

# Votre première application Apollo

par Olivier Bugalotto ([Mes articles](#))

Date de publication : 02/04/2007

Ce tutoriel vous guidera afin de réaliser votre première application Apollo : le célèbre Hello World !

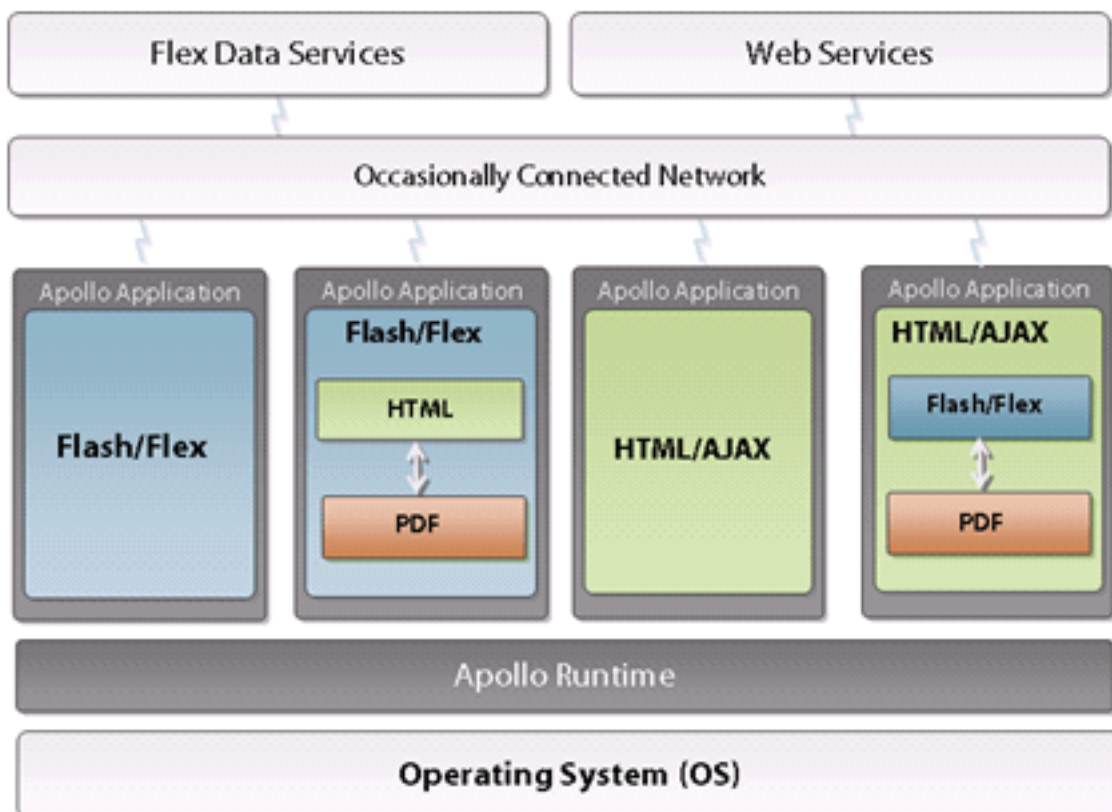
- I - Introduction
- II - Notre première application Apollo
- III - Des changements évidents
- IV - Le fichier de configuration
- V - Le déploiement
- VI - Sources

## I - Introduction

Apollo est un runtime d'exécution permettant la construction d'applications riches de bureau, à l'aide de technologies comme Flash, Flex, AJAX, HTML, JS. Apollo est concurrent à des technologies comme .net et Java pour ne citer qu'eux. Il en est qu'à ses débuts avec déjà de grandes possibilités :

- **Multiplateforme** : plus besoin de développer différentes versions pour les différents OS.
- **Gestion du HTML/JS/CSS/AJAX** : pas besoin d'en dire plus.
- **Gestion du PDF** : pas disponible dans cette alpha publique.
- **API de gestion IO** : Nous allons pouvoir enfin, lire, écrire et modifier des fichiers.
- **Extension Apollo pour Eclipse** : un environnement assez complet pour développer rapidement.

Voici un diagramme, nous présentant les différentes combinaisons de technologies citées plus haut :

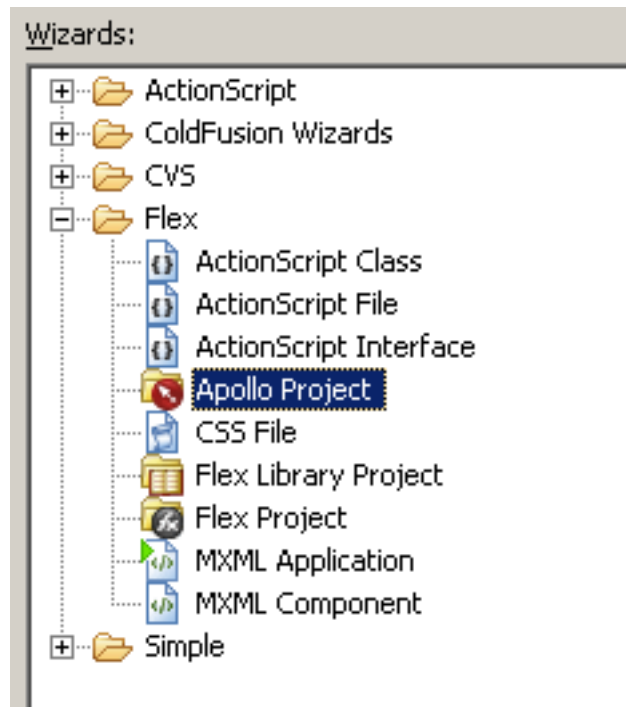


Et ces possibilités ne s'arrêtent pas là ...

## II - Notre première application Apollo

Ce tutorial utilise l'extension pour Eclipse.

Lançons Eclipse, créons un nouveau projet Apollo (Ctrl + N) :

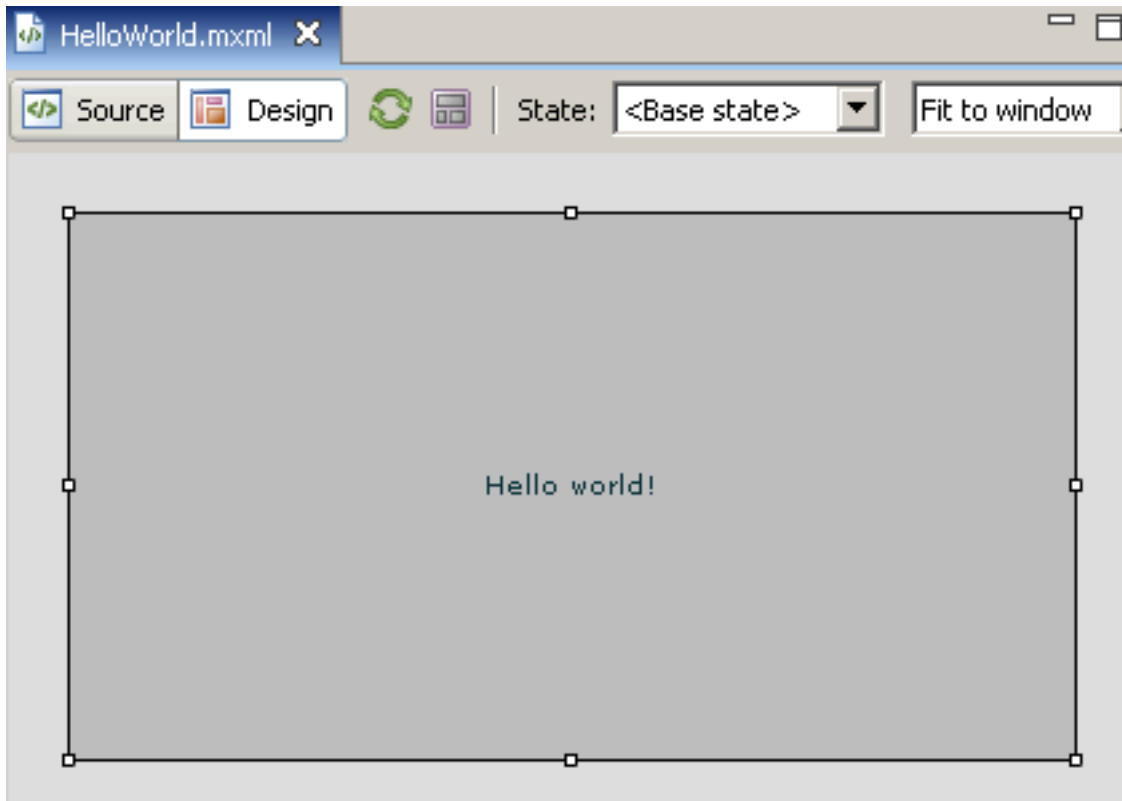


Nous allons construire un projet basique, et nous l'appellerons HelloWorld (tient pour changer).

### III - Des changements évidents

Une application Apollo est basée sur un composant conteneur ApolloApplication qui lui même hérite du composant Application (Flex 2), imaginez le portage de vos applications Flex 2 sous Apollo, il suffit de remplacer Application par ApolloApplication mais nous allons voir que ce n'est pas obligatoire.

Passons en mode design, pour travailler dans un environnement RAD :

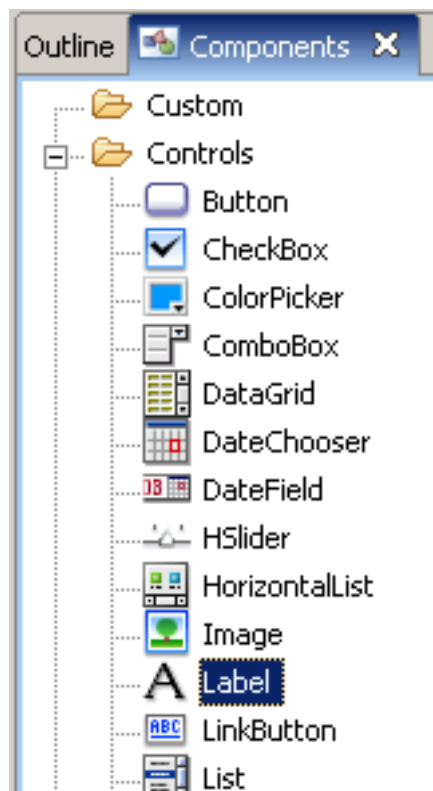


1. Sélectionnons le conteneur gris (ApolloApplication) et choisissons un mode de placement à *vertical* (par défaut *absolute*). Il y a 3 modes de placement dont un, le mode *absolute* est disponible uniquement pour les conteneurs : Panel, TitleWindow, Canvas et Application.

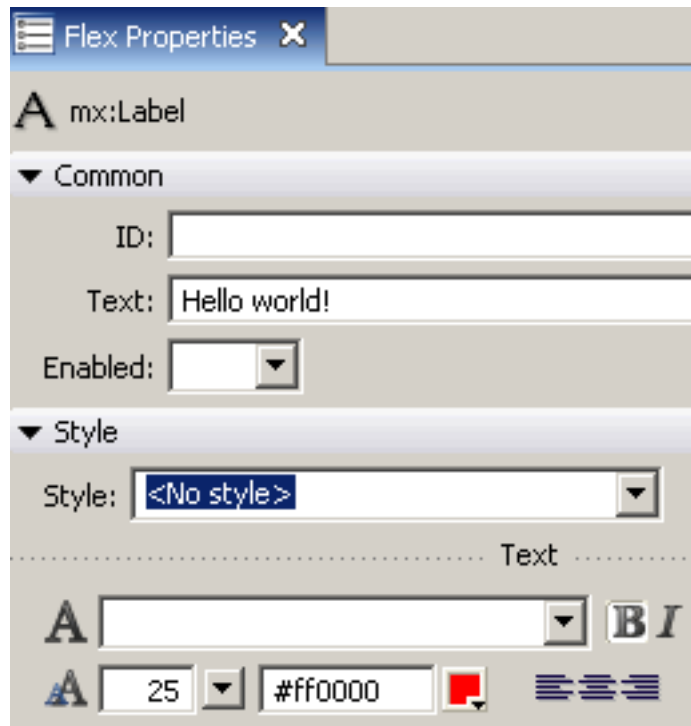
Ce mode permet de placer les composants à l'aide de leurs coordonnées x et y. Les deux autres permettent le placement des composants verticalement et horizontalement.



2. Ajoutons maintenant un contrôle Label que vous trouverez dans l'onglet composants (drag & drop) :

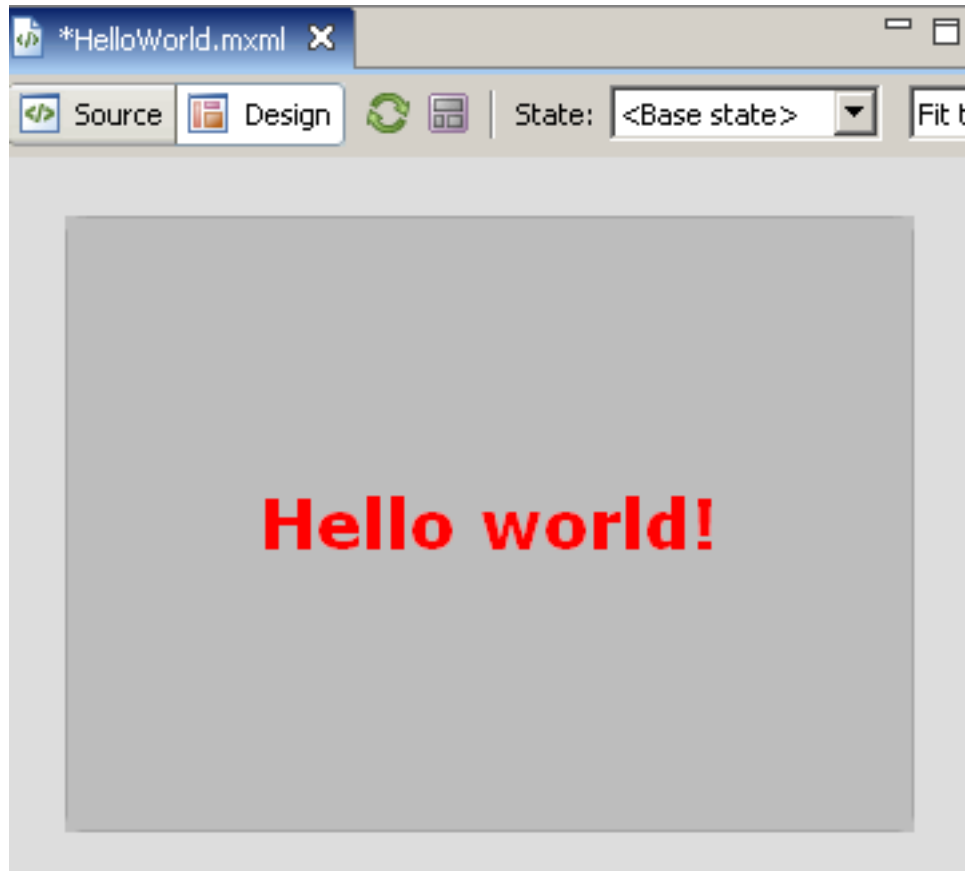


3. Modifions le texte et le style de notre Label



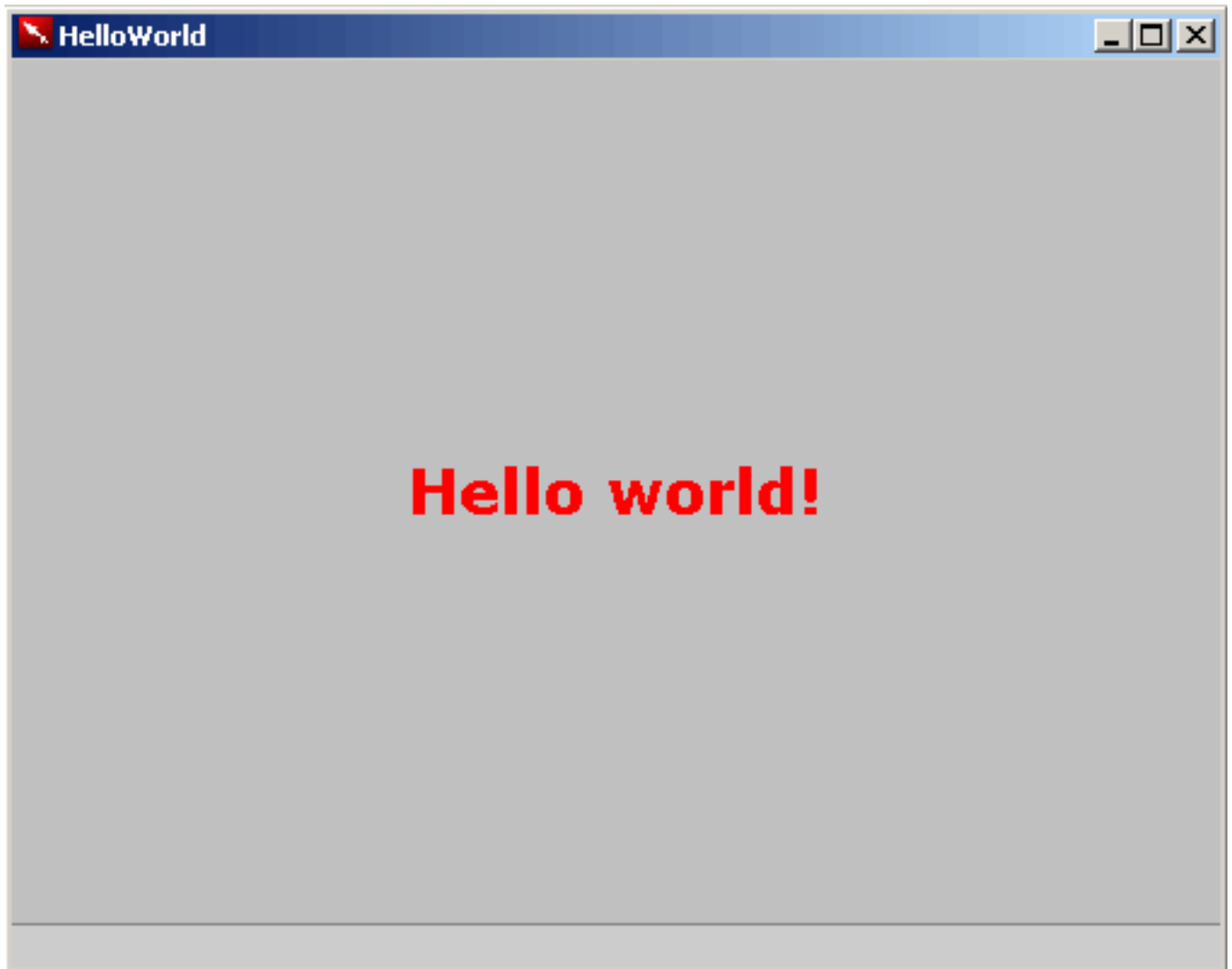
Le contrôle Label se trouve toujours en haut de votre application, nous allons y remédier.

4. Dans le panneau des propriétés, affichons les propriétés par categories. Déroulons la catégorie Styles et modifions la valeur du style VerticalAlign à middle



5. Testons notre application. Pour compiler, nous pouvons cliquer sur la flèche verte en haut ou en utiliser le raccourci (Ctrl + F11). Pour compiler en mode debug, il suffit de cliquer sur l'insecte vert ou d'utiliser le raccourci (F11).

Voici le rendu de notre application :



Notre application est une application bureau, ce qui implique la possibilité de réduire, d'agrandir et de fermer l'application.

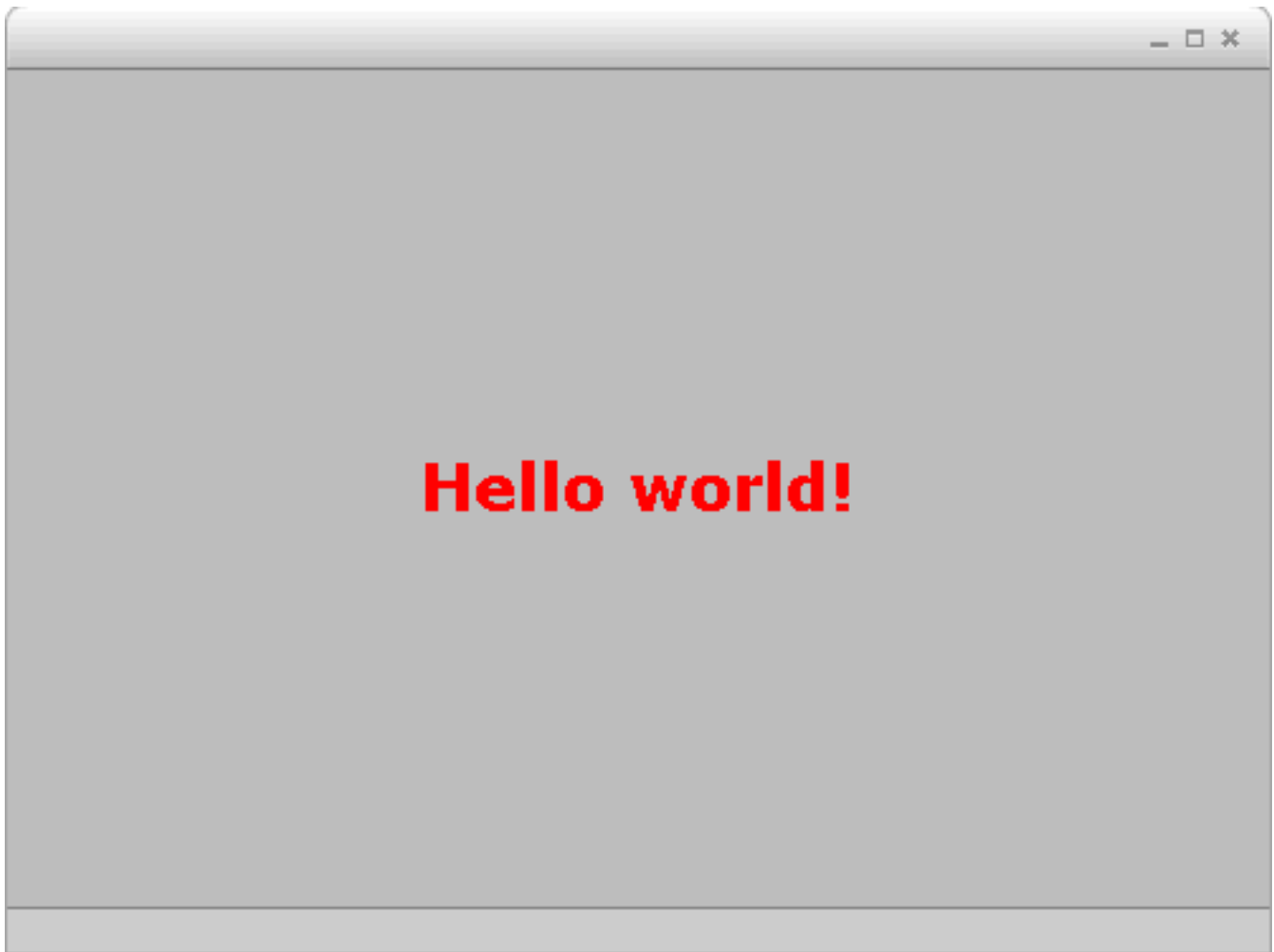
## IV - Le fichier de configuration

Il existe sous Apollo un fichier XML de configuration de votre application Apollo, donc voici un aperçu :

```
<application appId="HelloWorld" version="1.0"
  xmlns="http://ns.adobe.com/apollo/application/1.0.M3">
  <properties>
    <name>HelloWorld</name>
    <description>Notre première application</description>
    <publisher>Iteratif</publisher>
    <copyright/>
  </properties>
  <rootContent systemChrome="standard" transparent="false"
    visible="true">[SWF reference is generated]</rootContent>
  <icon>
    <image16x16>assets/IT16x16.png</image16x16>
    <image32x32>assets/IT32x32.png</image32x32>
    <image48x48>assets/IT48x48.png</image48x48>
  </icon>
</application>
```

Ce fichier nous permet de définir :

- un attribut *appId* : identifiant unique de l'application
- un attribut *version* : *version="major.minor"* - ex : *version="1.0"*
- un ensemble de propriétés indiquant : un nom, une description, un auteur et un copyright
- un noeud *rootContent* : définit l'apparence de votre fenêtre d'application à l'aide de l'attribut *systemChrome* avec deux valeurs possibles : *standard* (défaut) celle que nous avons actuellement pour notre application et aucune *none* à savoir l'apparence proposé par le composant *ApolloApplication* :

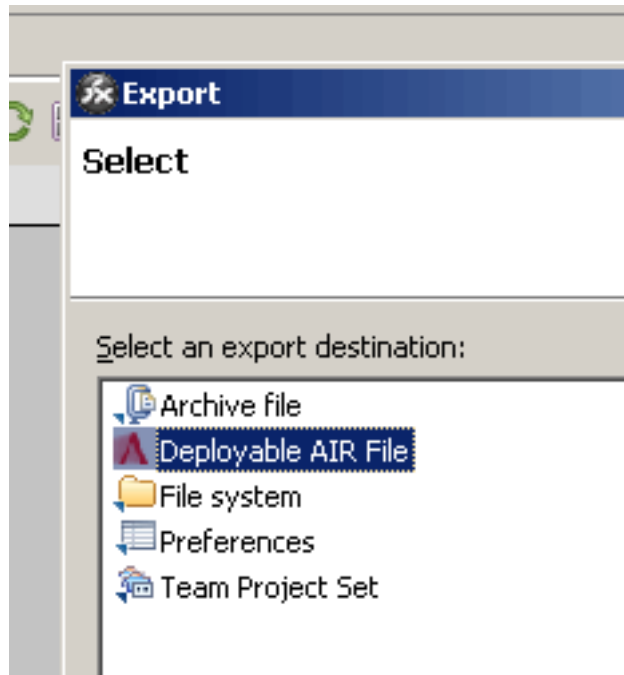


- des icônes : plusieurs formats, un format pour l'icône de la fenêtre, un autre pour l'icône de raccourci bureau et sur l'exé.

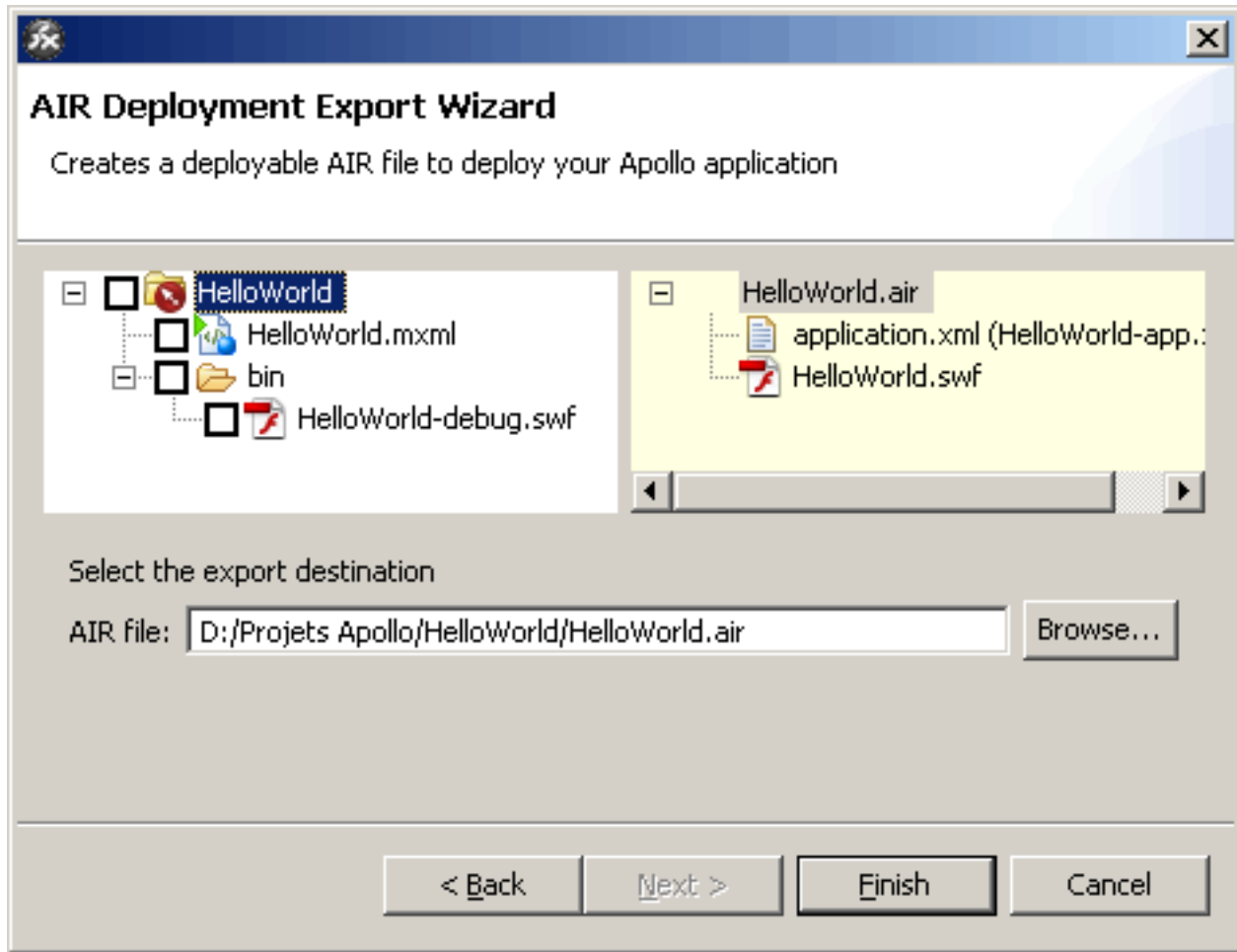
Si vous choisissez *none*, pensez à mettre l'attribut *transparent* à **true** pour éviter de laisser la couleur de fond par défaut de votre application.

## V - Le déploiement

Il y a rien de plus simple, là où vous devez développer différentes versions de votre application en fonction de l'OS visé, il suffit simplement de sélectionner le dossier du projet Apollo de cliquer sur le bouton droit de votre souris et de choisir d'exporter :



L'assistant vous permet d'ajouter le code source ou toutes autres ressources, ainsi que l'endroit de sauvegarde du fichier d'installation .air :



Ce fichier d'installation (.air) est multiplateforme ... Après avoir installé l'application, vous pouvez supprimer l'application en utilisant l'ajout et suppression de programmes dans le panneau de configuration si vous êtes sous Windows.

## VI - Sources

Les sources de l'application sont disponibles [ici](#) ( **miroir** )

