

Monoを使って Webアプリケーションを 走らせる

| 上村 亮

KAMIMURA, Ryo

UNIX環境でASP.NETを実行するための導入編

「ASP.NETを始めたいけど周りはみんなUNIXだし」

「IISのライセンスを買わずに無償でASP.NETアプリケーションを公開できないだろうか」

そんなふうに思ったことはないでしょうか？ 大学の研究室などではOSはUNIX、WebサーバーはApacheというところも多いと思います。UNIX環境しかないからといってASP.NETが使えないわけではありません。Linux + Apacheの環境があればASP.NETを動かすことができます。

本稿ではLinux上で.NET Frameworkを動作させるMonoを使ってASP.NETを実行する手順を紹介します。



Mono (<http://www.mono-project.com/>) はMono Projectが開発を進めている.NET Framework互換のオープンソースプロジェクトです。Monoを使うとLinux上で.NET Frameworkを利用できます。MonoはLinuxだけでなくWindows、Mac OS X、FreeBSDといった多くのプラットフォーム上で動作します。

このプロジェクトはオープンソフトウェアメーカーであるXimianの1プロジェクトとして2001年7月に始まりました。XimianはNovellに買収されましたが、開発はそのまま進められ、2004年の6月末にVer 1.0を正式リリースしました。

今も開発が続けられており、2005年1月の時点で安定版が1.05、開発版が1.13となっています。

Mono Projectでは以下のものが開発されています。

- **mcs** : コンパイラ、ランタイム、クラスライブラリ
- **MonoDoc** : Monoのドキュメント
- **Gtk#** : Gtk+やGNOMEモジュールを.NETにバインディングしたもの
- **Cocoa#** : MacOS Xで使われているフレームワークのバインディング
- **XSP** : ASP.NETをMonoで動かすためのサーバー

- **mod_mono** : ApacheからASP.NETを利用するためのモジュール



それでは早速Monoをインストールしてみましょう。ここではFedora Core 3でのインストールについて説明します。

●yumの設定

Monoはyum^[注1]を使ってインストールします。残念ながらMonoのダウンロードページにはFedora Core 3用のレポジトリがありません(2005年1月25日現在)。そこで今回はサードパーティのレポジトリを利用します。

まず/etc/yum.repo.dに「fedora-others.repo」を作成し、リスト1にあげた3行を加えます。

Fedora Core 3ではyumを使うのにGPG公開鍵をインストールしていないとエラーが出ます。したがって、GPG公開鍵もインストールします。yumを初めて使う方はもうひとつGPG公開鍵をインストールします

注1) 導入したいアプリケーション (UNIXではパッケージと呼ばれる) の依存関係を解決し、レポジトリと呼ばれるサーバーからインストールしてくれるツール。

リスト1: yumとPGPのインストール

```
fedora-others.repoに記述する
[dag]
name=Dag RPM Repository for Fedora Core
baseurl=http://apt.sw.be/fedora/$releasever/en/$basearch/dag

dagのPGP公開鍵のインストール
# rpm --import http://dag.wieers.com/packages/RPM-GPG-KEY.dag.txt

Fedora公式レポジトリのPGP公開鍵のインストール
# rpm --import /usr/share/rhn/RPM-GPG-KEY-fedora
```

● Mono本体のインストール

yumを使う準備ができたなら、yumコマンドでインストールします。先ほどの設定が済んでいれば、安定版のVer1.0.5がインストールされます。インストール中にはMonoの依存関係が表示されます。[y] キーを押して進めてください。

```
# yum install mono-complete
...
Dependencies Resolved
Transaction Listing:
  Install: mono-complete.i386 0:1.0.5-1.1.fc3.rf
(略)
Is this ok [y/N]:
```

同じようにmono-develとgtk-sharpもインストールしておきます。

```
# yum install mono-devel gtk-sharp
```

これでMono本体のインストールは終わりです。

http:// Monoの実行

Monoではmcsというコンパイラを使います。ソースファイルをコンパイルするとWindowsの実行形式であるPE形式(.exe)のファイルができあがります。これをmonoコマンドで実行します。

ソースコードの書式や名前空間などは、それほど大きく変わりません。Windowsベースで開発しているときと同様です。エディタでソースを記述し、コマンドラインでコンパイルすることになります。

ソースコードが「Hoge.cs」だった場合は、「mcs」というコマンドにファイルを渡せば「Hoge.exe」が生成されます。

ここで生成されたEXEファイルは中間コードなので、Javaと同じようにランタイム上で走らせることになります。

一連の流れは以下のようになります。

```
$ mcs Hoge.cs
$ mono Hoge.exe
```

http:// XSPとmod_monoのインストール

次にASP.NETを動かすために必要なXSPとmod_monoをインストールしましょう。MonoでASP.NETを実行するためにはサーバーのXSPが必要になります。また、ApacheからASP.NETを利用するためにはmod_monoが必要です。

Mono本体はyumでインストールできるのですが、この2つはyumでインストールできません。したがって、これらはソースからコンパイルしてインストールする必要があります。

ソースファイルをMono Project (<http://www.mono-project.com/downloads/>) のページからダウンロードしてきます。

本稿執筆時点のアーカイブは、

- XSP : xsp-1.0.5.tar.gz
- mod_mono : mod_mono-1.0.5.tar.gz

になります。

ダウンロードしたら、適当な場所に展開し、コンパイルしてインストールします。依存関係があるので、まずはXSPからです。

XSPをmake install

● 展開した場所に移動し、引数に「--prefix=/usr」を指定してコンフィギュアします。makeが成功したら、スーパーユーザーになってインストールします。

UNIXに馴れている人ならお馴染みの手順ですね。馴れていない方は、大文字/小文字に注意して導入してください。

```
$ cd xsp-1.0.5
$ ./configure --prefix=/usr
$ make
$ su -
# make install
```