

チュートリアル: MapXtreme 2005 WMS サーバの設定

このチュートリアルでは、MapXtreme 2005 で作成された WMS クライアントまたは既に WMS クライアントの機能を備えているアプリケーションから使用可能な WMS サーバをセットアップする方法について説明します。WMS クライアントまたはサービス互換クライアントの要件の詳細については、Open Geospatial Consortium (OGC) の仕様を参照してください (<http://www.opengeospatial.org/docs/01-068r3.pdf>)。

このチュートリアルの内容

- ◆ はじめに..... 2
- ◆ 要件..... 2
- ◆ IIS における WMS サービスの設定..... 3
- ◆ WMS サービスのカスタマイズ..... 14
- ◆ 親レイヤと子レイヤの作成..... 17
- ◆ WMS セットアップのトラブルシューティング..... 22
- ◆ ネットワークドライブへのアクセス..... 24

はじめに

WMS (Web Map Service) を利用することで、ソフトウェア クライアントからインターネットまたは私設イントラネット上にあるマップ イメージを参照することができます。Mapinfo WMS サーバおよびクライアントのインプリメントは、WMS 1.1.1 OGC 仕様にに基づいています。WMS サーバは HTTP リクエストを使用してクライアントに GIS データを提供し、クライアントはそのデータをイメージとして表示します。各イメージが独立したレイヤとして機能します。イメージは GIF、JPG、PNG などのイメージ形式を使用して提供できます。レンダリングされるイメージは、階層構造を持つレイヤで構成できます。つまり、複数のレイヤを集めて作成したレイヤでイメージを構成できます。WMS ではレイヤがレンダリングされるため、レイヤの外観 (スタイルや座標系など) を変更する MapXtreme の特性もレイヤに含めることができ、イメージのカスタマイズのレベルを提供できます。

独自クライアントの開発の詳細については、OGC WMS 仕様 (<http://www.opengeospatial.org/docs/01-068r3.pdf>) を参照してください。

要件

最新のソフトウェア (特にサーバベースのソフトウェア) では一般的なことですが、このシステムが動作するには前もって特定のオペレーティング システム コンポーネントを準備しておく必要があります。MapXtreme 2005 を Web Map Service として動作させるには、次のコンポーネントをインストールする必要があります。

- MapXtreme 2005
- インターネット インフォメーション サービス (IIS)
- ASP.NET v1.1.4322

注意: Windows ディレクトリの名前は、使用している Windows オペレーティング システムのバージョンによって異なります。このドキュメントでは、Windows ディレクトリが C:\Windows であると想定しています。

インストール手順で説明した方法を使って、最新バージョンの MapXtreme 2005 をインストールした場合、ASP.NET の必要なコンポーネントは既にコンピュータにインストールされ、適切に設定されています。

IIS における WMS サービスの設定

MapXtreme 2005 に付属する構成ファイルを使用すると、ほとんど変更を加えずに WMS サーバを設定できます。

ここでは、このファイルを使用して WMS サーバを簡単に設定し、起動する方法を説明します。

Web Map Service のセットアップに必要な手順の大きな流れは次のようになります。

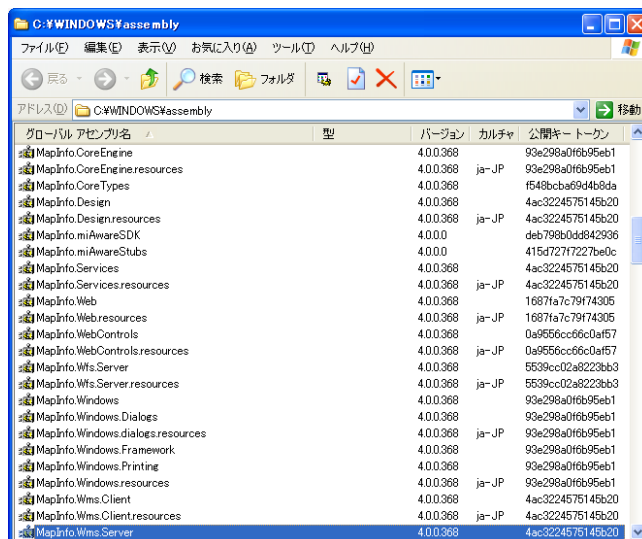
1. MapInfo.Wms.Server ソフトウェアのバージョン番号を確認します。
2. 構成ファイルを格納するフォルダを作成します。
3. サンプル構成ファイルを MapXtreme 2005 CD からコピーします。
4. 環境に合わせて構成ファイルを変更します。
5. 構成ファイルのフォルダを Web 共有ディレクトリとして使用可能にします。
6. IIS において、新しく作成した仮想ディレクトリのセキュリティプロパティを変更します。
7. 構成をテストします。
8. マップをリクエストします。

各手順については、以降の各セクションで詳しく説明します。

手順 1: ソフトウェアのバージョンを確認する

Web.config 構成ファイルに適切な情報を記述するには、MapInfo ソフトウェアのバージョン番号を確認しておく必要があります。バージョン番号を確認するには、以下の手順に従います。

1. Windows の [スタート] メニューの [ファイル名を指定して実行] を選択し、次のコマンドを入力します。 %windir%\assembly
Assembly ディレクトリの内容が表示されます。このウィンドウには、インストールされているすべての .NET グローバル アセンブリの一覧が表示されます。
2. 一覧から MapInfo.Wms.Server および MapInfo.CoreEngine アセンブリを探し、 [Version] 列と [Public Key Token] 列に表示されている番号を書き留めます。この後で説明する Web.config ファイルには、ここで確認したバージョン番号を正確に記述する必要があります。下記の場合、バージョン番号は 4.0.0.368 です。公開キー トークンは、MapInfo.Wms.Server が 4ac3224575145b20、MapInfo.CoreEngine が 93e298a0f6b95eb1 です。



手順 2: 構成ファイルを格納するフォルダを作成する

WMS 構成ファイルを格納するディレクトリをサーバ上に作成します。実際のディレクトリ名は何でもかまいません。このチュートリアルでは、`c:\¥WMS` というディレクトリ名を使用します。これは、サンプル構成ファイルで既に指定されているディレクトリです。

手順 3: サンプル構成ファイルを MapXtreme 2005 CD からコピーする

MapXtreme 2005 CD から、2 つのサンプル WMS 構成ファイルをコピーします。サンプル構成ファイルは WMS Config Files フォルダに格納されており、ファイル名は `Web.config` および `WMSSample.xml` です。これらのファイルは、MapXtreme 2005 に含まれているサンプルデータを使用してサーバを実行できるようにあらかじめ構成されています。

手順 4: 環境に合わせて構成ファイルを変更する

前の手順で説明したように、WMS サーバの設定に使用する構成ファイルは 2 つあります。`Web.config` は ASP プロセスの処理方法を定義するためのファイルなので、変更の必要がある箇所は比較的少数です。その機能の 1 つは、`WMSSample.xml` ファイルを参照することです。`WMSSample.xml` ファイルでは、使用されるデータ ソースとレイヤを定義します。

Web.config ファイル

任意のテキスト エディタで `Web.config` ファイルを開きます。

このファイルには、変更が必要な箇所が 3 つあります。その 1 つは、2 番目の構成ファイルを参照対象のファイルとして指定している箇所です。MapXtreme 2005 に付属するサンプル ファイルでは、次のように指定されています。

```
<configuration>
  <appSettings>
    <add key="configFile" value="C:\¥wms¥WMSSample.xml" />
```

ここではデフォルトのインプリメントを使用しているため、このパスが設定に合っているかどうかを確認するだけでかまいません。

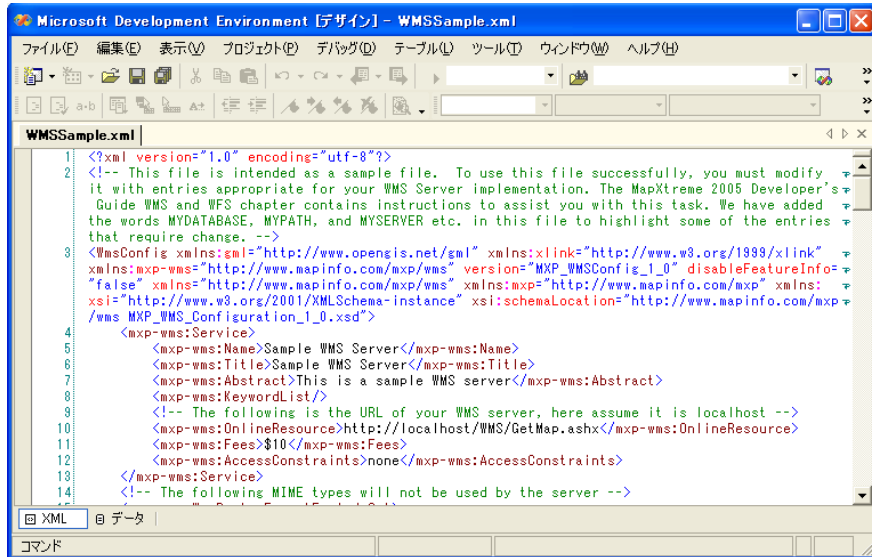
次に、[手順 1: ソフトウェアのバージョンを確認する](#)で確認した `MapInfo.Wms.Server` および `MapInfo.CoreEngine` アセンブリのソフトウェアのバージョン番号と公開キー トークンを指定する値 (下記の太字の箇所) を変更します。

```
<system.web>
  <httpHandlers>
    <add verb="GET,POST" path="*.ashx" type="MapInfo.Wms.WmsHttpHandler,
      MapInfo.Wms.Server, Version=4.0.0.352, Culture=neutral,
      PublicKeyToken=4ac3224575145b20" />
  </httpHandlers>
  <httpModules>
    <add type="MapInfo.Engine.WebSessionActivator, MapInfo.CoreEngine,
      Version=4.0.0.352, Culture=neutral, PublicKeyToken=93e298a0f6b95eb1"
      name="WebSessionActivator" />
  </httpModules>
```

このファイルに加えた変更を保存します。

WMSSample.xml

2 番目の構成ファイル (WMSSample.xml) をエディタで開きます。これは XML ファイルであり、Web.config ファイルよりはるかにサイズが大きいことに注意してください。このファイルの以下の各セクションの情報を変更します。



<mxp:wms:Service>

WMS サーバを指定する <mxp-wms:Name>、<mxp-wms:Title>、および <mxp-wms:Abstract> 要素の値を変更します。これらの要素で指定されている値がどのように使用されるかは、ユーザが使用するクライアントによって異なります。

次に、WMS 仮想ディレクトリの場所を変更します (後に詳細を説明します)。ここでは、先ほど作成したフォルダ名に合わせて、Web URL `http://YourMachineName/wms` を使用します。この URL は、後の段階で、Web 管理者の指示に基づいて適切な URL 定義に変更する必要があります。

注意: このドキュメントでは、すべての例と図でサーバに `MITestServer` という名前を使用します。MITestServer と書かれている部分は、使用中のサーバ名に置き換えてください。

以下のサンプルコードでは、変更が必要な箇所を太字で示しています。

```
<?xml version="1.0" encoding="utf-8"?>
<!-- This file is intended as a sample file. To use this file successfully, you
must modify it with entries appropriate for your WMS Server implementation. The
MapXtreme 2005 Developer's Guide WMS and WFS chapter contains instructions to
assist you with this task. We have added the words MYDATABASE, MYPATH, and
MYSERVER etc. in this file to highlight some of the entries that require change.
-->
<WmsConfig xmlns:gml="http://www.opengis.net/gml" xmlns:xlink="http://
www.w3.org/1999/xlink" xmlns:mxp-wms="http://www.mapinfo.com/mxp/wms"
version="MXP_WMSConfig_1_0" disableFeatureInfo="false" xmlns="http://
www.mapinfo.com/mxp/wms" xmlns:mxp="http://www.mapinfo.com/mxp"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://
/www.mapinfo.com/mxp/wms MXP_WMS_Configuration_1_0.xsd">
  <mxp-wms:Service>
    <mxp-wms:Name>Sample WMS Server</mxp-wms:Name>
    <mxp-wms:Title>Sample WMS Server</mxp-wms:Title>
    <mxp-wms:Abstract>This is a sample WMS server</mxp-wms:Abstract>
    <mxp-wms:KeywordList/>
    <!-- The following is the URL of your WMS server, here assume it is localhost -->
    <mxp-wms:OnlineResource>http://localhost/WMS/GetMap.ashx</mxp-wms:OnlineResource>
    <mxp-wms:Fees>$10</mxp-wms:Fees>
    <mxp-wms:AccessConstraints>none</mxp-wms:AccessConstraints>
  </mxp-wms:Service>
  <!-- The following MIME types will not be used by the server -->
```

```

<mxp-wms:KeywordList/>
<!-- The following is the URL of the WMS server -->
<mxp-wms:OnlineResource>http://MITestServer/WMS/GetMap.ashx
  </mxp-wms:OnlineResource>
<mxp-wms:Fees>none</mxp-wms:Fees>
<mxp-wms:AccessConstraints>none</mxp-wms:AccessConstraints>
</mxp-wms:Service>
<!-- The following MIME types will not be used by the server -->
<mxp-wms:WmsRasterFormatExcludeSet>
  <mxp-wms:Format>image/jpeg2000</mxp-wms:Format>
  <mxp-wms:Format>image/photoshop</mxp-wms:Format>
  <mxp-wms:Format>image/jp2</mxp-wms:Format>
</mxp-wms:WmsRasterFormatExcludeSet>

```

<DataSourceDefinitionSet>

次に、マッピング レイヤを登録します。データは WMSSample.xml ファイル内の 2 箇所で指定します。<DataSourceDefinitionSet> セクションと <WMSLayer> セクションです。WMSSample.xml ファイルの次の部分では、MapXtreme 2005 と共にインストールされるデータ ファイルを定義しています。

```

<DataSourceDefinitionSet xmlns="http://www.mapinfo.com/mxp/wms">
  <!-- The following data sources reference local TAB files, you need to replace
  MYPATH with the real path to those tab files. -->
  <TABFileDataSourceDefinition id="id1" readOnly="false" xmlns="http://
  www.mapinfo.com/mxp">
    <DataSourceName>STATES</DataSourceName>
    <FileName>MYPATH\USA.TAB</FileName>
  </TABFileDataSourceDefinition>
  <TABFileDataSourceDefinition id="id2" readOnly="false" xmlns="http://
  www.mapinfo.com/mxp">
    <DataSourceName>US_HIWAY</DataSourceName>
    <FileName>MYPATH\US_HIWAY.TAB</FileName>
  </TABFileDataSourceDefinition>
  <TABFileDataSourceDefinition id="id5" readOnly="false" xmlns="http://
  www.mapinfo.com/mxp">
    <DataSourceName>OCEAN</DataSourceName>
    <FileName>MYPATH\OCEAN.TAB</FileName>
  </TABFileDataSourceDefinition>

```

それぞれの <TABFileDataSourceDefinition> 要素について、id 属性と <DataSourceName> 要素の値を指定します。id 値には、アルファベットで始まる単純な文字列を指定する必要があります。上記の例では、id 値がそれぞれ id1、id2、id5 に指定されています。必要に応じて、これらの値を States、US_Highways、および Ocean に変更します。

各 <TABFileDataSourceDefinition> 要素内の <FileName> 要素で、実際のファイルのパスを指定します。たとえば、デフォルトのインストールでは、サンプルデータファイルのパスは c:\Program Files\MapInfo\MapXtreme\<version number>\Samples\Data になります。この値が正しいことを確認してください。

<wms:WmsLayerList>

最後に、WMS Layer List セクションで、クライアントアプリケーションで使用可能なレイヤを指定するために使用される実際のレイヤリストを定義します。レイヤをネストすることで、親レイヤを呼び出すだけですべての子レイヤを呼び出すことができます。このような構成の例については、14 ページの「WMSConfig.xml」を参照してください。

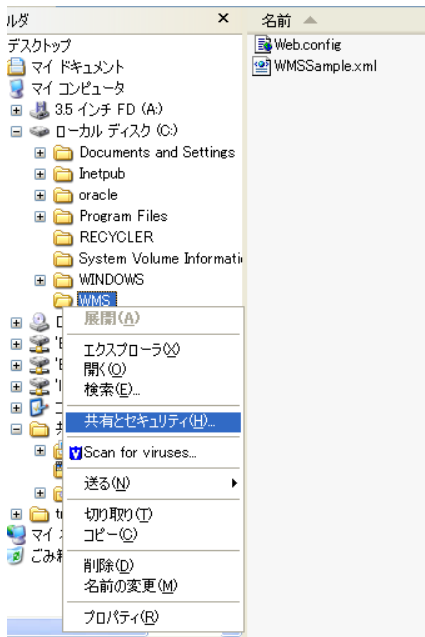
```
<mxp-wms:WmsLayer>
  <mxp-wms:Name>States</mxp-wms:Name>
  <mxp-wms:Title>States</mxp-wms:Title>
  <mxp-wms:SRSNameSet/>
  <mxp-wms:WmsStyleSet>
    ...style information removed for clarity
  </mxp-wms:WmsStyleSet>
  <FeatureLayer id="id8" name="STATES" alias="STATES" volatile="unknown"
xmlns="http://www.mapinfo.com/mxp">
    <DataSourceRef ref="States"/>
  </FeatureLayer>
</mxp-wms:WmsLayer>
<mxp-wms:WmsLayer>
  <mxp-wms:Name>USHighways</mxp-wms:Name>
  <mxp-wms:Title>US Highways</mxp-wms:Title>
  <mxp-wms:Abstract>The major highways for the United States</mxp-
wms:Abstract>
  <mxp-wms:SRSNameSet/>
  <mxp-wms:WmsStyleSet/>
  <FeatureLayer id="id9" name="US_HIWAY" alias="US_HIWAY" volatile="unknown"
xmlns="http://www.mapinfo.com/mxp">
    <DataSourceRef ref="US_Highway"/>
  </FeatureLayer>
</mxp-wms:WmsLayer>
  <mxp-wms:WmsLayer>
<mxp-wms:Name>Oceans</mxp-wms:Name>
<mxp-wms:Title>Oceans</mxp-wms:Title>
<mxp-wms:Abstract>The oceans of the world</mxp-wms:Abstract>
<mxp-wms:SRSNameSet/>
<mxp-wms:WmsStyleSet/>
<mxp:FeatureLayer id="id12" name="OCEAN" alias="OCEAN" volatile="unknown">
  <mxp:DataSourceRef ref="Ocean"/>
</mxp:FeatureLayer>
</mxp-wms:WmsLayer>
```

<DataSourceRef> 要素の値 (太字で示しています) が、<DataSourceDefinitionSet> で指定されている <TABFileDataSourceDefinition> の id 属性の値と対応していることに注意してください。これらの値が一致するようにしてください。

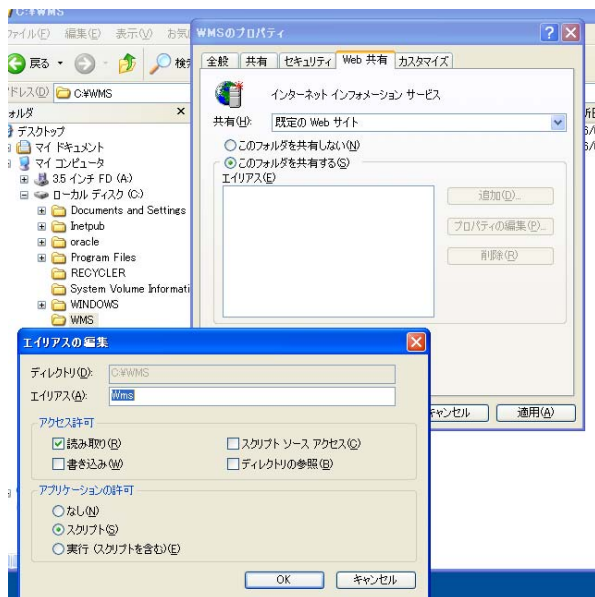
以上すべての変更を 2 つの構成ファイルに加えたら、ファイルを保存して閉じます。

手順 5: 構成ファイルのフォルダを Web 共有ディレクトリとして使用可能にする

次に、構成ファイルが格納されているフォルダを Web から使用できるようにする必要があります。この設定を行うには、フォルダ名を右クリックして、[共有とセキュリティ]を選択します。



[Web 共有] タブで、[このフォルダを共有する]を選択します。[エイリアスの編集]ダイアログボックスが表示されます。ここで必要に応じてエイリアス名を変更し、[OK]をクリックしてダイアログボックスを閉じます。[プロパティ]ダイアログボックスの[Web 共有]タブが更新され、共有するエイリアス名が表示されます。[OK]をクリックして、[プロパティ]ダイアログボックスを閉じます。



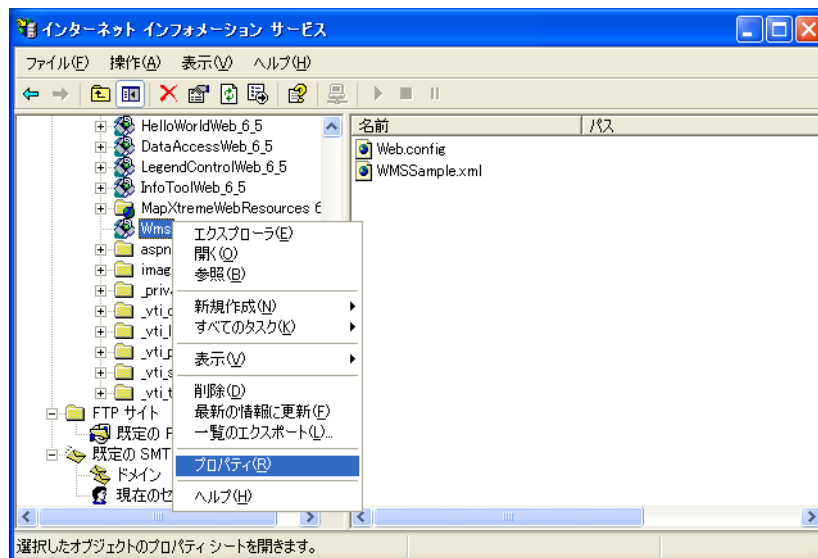
手順 6: IIS のセキュリティ プロパティを変更する

設定の最後の手順は、新しい WMS 仮想ディレクトリのセキュリティ プロパティを設定することです。この作業は IIS の管理画面で行います。WMS 仮想ディレクトリのセキュリティ プロパティを変更します。フォルダを Anonymous (匿名) に設定すると、アクセス時にユーザ名とパスワードを指定する必要がなくなります。

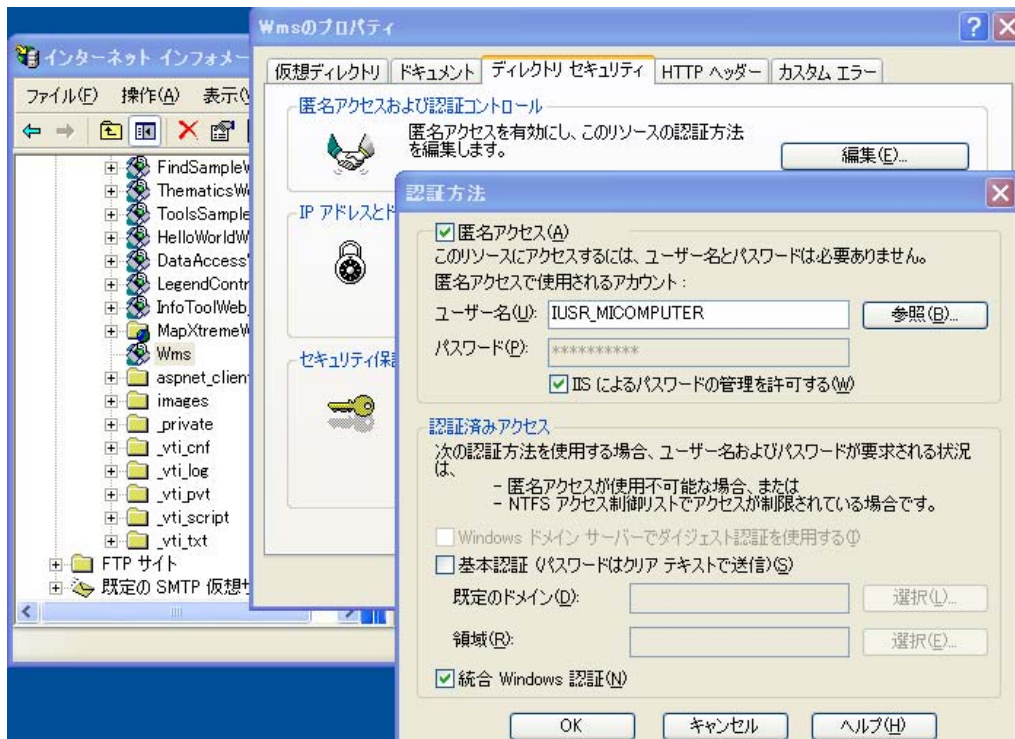
Windows の [スタート] メニューから、インターネット インフォメーション サービス (IIS) を起動します ([スタート] > [コントロール パネル] > [管理ツール] > [インターネット インフォメーション サービス])。



[既定の Web サイト] を展開すると、前の手順で作成した WMS 仮想ディレクトリが表示されます。WMS ディレクトリを右クリックし、[プロパティ] メニュー項目をクリックします。



次に、[ディレクトリ セキュリティ] タブを選択し、右上にある [編集] をクリックします。[認証方法] ダイアログボックスで、[匿名アクセス] チェックボックスをオンにします。これにより、WMS サービスのユーザはユーザ名とパスワードを指定する必要がなくなります。



手順 7: 構成をテストする

この時点で必要な設定と構成はすべて適切に完了しているので、Internet Explorer をクライアントとして使用して、簡単なクエリを実行してみましょう。Internet Explorer にはマップは表示されませんが、サーバの機能を XML で取得することができます。

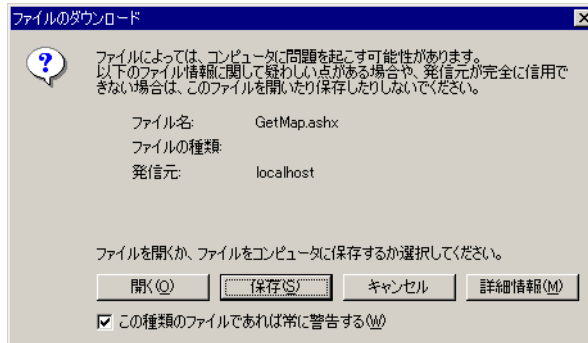
注意: WMS の機能を Internet Explorer で表示すると、必要なセクションを展開したり折りたたんだりすることができます。取得した XML をテキストエディタで表示することもできますが、決して見やすいとはいえません。

1. ブラウザを起動して、次の URL を入力します。

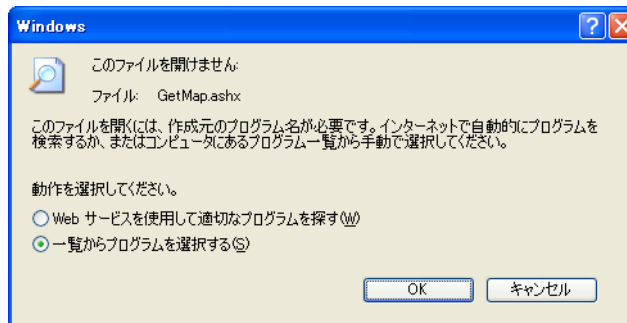
```
http://MITestServer/WMS/
GetMap.ashx?request=GetCapabilities&service=WMS&version=1.1.1
```

このリクエストは、サーバの機能 (使用可能なレイヤ、座標系、レイヤ スタイル、サポートされるイメージフォーマットなど) を問い合わせます。

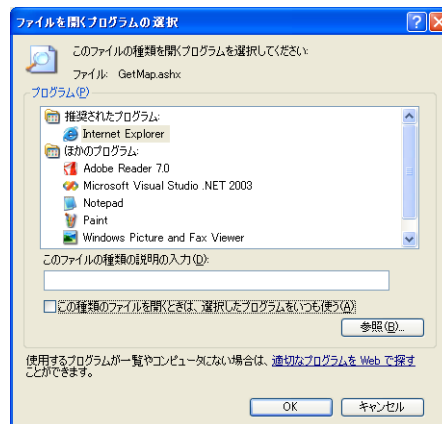
[ファイルのダウンロード] ダイアログ ボックスが表示されます。



2. [ファイルのダウンロード] ダイアログ ボックスで、[開く] をクリックします。
「このファイルを開けません:」 というメッセージを示すダイアログ ボックスが表示されます。



3. [一覧からプログラムを選択する] を選択して、[OK] をクリックします。
[ファイルを開くプログラムの選択] ダイアログ ボックスが開き、ファイルを開くために使用できるアプリケーションの一覧が表示されます。



4. リストの最初に表示されている Internet Explorer を選択して、[OK] をクリックします。
次の図に示すように、新しい Internet Explorer ウィンドウが開いて XML ファイルが表示されます。

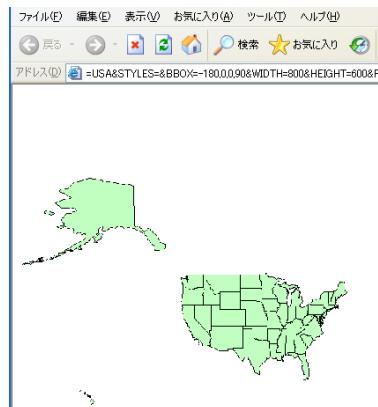
これにより、WMS サーバが適切に構成され、正常に動作していることがわかります。適切な結果が得られなかった場合は、これまでの手順を再確認して、もう一度テストしてください。その後で 22 ページの「WMS セットアップのトラブルシューティング」を参照して、詳細なチェック内容を確認してください。

手順 8: マップをリクエストする

サーバがリクエストに応答していることを確認できたので、次の URL を使用して、WMS サーバにマップイメージをリクエストしてみます。

Web ブラウザで次の URL を入力し、マップイメージをリクエストします。

```
http://MITestServer/WMS/
GetMap.ashx?service=WMS&VERSION=1.1.1&SRS=epsg:4267&REQUEST=GetMap&LAYERS=USA&STYLES=&BBOX=-180,0,90&WIDTH=800&HEIGHT=600&FORMAT=image/gif
```

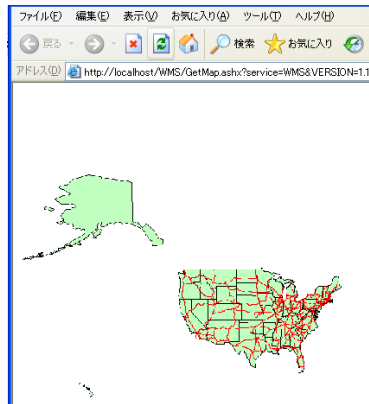


上記の URL の `LAYERS=USA` は、USA レイヤリストの下にあるレイヤのみを取得することを指定しています。この例では、States レイヤと US Highways が取得されます。しかし、Web ブラウザには US Highways が表示されていないことに注意してください。これは WMSample.xml ファイル内のレイヤの順序によるものです。US Highways は States レイヤの下に配置されています。サーバはレイヤを最初から描画するため、最上位のレイヤが境界レイヤの場合、その下のポイントレイヤと折れ線レイヤは見えなくなります。

US Highways と States レイヤの順序を変更するには、WMSample.xml ファイルで以下のブロックを States レイヤブロックの上に移動します。ファイルに加えた変更を保存したら、ファイルを閉じます。

```
<mxp-wms:WmsLayer>
  <mxp-wms:Name>USHighways</mxp-wms:Name>
  <mxp-wms:Title>US Highways</mxp-wms:Title>
  <mxp-wms:Abstract>The major highways for the United States</mxp-wms:Abstract>
  <mxp-wms:SRSNameSet/>
  <mxp-wms:WmsStyleSet/>
  <FeatureLayer id="id9" name="US_HIWAY" alias="US_HIWAY" volatile="unknown"
xmlns="http://www.mapinfo.com/mxp">
    <DataSourceRef ref="id2"/>
  </FeatureLayer>
</mxp-wms:WmsLayer>
```

Web ブラウザで [更新] ボタンをクリックし、URL を更新します。



GetCapabilities、GetMap、および GetFeatureInfo に関連するプロパティの詳細については、『MapXtreme 2005 開発者ガイド』の第 18 章、「WMS および WFS」を参照してください。

WMS サービスのカスタマイズ

これで、基本的なサービスを起動できました。次は、データと設定に合わせてカスタマイズします。ここでは、MapXtreme 2005 CD に収録されているサンプル構成ファイルを WMS ディレクトリにコピーして、新しいレイヤを追加します。

手順は前のセクションと同じですが、新しいサーバに合わせた値を使用します。必要な手順は、次のとおりです。

1. 構成ファイルを格納するディレクトリを作成します。
2. MapXtreme 2005 CD のサンプル ファイルを **手順 1** で作成した新しいディレクトリにコピーします。
3. 新しい環境に合わせてファイルを変更します。
4. 構成ファイルのフォルダを Web 共有ディレクトリとして使用可能にします。
5. IIS において、新しく作成した仮想ディレクトリのセキュリティプロパティを変更します。
6. 構成をテストします。
7. マップをリクエストします。

手順 1 および 2: サンプル ファイルをコピーする

C:\¥WMSDemoServer という新しいディレクトリを作成し、製品 CD からサンプル WMS ファイルをコピーします。前の例と区別するために、ファイル名を WMSsample.xml から WMSConfig.xml に変更します。

手順 3: 新しい環境に合わせてファイルを変更する

ここでは、WMS サーバに 2 つのレイヤを追加します。州都図 (state capitals) と郡境界図 (county boundaries) です。最初に、WMS サーバの新しいパスに合わせて Web.config ファイルを編集します。ソフトウェアのバージョン番号が現在のビルド環境と一致していることを確認します。

Web.config

```
<configuration>
  <appSettings>
    <add key="configFile" value="C:\¥WMSDemoServer¥WMSConfig.xml" />
    ...
  <system.web>
    <httpHandlers>
      <add verb="GET,POST" path="*.ashx" type="MapInfo.Wms.WmsHttpHandler,
        MapInfo.Wms.Server, Version=4.0.0.350, Culture=neutral,
        PublicKeyToken=4ac3224575145b20"/>
    </httpHandlers>
```

WMSConfig.xml

このファイルでは、新しい構成に合わせて <mxp-wms:Service> の各要素を編集します。<mxp-wms:Service> ブロックで、新しいサイトの名前、タイトル、要約、および OnlineResource を指定します。次の例では、変更する必要がある項目を太字で示しています。

```

<mxp-wms:Service>
  <mxp-wms:Name>USA WMS Server</mxp-wms:Name>
  <mxp-wms:Title>US Data WMS Server</mxp-wms:Title>
  <mxp-wms:Abstract>This is WMS server of US data
    </mxp-wms:Abstract>
  <mxp-wms:KeywordList/>
  <!-- The following is the URL of the WMS server -->
  <mxp-wms:OnlineResource>http://MITestServer/WMSDemoServer/GetMap.ashx
    </mxp-wms:OnlineResource>
  <mxp-wms:Fees>none</mxp-wms:Fees>
  <mxp-wms:AccessConstraints>none</mxp-wms:AccessConstraints>
</mxp-wms:Service>

```

注意: mxp: という名前空間 (接頭辞) を持つ要素は、MapInfo の WMS のインプリメントに固有の要素であり、OGC 仕様では定義されていません。

注意: システムを運用する際には、http://MITestServer/WMSDemoServer/ の部分を Web 管理者から指定された URL に変更してください。

次に、使用するレイヤが表示されるようにファイルをカスタマイズする必要があります。この例では、アメリカ合衆国データの 2 つのレイヤを追加します。州都と郡です。これらのファイルは、前のセクションで使用したファイルと同じフォルダに格納されています

(C:\Program Files\MapInfo\MapXtreme\6.5\Samples\Data)。

レイヤを追加するには、レイヤごとに新しい <TABFileDataSourceDefinition> 要素を定義する必要があります。これらは既存のデータに似ているので、<TABFileDataSourceDefinition> をコピーして内容を変更するのが最も簡単な方法です。それぞれのレイヤに新しい id 番号、データソース名、およびファイルのパスを指定する必要があります。

新しい要素を定義した WMSConfig.xml ファイルは次のようになります。

```

<TABFileDataSourceDefinition id="id1" readOnly="false" xmlns="http://
www.mapinfo.com/mxp">
  <DataSourceName>STATES</DataSourceName>
  <FileName>C:\Program Files\MapInfo\MapXtreme\6.5\Samples\Data\USA.TAB</
FileName>
</TABFileDataSourceDefinition>
<TABFileDataSourceDefinition id="id2" readOnly="false" xmlns="http://
www.mapinfo.com/mxp">
  <DataSourceName>US_HIWAY</DataSourceName>
  <FileName>C:\Program Files\MapInfo\MapXtreme\6.5\Samples\Data\US_HIWAY.TAB</
FileName>
</TABFileDataSourceDefinition>
<TABFileDataSourceDefinition id="id3" readOnly="false" xmlns="http://
www.mapinfo.com/mxp">
  <DataSourceName>US Capitals</DataSourceName>
  <FileName>C:\Program Files\MapInfo\MapXtreme\6.5\Samples\Data\usa_caps.TAB</
FileName>
</TABFileDataSourceDefinition>
<TABFileDataSourceDefinition id="id4" readOnly="false" xmlns="http://
www.mapinfo.com/mxp">
  <DataSourceName>US County Boundaries</DataSourceName>
  <FileName>C:\Program Files\MapInfo\MapXtreme\6.5\Samples\Data\us_cnty.TAB</
FileName>
</TABFileDataSourceDefinition>
<TABFileDataSourceDefinition id="id5" readOnly="false" xmlns="http://
www.mapinfo.com/mxp">
  <DataSourceName>OCEAN</DataSourceName>

```

```
<FileName>C:\Program Files\MapInfo\MapXtreme\6.5\Samples\Data\OCEAN.TAB</
FileName>
</TABFileDataSourceDefinition>
```

次に、XML の <WMSLayer> セクションを変更する必要があります。州都図のレイヤは既に WMSLayer セクションで指定されていますが、コメントアウトされています。US Capitals ブロックの先頭 (<!--) と末尾 (-->) からコメントを削除します。WMSSConfig.xml ファイルを保存してブラウザの表示を更新すると、新しいレイヤが表示されます。

```
<!-- Here depends on a valid id3 DataSourceDefinition, so it is commented out.
-->
<!--<mxp-wms:WmsLayer>
<mxp-wms:Name>USCapitals</mxp-wms:Name>
<mxp-wms:Title>US Capitals</mxp-wms:Title>
<mxp-wms:Abstract>The capitals for the United States.</mxp-wms:Abstract>
<mxp-wms:SRSNameSet/>
<mxp-wms:WmsStyleSet/>
<FeatureLayer id="id10" name="STATECAP" alias="STATECAP" volatile="unknown"
xmlns="http://www.mapinfo.com/mxp">
  <DataSourceRef ref="id3"/>
</FeatureLayer>
</mxp-wms:WmsLayer>-->
```

もう1つのレイヤについては、新しいセクションを追加する必要があります。先ほどと同様に、このセクションの既存のブロックをコピーして編集するのが最も簡単な方法です。US Capitals レイヤと States レイヤの間にブロックを挿入します。名前、タイトル、要約、フィーチャレイヤの要素と DataSourceRef id を指定します。編集後のブロックは次のようになります。

```
<<mxp-wms:WmsLayer>
<mxp-wms:Name>US Counties</mxp-wms:Name>
<mxp-wms:Title>US County Boundaries</mxp-wms:Title>
<mxp-wms:Abstract>County boundaries for the United States.</mxp-wms:Abstract>
<mxp-wms:SRSNameSet/>
<mxp-wms:WmsStyleSet/>
<FeatureLayer id="id11" name="County" alias="County" volatile="unknown"
xmlns="http://www.mapinfo.com/mxp">
  <DataSourceRef ref="id4"/>
</FeatureLayer>
</mxp-wms:WmsLayer>
```

手順 4: 構成ファイルのフォルダを Web 共有ディレクトリとして使用可能にする

この手順では、8 ページの「[手順 5: 構成ファイルのフォルダを Web 共有ディレクトリとして使用可能にする](#)」の説明に従って、まったく同じ操作を実行してください。

手順 5: IIS で、新しい仮想ディレクトリのセキュリティ プロパティを変更する

この手順では、9 ページの「[手順 6: IIS のセキュリティ プロパティを変更する](#)」の説明に従って、まったく同じ操作を実行してください。

手順 6: 構成をテストする

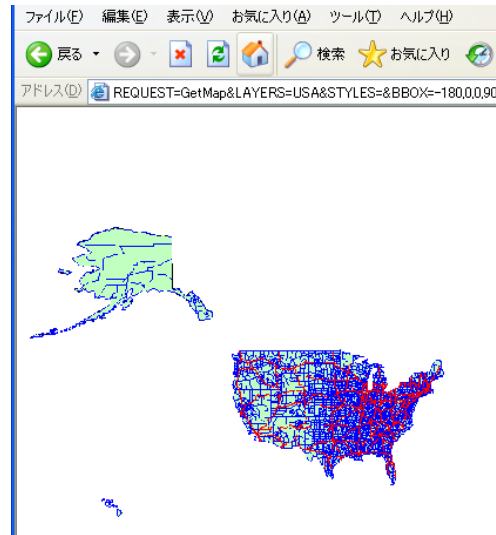
セットアップした構成が正常に機能することを確認するために、Internet Explorer を使用して、簡単なリクエストを送信してみましょう。ブラウザに次の URL を入力し、10 ページの「[手順 7: 構成をテストする](#)」で説明した手順に従ってテストを実行してください。

http://MITestServer/WMSDemoServer/
GetMap.ashx?request=GetCapabilities&service=WMS&version=1.1.1

参照先のセクションの画像と同じような結果が表示されるでしょう。

手順 7: マップをリクエストする

新しく追加した州都図と郡境界図のレイヤが含まれるマップイメージをリクエストするには、12 ページの「手順 8: マップをリクエストする」で説明した手順に従って、新しいレイヤが含まれるマップを表示します。



親レイヤと子レイヤの作成

上記の手順では、WMS サーバをカスタマイズし、サーバから追加のレイヤを提供する方法について説明しました。説明を簡単にするために、これらの新しいレイヤは、同じ親レイヤ (USA) の下に追加されていました。しかし実際には、類似のレイヤをグループ化する階層にデータを分類することが望ましい場合があります。それにより、親レイヤをリクエストするだけで、レイヤのグループを呼び出せるようになります。

ここでは、WMSConfig.xml ファイルでネストされた親レイヤと子レイヤを定義する方法について説明します。

親レイヤ

親レイヤへのリクエストが正しく処理されるようにするには、<SRSNameSet> 要素を定義し、その座標系を示す必要があります。この座標系は、親レイヤと、その下にあるすべての子レイヤに使用されます。

```

...
<mxp-wms:WmsLayerList>
  <mxp-wms:WmsLayer queryable="false">
    <mxp-wms:Name>USA</mxp-wms:Name>
    <mxp-wms:Title>United States</mxp-wms:Title>
    <mxp-wms:Abstract>Contains layers for the United States</mxp-wms:Abstract>
    <!-- The following SRS is available to this layer and all of the layers
underneath it -->
    <mxp-wms:SRSNameSet>
      <mxp:SRSName>EPSG:4267</mxp:SRSName>
    </mxp-wms:SRSNameSet>
    <mxp-wms:WmsStyleSet/>
  </mxp-wms:WmsLayer>
</mxp-wms:WmsLayerList>
...

```

これまで編集してきた WMSConfig.xml ファイルでは、USA レイヤは複数の州レベルのレイヤの親であると同時に、World グループ レイヤの子でもあります。上記の例では、1 行目の `<mxp-wms:WmsLayerList>` で USA が子レイヤであることがわかります。

さらに WMSConfig.xml ファイルを調べると、`<mxp-wms:WmsLayerList>` がもう 1 つあることがわかります。ここから USA グループ レイヤの子レイヤが開始されます。

```

...
<mxp-wms:WmsLayerList>
  <mxp-wms:WmsLayer>
    <mxp-wms:Name>USHighways</mxp-wms:Name>
    <mxp-wms:Title>US Highways</mxp-wms:Title>
    <mxp-wms:Abstract>The major highways for the United States</mxp-
wms:Abstract>
    <mxp-wms:SRSNameSet/>
    <mxp-wms:WmsStyleSet/>
    <FeatureLayer id="id9" name="US_HIWAY" alias="US_HIWAY" volatile="unknown"
xmlns="http://www.mapinfo.com/mxp">
      <DataSourceRef ref="id2"/>
    </FeatureLayer>
  </mxp-wms:WmsLayer>
  <!-- Here depends on a valid id3 DataSourceDefinition, so it is commented out. -
->
  <<mxp-wms:WmsLayer>
    <mxp-wms:Name>USCapitals</mxp-wms:Name>
    <mxp-wms:Title>US Capitals</mxp-wms:Title>
    <mxp-wms:Abstract>The capitals for the United States.</mxp-wms:Abstract>
    <mxp-wms:SRSNameSet/>
    <mxp-wms:WmsStyleSet/>
    <FeatureLayer id="id10" name="STATECAP" alias="STATECAP"
volatile="unknown" xmlns="http://www.mapinfo.com/mxp">
      <DataSourceRef ref="id3"/>
    </FeatureLayer>
  </mxp-wms:WmsLayer>
</mxp-wms:WmsLayerList>
...

```

すべてのレイヤは `<mxp-wms:WmsLayer></mxp-wms:WmsLayer>` 要素タグで囲まれます。レイヤのコレクションは、さらに `<mxp-wms:WmsLayerList></mxp-wms:WmsLayerList>` 要素タグを使ってグループ化されます。

まとめると、WMSConfig.xml ファイルの親レイヤと子レイヤは次のようになります。

```

...
<mxp-wms:WmsLayer queryable="false">
  <mxp-wms:Name>World</mxp-wms:Name>
  <mxp-wms:Title>The World</mxp-wms:Title>
  <mxp-wms:Abstract>This is a root layer that represents the whole world.</mxp-
wms:Abstract>
  <!-- The following SRS is available to all layers -->
  <mxp-wms:SRSNameSet>
    <mxp:SRSName>EPSG:4326</mxp:SRSName>
  </mxp-wms:SRSNameSet>
  <mxp-wms:WmsStyleSet/>
  <mxp-wms:WmsLayerList>
    <mxp-wms:WmsLayer queryable="false">
      <mxp-wms:Name>USA</mxp-wms:Name>
      <mxp-wms:Title>United States</mxp-wms:Title>
      <mxp-wms:Abstract>Contains layers for the United States</mxp-
wms:Abstract>
      <!-- The following SRS is available to this layer and all of the layers
underneath it -->
      <mxp-wms:SRSNameSet>
        <mxp:SRSName>EPSG:4267</mxp:SRSName>
      </mxp-wms:SRSNameSet>
      <mxp-wms:WmsStyleSet/>
      <mxp-wms:WmsLayerList>
        <mxp-wms:WmsLayer>
          <mxp-wms:Name>USHighways</mxp-wms:Name>
          <mxp-wms:Title>US Highways</mxp-wms:Title>
          <mxp-wms:Abstract>The major highways for the United States</mxp-
wms:Abstract>
          <mxp-wms:SRSNameSet/>
          <mxp-wms:WmsStyleSet/>
          <FeatureLayer id="id9" name="US_HIWAY" alias="US_HIWAY"
volatile="unknown" xmlns="http://www.mapinfo.com/mxp">
            <DataSourceRef ref="id2"/>
          </FeatureLayer>
        </mxp-wms:WmsLayer>
        <!-- Here depends on a valid id3 DataSourceDefinition, so it is
commented out. -->
        <<mxp-wms:WmsLayer>
          <mxp-wms:Name>USCapitals</mxp-wms:Name>
          <mxp-wms:Title>US Capitals</mxp-wms:Title>
          <mxp-wms:Abstract>The capitals for the United States.</mxp-
wms:Abstract>
          <mxp-wms:SRSNameSet/>
          <mxp-wms:WmsStyleSet/>
          <FeatureLayer id="id10" name="STATECAP" alias="STATECAP"
volatile="unknown" xmlns="http://www.mapinfo.com/mxp">
            <DataSourceRef ref="id3"/>
          </FeatureLayer>
        </mxp-wms:WmsLayer>
        <<mxp-wms:WmsLayer>
          <mxp-wms:Name>US Counties</mxp-wms:Name>
          <mxp-wms:Title>US County Boundaries</mxp-wms:Title>
          <mxp-wms:Abstract>County boundaries for the United States.</mxp-
wms:Abstract>
          <mxp-wms:SRSNameSet/>
          <mxp-wms:WmsStyleSet/>
          <FeatureLayer id="id11" name="County" alias="County"
volatile="unknown" xmlns="http://www.mapinfo.com/mxp">

```

```

        <DataSourceRef ref="id4"/>
    </FeatureLayer>
</mxp-wms:WmsLayer>
<mxp-wms:WmsLayer>
    <mxp-wms:Name>States</mxp-wms:Name>
    <mxp-wms:Title>States</mxp-wms:Title>
    <mxp-wms:SRSNameSet/>
    <mxp-wms:WmsStyleSet>
... style information removed for clarity.
    </mxp-wms:WmsStyleSet>
    <FeatureLayer id="id8" name="STATES" alias="STATES"
volatile="unknown" xmlns="http://www.mapinfo.com/mxp">
        <DataSourceRef ref="id1"/>
    </FeatureLayer>
</mxp-wms:WmsLayer>
</mxp-wms:WmsLayerList>
</mxp-wms:WmsLayer>
<mxp-wms:WmsLayer>
    <mxp-wms:Name>Oceans</mxp-wms:Name>
    <mxp-wms:Title>Oceans</mxp-wms:Title>
    <mxp-wms:Abstract>The oceans of the world</mxp-wms:Abstract>
    <mxp-wms:SRSNameSet/>
    <mxp-wms:WmsStyleSet/>
    <mxp:FeatureLayer id="id12" name="OCEAN" alias="OCEAN"
volatile="unknown">
        <mxp:DataSourceRef ref="id5"/>
    </mxp:FeatureLayer>
</mxp-wms:WmsLayer>
<!-- Here depends on a valid id6 DataSourceDefinition, so it is commented
out. -->
<!--<mxp-wms:WmsLayer>
    <mxp-wms:Name>Asia</mxp-wms:Name>
    <mxp-wms:Title>Asia</mxp-wms:Title>
    <mxp-wms:SRSNameSet/>
    <mxp-wms:WmsStyleSet/>
    <FeatureLayer id="id13" name="ASIA" alias="ASIA" volatile="unknown"
xmlns="http://www.mapinfo.com/mxp">
        <DataSourceRef ref="id6"/>
    </FeatureLayer>
</mxp-wms:WmsLayer> -->
</mxp-wms:WmsLayerList>
</mxp-wms:WmsLayer>
...

```

レイヤ要素

WMSConfig.xml ファイルを編集するとき、構成する各レイヤに対していくつかの値を指定します。この値がクライアントアプリケーションでどのように使用されるかについては、特定のガイドラインはありません。値の使用方法は、その情報を読み取るクライアントによって異なります。このような値の使用方法をインプリメントする方法の詳細については、OGC WMS 仕様 (<http://www.opengeospatial.org/docs/01-068r3.pdf>) を参照してください。

このような値の例としては、<Name> と <Title> の 2 つの要素があります。OGC WMS 仕様によれば、Name 値はマシン間の通信に使用され、Title 値は人との対話操作に使用されます。Name 値はクライアントがサーバにリクエストするものであり、Title 値はユーザに表示されるものです。

次に、構成ファイルの <wms:WMSLayer> 部分に含まれる各要素を示し、その使用方法を説明します。

注意: mxp: という名前空間 (接頭辞) を持つ要素は、MapInfo の WMS のインプリメントに固有の要素であり、OGC 仕様では定義されていません。

<Name>

この要素の値は、一般的にはマシン間の通信に使用される名前です。

例: <Name>NYStreets</Name>

<Title>

この要素の値を利用して、レイヤのフレンドリ名を指定します。

例: <Title>New York State Streets</Title>

<Abstract>

オブジェクトの注釈的な説明です。

例: <Abstract>Streets and roads in New York State</Abstract>

<SRSNameSet>

<mxp:SRSName> 要素をグループ化します。

<mxp:SRSName>

特定のレイヤで使用する座標系を指定します。この値が空の場合、そのレイヤは親レイヤの <mxp:SRSName> 値を継承します。1 つのレイヤで複数の <mxp:SRSName> 要素を使用できます。重複している場合は無視されます。

例: <mxp:SRSName>epsg:4326</mxp:SRSName>

<WmsStyleSet>

個々の <WmsStyle> 要素のコンテナの役割を果たします。

<WmsStyle>

各 <WmsStyle> には、ワークスペースで作成した特定のスタイルを表す <Name>、<Title>、および <Abstract> 要素が含まれます。さらに、スタイルの詳細を入力します。要素名の前に名前空間接頭辞 mxp: を追加するようにしてください。

例:

```
<WmsStyleSet>
  <WmsStyle>
    <Name>StreetStyle</Name>
    <Title>Street Style</Title>
    <Abstract>This is the line style for displaying streets.</Abstract>
    <mxp:LineStyle stroke="black" width="1"
      width-unit="mapinfo:image size pixel">
      <Pen>mapinfo:pen 2</Pen>
    </mxp:LineStyle>
  </WmsStyle>
</WmsStyleSet>
```

<mxp:FeatureLayer>

この要素により、レンダリングされるレイヤが指定されます。各 <mxp:FeatureLayer> 要素の id 属性に一意的な値を指定する必要があります。この要素には他に name と alias という属性もありますが、この 2 つの値は WMS には影響しません。

例: <mxp:Featurelayer id="NYStreets" name="Streets" alias="streets">

<mxp:DataSourceRef>

この空の要素に含まれる ref という属性は、構成ファイルの <mxp:DataSourceDefinitionSet> セクションで指定されているレイヤ (<mxp:TABFileDataSourceDefinition> 要素) と一致している必要があります。

例: <mxp:DataSourceRef ref="Streets"/>

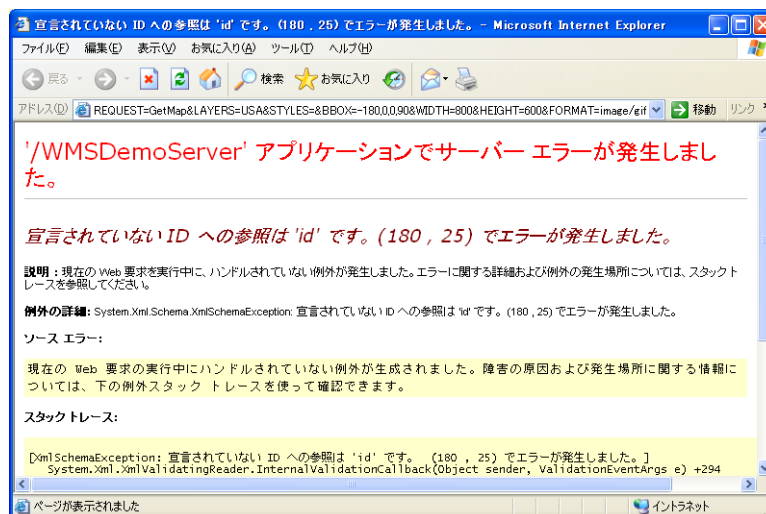
WMS セットアップのトラブルシューティング

セットアップの問題の解決

Web サービス方式のすべてのアプリケーションの場合と同様に、WMS でも構成ファイルが非常に重要です。わずかに 1 か所の小さな入力ミスが原因で、システム全体の機能が停止してしまう場合があります。次に、作成した WMS サーバへの接続に失敗した場合によく表示されるエラーメッセージを示します。



このエラーメッセージでは、WMS サーバでエラーが発生したことはわかりますが、問題が発生した具体的な場所や、そのエラーの対処方法はわかりません。エラーについてさらに詳しい情報を得るには、Internet Explorer を使ってサービスに直接接続します。この方法により、エラーの詳細を確認できます。



この例の場合、Internet Explorer を使用してサービスに直接接続すると、WMSSchema.xml ファイルの 180 行目で問題が発生したことがわかります。問題のある箇所が判明したら、XML ファイルのその箇所に移動してトラブルシューティングを開始することができます。

Web.config ファイルまたは WMSConfig.xml ファイルを修正した後は、エディタでそのファイルを閉じたことを確認する必要があります。どちらかのファイルが別のアプリケーションで使用されているときに、サーバがファイルにアクセスしようとすると、エラーが発生します。

注意: Windows にバグが存在するため、Web サービスに再度接続するたびに aspnet_wp.exe (Windows 2003 Server の場合は w3wp.exe) プロセスを終了する必要があることもあります。

エラーを解決する手順を次に示します。

1. 構成ファイル (web.config または wmsconfig.xml) を編集します。
2. ファイルを閉じます (必要に応じて aspnet_wp.exe または w3wp.exe プロセスを再起動します)。
3. IE または自分のサンプルクライアントを使用して、WMS サーバに再度接続します。

セットアップのトラブルシューティング

セットアップ後にサーバが正常に機能しないときは、以下の点を見直して、構成情報が適切に指定されていることを確認してください。

Web.config ファイル

1. ConfigFile の値は正しく指定されていますか？
サーバを正常に機能させるには、wmsconfig.xml ファイルへのパスを、Web.config ファイル内の <appSettings>|<add> 要素の value 属性の値として正確に指定する必要があります。
2. ソフトウェアのバージョン番号が正しく指定されていますか？
手順 1: ソフトウェアのバージョンを確認するで確認したバージョン番号が、Web.config ファイルの <httpHandlers>|<add> 要素の type(Version) 属性の値として正しく指定されているかどうか確認してください。
3. 公開キー トークンが正しく指定されていますか？
公開キー トークンの値が、Web.config ファイルの <httpHandlers>|<add> 要素の type(PublicKeyToken) 属性の値として正しく指定されているかどうか確認してください。この情報は、ソフトウェアのバージョン番号と同じ場所で確認できます ([スタート]> [ファイル名を指定して実行]> Assembly)。

これらの指定にわずかでも不正確な部分があると、システムは正常に動作できずに、エラーが発生します。上記の各設定を再確認して問題点を修正すれば、システムは常に適切に機能するようになります。

wmsconfig.xml ファイル

1. <mxp-wms:OnlineResource> 要素の値が適切に指定されていますか？
この値には、末尾が getmap.ashx の URL を正しく指定する必要があります。
2. <DataSourceDefinitionSet> 要素にデータ ソースが正しく指定されていますか？
データ ソースについては、各ソースを定義する要素を個別に記述して、それぞれのデータ ファイルを示す有効なパスを指定する必要があります。ファイル内で id 属性の値や <DataSourceName> 要素の値が重複していないかどうか確認してください。また、各データ ソースの <FileName> 要素の値には、有効なファイルを示す有効なパスを指定する必要があります。
3. <WmsLayerList> プロパティの要素が正しく指定されていますか？
 - a. 名前やタイトルが重複していませんか？
名前またはタイトルが重複している場合、エラーが発生します。
 - b. <mxp:FeatureLayer> 属性が重複していませんか？
この属性のいずれかの値が重複している場合、エラーが発生します。

- c. <DataSourceRef> 要素の ref 属性と、対応する <TABFileDataSourceDefinition> 要素の属性が正確に一致していますか？
- システムが適切に動作するためには、それぞれの ref 属性の値が、対応する <TABFileDataSourceDefinition> 要素の id 属性と正確に一致している必要があります。

IIS のセットアップ

1. サービスで使用するディレクトリは仮想ディレクトリとして設定されていますか？
 2. IIS サーバが実行されていますか？
 3. aspnet_wp.exe または w3wp.exe のプロセスを終了しましたか？
- これらのプロセスは強制終了された直後に再起動されるので、終了操作を実行した時点ではプロセスの終了を確認しにくいことがあります。実際にいったん終了した場合は、メモリ使用量が大幅に減少します。

ネットワークドライブへのアクセス

ASP.NET のログイン アカウントは、ネットワーク ファイルへのアクセス権限を持たないローカル マシンアカウントです。そのため、WMS サーバを実行する ASP.NET プロセスは、ネットワーク フォルダにアクセスする権限を持つユーザとして実行されるように設定する必要があります。また、ネットワーク フォルダを指定する場合は、マッピングされたドライブ名を使用せずに、UNC (Universal Naming Convention) 名を使用した完全なパスで指定する必要があります。たとえば、マッピングされたドライブ名を含む "Z:\data" は使用できないので、\\server\data のように指定してください。

ASP.NET プロセスのユーザを設定するには、いくつかの方法や選択肢が考えられます。各方法を使用した適切なソリューションの詳細については、以下の説明の中で示しているリンク先の情報を参照してください。

IIS 5.0/5.1 でのネットワーク ドライブへのアクセス

単純なパスワード

ASP.NET プロセスを実行するネットワーク ユーザを選択し、そのユーザがネットワーク フォルダにアクセスできるようにします。このとき、データに予想外のアクセスや悪意あるアクセスが行われるのを防止するために、このユーザの権限を制限してください。

たとえば、ユーザ名が "MYDOMAIN\WMSUser"、パスワードが "WMSUserPassword" であるとします。

1. Web.config ファイルのバックアップ コピーを作成します。
2. Web.config ファイルの <system.web> セクションに、次の行を追加します。

```
<identity password="WMSUserPassword" userName="MYDOMAIN\WMSUser" impersonate="true" />
```
3. WMS サーバを再起動します。

この方法の場合、指定したユーザのパスワードは何の暗号化もせずに保存されるため、他のユーザが web.config ファイルにアクセスしてこのパスワードを調べることができます。よりセキュリティの高い方法が必要な場合は、次の「暗号化パスワード」で説明している方法を使用してください。

暗号化パスワード

暗号化パスワードの作成手順は、次の MSDN 記事に記載されています。

<http://support.microsoft.com/default.aspx?scid=kb;en-us;329290>.

暗号化レジストリ パスワードの使用を有効にするには、次の記事に記載されているホットフィックスを使用する必要がある場合があります。

<http://support.microsoft.com/default.aspx?scid=kb;EN-US;329250>.

IIS 6.0 でのネットワーク ドライブへのアクセス

上記の IIS 5.0/5.1 用の方法は IIS 6.0 でも有効です。IIS 6.0 では、ASP.NET プロセスのユーザ識別情報および関連のセキュリティ設定をより簡便に設定できる方法も使用できます。たとえば、次の記事の「userName and password Equivalent Settings in IIS 6.0 Worker Process Isolation Mode」を参照してください。

<http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/Library/IIS/26d8cee3-ec31-4148-afab-b6e089a0300b.mspx>.