverlight* sunt plasate controalele specifice browserului pentru input de la utilizator. Suportul pentru formatele de date este limitat la *XML* și *JSON (JavaScript Object Notation)*.

Silverlight 2 (referit inițial ca *Silverlight 1.1*) include o versiune a platformei *.NET* aceeași versiune completă a *CLR (Common Language Runtime)* din *.NET Framework 3.0*. Totuși, nu poate referi *assembly-uri* compilate cu *.NET Framework* obișnuit. Spre deosebire de *CLR* inclus cu *.NET Framework 3.5* și mai devreme, dar asemănător cu *.NET Framework 4.0*, multiple instanțe de *CoreCLR* pot fi găzduite într-un singur proces. Folosind aceasta, fișierul *XAML* mărit folosind cod în spatele unui cod, scris în orice limbaj *.NET*, care conține logica aplicație. Poate fi folosit pentru a manipula programatic atât aplicația *Silverlight* cât și pagina *HTML* ce găzduiește controlul *Silverlight*. Atât documentul *XAML*, cât și codul aplicației sunt compilate în *assembly-uri* și apoi compresate folosind *ZIP*, și sunt memorate în fișiere *.xap*.

Silverlight 2 este distribuit cu o librărie de clase foarte ușoară ce include trăsături ca servicii web *XML*, controale extensibile, componente de rețea și *LINQ (Language Integrated Query)*. Această librărie de clase este o submulțime, și considerabil mai mică decât *.NET Framework Base Class Library (BCL)*.

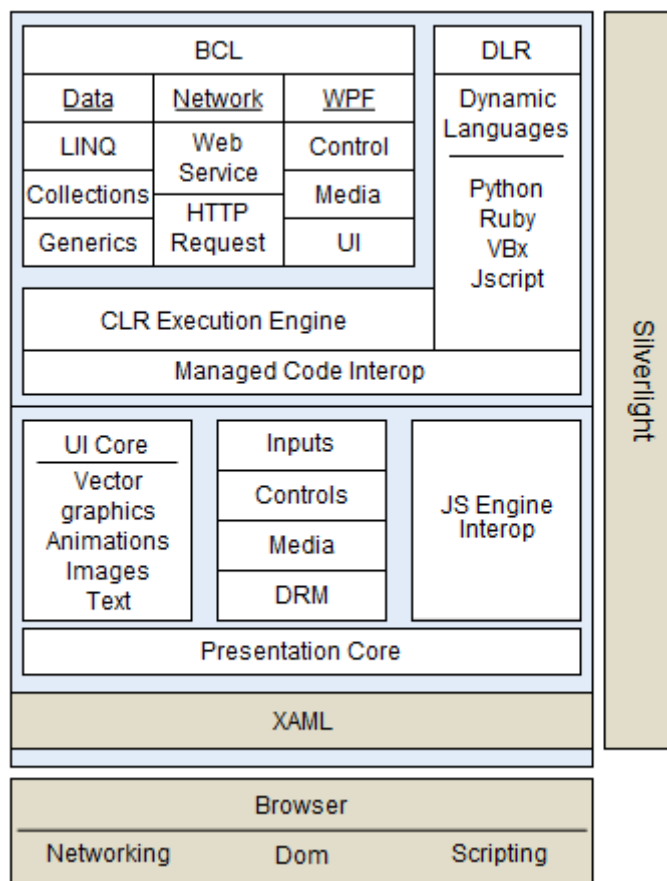


Fig 2.2 – Arhitectura Silverlight 2.0

CoreCLR folosește un model de securitate bazat pe atribute, spre deosebire de .NET Framework ce folosește CAS (Code Access Security). Totuși, doar codul de platformă este marcat ca și *critical* sau *safecritical*. Mediul de rulare Silverlight asigură faptul ca *assembly*-urile specifice platformei sunt încărcate din directorul de instalare Silverlight, și sunt semnate digital de către Microsoft. Acest lucru presupune că *assembly*-urile utilizator-aplicație pot fi doar cod transparent (rulat cu încredere parțială și drepturi limitate). Codul de platformă poate fi marcat cu oricare din atribute. Metodele BCL ale platforme .NET ce au setat atributul *Internet* (adică sunt expuse pentru a fi rulate din internet de către cod necunoscut), în BCL Silverlight ca metode transparente.

Silverlight 3 mai multe tipuri de controale, incluzând *DataGrid*, *TreeView*, diverse paneele de afișare, *DataForm* pentru diverse aplicații de manipulare a datelor și *DataPager* pentru vizualizarea datelor paginate. Câteva din aceste controale aparțin de *Silverlight Toolkit*. În plus față de acestea, Silverlight include o platformă de navigare pentru a permite aplicațiilor Silverlight să folosească navigarea prin *hyperlink*-uri.

Pe parte de multimedia, Silverlight suportă codarea audio AAC, și de asemenea, decodarea hardware a conținutul video comprimat folosind H.264. Suportă și streaming video la rezoluția HD de 1080p. Clasele multimedia sunt expuse programatic, pentru ca alte formate sa poată fi implementate folosind decodoare scrise folosind *managed code*. Silverlight 3 suportă și perspective 3D, ce permite transformare 3D a elementelor 2D. Aceste transformări, la fel ca și alte operații de transformare 2D, efecte de transparență, etc., sunt accelerate hardware. Animații personalizate pot fi create, incluzând

transformări și adaptări, folosind *HLSL (High Level Shader Language)* pentru programarea *pixel shader*. Un *API* este distribuit cu *Silverlight 3* pentru a permite manipularea imaginilor bitmap. *Silverlight* folosește acum *GPU-ul (Graphical Processing Unit)* pentru redarea și compunerea de *Visual Tree*. Opțional, *Silverlight* poate folosi *Binary XML* pentru a comunica cu servicii *WCF (Windows Communication Foundation)*.

Cu această versiune, aplicațiile *Silverlight 3* pot fi instalate pentru a fi folosite *offline*, dacă dezvoltatorul dorește acest lucru. În plus față de rularea lor în browser, aplicațiile pot folosi tastele F, dar nu mai pot folosi serviciile *HTML interop*. Aplicațiile instalate local verifică automat la fiecare lansare ultima versiune disponibilă și o instalează dacă aceasta este disponibilă.

Silverlight 4 a fost lansat în noiembrie 2009 ca versiune beta, fiind lansată oficial apoi în aprilie 2010. Această versiune aduce în plus suport pentru browserul *Google Chrome*, suport pentru microfon și cameră web, suport pentru tipărire la imprimantă, plus multe alte îmbunătățiri la librăria de clase de bază, buguri reparate.

2.2. Moonlight

Moonlight reprezintă implementarea alternativă, gratis și cu surse deschise a *Microsoft Silverlight* pentru sistemele de operare *Linux* și variante de *Unix*. Este dezvoltat de către *Mono Project*. *Moonlight 1.0* a fost lansat în ianuarie 2009, iar *Moonlight 2.0* a fost lansat în decembrie 2009.



Fig 2.3 – Logo Moonlight

Un *IDE (Integrated Development Environment)*, numit *Lunar Eclipse*, a fost, de asemenea, creat pentru a crea și edita fișiere *XAML* și pentru a compila aplicații *Moonlight*. Se așteaptă ca *Lunar Eclipse* să fie integrat în *MonoDevelop* ca și *add-on*.

Moonlight folosește *Cairo* pentru randare și este distribuit ca plugin pentru *Firefox* pe câteva versiuni *Linux*. Pluginul nu conține codecurile media, dar când detectează conținut media, utilizatorii sunt direcționați către situl *Microsoft* pentru a le descărca.

Moonlight 2.0 urmărește momentan specificațiile *Silverlight 2.0*, dar și câteva din specificațiile *Silverlight 3.0*, cum ar fi platforma media ce permite *Moonlight*-ului să lucreze cu codecuri *open-source* cum ar fi *Theora* sau *Dirac*. Un preview pentru *Moonlight 3.0* este planificat pentru primul sfert de an 2010.

În *Silverlight*, tehnologia proprietară *Netflix* se bazează foarte mult pe *DRM*. Deoarece prin definiție, *DRM* nu poate fi implementată ca o soluție *opensource*, *Moonlight* nu oferă suport pentru acest lucru. Singurul mod de a le suporta este de a introduce un sistem proprietar *DRM* cu surse închise pentru *GNU/Linux*, ceea ce ar deteriora imaginea de mișcare *opensource*.

Microsoft și *Novell* au început cooperarea pentru a construi *Moonlight*. Cooperarea constă în accesul exclusiv pe care îl are *Novell* la artefactele *Silverlight*:

- Suita de teste *Microsoft* pentru *Silverlight*;
- Detalii de specificații, dincolo de cele de pe web
- Codec-uri cu surse închise oferite gratis pentru *Windows Media Video* și *Audio*, pentru *VC-1* și *MP3*, *H.264* și *AAC*, pentru a fi folosite de către *Moonlight* rulat în browsere web.

Microsoft a lansat două acorduri pentru a nu da în judecată pentru încălcarea patentului atunci când se folosește *Moonlight*. În primul sunt prevăzute *Moonlight 1.0* și *2.0*, este destul de restrictiv și acoperă doar folosirea *Moonlight* ca plugin de browser, nu este implementare *GPL3* și implementarea *Moonlight* a fost obținută de la *Novell*. De asemenea, este menționat ca *Microsoft* poate renunța oricând la drepturile de utilizare.

Al doilea acord este unul actualizat și mai general ce nu mai limitează utilizatorii să obțină *Moonlight* de la *Novell*, și privește produsul *Moonlight* indiferent de proveniența implementării. Acordul nu privește *fork*-uri ale *Moonlight*.

3. JavaFX

JavaFX este o platformă de creare și distribuire a aplicațiilor de tip *RIA* (*Rich Internet Application*) ce pot rula pe o varietate mare de dispozitive. Versiunea curentă (*JavaFX 1.3*, din aprilie 2010) permite construirea aplicațiilor desktop, aplicații în browser și pentru terminale mobile. Televizoarele, consolele de jocuri, playere *Blu-ray* și alte platforme sunt planificate deja.



Fig 3.1 – Logo *JavaFX*

JavaFX construiește pe o fundație *Java*. Pentru a construi aplicații *JavaFX* dezvoltatorii folosesc un limbaj declarativ și cu tipuri statice de date numit *JavaFX Script*; în programele *JavaFX* se poate integra și cod *Java*. O aplicație *JavaFX* este compilată în *bytecode Java*, deci poate rula pe orice mașină ce are *JRE* (*Java Runtime Environment*) instalat, iar pentru alte dispozitive *Java ME*.

În ceea ce privește aplicațiile desktop, versiunea curentă suportă sistemele de operare *Windows XP*, *Windows Vista* și *Mac OS X*. Începând cu *JavaFX 1.2*, *Sun* a lansat variante beta pentru *Linux* și *OpenSolaris*.

Pe terminalele mobile, *JavaFX* este capabil să ruleze pe mai multe sisteme de operare pentru telefoane mobile, inclusiv *Symbian OS*, *Windows Mobile*, și sisteme de operare *real-time* proprietare.

Varianta curentă de platformă *JavaFX* include:

1. *JavaFX SDK*: compilatorul *JavaFX* și utilitare pentru mediul de rulare. Grafică, servicii web pentru media, și librării *rich text*
2. *NetBeans IDE* pentru *JavaFX*: *NetBeans* cu paletă *drag-and-drop* pentru adăugarea de obiecte cu transformări, efecte și animații plus un set de tutoriale și sfaturi pentru cele mai bune practici. Pentru utilizatorii *Eclipse* există un plugin dezvoltat și întreținut de comunitate
3. Unelte și pluginuri pentru utilitatea creative (cunoscut și ca *Production Suite*): Pluginuri pentru *Adobe Photoshop* și *Adobe Illustrator* ce pot exportă obiecte grafice ce pot fi folosite în codul *JavaFX Script*, utilitare ce convertesc fișiere grafice *SVG* în cod *JavaFX Script*.

Unii critici au afirmat ca *JavaFX* va putea concura pe segmentul desktop cu *Adobe Flash Player*, *Adobe AIR*, *OpenLaszlo* și *Microsoft Silverlight*.

JavaFX se bazează pe conceptul de *profil comun* (*common profile*) ce se intenționează a fi răspândit peste toata gama de dispozitive ce sunt sau vor fi suportate de către *JavaFX*. Această abordare facilitează un model de programare asemănător atunci când se dorește atât dezvoltarea de aplicații desktop, cât și variantă pentru alte dispozitive. Se dorește să se poată folosi cât mai mult din codul și obiectele grafice când se trece de la dezvoltarea de pe o platformă pe alta. Pentru desktop, de exemplu, se folosește librăria *Swing* pentru obiecte grafice și redarea transformărilor și efectelor aplicate acestora.

Din punctul de vedere al utilizatorului final, conceptul de „*drag-to-install*” le permite să mute o aplicație cu mouseul din fereastra browserului pe desktop, făcându-se o scurtătură în mod automat. Acest lucru oferă avantajul că aplicația nu își pierde contextul sau starea atunci când este lansată făcând dublu-click pe scurtătură. Acest comportament nu trebuie programat, este disponibil fără a fi luat în considerare de către dezvoltatori atunci când se face proiectarea sau implementarea aplicației, fiind inclus în mecanismul de applet a aplicației.

JavaFX include un set de pluginuri pentru *Adobe Photoshop* și *Illustrator* ce permit includerea în aplicație a unei grafici mai avansate. Pluginurile generează cod *JavaFX Script* ce reține layerele și structura obiectelor grafice. Apoi, dezvoltatorii pot ușor adăuga animații sau efecte obiectelor grafice statice importate. Există, de asemenea, un utilitar de conversie a fișierelor grafice de tip *SVG* (*Media Factory*) ce permit importul de grafică și vizualizarea obiectelor convertite în format *JavaFX*.

4. HTML5

HTML5 este în momentul de față în stadiul de dezvoltare fiind remarcat ca următorul pas și marea revizie a standardului *HTML*. Le fel ca și predecesorii lui imediați, *HTML 4.01* și *XHTML 1.1*, *HTML* este un standard folosit în structurarea și prezentarea conținutului *World Wide Web*. Noul standard include facilități ca streaming video și *drag-and-drop* ce erau inițial dependente de pluginuri dezvoltate de terți: *Adobe Flash*, *Microsoft Silverlight* și *Google Gears*.

HTML5 introduce un număr mare de elemente și atribute noi ce reflectă modul modern de utilizare a siturilor. Unele din ele sunt înlocuitori semantici pentru utilizările dese ale unui bloc de elemente generic e(*<div>*) și *inline* (**), de exemplu *<nav>* (bloc de navigare situri web) și *<footer>* (referindu-se de obicei la partea de jos a paginii web). Alte elemente oferă funcționalități noi printr-o interfață standardizată, cum ar fi elementele multimedia *audio* și *video*. Unele elemente depășite din standardul *HTML 4.01* au fost eliminate complet, inclusiv etichetele cu rol pur de prezentare ca *font* și *center*, a căror efecte pot fi obținute folosind *CSS (Cascading Style Sheets)*. De asemenea, se pune accentul pe scriptingul *DOM (Document Object Model)* în *JavaScript* sau alte limbaje.

Sintaxa *HTML5* nu se mai bazează pe *SGML*, în ciuda similarității cu acesta. Totuși, a fost gândit pentru a fi invers-compatibil cu standardele anterioare de *HTML*. *HTML5* suportă *Web Forms 2.0*, o specificație *WHATWG (Web Hypertext Application Technology Working Group)*.

Diferențele dintre *HTML5* și standardele imediat anterioare lui *HTML 4.01* și *XHTML 1.1* sunt următoarele:

- Noi reguli de parsare: parsarea este orientată către flexibilitate și compatibilitate; nu se bazează pe *SGML*
- Abilitatea de folosi obiecte grafice *SVG* și *MathML* în partea de *text/html*
- Noi etichete: *article*, *aside*, *audio*, *canvas*, *command*, *datalist*, *figure*, *footer*, *header*, *keygen*, *progress*, *section*, *source*, *time*, *video*, etc.
- Noi tipuri de controale: dată și oră, email, url, căutare, colorare
- Noi atribute: *ping*, *charset*, *async*

dev.w3.org furnizează ultimul draft al editorului de diferențe dintre *HTML5* și *HTML4*, ce listează toate adăugările, eliminările și schimbările la *HTML5*.

Un browser *HTML5 (text/html)* va fi flexibil în tratarea sintaxei incorecte. *HTML5* este gândit în așa fel încât browserele vechi să ignore fără probleme noile construcții *HTML5*. În comparație cu *HTML 4.01*, specificațiile *HTML5* oferă reguli detaliate pentru lexare și parsare, cu intenția ca diferite browsere incompatibile să producă același rezultat în cazul unei sintaxe incorecte. Deși *HTML5* definește acum un comportament nou pentru documentele cu etichete plasate prost, acele documente nu sunt privite ca conforme cu standardul *HTML5*.

5. BIBLIOGRAFIE

[1] – „Adobe Flash CS4 Professional – Classroom in a book” – Russel Chun, Adobe Systems 2009

[2] – „Microsoft Silverlight 4 for dummies” – Mahesh Krishnan, Philip Beadle, Wiley Publishing 2010

[3] – „Pro JavaFX Platform” – James L. Weaver, Weiqi Gao, APRESS 2009

[4] – Referință web – <http://en.wikipedia.org/wiki/Html5>