

# Introducció i conceptes generals

Marcos González Sancho

PID\_00192287



*Els textos i imatges publicats en aquesta obra estan subjectes –llevat que s'indiqui el contrari– a una llicència de Reconeixement-Compartir igual (BY-SA) v.3.0 Espanya de Creative Commons. Podeu modificar l'obra, reproduir-la, distribuir-la o comunicar-la públicament sempre que en citeu l'autor i la font (FUOC. Fundació per a la Universitat Oberta de Catalunya), i sempre que l'obra derivada quedi subjecta a la mateixa llicència que el material original. La llicència completa es pot consultar a <http://creativecommons.org/licenses/by-sa/3.0/es/legalcode.ca>*

# Índex

<b>1. Aplicacions Rich Media</b> .....	5
1.1. Concepte i definició .....	5
1.2. Característiques d'una ARM .....	5
1.3. ARM i Rich Internet Applications .....	7
<b>2. Tecnologies relacionades</b> .....	8
2.1. JavaFX .....	8
2.2. Silverlight .....	8
2.3. HTML5, CSS3 i Ajax .....	9
2.3.1. Extensió del navegador .....	11
2.4. Flash .....	12
2.5. Flex Framework .....	13
2.6. Adobe AIR .....	14
<b>3. Plataforma Flash per a la creació d'ARM</b> .....	16
3.1. Capacitats multimèdia .....	16
3.2. Capacitats multiplataforma .....	17
3.3. Capacitats multidispositiu .....	17
3.4. Capacitat de gestió de recursos externs .....	18
<b>4. Consideracions</b> .....	20



## 1. Aplicacions Rich Media

### 1.1. Concepte i definició

El terme Rich Media fa referència a l'ús conjunt de diferents mitjans (text, imatge, so, vídeo, animació, etc.) i és freqüent trobar-lo en l'àmbit del màrqueting i la creativitat, però no tant en el camp de les aplicacions.

Actualment, s'estan establint molts vincles entre els mitjans de comunicació tradicionals i Internet, i una cosa semblant comença a succeir quan parlem d'aplicacions. Internet ha evolucionat, amb el pas dels anys, des d'una visió molt limitada dels mitjans a ser-ne el referent pel que fa a suport i varietat i, a més, és l'entorn en el qual es fa un ús més gran d'aquests mitjans.

Les aplicacions també s'estan adaptant, en moltes àrees, al concepte que existeix darrere d'Internet i, per tant, al concepte de *multimèdia*, amb un enfocament clar cap a l'usuari. Això ha donat lloc a una tipologia d'aplicacions que seria la que englobaríem sota el concepte d'*aplicacions Rich Media* (ARM).

Les aplicacions Rich Media són totes aquelles que permeten i potencien l'ús de diferents elements i continguts multimèdia com a essència de l'aplicació i també com a mètode d'interacció amb l'usuari, amb independència de la plataforma o entorn al qual van destinades (navegador, escriptori, dispositius, etc.).

Quan parlem de Rich Media estem englobant aspectes que afecten la planificació, el disseny, el desenvolupament i el concepte d'aplicacions, amb una clara orientació cap als mètodes d'interacció i comunicació amb l'usuari. Si bé el concepte *multimèdia* no necessàriament implica interactivitat, avui dia l'usuari ha deixat de tenir un paper passiu en les aplicacions, per la qual cosa el disseny orientat a l'usuari és una màxima en aquest tipus d'aplicacions.

### 1.2. Característiques d'una ARM

Per a comprendre d'una manera més completa el concepte i les funcions d'una ARM, és interessant analitzar quines característiques són desitjables (en part o íntegrament) quan es decideix crear una ARM, i quines s'apliquen de forma incorrecta.

Quan treballem en la creació d'una ARM, és desitjable, i habitualment necessari, tenir en compte les característiques següents:

- **Disseny i desenvolupament orientats a l'usuari** (recordem que el contingut multimèdia enriquit hi està clarament enfocat).
- **Ús de diferents mitjans**, ja sigui com a continguts en si mateixos o com a mecanismes per a mostrar de manera millorada la informació a l'usuari.
- **Interactivitat** en disseny, interfície i interacció.

Per contra, existeixen consideracions que no necessàriament són certes quan s'apliquen a una ARM (en moltes ocasions tenen sentit, però no de forma obligatòria), com ara:

- Una ARM ha de manipular la màxima quantitat possible de tipus de mitjans.
- Una ARM ha de ser multiplataforma.
- Una ARM ha de ser multidispositiu.
- Una ARM ha d'estar basada en continguts en el núvol.
- Una ARM no pot ser una aplicació basada en dades.

Avui dia el concepte Rich Media ha estat expandit amb **nous tipus de continguts** "no tradicionals", que van molt més enllà dels clàssics (text, imatge, animació, àudio i vídeo). Aquests continguts, que es poden considerar nous, són els següents:

- Mapes
- Publicitat
- Codis QR o similars
- Realitat augmentada
- 3D
- Contingut social o col·laboratiu
- Contingut interactiu (educatiu, lúdic...)

A més, la **interactivitat**, entesa com un aspecte més que habitual i fonamental en les ARM (atès que aquestes aplicacions estan clarament orientades a l'usuari), té nous mecanismes que pot oferir:

- Entrada de vídeo / càmera web.
- Entrada d'àudio / micròfon.
- GPS i geolocalització.
- Acceleròmetre.
- Brúixola.
- Sistemes de valoració.
- Compartició en xarxes socials.

### 1.3. ARM i Rich Internet Applications

Atès que la definició aportada és bastant oberta, pot donar lloc a confusions amb altres termes que, des de fa un temps, són molt presents en la comunitat de desenvolupament, com és el cas de les **Rich Internet Applications (RIA)**.

Podem veure les RIA com un subconjunt de les ARM, ja que compleixen les condicions de les ARM però en limiten en mitjà a un navegador.

No hem de perdre de vista que quan es parla d'ARM, no es parla d'un mitjà o d'una plataforma en concret, sinó que una ARM pot estar desenvolupada per tal de treballar en un navegador (Internet), en una televisió, en un dispositiu mòbil, etc., o fins i tot en una combinació de diverses plataformes.

## 2. Tecnologies relacionades

Dins del conjunt de tecnologies que ens permeten crear ARM, existeixen algunes que tenen una rellevància especial, a causa del volum de desenvolupadors que hi ha al darrere o per les capacitats enfocades especialment al treball amb continguts i interaccions basades en continguts multimèdia.

### 2.1. JavaFX

**JavaFX** és un conjunt de productes i tecnologies de Sun Microsystems que permet la creació d'ARM i RIA sota desenvolupament en Java. Els seus creadors el defineixen com el pas següent en l'evolució de Java cap al món dels continguts enriquits multimèdia. Va ser llançat oficialment el desembre de 2008.

JavaFX està dissenyat per a proporcionar una plataforma lleugera amb capacitats gràfiques d'acceleració per maquinari per al desenvolupament d'aplicacions de negoci. Amb JavaFX, els desenvolupadors podran reutilitzar biblioteques emprades en desenvolupaments previs (realitzats, òbviament, en Java) i fins i tot tenir accés a capacitats natives del sistema.

JavaFX té com a components destacats JavaFX Script (llenguatge script orientat a objectes i amb una sintaxi simplificada pel que fa a Java) i JavaFX Mobile (sistema de programari complet per a dispositius mòbils), que van ser llançats a inicis de 2009.

#### **Enllaços relacionats**

<http://javafx.com/>

<http://en.wikipedia.org/wiki/javafx>

<http://es.wikipedia.org/wiki/javafx>

<http://www.oracle.com/technetwork/java/javafx/overview/index.html>

<http://www.slideshare.net/ibantxuyn/introduccion-a-java-fx>

### 2.2. Silverlight

**Silverlight** és una eina de desenvolupament dissenyada per a permetre crear experiències interactives atractives que puguin funcionar en diferents plataformes, com ara navegador i dispositius. Silverlight disposa d'un connector gratuït creat amb l'entorn .NET i compatible amb molts navegadors, dispositius i sistemes operatius.



Silverlight es va considerar des del seu llançament (setembre de 2007) com l'eina creada per Microsoft per a competir amb l'omnipresent Adobe Flash. Per això, l'eina permet treballar amb funcions multimèdia com ara la reproducció de vídeo, suport per a gràfics vectorials, animacions i interactivitat.

Malgrat les seves capacitats i les seves eines de desenvolupament reconegudes, no ha aconseguit la mateixa popularitat que el seu competidor de referència.

#### **Enllaços relacionats**

<http://www.microsoft.com/silverlight/>

<http://www.silverlight.net/>

[http://en.wikipedia.org/wiki/Microsoft\\_Silverlight](http://en.wikipedia.org/wiki/Microsoft_Silverlight)

[http://es.wikipedia.org/wiki/Microsoft\\_Silverlight](http://es.wikipedia.org/wiki/Microsoft_Silverlight)

<http://msdn.microsoft.com/es-es/silverlight/bb187358>

[http://msdn.microsoft.com/es-es/library/cc838158\(v=vs.95\).aspx](http://msdn.microsoft.com/es-es/library/cc838158(v=vs.95).aspx)

### **2.3. HTML5, CSS3 i Ajax**

**Ajax** (Asynchronous Javascript and XML) és una tècnica de desenvolupament web per a la creació d'aplicacions interactives enriquides, que s'empra en la part del client (navegador) i que permet establir una comunicació amb el servidor sense canviar de pàgina, fet que repercuteix de forma directa en la interacció amb l'usuari, ja que és possible realitzar canvis en les interfícies sense realitzar una recàrrega.

Ajax és una de les tecnologies o mecanismes que han estat fonamentals en el denominat *Web 2.0* i en les RIA, i està íntimament relacionada amb altres llenguatges, formats i estàndards com XHTML, HTML, DOM, CSS, XML, JSON, etc.

Les peticions del costat del client i l'actualització de la interfície normalment es realitzen en Javascript i la comunicació amb el servidor es pot dur a terme fent ús de l'objecte XMLHttpRequest (disponible en la majoria de navegadors actuals). La resposta, si bé originalment va ser pensada en XML, no requereix de forma obligatòria aquest format, i actualment és molt habitual l'ús d'altres formats generalitzats com ara JSON, HTML o text net.

L'objecte XMLHttpRequest encara no ha estat estandarditzat per la W3C (<http://www.w3.org/tr/xmlhttprequest/>) i, per tant, a dia d'avui existeixen algunes diferències entre les diferents implementacions que s'han anat elaborant per part dels diferents fabricants de navegadors.

Tot i que Ajax com a tal no és un exponent clar de desenvolupament d'ARM, l'arribada d'**HTML5** i **CSS3**, amb algunes característiques especialment interessants per al desenvolupament d'aquest tipus d'aplicacions (principalment per

a dispositius en forma d'aplicacions web), l'ha potenciat encara més, ja que combinant Ajax amb aquests dos estàndards es poden crear aplicacions molt interessants que tenen com a principal i gran avantatge la seva compatibilitat amb una gran quantitat de dispositius.

La creació d'una aplicació web basada en HTML5 (i CSS3 per a la seva estètica) té els avantatges que implica el desenvolupament web, com ara una presentació flexible que permet la seva visualització en diferents pantalles, la compatibilitat amb gairebé la totalitat de sistemes operatius per a telèfons intel·ligents (que tenen la qualitat d'anar per davant dels navegadors d'escriptori en compatibilitat amb HTML5) i, gràcies a les noves característiques del llenguatge, capacitat de treballar amb:

- Àudio (mitjançant l'etiqueta <audio>).
- Vídeo (mitjançant l'etiqueta <video>).
- Llenços 2D i 3D per al treball amb gràfics avançat.
- Geolocalització.
- Mode de treball fora de línia (mitjançant un emmagatzematge en memòria cau especialment interessant).
- Emmagatzematge persistent (fins i tot amb base de dades).
- Millores en la formatació de camps d'entrada (fins i tot per al teclat virtual).
- Millores en la validació de formularis (per tipus: correu electrònic, telèfon, etc.).
- Possibilitat de treball amb sòcols (WebSockets).
- Possibilitat de treball amb execució multitasca (Web Workers).
- Millores en interacció com l'API de Drag & Drop.

Malgrat totes aquestes característiques, cal destacar que a dia d'avui no tot és tan bonic com pugui semblar, ja que continuen existint diferències entre navegadors (com en el cas del vídeo), característiques que no tenen molt clares el seu futur (com el *web database* per a emmagatzematge persistent en base de dades, que ha estat discontinuat i enfocat, en canvi, cap a la Indexed DataBase API -<http://www.w3.org/tr/indexeddb/>-, etc.).

En general sempre que es vulgui treballar sobre HTML5, cal afinar molt bé els dispositius o navegadors destinació, per a tenir clars els *fallbacks* que cal implementar per a donar suport a tots els usuaris. En conseqüència, és habitual comptar amb eines que augmenten la compatibilitat dels navegadors amb HTML5 com HTML5Shiv (<http://code.google.com/p/html5shiv/>), que permet ampliar el suport d'HTML5 a navegadors Internet Explorer antics, o Modernizr (<http://modernizr.com/>), que simplifica en gran mesura la detecció de totes les capacitats del navegador per a HTML5 i CSS3, de manera que el desenvolupador pugui oferir una versió compatible si no en suporta algunes (Modernizr incorpora, per cert, HTML5Shiv).

### **Enllaços relacionats amb Ajax**

<http://es.wikipedia.org/wiki/AJAX>

[http://en.wikipedia.org/wiki/Ajax\\_\(programming\)](http://en.wikipedia.org/wiki/Ajax_(programming))

<http://www.librosweb.es/ajax/>

<http://www.w3schools.com/ajax/default.asp>

<http://jquery.com/>

Com a biblioteca destacada que disposa de gestió d'Ajax: <http://api.jquery.com/category/ajax/>

### **Enllaços relacionats amb HTML5**

HTML5:

<http://dev.w3.org/html5/spec/spec.html>

<http://diveintohtml5.info/>

<http://www.html5rocks.com/en/>

<http://5stepstohtml5.com/>

Compatibilitat HTML5 i CSS3:

<http://html5please.com/>

<http://caniuse.com/>

<http://www.findmebyip.com/litmus/>

<http://modernizr.com/>

Demos HTML5:

<http://html5demos.com/>

Llenços HTML5:

<http://blog.nihilogic.dk/2009/02/html5-canvas-cheat-sheet.html>

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

Animació amb HTML5 i llenços:

<http://createjs.com/#!/CreateJS>

Javascript Game Engines List:

<https://gist.github.com/768272>

Box2D Javascript Game Engine:

<http://code.google.com/p/box2dweb/>

#### **2.3.1. Extensió del navegador**

Finalment hi ha un tema especialment delicat que, en el cas d'aplicacions, es converteix en vital, com és el fet que la necessitat de les aplicacions web de treballar sobre un navegador fa que necessitin un canal addicional per a estendre les capacitats que s'espera d'una aplicació. Existeixen diverses solucions

en el mercat que actuen com a pont entre el navegador i el dispositiu, de manera que des d'un llenguatge com Javascript es poden fer crides a biblioteques natives que estenen les funcionalitats.

#### **Enllaços relacionats amb l'extensió d'HTML5**

<http://phonegap.com/>

<http://www.appcelerator.com/>

<http://www.sencha.com/products/touch/>

<http://www.markus-falk.com/mobile-frameworks-comparison-chart/>

Una de les més conegudes és PhoneGap, que actualment és sota el control d'Adobe. Mitjançant aquesta eina podem crear una aplicació web que actualment es podria executar com a aplicació nativa (amb algunes limitacions: <http://phonegap.com/about/features>) en molts sistemes operatius, però unificant les crides a accions natives com, per exemple, consultar el GPS, obtenir una imatge de la càmera del dispositiu, el sistema de fitxers, etc.

Igual que ocorre amb qualsevol tecnologia d'aquest tipus, una vegada creada la base amb la tecnologia comuna, s'ha de realitzar un procés de compilació per a les diferents plataformes de manera independent, i així poder obtenir com a resultat instal·ladors compatibles amb els corresponents sistemes operatius.

Com diem, PhoneGap és solament una de moltes, com poden ser també Titanium Appcelerator, Sencha Touch, Corona, etc.

## **2.4. Flash**

Es coneix com a **Flash** (en realitat Flash Player) el connector gratuït disponible en gairebé tots els navegadors, que permet renderitzar contingut creat en format SWF (actualment obert per l'Open Screen Project). Adobe Flash Professional és l'eina o IDE de desenvolupament creada originalment per Macromedia (actualment Adobe) per a generar aquesta mena de contingut, que des dels seus inicis ha anat solucionant de forma anticipada les limitacions o problemàtiques de la web basada en estàndards, sempre amb la creació avançada de continguts multimèdia i interactivitat.

Actualment, Flash és present en una gran quantitat d'àrees de negoci a Internet, com ara publicitat, televisió web, reproducció en temps real (*streaming*), desenvolupament de jocs, interactius i infografies, animació, etc., i s'usa com a eina complementària en determinats camps en una gran quantitat de mitjans.

Per l'índex de penetració que ha aconseguit a Internet (al voltant del 99% segons les estadístiques de la companyia), s'ha convertit en un estàndard *de facto* per a continguts multimèdia a Internet i avui dia és un clar dominador en vídeo, àudio, jocs (amb suport 3D i acceleració gràfica) i publicitat. Si bé les capacitats que van adquirint a poc a poc els estàndards com HTML5 comen-

cen a permetre substituir Flash en determinats continguts, aquest connector encara hi està per davant en la majoria d'aquests camps quan es requereixen característiques avançades.

Finalment, Flash Professional (l'eina o IDE) permet generar aplicacions independents (*standalone*) d'escriptori, compatibles amb sistemes Windows i Mac i amb una reutilització de pràcticament el 100% de codi que per a la mateixa versió d'Internet, encara que aquesta característica s'ha obviat en els últims anys amb l'aparició d'Adobe AIR.

## 2.5. Flex Framework

Amb el nom actual d'**Apache Flex** (sota el control de la Fundació Apache des de final de l'any 2011) i abans conegut com a Adobe Flex, és un *framework* per al desenvolupament i la creació d'aplicacions enriquides sobre diferents plataformes, que es basa en la Plataforma Flash.

La principal característica de Flex és que aporta un conjunt de desenvolupament de programari (SDK, *software development kit*) que afavoreix i agilita el desenvolupament d'aplicacions pensades principalment per al consum de grans quantitats de dades, i que té un vessant més empresarial que les aplicacions per a les quals estava dissenyat tradicionalment Flash Professional.

Com que a partir de 2008 Flex va ser alliberat com a font pública (*open source*), és possible desenvolupar-hi usant IDE convencionals, com per exemple Eclipse, o amb l'IDE propietari d'Adobe, anomenat Flash Builder (que internament està basat en Eclipse).

Flex es combina amb el llenguatge MXML (similar a XML) que permet definir elements i comportaments per a una vista i que disposa d'una gran quantitat de components especialment dissenyats i optimitzats per al desenvolupament d'aplicacions empresarials basades en dades, en combinació amb sistemes i eines com BlazeDS, Live DataCycle Services, ColdFusion, etc.

En la seva versió 4.5 comença a admetre el desenvolupament multidispositiu i en versions *premium* aporta eines addicionals que són clau per a l'optimització d'aplicacions, com ara *profilers* de memòria i rendiment, sistemes de prova automatitzats, suport per a Flex Unit, capacitat de compilació des de línia d'ordres, etc.

### Enllaços relacionats

Apache Flex:

<http://incubator.apache.org/flex/>

[http://en.wikipedia.org/wiki/Apache\\_Flex](http://en.wikipedia.org/wiki/Apache_Flex)

Adobe Flex:

<http://www.adobe.com/devnet/flex.html>

<http://labs.adobe.com/technologies/flex/>

[http://es.wikipedia.org/wiki/Adobe\\_Flex](http://es.wikipedia.org/wiki/Adobe_Flex).

<http://flex.org/>

Flash Builder:

<http://www.adobe.com/es/products/flash-builder.html>

[http://en.wikipedia.org/wiki/Adobe\\_Flash\\_Builder](http://en.wikipedia.org/wiki/Adobe_Flash_Builder)

[http://es.wikipedia.org/wiki/Adobe\\_Flash\\_Builder](http://es.wikipedia.org/wiki/Adobe_Flash_Builder)

## 2.6. Adobe AIR

**AIR** és una versió d'execució o *runtime* (Adobe Integrated Runtime) multiplataforma desenvolupat per Adobe amb l'objectiu de permetre la creació de RIA que vagin més enllà d'un navegador, principalment al voltant de la Plataforma Flash. Actualment se'n pot trobar la darrera versió de desenvolupament a Adobe Labs (<http://labs.adobe.com/technologies/flashplatformruntimes/>).

Mitjançant Adobe AIR es mantenen les eines i els llenguatges de desenvolupament emprats habitualment per a la creació de RIA en el navegador (Flash, Flex, HTML, Ajax, ActionScript, Javascript, etc.) i s'estenen les capacitats de Flash Player a altres plataformes, com l'escriptori i els dispositius (en diferents sistemes operatius), sota una API unificada per tal que el desenvolupador disposi de funcions generalment reservades a aplicacions natives.

Actualment admet la creació d'aplicacions amb funcionalitats natives per a sistemes operatius Windows, Mac, Linux, iOS, Android i BlackBerry Tablet, amb un percentatge de reutilització de codi molt alt. Les aplicacions realitzades amb Adobe AIR són empaquetades, signades digitalment i instal·lades en el sistema local de fitxers, i s'integren de manera similar a una aplicació nativa en el sistema d'arxius, acoblador, maneig de fitxers locals, bases de dades, etc.

Actualment, amb Adobe AIR es té accés a una sèrie de funcionalitats comunes que, tret que el sistema operatiu les limiti, són totalment funcionals, com per exemple:

- Instal·lació nativa de l'aplicació.
- Accés al sistema de fitxers del sistema (lectura i, si ho permet el sistema operatiu, escriptura).
- Emmagatzematge persistent bàsic i avançat amb base de dades local.
- Accés a la càmera.
- Accés al micròfon.
- Accés al GPS para geoposicionament.
- Accés a sensors.
- Compatibilitat amb l'ús de sòcols.
- Descodificació de vídeo amb multifil i reproducció en temps real avançats.

### Enllaç relacionat

Aquestes són solament algunes d'aquestes funcionalitats. Per a conèixer la llista completa podeu consultar la pàgina web <http://www.adobe.com/products/air/features.html>.

- Capacitat de renderització HTML en el seu interior (instància del navegador).
- Suport per a 2D i 3D amb acceleració gràfica.
- API per a treball amb gestos i sistemes multitàctils.
- Depuració per USB i Wi-Fi en equip de sobretaula des d'un terminal.

En les seves últimes versions, incorpora funcionalitats realment interessants, com per exemple comunicació amb extensions natives ([http://help.adobe.com/en\\_US/air/extensions/index.html](http://help.adobe.com/en_US/air/extensions/index.html)) desenvolupades “a mida” per a cada plataforma, suport per a arquitectura nativa en 64 bits, renderització de gràfics amb acceleració per maquinari, compressió LMZA per als fitxers SWF i codificació H.264. En el camp dels dispositius mòbils, integra accés a GPS, acceleròmetre, càmera i micro, etc.

#### **Enllaços relacionats**

<http://www.adobe.com/es/products/air.html>

<http://www.adobe.com/devnet/air.html>

[http://help.adobe.com/en\\_US/FlashPlatform/reference/actionscript/3/index.html](http://help.adobe.com/en_US/FlashPlatform/reference/actionscript/3/index.html)

[http://en.wikipedia.org/wiki/Adobe\\_Integrated\\_Runtime](http://en.wikipedia.org/wiki/Adobe_Integrated_Runtime)

[http://es.wikipedia.org/wiki/Adobe\\_Integrated\\_Runtime](http://es.wikipedia.org/wiki/Adobe_Integrated_Runtime)

## 3. Plataforma Flash per a la creació d'ARM

Sense perdre de vista les diferents tecnologies que s'han avaluat en l'apartat anterior, ens centrarem ara en les relacionades amb la denominada **Plataforma Flash** per a analitzar quins avantatges ens aporten a l'hora de treballar amb aplicacions Rich Media.

### 3.1. Capacitats multimèdia

Una de les raons més importants per les quals la Plataforma Flash (d'ara endavant ens hi podrem referir simplement com a Flash) és un referent en la creació d'ARM, és el fet que des dels seus inicis ha estat **enfocada al suport avançat de diferents mitjans**, com ara vectors, imatge, animació, àudio, vídeo, etc.

Aquesta realitat ha provocat que totes les eines i llenguatges que giren al voltant de la Plataforma Flash estiguin clarament orientats a aquest fet, la qual cosa en l'actualitat la converteix, possiblement, en el millor entorn global per a la creació d'aplicacions multimèdia.

Tant l'IDE Flash Professional com el llenguatge de programació ActionScript estan enfocats a la creació i la interacció amb aquests tipus de continguts, cosa que es tradueix en una agilitat i velocitat més grans en els cicles de desenvolupament.

L'IDE permet combinar amb gran efectivitat animació tradicional basada en línia de temps amb programació avançada en un llenguatge madur i orientat a objectes, com és ActionScript 3. La combinació treballa de la mateixa manera amb contingut vectorial, mapes de bits, etc., i permet treballar amb els recursos de manera interna a l'aplicació principal o carregar-la externament.

La llista de tipus de recursos que pot gestionar una aplicació Flash és realment extensa i, avui dia, és probablement la plataforma amb la millor relació capacitat/rendiment global de tots aquests tipus de recursos.

Aquestes característiques bàsiques del Flash Player ampliades mitjançant Adobe AIR amb les seves funcionalitats natives, fan que es converteixi en una opció prou interessant quan volem treballar amb aplicacions basades en mitjans enriquits múltiples.



### 3.2. Capacitats multiplataforma

A més de l'àmplia capacitat de gestió de diferents tipus de recursos, una altra característica que fa que Flash sigui una gran elecció per a la creació d'ARM és la seva **compatibilitat amb múltiples plataformes**.

Avui dia no es posen en dubte les capacitats que ofereix Flash en el navegador, que estan per davant de qualsevol altra tecnologia quan es treballa amb molts tipus de continguts. Conjuntament amb Adobe AIR, aquest plantejament s'estén a aplicacions d'escriptori i, fins i tot, a aplicacions per a dispositius (no solament mòbils, sinó actualment altres com TV, per exemple).

Hi ha un ventall de característiques que potencien aquesta capacitat multiplataforma de Flash i que li donen un avantatge important enfront d'altres tecnologies:

- L'entorn de desenvolupament és comú per a treballar amb les diferents plataformes.
- El llenguatge de programació és el mateix per a treballar amb les diferents plataformes, i s'amplia mitjançant API i funcions esteses específiques, però amb una base comuna.
- El fet de treballar sobre una mateixa versió d'execució (*runtime*) redueix considerablement les incompatibilitats entre plataformes, ja que és l'entorn qui les gestiona internament.

Per exemple, l'ús d'una mateixa classe per al treball amb el sistema local de fitxers, independentment de si és un sistema operatiu Windows o Mac.

- El procés de desenvolupament i producció de l'aplicació final és molt similar en general entre totes les plataformes admeses.

Totes les característiques anteriors repercuteixen molt positivament en la corba d'aprenentatge necessària perquè un desenvolupador (o un equip de desenvolupament) reutilitzi els seus coneixements per a començar a treballar sobre una nova plataforma. De fet, amb Flash els canvis més importants en treballar per a diferents plataformes es troben en la mentalitat necessària per a cadascuna d'elles, més que no pas en el fet d'haver d'adaptar coneixements tècnics.

### 3.3. Capacitats multidispositiu

Quan s'analitzen les capacitats multidispositiu d'una eina, avui dia entren en joc molts factors que cal tenir en compte. Depenent de les capacitats que aporta una eina, es poden tenir avantatges importants en el desenvolupament.

Alguns dels aspectes que afecten notablement el treball amb desenvolupaments que es pretenen executar en diferents dispositius són:

- Dimensions físiques de la pantalla.
- Densitat de píxels de la pantalla.
- Capacitats d'interacció (tàctil, teclat, etc.).
- Limitacions de memòria.
- Limitacions de processador.
- Accés a capacitats específiques del dispositiu.
- Connectivitat i optimització de càrrega de dades.
- Capacitat d'instal·lació "nativa".

Pel que fa a algun d'aquests punts, les característiques innates de Flash aporten beneficis importants, com per exemple el poder treballar amb contingut vectorial amb diferents grandàries de pantalla i densitat de píxels, o gestionar la qualitat de la renderització en funció de les limitacions del processador.

A més, Adobe AIR ha aconseguit (en les seves últimes versions) homogeneïtzar l'accés a les capacitats específiques del dispositiu entre sistemes operatius iOS, Android i BlackBerry Tablet, i apropar el rendiment de les aplicacions entre aquests diversos sistemes.

Per exemple, la classe `flash.sensors.Geolocation` permet treballar amb una mateixa classe i mètodes en diferents plataformes mòbils.

En el desenvolupament actual sota la Plataforma Flash, i salvant les limitacions imposades específicament per alguns fabricants (la més destacada de les quals és la prohibició d'Apple de poder carregar un SWF extern i interactuar-hi), es poden unificar en gran manera desenvolupaments creats conjuntament per a iOS i Android, i en alguns casos la compatibilitat arriba al 100%.

Finalment, la Plataforma Flash soluciona el tema de la creació d'instal·ladors "nadius", que és un punt que, malgrat que sigui simple, no deixa de ser de vital importància, ja que és el primer contacte que l'usuari té amb la nostra aplicació i ha de ser tan estandarditzada com sigui possible en cada plataforma.

### 3.4. Capacitat de gestió de recursos externs

Un altre factor important a l'hora de desenvolupar ARM és la **comunicació i càrrega de continguts externs**. En una època en què Internet està marcant una filosofia de vida, la majoria de les aplicacions necessiten connectivitat per a desplegar totes les seves funcionalitats, i no estem parlant solament de dades, sinó dels continguts mateixos en diferents formats.

Quan es treballa amb diferents formats externament, cal disposar de mecanismes eficients per a sol·licitar la càrrega, gestionar-la i monitorar-la. Flash realitza aquests processos de manera molt semblant, independentment del tipus de contingut que es carregui, i permet controlar la càrrega en la petició, durant la petició i després de la petició.

Pel que fa a la càrrega d'informació entesa com a dades, Flash admet molts formats, com ara text net, parelles variable / valor, XML i JSON, entre d'altres. Aquests mecanismes bàsics s'amplien amb la compatibilitat per a sòcols (que permeten el desenvolupament d'aplicacions multiusuari en temps real), serveis web i, fins i tot, dades serialitzades en format binari per mitjà d'AMF (Action Message Format), que potencien notablement la comunicació entre client i servidor.

Per si això fos poc, Flash també compta amb mecanismes de treball amb determinats tipus de contingut que són vitals per a una millor experiència d'usuari, com és el cas de la compatibilitat amb càrrega d'àudio i vídeo amb reproducció en temps real. A dia d'avui, Flaix és un dominador clar en aquests camps, ja que encara que l'HTML5 hagi millorat notablement les capacitats dels estàndards en aquests fronts, la seva compatibilitat i les seves capacitats són molt limitades pel que fa a la tecnologia d'Adobe.

## 4. Consideracions

Per tots aquests factors, i més característiques que seria impossible enumerar detalladament, la Plataforma Flash és, avui dia, una possibilitat molt interessant i real en el desenvolupament d'ARM.

Des del punt de vista tècnic i de desenvolupament, ofereix avantatges importants com són les seves capacitats multimèdia, multiplataforma i multidispositiu. A més, i gràcies a la reutilització d'eines i codi, des del punt de vista empresarial (en equips de desenvolupament o de professionals individuals) també ofereix grans avantatges gràcies a les reduccions de temps i de costos que comporta el fet de poder aprofitar en gran mesura els coneixements existents en un entorn a l'hora d'endinsar-se en altres diferents.

No obstant això, sempre és fonamental tenir en compte que **no existeix una tecnologia preferible de forma absoluta**. Cada projecte, en funció de les seves característiques, objectius i condicionants, pot obligar-nos a prendre una decisió diferent quant a la tecnologia i a les eines que cal emprar.

La tecnologia (sobretot en la part del client) és molt canviant i està exposada a les tendències que imposen fabricants, sistemes operatius, etc., de manera que és important tenir un coneixement global ampli, per a poder prendre decisions encertades a l'hora d'escollir les nostres eines de treball.