

実践! 色々なモバイルアプリを作る

背景が毎回変わる「Flash 待ち受け」を作る

■ケータイの「Flash 待ち受け」を作る

最近のモバイル市場ではスマートフォンが全盛ですが、国内では、ガラケー(ガラパゴスケータイ)と呼ばれるフィーチャーフォン(ケータイ)も、まだまだ根強い人気があります。本ガイドでは、これらケータイ向けに、待ち受け用の Flash コンテンツを動的に生成・出力するプログラムを作成します。開発済みの待ち受け Flash ファイルの背景画像を、ケータイからダウンロードの要求があるごとにランダムな画像に入れ替えます。このアプリケーションを、Windows Azure にデプロイします。

■前提とする知識と開発環境

このサンプルは、以下の知識や開発環境が構築できていることを前提としています。

□FlashLite と PHP の知識

1. FlashLite 1.1 についての知識
2. PHP についての知識

□Azure 向けの PHP 開発環境

1. Windows Azure をデプロイできるアカウントとサブスクリプション
2. Visual Web Developer、Eclipse for Windows Azure、エディタ等で PHP を Windows Azure 向けに開発・実行できる環境

□ケータイ向けの FlashLite を使う

FlashLite は、携帯端末等のモバイルデバイス向けに実装された Flash のサブセット規格です。PC 向けの Flash と比べて、単にアプリケーションサイズやメモリ、解像度の違いにとどまらず、fscommand/fscommand2 と呼ばれる携帯独自の関数群が実装されています。PC 向けの Flash 同様に、Adobe Flash Professional CS5 などのツールを使って開発できます。

FlashLite は 2011 年 4 月現在で、バージョン 4 まで発表されていますが、対応端末の普及事情から、多くの端末で実行可能なバージョン 1.1 を前提にして開発しています。このバージョン 1.1 は PC 向け Flash バージョン 4 のサブセットとなります。

本稿では、FlashLite を使って作成した Flash コンテンツをサンプルとして同梱していますが、FlashLite や ActionScript については、前提の知識として進めます。Flash は経験があるけれど FlashLite の経験が無い方は、以下の記事を一読することを推奨します。

Think IT 「出遅れるな! Flash Lite 入門」 (<http://thinkit.co.jp/book/2008/09/24/170>)

□swfmill で Flash を動的に書き換える



図 1 (swfmill.org)

今回作る Flash 待ち受けは、すでに開発済みの待ち受け Flash ファイル (SWF) をそのまま利用しながら、ケータイからダウンロードの要求があるごとに、背景画像を切り替えた待ち受け Flash を動的に生成します。この仕組みを実現するために今回利用するソフトウェアが、swfmill です。

swfmill (<http://swfmill.org/>)は、Daniel Turing さんによって開発され、2011 年 4 月現在では Daniel Cassidy さんによってメンテナンスされているオープンソースのソフトウェアです。今回は最新安定版の swfmill-0.3.1-win32.zip を使っています。

一般的に、出力した Flash ファイル (SWF)を編集するには、作成したオーサリング環境で再編集を行い、再度パブリッシュする必要がありますが、swfmill を使うと、パブリッシュ済みの SWF から XML、XML から SWF へ相互変換することができます。

今回作る待ち受け Flash では、この機能を生かして、図 2 に示した流れで SWF を編集し、直接端末へ出力します。



図 2 (swfmill の編集の流れ)

1. ユーザに提供するパブリッシュ済の SWF を用意します
2. swfmill を使って SWF から XML へ変換します
3. PHP で 2 の XML を置換します
4. swfmill を使って 3 の XML から SWF へ変換します
5. PHP で SWF をブラウザへ出力します

swfmill は、Windows 向けに Win32 実行バイナリが提供されており、Windows Server 2008 R2 をベースにした Windows Azure の各種インスタンスで、そのまま実行することができます。

□PHP の開発環境を用意する

本稿では PHP の実行と開発環境については、事前に準備がされているものとします。環境構築については Windows Azure で PHP を実行する各種資料を参考にしてください。

筆者の環境では

- Windows7 Ultimate Edition (x64)
- Windows Azure SDK 1.4
- Eclipse 3.6 (Helios) PDT
- Windows Azure Tools for Eclipse Plugin

を用いて開発していますが、Windows Azure で PHP を実行できる環境であれば、エディタのみや Visual Web Developer 2010 Express Edition でも、同様にコードを実行できるでしょう。

■待ち受け画面を作成する

□Flash 画面の作成



図 3 (tmpl.swf)

添付の SWF ファイル (tmpl.swf)は、図 3 に示したのシミュレーター実行画面のように、とてもシンプルな待ち受けです。

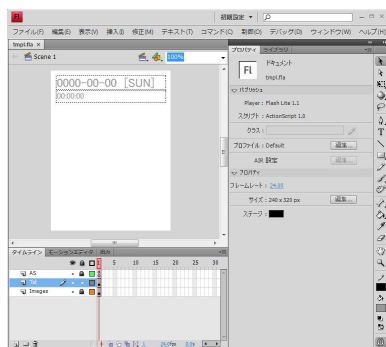


図 4 (Flash 作成画面)

この SWF は図 4 に示した作成画面のように

- ・ 背景は横 240px×縦 320px の無地の jpg (tmpl.jpg)
- ・ ActionScript と fscommand を使った時刻、曜日などの表示

で構成されています。

サンプルでは Windows Azure 用の壁紙を tmpl.jpg と同じ横 240px×縦 320px で 7 枚作成し wallpaper フォルダに 0.jpg~7.jpg として連番で配置しています。

今回サンプルとして同梱している壁紙は、カラチとパキスタンで積極的にコミュニティ活動をしている、Ali Raza Shaikh 氏がデザインしたものです。サンプルとしての利用を快諾してくれた氏に謝意を述べるとともにサイトを紹介します。

Microsoft Feed

10+ Beautiful Microsoft Windows Azure Wallpapers

<http://microsoftfeed.com/2011/10-beautiful-microsoft-windows-azure-wallpapers/>

By Ali Raza Shaikh (Microsoft MVP)

今回の同梱画像はトリミングをしていますが、是非 PC 用の壁紙としても使ってください。

□Eclipse でプロジェクトを作成



図 5 (Eclipse でのプロジェクトツリー)

Windows Azure 向けにプロジェクトを作成しましょう。例は Eclipse でプロジェクト作成を行い、必要なファイルを追加した画面です。

1. 用意した壁紙は wallpaper フォルダを作り、その中に 0~7.jpg を配置します。
2. swfmill.exe / tmpl.jpg / tmpl.swf をプロジェクトルートに配置します。
3. Web.config の defaultDocument に index.php が設定されていることを確認します。

```
<!-- Example WebRole IIS 7 Configuration -->
<defaultDocument>
  <files>
    <clear />
    <add value="index.php" />
  </files>
</defaultDocument>
```

□プロジェクトに PHP コードを追加、作成する

次に、swf を変換、生成する PHP コードを、以下のような内容で作成します。

```
---index.php---
<?php
// swfmill.exe 実行 swf から xml へ変換
$templateSwf = "tmpl.swf";
$xml = shell_exec("swfmill.exe -e cp932 swf2xml $templateSwf");

// 画像の置換 tmpl.jpg からランダムに選択した jpg へ
$tplImg = base64_encode("%xFF%D9%FF%D8" . file_get_contents("tmpl.jpg"));
```

```
$wallpaper = "wallpaper¥¥" . mt_rand(0, 6) . ".jpg";
$replaceImg = base64_encode("¥xFF¥xD9¥xFF¥xD8" . file_get_contents($wallpaper));
$xml = str_replace($tmplImg, $replaceImg, $xml);

// ユニークなファイル名を生成して、置換結果を xml に保存
$xmlFilename = uniqid(rand()) . '.xml';
file_put_contents($xmlFilename, $xml);

// 保存した XML から SWF に変換
$outFilename = uniqid(rand()) . '.swf';
$swfOutput = shell_exec("swfmill.exe -e cp932 xml2swf $xmlFilename $outFilename");

// SWF を出力
header('Content-type: application/x-shockwave-flash');
echo file_get_contents($outFilename);

// 一時生成したファイルの削除
unlink($xmlFilename);
unlink($outFilename);
```

Flash Lite 1.1 で画像を置き換える際に注意すべき点として、2 つ挙げることができます。

1. swfmill.exe のオプションに -e cp932 を指定する
2. FlashLite 1.1 の場合の JPEG には、バイナリ FF D9 FF D8 を先頭に付与する

1 と 2 の注意点は、Flash Lite1.1 が Flash4 のサブセットであるために生じます。詳しい仕様は SWF の仕様書などに譲りますが、Flash4 のサブセットである Flash Lite1.1 では、ファイル内の文字エンコードとして cp932 が扱われています。また、JPEG の埋め込みは DefineBitsJPEG2 という SWF タグで埋めこまれており、erroneous header (FFD9 FFD8) が先頭に追加されています。

コーディングが済んだら、早速パッケージングして、Windows Azure ヘデプロイしましょう。



図 6 （実機で実行）

`http://デプロイしたインスタンスの FQDN/`に、対応端末からアクセスすると、画像が置換されて表示されるでしょう。ランダムで画像が変化することもありロードして確認してみてください（図 6）。

いかがでしょうか？今回は、コードをシンプルにするために固定の画像で変換、生成を行っていますが、パフォーマンスなどを考慮して、以下のような改良もできるでしょう。

1. 生成済みの SWF を Windows Azure Storage（Blob サービス）にキャッシュさせる
2. 画像を Windows Azure Storage（Blob サービス）から呼び出して動的に置換

このような改良を加えて、クラウドが本来持つ機能を積極的に活用すれば、変換の負荷や稼働率に応じて柔軟にスケールリングさせることが可能です。

■最後に

ここまで、フィーチャーフォン向けの Flash 待ち受けを Windows Azure を使って動的に生成するアプリケーションを開発してきました。swfmill.exe のような、任意の 32 ビット、64 ビットアプリケーションを実行させられる点や、ASP.NET だけでなく PHP や各種スクリプト言語を扱うことができる点において、Windows Azure は柔軟で扱いやすく、とても魅力的な PaaS プラットフォームです。PHP を簡単に扱える Windows Azure のプラットフォームの良さを、是非体感してください。