

# Préambule

## Autour de la création d'une application ou sites flash AS3

L'as3 pourquoi ?

Construire une application (site, jeu...), ne signifie pas se lancer tout de suite dans une création de code à tout va

Une publication papier, par exemple, est pensée en termes de grille, de charte graphique, de hiérarchisation du contenu. Avec flash la chose n'est pas différente : Il y a des étapes à ne pas négliger avant de se lancer dans des lignes de codes au risque de se perdre sous les traces() et les impasses !

En écartant la charte et la création graphique, il reste quatre étapes indispensables à la création d'une application flash. Ou plutôt : il va falloir envisager son application de trois manières différentes. La première concerne le temps de la visite. La seconde définit l'espace de l'application et l'agencement des éléments graphiques et interactifs des uns vers les autres. Enfin la dernière étape de pré-projet consiste en un listage des fonctionnalités et actions, et concerne donc tant l'ergonomie pour le visiteur que l'architecture des différentes classes qui composeront le programme.

Toutes ces étapes nécessitent plus le papier, la règle, et le stylo que les touches de votre clavier. C'est le moment de prendre un bol d'air, et de penser sous un arbre - comme le veut la tradition -, avant de se lancer dans les tréfonds d'ActionScript !

### **Première chose à faire : se documenter.**

La formation que vous suivez est avant tout destinée à vous permettre de voler de vos propres ailes. Les possibilités liées à ActionScript sont gigantesques qu'il s'agisse d'effets visuels ou d'applications complexes. En somme, vous pouvez dire à vos futurs clients que tout est réalisable, qu'il s'agit seulement d'une question de budget !

En sortant de cette formation vous devriez être au fait sur l'organisation et la mise en place d'une application, et surtout vous devriez avoir pris l'habitude de savoir comment vous diriger pour manipuler une technique ou un objet que vous n'aurez pas eu l'occasion de voir durant le cours.

Adobe met à la disposition de l'utilisateur d'ActionScript toute une gamme de documentation. Les deux plus importantes sont l'aide à la programmation AS3 en français. Vous y trouverez des démarches étapes par étapes pour réaliser certaines animations ou applications. Vous y trouverez souvent les codes nécessaires à leur application.

Autre source possible, le livre, Pratique de l'ActionScript 3 de Tibault Imbert ( Entré cette année dans l'équipe de concepteurs du logiciel) trouvable en libre accès. Il traite des grands aspects de l'ActionScript 3 avec de nombreux codes en exemple

L'autre outil indispensable est le dictionnaire de référence AS3, malheureusement encore non traduit en français. Il est également pourvu de nombreux exemples utilisables directement.

## Se fabriquer sa boîte à outils permanente

Au fil des projets, nous nous rendons compte que certaines classes seront utilisées régulièrement alors que d'autres n'auront d'utilité que pour le projet en cours. Principalement les premières concerneront :

- Les fonctions redondantes qui pourront être stockées en tant que fonctions statiques dans une classe thématique (comme une classe de placement automatique d'objets, outils de calcul divers...).
- Les classes de chargement (texte et graphiques).
- Les classes de communication externes.
- Certains évènements personnalisés...

Nous pouvons alors faire le choix de séparer ces classes dans un répertoire à part et d'en donner le chemin à flash et ou flashDevelop et qui deviendront disponibles de façon permanent au fil des projets. Nous pouvons alors capitaliser une lignée importante de codes qui nous resserviront à travers nos différents projet sans avoir de copie et donc de versions différentes de ces classes.

Nous configurons flash CS3 en ce sens :

Menu : Modifier / préférences / actionScript / paramètres d'actionScript 3 / « ajouter le répertoire cible » ;

Pour flash develop

Menu : tools / global Classpaths / "ajouter le répertoire cible"

## Création de Sites :

Choisir une stratégie de site full flash : Incorporé sur une page ou sur plusieurs ? Voici une petite liste des avantages de l'un et de l'autre pour vous aidez à faire valoir vos choix auprès de vos clients.

### Avantages de l'incorporation

- Maîtrise de processus de chargement. Et scénarisation de l'affichage
- Chargement unique des parties principales de l'application.
- Référencement possible et « apparence de navigation » à l'aide d'un système de « deep linking ».

### Avantages de la séparation des pages

- Sectorisation de l'application.
- Comportement du site plus proche des habitudes d'un internaute (utilisation des outils d'historiques).
- Plus de clics sur le site.
- Meilleur référencement naturel.

### Le point sur le référencement Flash

Le principal problème d'un site full flash est son référencement naturel.

Dans l'état actuel de choses, le moteur de recherche n'est capable de voir que les textes et les liens inscrit en dur dans l'environnement auteur. Aucun document lié comme un xml n'est interprété comme faisant partie de l'application.

La seule solution actuellement pour cela est d'utiliser une interface flash / javascript assurant le référencement et la navigation dans l'historique donc l'accès direct à une page ou l'utilisation des fonctions du navigateur. On citera parmi ces systèmes « swfaddress ». Malheureusement l'adresse dispensée contient un « # » ce qui fait croire au moteurs que le lien concerné est une ancre de la même page. Pour contourner cela vous travaillez en conjonction avec un développeur php qui préparera un fichier d'url rewriting.

Il en reste que la meilleure solution reste de proposer aux moteur un contenu alternatif dans la balise dédiée au swf dans la page html.

## Le poids de l'application

La question du poids de fichier se pose dès lors où le fichier nécessitera plus de deux à trois secondes avant d'avoir le moindre affichage. A partir de 500 ko (dépendant de l'audience visée) pour une enveloppe de site full flash, le temps commence à paraître long sans le moindre petit élément graphique pour divertir l'internaute.

Il convient alors de placer une intro scénarisée dans le fichier de départ... Ce qui permettra de pouvoir télécharger les ressources principales externes sans donner l'impression de vide...

## Intégration de l'application avec SWFObject

SWFObject est la manière la plus pertinente d'intégrer du flash sur une page html. Il est en cours de généralisation dans dreamweaver.

Il fonctionne à l'aide d'un fichier javascript externe qui met en place le swf quel que soit le navigateur employé. Seules subsistent sur la page html, les instructions modifiables ou utilisables par le créateur (passage de variables, taille fixe ou non...).

Pour mettre en place ce système deux solutions :

- Si vous employez flashDevelop vous pouvez récupérer le modèle de la page en créant un nouveau projet AS3. Attention ne faites pas de celui-ci votre projet. Un projet flashDevelop AS3 est compilé via flex sdk et non le compilateur du logiciel. La classe principale n'étend donc pas un fichier fla mais est en réalité le point de départ de l'application.
- Vous pouvez aussi récupérer les sources et la documentation à l'adresse suivante : <http://code.google.com/p/swfobject/> vous aurez ainsi accès à une connaissance un peu plus intime du fonctionnement et des possibilités de cet objet javascript.

## Utilisation.

Vous aurez besoin dans votre dossier, en plus du swf, d'un dossier « js » dans lequel se trouve le fichier swfObject. Et du lien de téléchargement du player « expressInstall ». Vous pouvez bien sûr ranger ces objets de manière différente en modifiant les chemins d'accès dans votre page html.

La page suivante peut être utilisée en tant que modèle pour le placement d'un swf en « pleine page ».

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>
<head>
  <title>Classes</title>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <meta name="language" content="en" />
```

```

<meta name="description" content="" />
<meta name="keywords" content="" />
<!--importation de swf object -->
<script src="js/swfobject.js" type="text/javascript"></script>
<script type="text/javascript">
<!-- vous pouvez passer des paramètres au travers de l'objet
flashvars syntaxe > id1 : « valeur 1 », id2 : « valeur 2 », ... -->
    var flashvars = {
    };
<!-- ici les paramètres de réglages du player (comportement dans la
page, apparition du menu complet... ) cf. swfObjet doc -->
    var params = {
        menu: "false",
        scale: "noScale",
        allowFullscreen: "true",
        allowScriptAccess: "always",
        bgcolor: "#FFFFFF"
    };
<!-- paramètre utilisé par mozilla pour localiser l'animation dans la
page. Utiliser le nom du fichier sans l'extension -->
var attributes = {
    id:"Classes"
};
<!-- instruction d'insertion (on y reconnaît une fonction) 1 : fichier / 2
: Div d'insertion / 3,4 : largeur, hauteur / 5 : version du player /
fichier de lien vers l'installation du player requis / 6, 7, 8 : objets mis
en place ci-dessus -->
    swfobject.embedSWF("Classes.swf", "altContent", "100%", "100%",
"9.0.0", "expressInstall.swf", flashvars, params, attributes);
</script>

<!-- les styles sont ici définis afin de faire apparaître le swf sur la
fenêtre entière du navigateur. -->
<style>
    object:focus { outline: none; }
    html, body { height:100%; margin:0; }
    body { margin-top:0px; background-color : #ffffff}
    h1{font-family:Arial, Helvetica, sans-serif; color:#99CC33}
    p{font-family:Arial, Helvetica, sans-serif; color:#fcfcfc}
</style>
</head>
<body>
<!-- div d'insertion : Attention le div est REMPLACE par le swf. Si vous
voulez le placer dans une page composée vous devez l'insérer dans un autre
div et gérer le placement par celui-ci -->
    <div id="altContent">
        <h1>Classes</h1>
        <p>Alternative content</p>
        <p><a href="http://www.adobe.com/go/getflashplayer"></a></p>
    </div>
</body>
</html>

```