



Ładowanie zasobów w środowisku Adobe Flex 3

Autor: Marcin Klimek

Kraków, 12 sierpnia 2008



Polecenie kompilator Embed

Jest to polecenie kompilatora pozwalające **osadzić zasoby bezpośrednio w aplikacji**. Oznacza to, że nie będą one doładowywane w trakcie działania aplikacji, a zostaną one załadowane na samym początku jej uruchamiania.

Za pomocą dyrektywy Embed można ładować **różnego typu pliki**. Bezpośrednio wspierane są:

- ▶ pliki graficzne jpeg, png, gif oraz svg
- ▶ pliki dźwiękowe mp3
- ▶ pliki swf (również kompilowane dla FP 8 i wcześniejszych)
- ▶ pliki czcionek ttf oraz otf
- ▶ pliki innego typu (np. XML)

Za pomocą polecenia Embed można osadzać pliki w ActionScript, MXML oraz CSS.

Polecenie kompilator Embed

Najogólniej polecenie *Embed* ma następującą postać:

```
Embed(parameter1, paramater2, ...)
```

Do możliwych do zdefiniowania parametrów zaliczają się:

- ▶ plik źródłowy (*source*)
- ▶ typ pliku (*mimeType*)
- ▶ położenia plasterków (*scaleGridTop*, *scaleGridBottom*, *scaleGridLeft*, *scaleGridRight*)
- tylko w przypadku plików graficznych
- ▶ *instance name* symbolu (*symbol*) - tylko w przypadku pliku swf

Polecenie kompilator Embed

Polecenie Embed dopuszcza następujące *typy mime* plików:

- ▶ application/octet-stream
- ▶ application/x-font
- ▶ application/x-font-truetype
- ▶ application/x-shockwave-flash
- ▶ audio/mpeg
- ▶ image/gif
- ▶ image/jpeg
- ▶ image/png
- ▶ image/svg
- ▶ image/svg+xml

Pierwszy typ - *application/octet-stream* służy do definiowania danych nie wspieranych bezpośrednio przez środowisko Adobe Flex.

Załączenie pliki w języku ActionScript ma postać:

```
<mx:Script>
  <![CDATA[
    [Embed(source="test.jpeg")]
    public var imgClass:Class;

    public var imgObj:BitmapAsset = new imgClass() as BitmapAsset;
  ]]>
</mx:Script>
```

Importowane w ramach ActionScript elementy są również dostępne z poziomu MXML:

```
<mx:Image id="obrazek" source="{imgClass}"/>
```

Załączając pliki w języku MXML procedura Embed musi zostać poprzedzona znakiem "@".

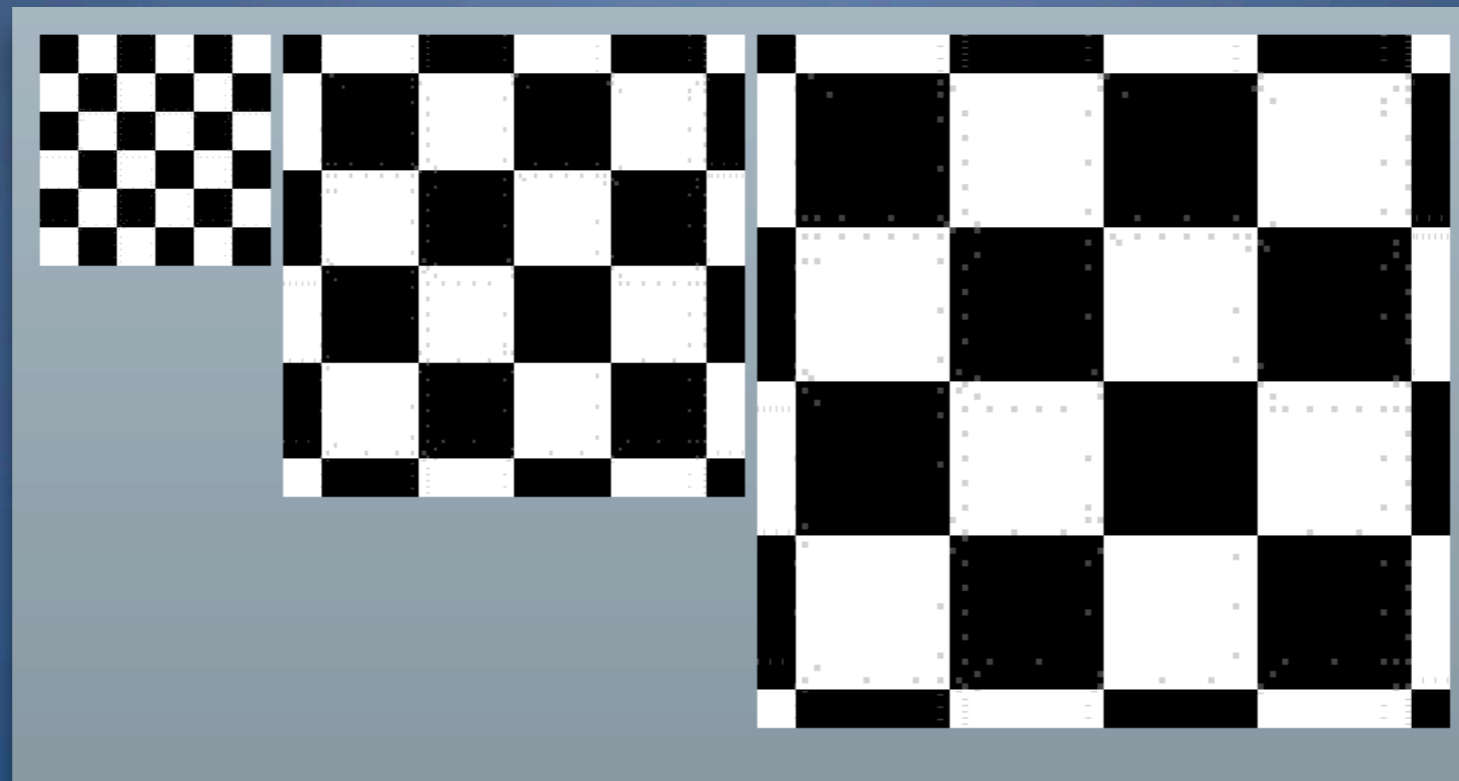
```
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml">  
  <mx:Image id="obrazek" source="@Embed(source='test.jpg')" />  
</mx:Application>
```

Stosując procedurę *Embed* w CSS można na przykład osadzić zasoby związane z wyglądem komponentów.

```
<mx:Style>
  .myCustomButton {
    overIcon:Embed("overIconImage.gif");
    upIcon:Embed(source="upIconImage.gif");
    downIcon:Embed(source="downIconImage.gif");
  }
</mx:Style>
```

Skalowanie 9-plasterkowe

Dodatkową możliwością przy załączaniu grafik w środowisku flex jest możliwość stosowania skalowanie 9-plasterkowego (9-slice scaling). Pozwala ono na zdefiniowanie 9 obszarów, które w raz ze skalowanie będą się zmieniały tylko w pionie, w poziomie lub w obu tych wymiarach.



Skalowanie 9-plasterkowe

Definiowanie prostych określających granice plasterków wymaga **zdefiniowania 4 parametrów** procedury *Embed* określających ich położenie:

```
<mx:Script>
  <![CDATA[
    [Embed(source="test.gif",
      scaleGridTop="25", scaleGridBottom="125",
      scaleGridLeft="25", scaleGridRight="125")]
    public var imgClass:Class;

    public var imgObj:BitmapAsset = new imgClass() as BitmapAsset;
  ]]>
</mx:Script>
```

Załączanie plików SWF

Polecenie *Embed* pozwala na *pracę z plikami SWF*. Możliwe jest załadowanie zarówno całego pliku jak i poszczególnych symboli opisanych za pomocą parametru *Class* definiowanego w właściwościach obiektu.

Gdy załączany jest symbol, uwzględniane są również wszystkie jego zależności - czyli wszystkie zagnieżdżone obiekty.

Możliwe jest załączanie całych aplikacji flex. Ten sposób zakłada jednak brak interakcji z ładowaną aplikacją. Gdy interakcja jest wymagana należy użyć ładowanego zasobu jako komponentu lub załadować go w trakcie działania aplikacji (run-time).

Możliwy jest natomiast dostęp do wizualnej struktury za pomocą metod `getChildByName()` lub `getChildAt()`.

Załączanie plików SWF

Załączanie całych plików swf ma postać:

```
<mx:Script>  
  <![CDATA[  
    [Embed(source='shapes.swf')]  
    public var imgCls:Class;  
  ]]>  
</mx:Script>
```

Załączanie symbolu polega na podaniu pliku źródłowego i symbolu, który ma zostać załączony. Można to zrobić na dwa sposoby.

```
<mx:Script>  
  <![CDATA[  
    [Embed(source='shapes.swf', symbol='greenSquare')]  
    public var imgCls:Class;  
  ]]>  
</mx:Script>
```

```
<mx:Script>  
  <![CDATA[  
    [Embed('shapes.swf#greenSquare')]  
    public var imgCls:Class;  
  ]]>  
</mx:Script>
```

Dobłą praktyką jest załączanie używanych czcionek do aplikacji flex.
Podstawowymi **zaletami** tego rozwiązania są:

- ▶ czcionka nie musi być zainstalowana na komputerze klienta,
- ▶ czcionka obsługuje anti-aliasing, napisy są wygładzone i bardziej czytelne,
- ▶ napis obsługuje przezroczystość (możliwość animowania alpha),
- ▶ napis może być płynnie skalowany i obracany

Podstawową **wadą** tego rozwiązania jest często znaczne zwiększenie rozmiaru pliku SWF.

http://livedocs.adobe.com/flex/3/html/help.html?content=fonts_04.html#133099

Czcionki mogą zostać załączone w stylach CSS, poniżej odpisane podstawowymi parametrami

```
@font-face {  
  src: url("location") | local("name");  
  fontFamily: alias;  
  [fontStyle: normal | italic | oblique;]  
  [fontWeight: normal | bold | heavy;]  
  [advancedAntiAliasing: true | false;]  
}
```

gdzie parametr **src** definiuje lokalizację fontu po ścieżce do pliku (url) lub nazwie systemowej (local), **font-family** określa używaną we flexie nazwę czcionki, **fontStyle** oznacza styl formatowania, **fontWeight** grubość. Możliwe jest też ustawienie zaawansowanego wygładzania za pomocą paramteru **advancedAntiAliasing**.

Załączanie Czcionek

Czcionki mogą zostać załączone w ActionScript.

Dodatkowo może zostać użyty parametr *unicodeRange* określający zakres embedowanych znaków.

```
<mx:Script>
  /*
   * Embed a font by location.
   */
  [Embed(source='../assets/MyriadWebPro.ttf',
    fileName='myMyriadFont',
    mimeType='application/x-font'
  )]
  // You do not use this variable directly. It exists so that
  // the compiler will link in the font.
  private var font1:Class;

  /*
   * Embed a font with bold typeface by location.
   */
  [Embed(source='../assets/MyriadWebPro-Bold.ttf',
    fontWeight='bold',
    fileName='myBoldMyriadFont',
    mimeType='application/x-font',
    advancedAntiAliasing='true'
  )]
  private var font2:Class;

</mx:Script>
```

Warto przeczytać:

http://livedocs.adobe.com/flex/3/html/help.html?content=embed_4.html#178459

http://livedocs.adobe.com/flex/3/html/help.html?content=fonts_04.html#133099

http://livedocs.adobe.com/flex/2/docs/wwhelp/wwhimpl/common/html/wwhelp.htm?context=LiveDocs_Parts&file=00000968.html