



Adobe Flex 3

présenté par Pierre Schambacher



Remerciements

- Merci à Mme Dery
- Merci à Camille Roux The logo for Camille Roux's blog, featuring a blue speech bubble icon above the text 'CamilleRoux' in a blue, rounded font, with the word 'blog' in a smaller, orange font to the right.
- Merci à Christian Brel et Sandie Dupain



Quel programme ?

- Pierre Schambacher ... qui c'est ?
- Qu'est qu'un RIA ?
- Flex in a Nutshell
- TP client Twitter: PolyTweet



Pierre Schambacher ... qui c'est ?

à venir: Qu'est qu'un RIA ?



Mini CV

Diplômes

- Bac S à Nice en 2004
- DUT Informatique à Nice en 2006
- Ingénieur en Informatique à Polytech'Nice Sophia en 2009

Expériences

- Application Java pour Windows en 2006 à l'Observatoire Océanographique de Villefranche
- Web-Application ASP.Net-C# en 2008 chez Single Buoy Moorings à Monaco
- Application Flex-WebObjects en 2008-2009 chez bebook™ à Sophia Antipolis



Expérience Flex

- Un an de développement
- Application de grande taille
- Utilisation des modules de Flex
- Participation au projet WOX



et quoi d'autre ?

- Recherche actuellement du travail
- Site personnel de veille et partage d'expériences

<http://www.pierreschambacher.com>

- Contact mail: schambac@gmail.com



Qu'est-ce qu'un RIA ?

à venir: Flex in a nutshell



Desktop

- ✓ Bonne exploitation des ressources
- ✓ Déconnecté
- ✓ Confidentialité des données
- ✓ Continuité du contexte
- ✓ Gestion/affichage possible de grandes quantités de données

Web

- ✓ Pas d'installation/update
- ✓ Portable
- ✓ Fortement connecté
- ✓ Traçage de l'utilisateur possible
- ✓ Pas de piratage de licence



Rich Desktop Application
(standalone)

RDA

RIA

Rich Internet Application
(embarqué dans le navigateur)

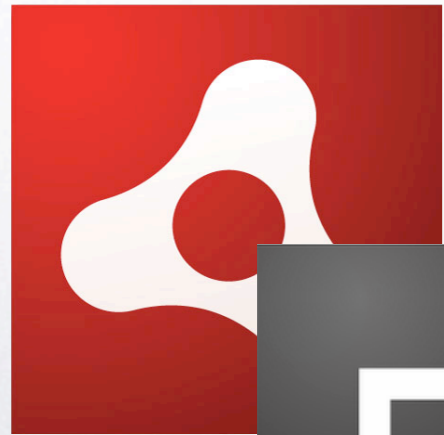


Caractéristiques des RIA

- Utiliser les ressources du client
- Augmenter la logique métier du client
- Diminuer le traitement du serveur



Les technologies RIA



<xul/>





Deux types de RIA

- RIA reposant sur le navigateur (type AJAX)
- RIA reposant sur un plug-in



Les RIA sur plug-in

- Adobe Flex reposant sur Flash (9 et +)
- Java FX reposant sur Java (1.6 et +)
- Microsoft Silverlight reposant sur Silverlight



Caractéristiques

- Multi OS / Multi Navigateur
- Langage UI et langage de comportement
- Repose sur la programmation événementielle
- Permet la création d'UI riches, animées, «glossy»



Différences

- **Public visé**
- **Performance / Complexité de l'application**
- **Communauté de développeur**



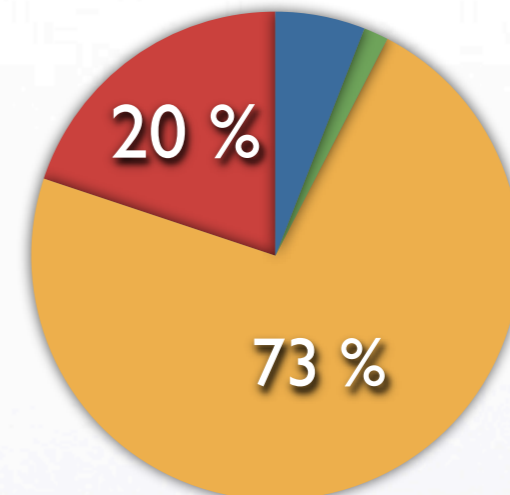
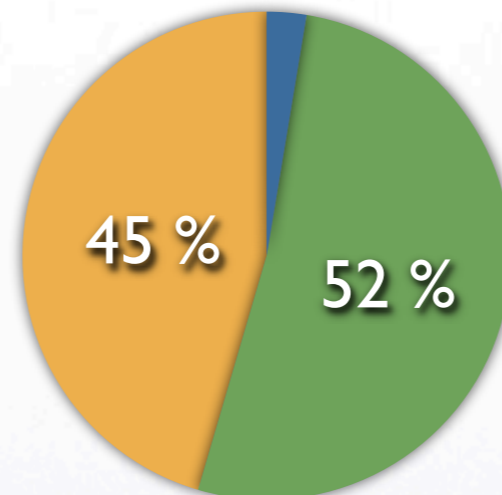
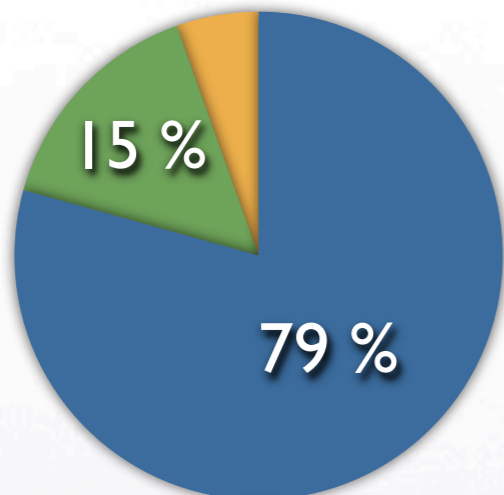
Taux d'installation

Adobe Flash

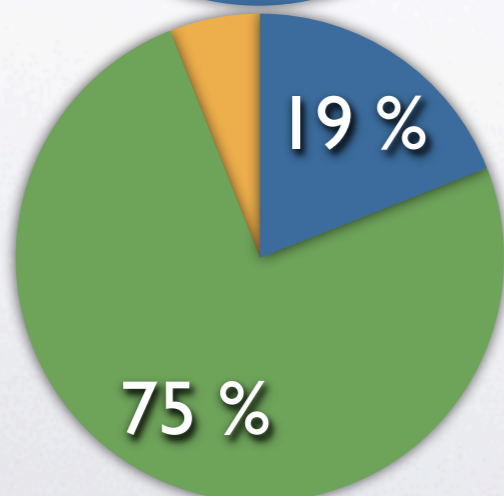
Java

Microsoft Silverlight

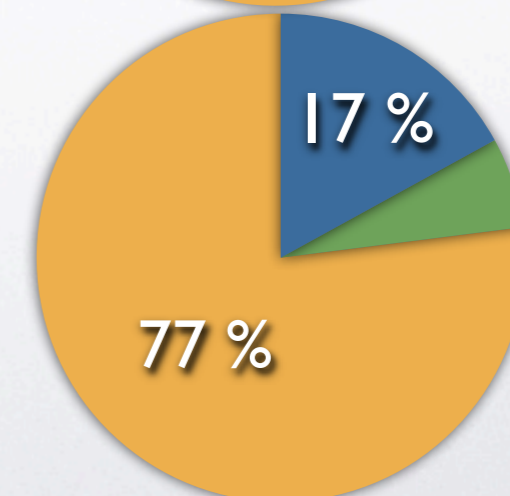
2009



2008



- Version 1.8
- Version 1.6
- Non valide



- Version 10
- Version 9
- Non valide

- Version 2
- Non valide
- Version 1
- Version 3



Flex in a nutshell

à venir: TP client Twitter: PolyTweet

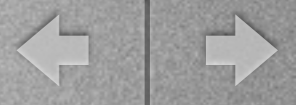


<MXML />



Qu'est-ce que c'est ?

- Format basé sur XML
- Description d'interfaces graphique
- Définition d'états dans l'interface
- Modélisation d'effets de transition
- Insertion de code comportemental

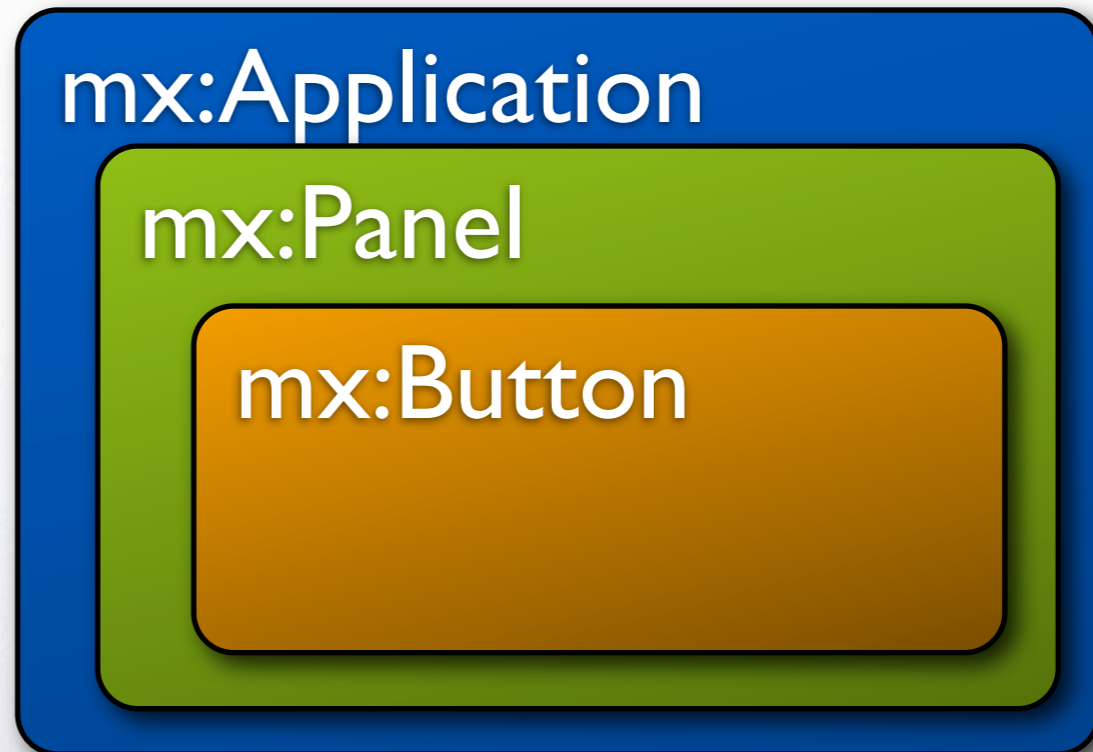


Du vrai XML

Hello World

Espace de
nommage

```
<?xml version="1.0" encoding="utf-8"?>
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml">
  <mx:Panel width="200" height="200">
    <mx:Button label="Hello world"/>
  </mx:Panel>
</mx:Application>
```





Une balise MXML

- Représente un composant graphique (DisplayObject)
- Comporte trois types d'attributs
 - propriétés (**id**, width, height, label, ...)
 - styles (paddingBottom, color, ...)
 - événements (click, enter, focusOut, ...)



La balise Script

- Syntaxe spéciale

```
<mx:Script>  
  <![CDATA[  
    ... // Action Script 3  
  ]]>  
</mx:Script>
```

- Permet d'injecter du code ActionScript 3



ActionScript 3



AS3 en un slide

- Syntaxe de type Eiffel
- Typé mais supporte l'absence de type
- Orienté Objet
- Langage dynamique (ajout de méthodes à la volée)
- Langage Réflexif



À savoir (1/2)

- **Déclarer une variable**

```
public var maVariable:int;
```

- **Déclarer une fonction**

```
public function helloWorld(param:String):void { ... }
```



À savoir (2/2)

- **Déclarer un getter**

```
public function get name():String { ... }
```

- **Déclarer un setter**

```
public function set name(value:String):void { ... }
```



Objet ? ... presque !

- Possibilité de définir des interfaces
- Possibilité de définir des classes
- Possibilité de faire de l'héritage simple

MAIS

- Pas de classes ni de méthodes abstraites
- Pas de constructeur privé



Où placer l'ActionScript ?

- Dans une balise Script
- Dans une balise d'événement du composant (creationComplete par exemple)
- Dans une classe séparée du MXML



Hello World 2

```
<?xml version="1.0" encoding="utf-8"?>
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml">
  <mx:Script>
    <![CDATA[
      import mx.controls.Alert;
      public function helloWorld():void {
        Alert.show("Hello World !");
      }
    ]]>
  </mx:Script>
  <mx:Panel width="200" height="200">
    <mx:Button label="Click me" click="helloWorld()"/>
  </mx:Panel>
</mx:Application>
```



Bindings



Bindings ?!

- Flex est basé sur la programmation événementielle
- Telle variable vaut la valeur de cette autre variable
- En pratique: abonnement aux modification d'une variable pour mise à jour de valeur



Syntaxe

- Deux méthodes

```
<mx:Label id="userName" text="{user.name}" />
```

```
<mx:Binding source="user.name"  
destination="userName.text" />
```



Déclarer un attribut Bindable

[Bindable]

```
public var name:String;
```



Bilan à l'usage

- Très pratique pour sortie brute de données
- Très pratique pour écriture de données sans contrôle
- Permet une bonne séparation MVC



States



Pourquoi ?

- Permet la définition d'états dans l'interface
- Simplification de l'ajout d'éléments
- Manière élégante pour représenter les changements de l'interface graphique



Actions Possibles

- **AddChild**: ajouter un élément
- **setProperty**: changer une propriété
- **setEventHandler**: changer le comportement sur un événement
- ...



Syntaxe

```
<mx:states>  
  <mx:State name="logged">  
    <mx:AddChild relativeTo="{panel}">  
      <mx:Button id="logout" label="Logout" />  
    </mx:AddChild>  
    <mx:SetProperty target="{label}"  
      property="text" value="Loggued" />  
  </mx:State>  
</mx:states>
```



Effets et Transitions



Pourquoi ?

- **Animer l'interface**
- **Adoucir les changements d'état**
- **Améliorer l'expérience de l'utilisateur**



Syntaxe

```
<mx:transitions>  
  <mx:Transition fromState="*" toState="*">  
    <mx:Parallel targets="{[windowA, windowB]}">  
      <mx:Move />  
      <mx:Resize />  
    </mx:Parallel>  
  </mx:Transition>  
</mx:transitions>
```

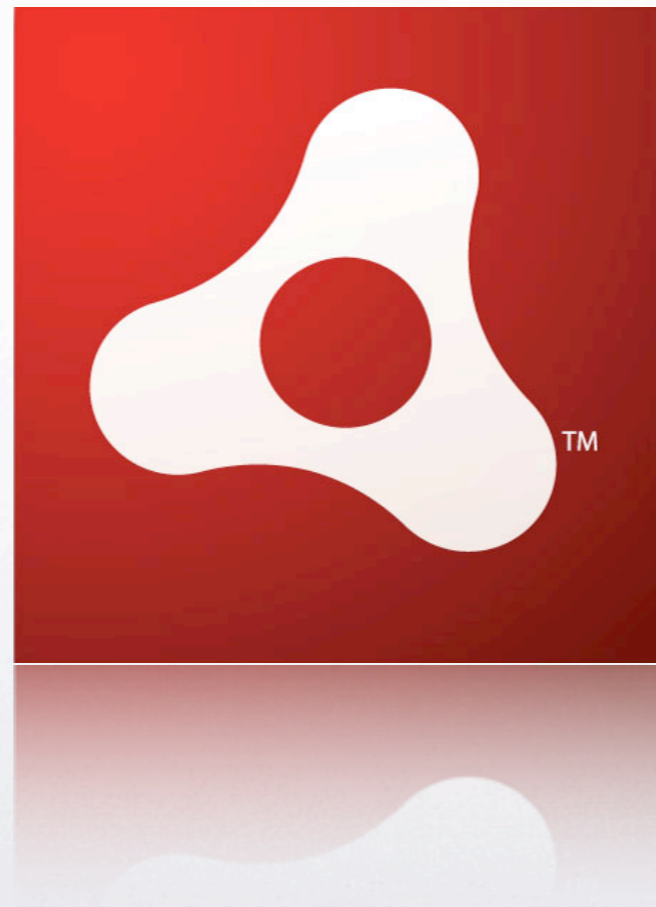


En pratique

- Alourdit l'interface
- Les transitions doivent être soigneusement dosées
- Difficile de faire l'effet voulu
- Peu utilisées



R I A





Quoi de plus dans AIR?

Accès au système de
fichier

UI natif
(fenêtres et menus)

Processus en
tâche de fond

PDF

SQLite

WebKit

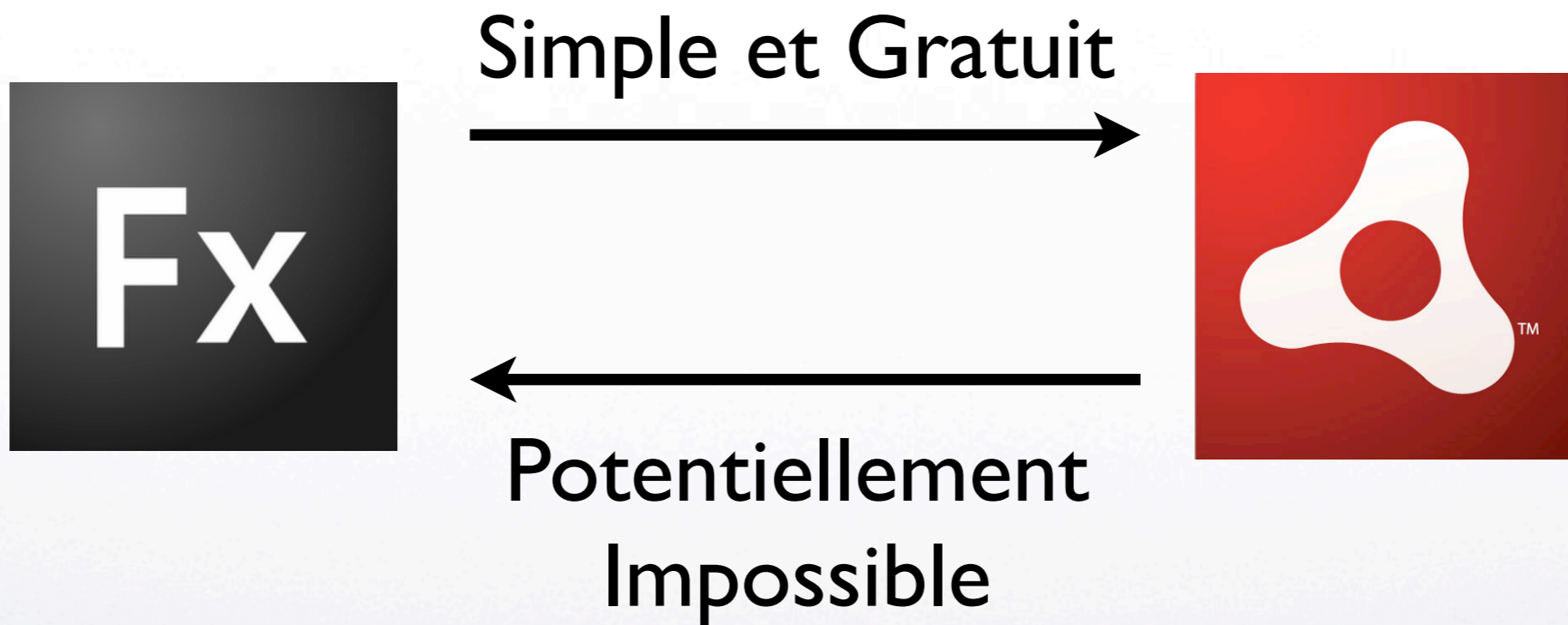
API pour les
mises à jour

Stockage local

Monitoring de la
connexion réseau

Contrôle du
chrome

Système de notification





Autour de Flex



Flex 4 (1/2)

- **Nouvelle librairie de composants SPARK**
- **Styles remaniés**
- **Bindings bi-directionnels**
- **Changement de fonctionnement des états**



Flex 4 (2/2)

- Bindings bi-directionnel

```
<s:TextInput id="myText2" text="@{myText1.text}" />
```

- Gestion des états

```
<s:states>  
  <mx:State name="blue" />  
  <mx:State name="red" />  
</s:states>
```

```
<s:Label text="Hello World !"  
  color.blue="#0000FF" color.red="#FF0000" />
```



Flash Builder 4



- Eviter la confusion Flex/Flash
- Intégration avec *Flash Catalyst*
- Moins de bug et mieux optimisé
- Génération de getter/setter
- Génération de handler d'événements
- Evalueur d'expression et breakpoints conditionnels



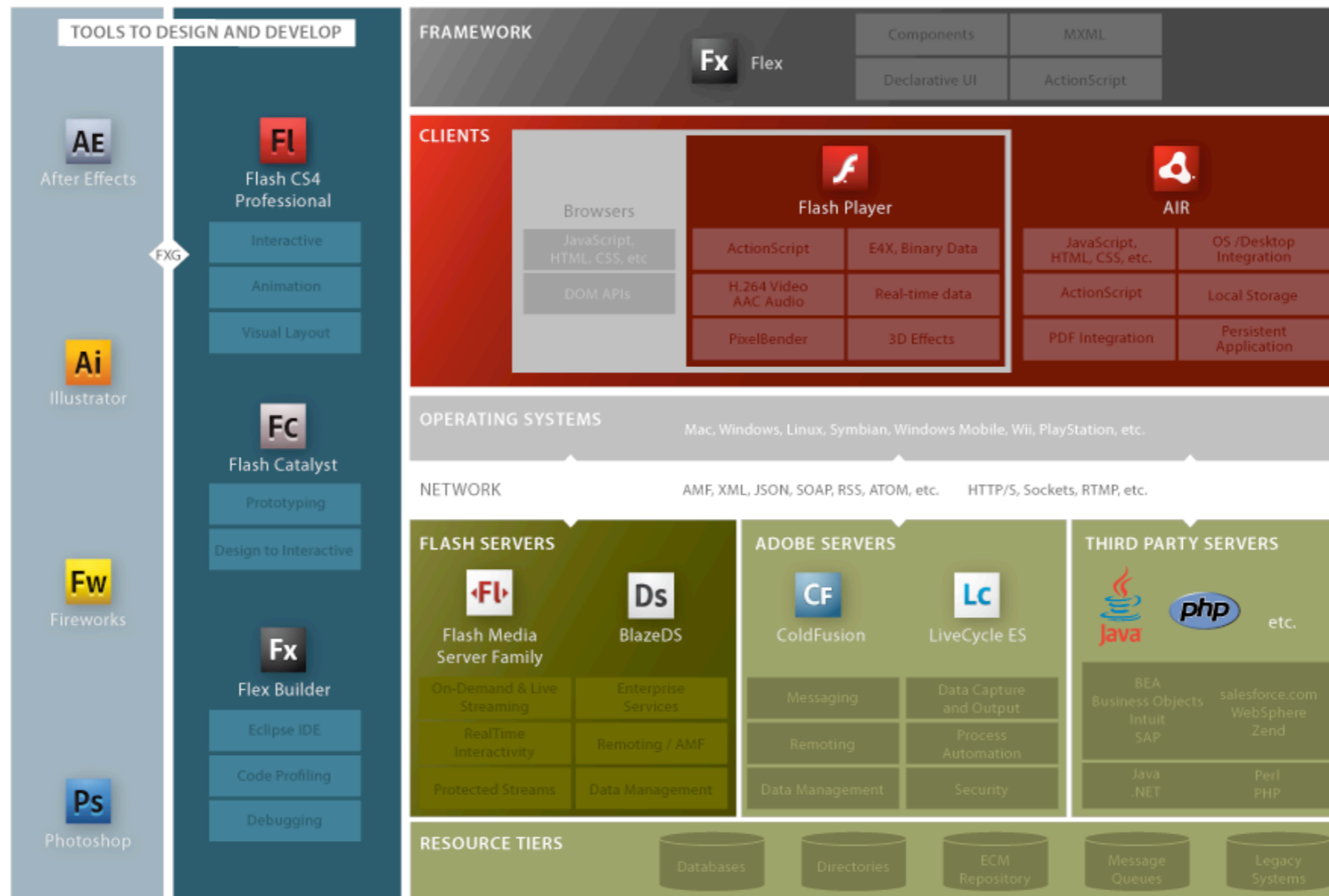
BlazeDS, LCDS, GraniteDS

- Data Services plutôt orientés JavaEE
- Permet une connexion avancée avec un serveur au lieu de services XML
- Basés sur le protocole AMF
- **!!\ Attention à la décompilation /\!**



Flash Catalyst

Adobe Flash Platform and web technologies





I an de Flex



Points négatifs

- API bancaire
 - pas de vrai dictionnaire
 - nouveauté de Flex 3.4: addAll dans ArrayCollection !!!!
- Tentation de briser MVC
- Difficile de créer des interfaces complexes
- Mono-threadé
- IDE en beta ?
- Problèmes mineurs (bugs, scroll sous mac, ...)



mais ...

- **Systeme de Bindings très puissant**
- **Parfait pour des interfaces prédéfinies**
- **Peu de problèmes pour de petites applications**
- **Librairie e4x fonctionne bien avec les services XML**



Autres



Bibliothèques

- **FlexUnit** (<http://code.google.com/p/as3flexunitlib/>)
Tests unitaires
- **Thunderbolt** (<http://code.google.com/p/flash-thunderbolt/>)
Logger extension
- **Flexlib** (<http://code.google.com/p/flexlib/>)
Composants graphiques
- **Degrafa** (<http://www.degrafa.org/>)
Framework graphique



Frameworks

- **Cairngorm** (<http://opensource.adobe.com/wiki/display/cairngorm/>)
- **PureMVC** (<http://puremvc.org/>)
- **Mate** (<http://mate.asfusion.com/>)
- **Tide** (<http://graniteds.blogspot.com/2008/10/gdstide-flex-3-framework.html>)
- **Architectures et Framework Flex / Air**
par Matthieu Segret
<http://techtalks.intellicore.net/2008/05/29/architectures-frameworks-flex/>



Liens à connaître

- **Flex Language Reference** (<http://livedocs.adobe.com/flex/3/langref/>)
- **Flex ressources** (<http://www.adobe.com/support/documentation/en/flex/>)
- **Flex in a week** (<http://www.adobe.com/devnet/flex/videotraining/>)
Vidéos et tutoriels pour apprendre Flex en 1 semaine
- **Flex.org** (<http://flex.org/>)
Agrégation d'actualités provenant de la sphère Flex/Air
- **Adobe Flex Tutorial** (<http://www.flex-tutorial.fr/>)
Site français contenant de nombreux tutoriaux sur Flex
- **TourDeFlex** (<http://flex.org/tour>)
RDA en Air pour explorer des ressources et des composants Flex



Sources

- **Rich Internet Applications (RIA): A Convergence of User Interface** - *Florian Moritz*
<http://www.flomedia.de/diploma/documents/DiplomaThesisFlorianMoritz.pdf>
- **Flash, Flex & AIR: A brief survey** - *Travis Isaacs*
<http://www.slideshare.net/tbisaacs/flash-flex-air-a-brief-survey>
- **Pratique d'ActionScript 3** - *Thibault Imbert*
<http://pratiqueactionscript3.bytearray.org>
- **AdvancED Flex 3** - *Shashank Tiwari & Elad Elrom*



Questions ?





TP client Twitter: PolyTweet



Utilisation de HTTPService

```
<mx:HttpService id="service"  
    url="http://service.com"  
    result="handler(event)"  
    fault="logger(event)" />
```

```
service.send([parameters]);
```