

C#でつくるLDAPブラウザ

Microsoft C#にはディレクトリサービスにアクセスするためのクラスが用意されていて、ASP.NETの中から呼び出してディレクトリ情報を取得することができる。これを利用して、Webブラウザを利用してディレクトリ情報を閲覧するためのユーザインターフェースをMicrosoft IIS(Internet Information Server)上で稼動するWebアプリケーションとして簡単に作成することができる。

Microsoftは2007年になり、自社のVisual Studio系の開発環境製品の無料版を配布しはじめた。その中には、C#やASP.NETなどのWebの開発環境も含まれている。また、Novellはmono projectでLinux上で稼動するC#と.NETの開発環境とサーバを開発して公開している。monoは最新の.NETフレームワークへの追従は遅れているものの、既存の.NETアプリケーションをある程度動かすことができる。逆に、monoで開発を行いIISで稼動させることも可能である。ただし、クロス開発を行ったアプリケーションの適応性に関しては実際に運用環境でのテストを行うまでわからないことも多いため、開発に際して十分にテストを行い、ノウハウについてのガイドラインを設けるなどの工夫が必要である。

Novellは自社のWebサイトにソリューションとして、monoとVisual Studioの双方で開発して実行することのできるディレクトリサービスブラウザを紹介している。

(<http://www.novell.com/cool solutions/appnote/1673.html>)

C#のディレクトリアクセス

ADSI (Active Directory Service Interfaces) は、Microsoft Windows Active Directory のプログラミング インターフェイスで、Visual Studio .NET および .NET Framework により、**DirectoryEntry** コンポーネントおよび **DirectorySearcher** コンポーネントが備わった ADSI 機能を簡単に追加できる。

([http://msdn2.microsoft.com/ja-jp/library/system.directoryservices.directoryentry\(VS.80\).aspx](http://msdn2.microsoft.com/ja-jp/library/system.directoryservices.directoryentry(VS.80).aspx))

DirectoryEntry クラスは、“プロバイダ://プロバイダパス” の書式でバインドするリソースを指定するが、次のプロバイダをサポートしている。

- LDAP://
- WinNT://
- NDS://
- NWCOMPAT://

次の例は、DirectoryEntry クラスを利用してLDAPディレクトリサーバにアクセスし、記載情報を取得してリスト出力する。

```

<%
    DirectoryEntry de = new DirectoryEntry(
        "LDAP://127.0.0.1:389/ou=User,dn=example,dn=com"
        , "cn=Admin,dn=example,dn=com", "password"
        , System.DirectoryServices.AuthenticationTypes.ServerBind);
    foreach(DirectoryEntry child in de.Children)
    {
        Response.Write("<br>" + child.Name.ToString());
    }
%>

```

例題では、"<LDAP://127.0.0.1:389/ou=User,dn=example,dn=com>" のように、ローカルマシンの LDAP サーバにアクセスし、example.com ドメインのユーザ情報にバインドしている。

.NET C# の LDAP ブラウザの mono への移植

Novell の Cool Solutions サイト:<http://www.novell.com/cool solutions/appnote/1673.html> には、.NET C# の LDAP ブラウザアプリケーションを Linux 上の Mono で動かす例が紹介されている。ここにあるサンプルプログラムは ASP.NET 1.1 用に書かれたもので、実際に mono にて稼動することは、SLES10 にて確認した。SuSE Linux 上で、ソースコード (.cs ファイル)を修正して mcs コンパイラでコンパイルし直すことも可能である。

```

mcs /t:library /out:bin/WebApplication1.dll -r:System.Web -r:System.Xml -r
:System.Data -r:System.Drawing -r:System.DirectoryServices AssemblyInfo.cs
WebForm1.aspx.cs WebForm2.aspx.cs WebForm3.aspx.cs Global.asax.cs

```

Windows 上の ASP .NET 開発環境

Microsoft が無料提供をしている Web 開発環境は、

[Microsoft Visual Web Developer 2005 Express](#)

で、C# .NET のアプリケーションの開発環境の

[Microsoft Visual C# 2005 Express](#)

や、.NET Framework 開発キットの

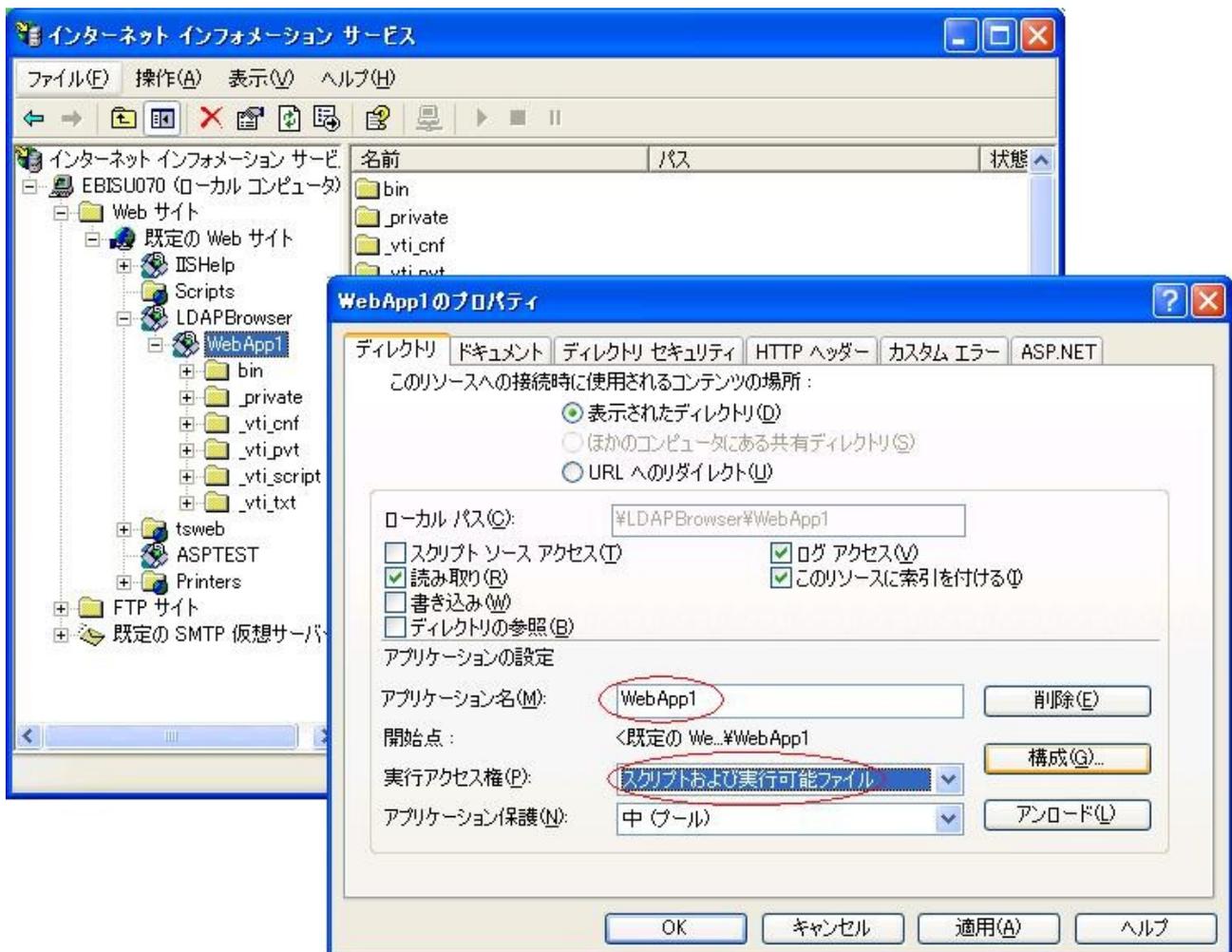
[Microsoft .NET Framework SDK](#)

が含まれているようである（当初、.NET 1.1 用の開発環境を構築しようとして、いくつか別々にインストール、アンインストールをしたため未確認）。

Visual Web Developer 2005 Express は、ASP.NET 2.0 用にデザインされていて、ASP.NET 1.1 用のアプリケーションは 2.0 用のアプリケーションに変換すれば動くようになる。



Webアプリケーションの開発でありながら、デバッガが利用可能なためとても便利である。IISにCool Solutions サイトからダウンロードしたWebApp1という例題を設定する際に、実行アクセス権に関して「スクリプトおよび実行可能ファイル」を選択する。



しかし、Visual Web Developer 2005 Express で ASP.NET 2.0 に変換してしまったプログラムは、SLES 10 の mono 1.0 環境では実行エラーを起こしてしまい、mono の C#コンパイラの mcs でコンパイルし直そうにもコンパイルエラーとなってしまった。(この問題は、mono 1.2 バージョンで解決している)。

Mono 最新版

最新版の mono-1.2.x は APS.NET 2.0 対応が強化されて、Visual Web Developer 2005 Express で作成したプログラムがほぼそのまま稼動する。Mono のソースあるいはバイナリパッケージが <http://www.mono-project.com/Downloads> からダウンロードできる。

たとえば、以下のブラウザ表示は、実際に Linux の mono で起動した ASP.NET サーバにアクセスして表示したものである。具体的は、Plamo Linux 4.2 に mono-1.2.3.1 のバイナリパッケージをインストールし、Visual Web Developer 2005 Express で ASP.NET 2.0 としてリビルドした WebApp1 というアプリケーションをプロジェクトごとコピーして、次のコマンドラインにて ASP.NET サーバを起動し、同一ホスト上でブラウザからアクセスして操作したものである。このアプリケーションの場合、テーブルのセルの背景色以外の変更はしていない。

```
$ xsp2 --root ~/mono/WebApp1
```

The screenshot shows a web browser window titled "WebForm2 - Deer Park". The address bar contains "http://localhost:8080/WebForm2.aspx". The page content is divided into two main sections:

Property Name	Property Value
cn	Mitya Kovalev
sn	Kovalev
seealso	documentTitle=book1,dc=example,dc=com
adspath	Idap://127.0.0.1/cn=Mitya Kovalev,dc=example,dc=com
givenname	Mitya
telephone number	222-3234

Below the table, there is a list of names, each with a "Select" link:

- [Select](#) Mitya Kovalev
- [Select](#) Torvlobnor Puzdoy
- [Select](#) Akakiy Zinberstein

The status bar at the bottom of the browser window displays "完了".