

教程：设置 MapXtreme 2005 WMS 服务器

本教程描述了如何设置 WMS 服务器，使其能够与自己创建的 MapXtreme 2005 WMS 客户端或已具有 WMS 客户端功能的应用程序一起使用。更多关于 WMS 要求或服务兼容客户端的详细信息，请参阅 Open Geospatial Consortium (OGC) 规范 (<http://www.opengeospatial.org/docs/01-068r3.pdf>)。

在本教程中 ...

- ◆ 简介 2
- ◆ 先决条件 2
- ◆ 设置 IIS 下的 WMS 服务 2
- ◆ 定制 WMS 服务 12
- ◆ 创建父图层和子图层 16
- ◆ WMS 设置的故障诊断 20
- ◆ 访问网络驱动器 22

简介

Web 地图服务 (WMS) 使软件客户端可以通过因特网或专用内联网参考地图图像。Mapinfo WMS 服务器和客户端实现基于 WMS 1.1.1 OGC 规范。WMS 服务器使用 HTTP 请求向客户端提供 GIS 数据, 该客户端将该数据以图像形式显示。每个图像都可以作为独立的图层。图像可以是 GIF、JPG、PNG 和其它多种图像格式。渲染的图像可由分层图层组成 - 可服务的图像, 由其它图层集合中的图层组成。由于 WMS 渲染图层, 因此在提供的图层中也可包括修改图层视图 (样式、坐标系等) 的 MapXtreme 特性, 这些特性允许提供图像的自定义级别。

有关开发自己的客户端的详细信息, 请参阅 OGC WMS 规范 (<http://www.opengeospatial.org/docs/01-068r3.pdf>)。

先决条件

由于这些系统包含最现代的软件, 特别是基于服务器的软件, 所以在正确运行前必须在适当位置设置所需的操作系统组件。要使 MapXtreme 2005 作为 Web 地图服务运行, 则需要安装以下内容:

- MapXtreme 2005
- Internet 信息服务 (IIS)
- ASP.NET v1.1.4322

注: Windows 目录的名称会根据所运行操作系统版本的不同而改变。本文档的其余部分我们使用 C:\Windows 表示该目录。

如果已按安装指示中所述方式安装了最新版本的 MapXtreme 2005, 则 ASP.NET 的必要组件应该已在您的计算机中安装并配置好。

设置 IIS 下的 WMS 服务

MapXtreme 2005 包含一组配置文件, 有了这些配置文件, 您只需通过几项修改就可以设置 WMS 服务器。本节描述了如何使用这些文件迅速设置好 WMS 服务器。

对设置 Web 地图服务所需步骤的高级描述具体如下:

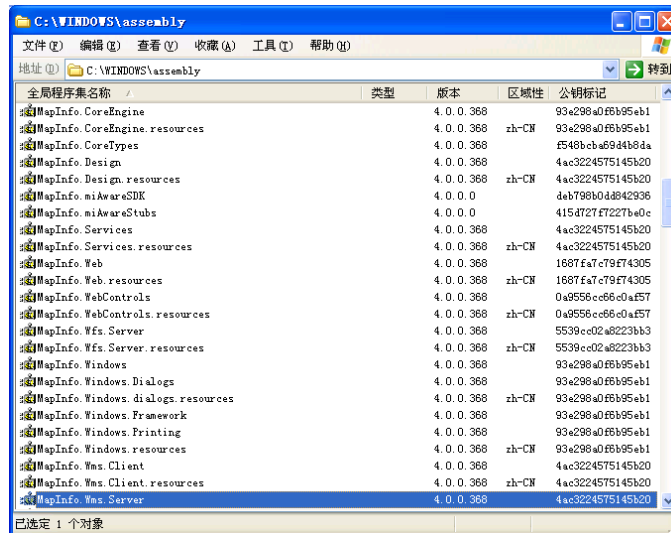
1. 验证 MapInfo.WMS.Server 软件版本号。
2. 创建文件夹, 用于存放配置文件。
3. 从 MapXtreme 2005 光盘复制示例配置文件。
4. 修改配置文件以适应运行环境。
5. 可将该文件夹用作 Web 共享目录。
6. 在 IIS 中, 更改新虚拟目录的安全属性。
7. 测试配置。
8. 请求地图。

以下部分详细描述了上述每个步骤。

步骤 1：验证软件的版本

要在 Web.config 配置文件中包括正确的信息，则需要验证 MapInfo 软件的版本号。要检查版本号，请执行以下步骤：

1. 从 Windows **开始** 菜单中选择 **运行 ...**，然后输入以下命令：`%windir%\assembly`
此时将显示 **ASSEMBLY** 目录。该窗口列出已安装的所有 .NET 全局程序集。
2. 在列表中找到 MapInfo.Wms.Server 和 MapInfo.CoreEngine 程序集并查看版本和 Public Key Token 列中的数字。必须在 Web.config 文件中正确指定这些数字，我们不久后会检查这一点。以下示例中的版本为 4.0.0.368。MapInfo.Wms.Server 和 MapInfo.CoreEngine 的 Public Key Token 分别为 4ac3224575145b20 和 93e298a0f6b95eb1。



步骤 2：创建用于保存配置文件的目录

在服务器上创建目录，用于存储 WMS 配置文件。实际名称并不重要。在本教程中，我们将使用 `c:\WMS`。该目录已在示例配置文件中指定。

步骤 3：从 MapXtreme 2005 光盘复制示例配置文件

从 MapXtreme 2005 光盘复制两个示例 WMS 配置文件。文件位于“WMS 配置文件”文件夹，名为 `Web.config` 和 `WMSample.xml`。这些文件已被配置为使用与 MapXtreme 2005 同时提供的示例数据工作。

步骤 4：修改配置文件以适应运行环境

如上所示，在设置 WMS 服务器的过程中使用了两个配置文件。文件 `Web.config` 需要的修改较少，因为其目的是定义 ASP 进程的处理方式。其中一个函数将参考 `WMSample.xml` 文件。`WMSample.xml` 文件定义您要使用的数据源和图层定义。

Web.config 文件

在任意文本编辑器中打开 `Web.config` 文件。

您需要在三个位置对该文件进行修改。首先是 `Web.config` 文件指向第二个配置文件的位置。在与 MapXtreme 2005 同时提供的示例文件中，该行显示为：

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

由于使用默认实现, 因此只需验证该路径是否与设置匹配。

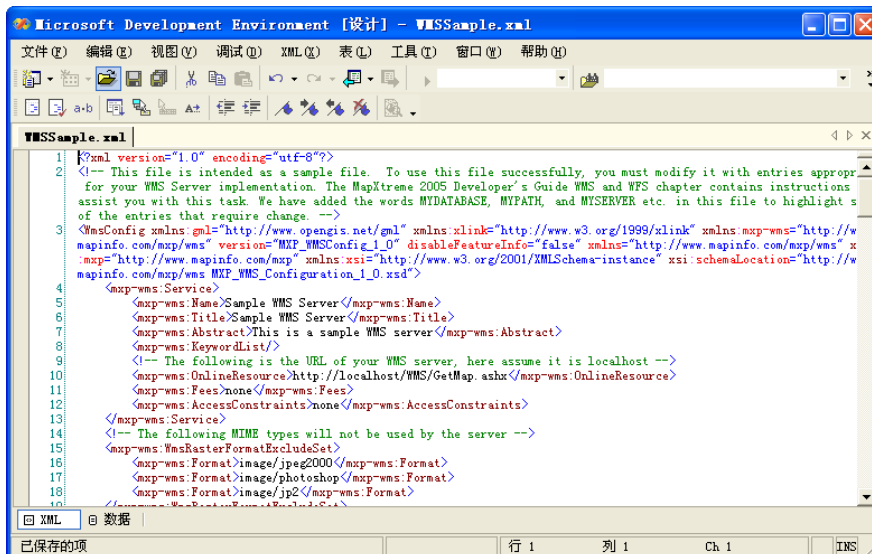
第二, 修改文件 (下文中显示为黑体), 这些文件可指定 MapInfo.Wms.Server 和 MapInfo.CoreEngine 程序集的软件版本和 Public Key Tokens (在步骤 1: 验证软件的版本中验证)。

```
<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>
```

保存对该文件的更改。

WMSsample.xml

在编辑器中打开第二个配置文件 (WMSsample.xml)。注意, 该文件为 XML 文件, 远远大于 web.config。您要修改该文件中多个部分的信息:



<mxp:wms:Service:>

修改 <mxp:wms:Name>、<mxp:wms:Title> 和 <mxp:wms:Abstract> 元素 (这些元素用于描述 WMS 服务器) 的值。如何使用在该部分指定的值完全由正使用的客户端规定。

其次, 更改 WMS 虚拟目录的位置 (具体如下)。现在, 我们将使用 web URL `http://YourMachineName/wms` 与先前创建的文件夹名称匹配。稍后, 应根据 Web 管理员的需要更改为正确的 URL 定义。

注： 本文的其余部分 MITestServer 将用作示例和插图的服务器。请确保在任何出现 MITestServer 的位置将其替换为您的服务器的名称。

在以下代码片断中要更改的行显示为黑体。

```
<?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.
-->
<mxp-wms: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_WMSSampleConfig_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_WMSSample_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 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>

接下来，注册映射图层。数据在 WMSSample.xml 中的两个位置指定：即，<DataSourceDefinitionSet> 和 <WMSLayer> 部分。WMSSample.xml 的以下部分定义了与 MapXtreme 2005 版本同时安装的数据文件。

```
<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>
```

对于每个 <TABFileDataSourceDefinition> 元素, 请指定 id 属性值和 <DataSourceName> 元素值。Id 值必须是以字母开头的简单字符串。在以上示例中, id 值为 id1、id2 和 id5。也可以分别将其更改为 States、US_Highways 和 Ocean。

每个 <TABFileDataSourceDefinition> 元素中的 <FileName> 元素都应包含到实际文件的路径。例如, 对于默认安装, 到示例数据文件的路径为

c:\Program Files\MapInfo\MapXtreme\<version number>\Samples\Data。请确保该值正确。

<wms:WmsLayerList>

最后, 在“WMS 图层列表”部分, 定义在客户端应用程序中使用的实际图层列表以显示可用图层。可将图层嵌套, 这样就可以通过请求父图层将所有子图层包括在响应中。有关此类配置的示例, 请参阅第 13 页中的 *WMSSConfig.xml*。

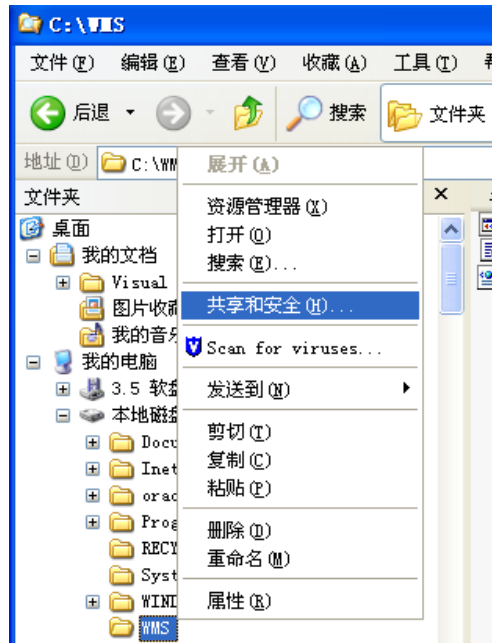
```
<mxp-wms:WmsLayer>
  <mxp-wms:Name>States</mxp-wms:Name>
  <mxp-wms:Title>States</mxp-wms:Title>
  <mxp-wms:SRSNameSet/>
  <mxp-wms:WmsStyleSet>
    ...style information removed for clarity
  </mxp-wms:WmsStyleSet>
  <FeatureLayer id="id8" name="STATES" alias="STATES" volatile="unknown"
xmlns="http://www.mapinfo.com/mxp">
    <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>
```

注意, <DataSourceRef> 元素 (以上黑体部分) 中的值是指在 <DataSourceDefinitionSet> 中提供的 <TABFileDataSourceDefinition> id 属性值。请确保这些值相匹配。

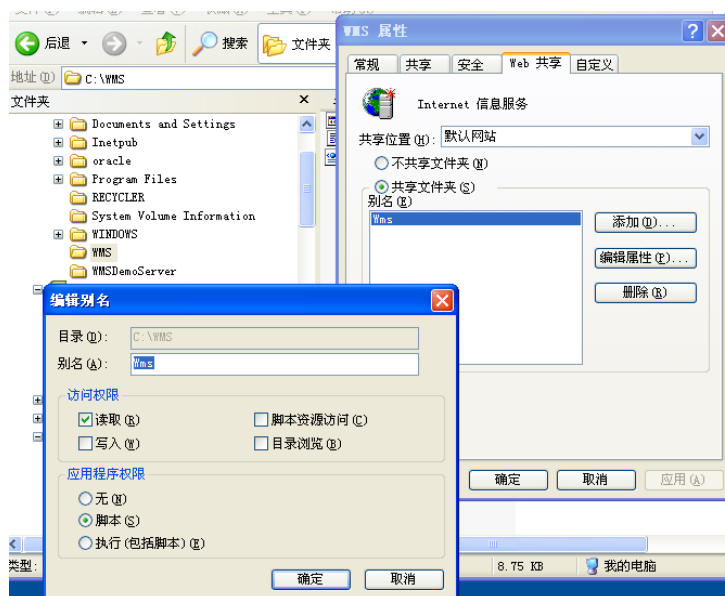
执行对两个配置文件的所有更改后, 请保存并关闭所有文件。

步骤 5：将该文件夹用作 Web 共享目录

接下来，必须将包含配置文件的文件夹设置为可通过 web 使用。要执行该操作，请右键单击文件夹名称，然后选择**共享和安全...**。



在 **WEB 共享** 选项卡上，选择**共享文件夹** 单选按钮。此时将显示**编辑别名** 对话框。可以在此处更改别名，然后单击**确定** 关闭该对话框。**属性** 对话框的 **WEB 共享** 选项卡会更新，显示要共享的别名。单击**确定**，关闭**属性** 对话框。



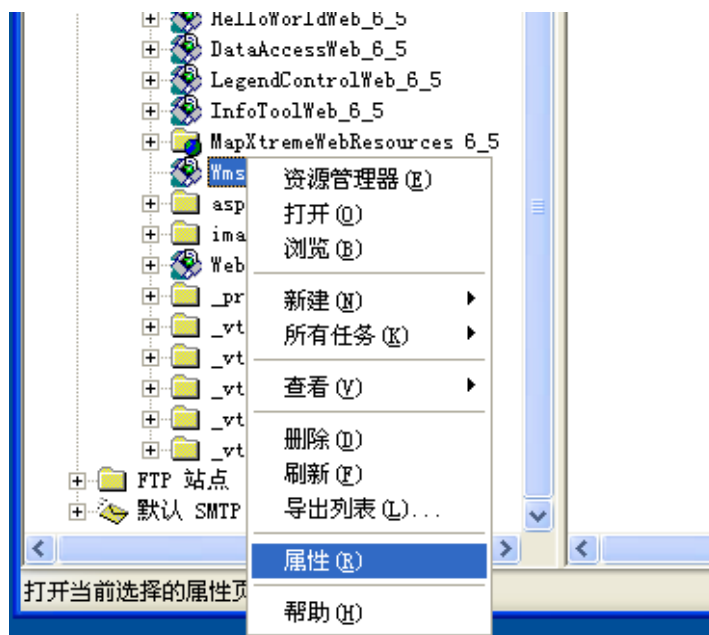
步骤 6: 更改 IIS 安全属性

最后一个安装步骤是设置新 WMS 虚拟目录的安全属性。该任务通过 IIS 管理完成; 可以更改 WMS 虚拟目录的安全属性。将文件夹设置为匿名意味着您不必再为用户名和密码担心。

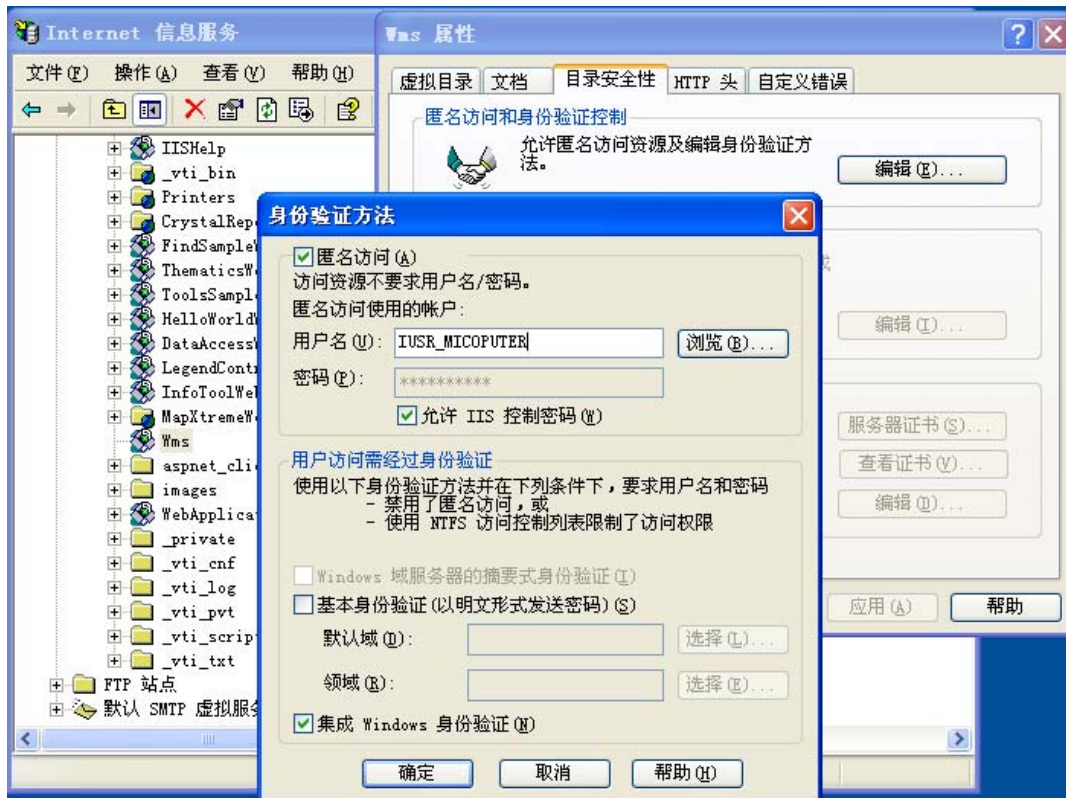
通过 Windows “开始” 菜单启动 Internet 信息服务 (IIS) (**开始 > 控制面板 > 管理工具 > INTERNET 信息服务**)。



展开“默认 Web 站点”项则可以查看刚刚创建的 WMS 虚拟目录。右键单击 WMS 目录并单击**属性**菜单项。



接下来，选择**目录安全性**选项卡，然后单击右上角的**编辑**按钮。在**身份验证方法**对话框中，选择**匿名访问**复选框，以便允许 WMS 服务用户跳过用户名 / 密码步骤。



步骤 7：测试配置

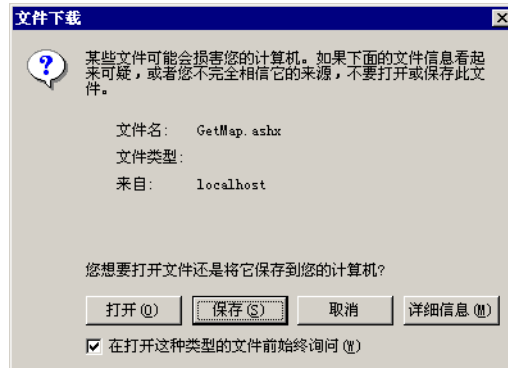
全部启用并正确配置后，让我们以 Internet Explorer 为客户端执行一个简单的查询。Internet Explorer 将不会显示地图，但我们获得 XML 格式的服务器功能。

注： 在 Internet Explorer 中查看 WMS 功能使您可以展开或折叠感兴趣的部分。您也可以使用文本编辑器显示生成的 XML，但结果不易查看。

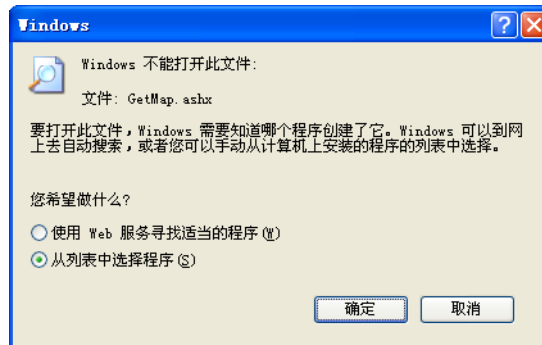
1. 启动浏览器并键入以下 URL:

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

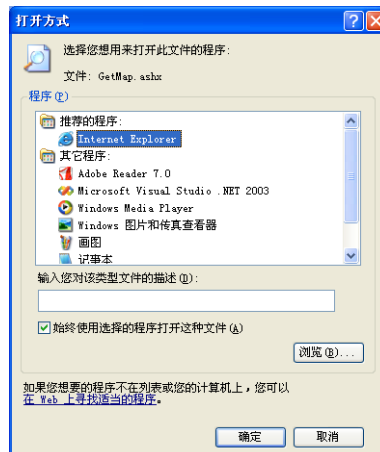
该请求可向服务器请求其功能，如可用图层、坐标系、图层样式和支持的图像格式。
此时将显示**文件下载**对话框。



2. 在**文件下载**对话框中，单击**打开**。
出现该对话框，表明 Windows 无法打开该文件：

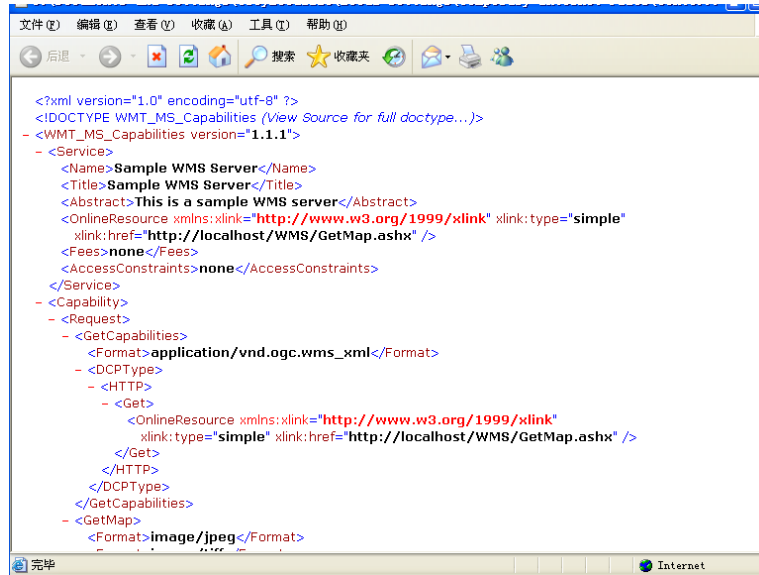


3. 选择**从列表中选择程序**并单击**确定**。
此时将出现**打开方式**对话框，显示可打开该文件的可用应用程序列表。



4. 选择列表顶部的 Internet Explorer 并单击**确定**。

在打开的新 Internet Explorer 窗口中显示如下屏幕图像所示的 XML 文件。



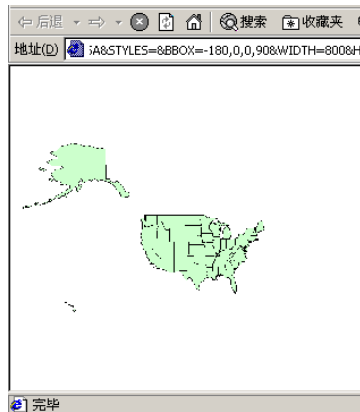
这说明 WMS 服务器已正确配置和运行。如果未出现正确结果，请双击查看以上步骤并重试。有关要查看内容的更详细列表，请参阅第 20 页中的 *WMS 设置的故障诊断*。

步骤 8：请求地图

了解到服务器正在对请求作出响应后，可以使用以下 URL 从 WMS 服务器中请求地图图像。

在 web 浏览器中，键入用于请求地图图像的 URL。

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

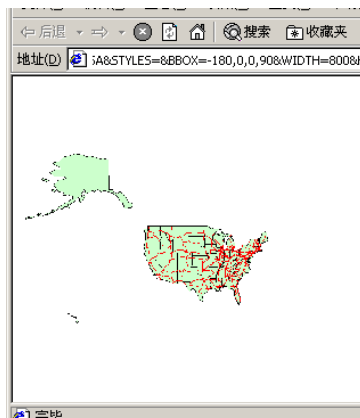


上述 URL 中的 `LAYERS=USA` 意味着您只需检索 USA 图层列表下的图层。在这种情况下，即指 States 和 US Highways 图层。但是请注意，web 浏览器中的 US Highways 不可见。原因是 `WMSample.xml` 中图层的顺序；US Highways 图层列于 States 图层之下。服务器从上至下绘制图层，因此如果顶部图层为边界图层，则下方的点和折线图层将不可见。

要将 US Highways 和 States 图层重新排序, 请在 WMSample.xml 中将以下块移至 States 图层块之上。然后保存并关闭该文件。

```
<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>
```

在 web 浏览器中, 单击“刷新”按钮刷新 URL。



有关 GetCapabilities、GetMap 和 GetFeatureInfo 相关属性的详细信息, 请参阅《MapXtreme 2005 开发人员指南》中的第 20 章“WMS 和 WFS”。

定制 WMS 服务

基本服务可以运行之后, 可以对其进行定制以匹配您的数据和设置。本节中我们将从 MapXtreme 2005 光盘中相同的示例配置文件 (已复制到 WMS 目录) 开始, 然后添加新图层。

我们将执行与上节相同的步骤, 只使用值来匹配新服务器。步骤具体如下:

1. 创建用于存储配置文件的目录。
2. 将示例文件从 MapXtreme 2005 光盘复制到在步骤 1 中创建的新目录下。
3. 修改文件以适应新环境。
4. 可将该文件夹用作 Web 共享目录。
5. 在 IIS 中, 更改新虚拟目录的安全属性。
6. 测试配置。
7. 请求地图。

步骤 1 和 2：复制示例文件

创建新目录 C:\WMSDemoServer 并从产品光盘中复制示例 WMS 文件。将 WMSsample.xml 重新命名为 WMSConfig.xml，以便与上一示例进行区分。

步骤 3：修改文件以适应新环境

在本部分教程中，要向 WMS 服务器添加两个图层：州首府和郡边界。首先，编辑 Web.config 文件以匹配到 WMS 服务器的新路径。验证软件版本号与编译环境是否匹配。

Web.config

```
<configuration>
  <appSettings>
    <add key="configFile" value="C:\WMSDemoServer\WMSConfig.xml" />
    ...
  </appSettings>
  <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>
  </system.web>
</configuration>
```

WMSConfig.xml

在该文件中，我们必须编辑 WMSConfig.xml 文件的 <mxp-wms:Service> 元素，使之与新配置相匹配。在 <mxp-wms:Service> 块中，请为新站点指定名称、标题、摘要和 OnlineResource。要更改的项在以下示例中突出显示为黑体。

```
<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： 带有 mxp: 命名空间（前缀）的元素特定于 MapInfo WMS 实现，不在 OGC 规范中出现。

注： 应用到您的系统时，请确保将 http://MITestServer/WMSDemoServer/ 更改为由 Web 管理员指定的 URL。

接下来，需要定制文件以查看所服务的图层。在该示例中，我们要添加两个 US 数据图层：首府和郡。这些文件与上一部分使用的文件位于相同目录 (C:\Program Files\MapInfo\MapXtreme\6.5\Samples\Data)。

要添加这些图层，则必须为每个图层定义新的 <TABFileDataSourceDefinition> 元素。由于它们与现有数据类似，因此最简单的方法就是复制 <TABFileDataSourceDefinition> 并对其进行修改。每个图层都必须具有新的 id 编号、数据源名称和到文件的路径。

此时，生成的 WMSConfig.xml 将如下所示：

```

<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>

```

此时, 需要修改 XML 的 <WMSLayer> 部分。州首府图层已在 WMSLayer 部分提供, 但标为注释。删除 US Capitals 块开始 (<!--) 和结束 (-->) 位置的注释, 保存 WMSConfig.xml 并刷新浏览器以查看新图层。

```

<!-- 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>-->

```

对于第二个图层, 则必须添加新的部分。同样, 最简单的方法是复制和编辑该部分中现有的块。将该块放置在 US Capitals 和 States 图层之间。指定名称、标题、摘要、图元图层元素和 DataSourceRef id。结果如下所示:

```

<<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>
```

步骤 4：将该文件夹用作 Web 共享目录

该步骤和第 7 页中的步骤 5：将该文件夹用作 Web 共享目录中所述步骤相同。

步骤 5：在 IIS 中，更改新虚拟目录的安全属性

该步骤和第 8 页中的步骤 6：更改 IIS 安全属性中所述步骤相同。

步骤 6：测试配置

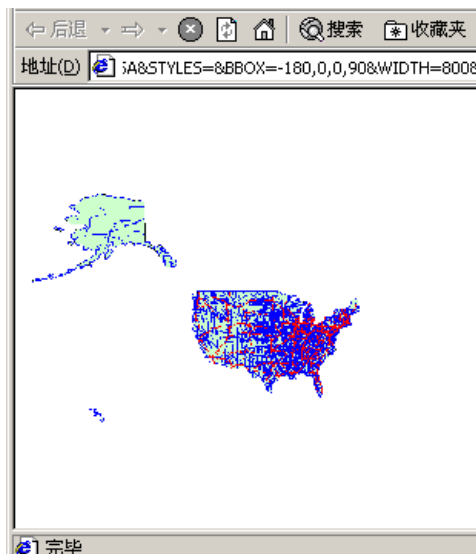
要确保配置正常工作，需要使用 Internet Explorer 发送示例请求。在浏览器中键入以下 URL，然后执行在第 9 页中的步骤 7：测试配置中指定的其它步骤。

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

结果应与以上引用部分中的图像类似。

步骤 7：请求地图

要请求包含新添加首府和郡图层的地图图像，请遵循第 11 页中的步骤 8：请求地图，以显示具有新图层的地图。



创建父图层和子图层

以上步骤说明了如何通过定制 WMS 服务器来使用附加图层。简单来说, 是将这些附加图层添加到父图层 (USA) 下。但实际上, 可能要将数据组织到容纳类似图层的层次结构中。这允许用户只通过一个对父图层的请求即可请求一组图层。

本节说明了如何使用 WMSConfig.xml 中的父图层和子嵌套图层。

父图层

要使对父图层的请求正确运行, 则必须定义 `<SRSNameSet>` 元素用于指示其坐标系。该坐标系适用于父图层及其下面的所有子图层。

```
...
<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>
...
```

在已编辑的 WMSConfig.xml 中, USA 图层既是多个州级图层的父图层, 又是 World 组图层的子图层。在以上示例中, 第一行 `<mxp-wms:WmsLayerList>` 表明 USA 为子图层。

继续检查 WMSConfig.xml 文件将查看到另一个 `<mxp-wms:WmsLayerList>`。它引入了 USA 组图层的子图层。

```
...
<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/>
  </mxp-wms:WmsLayer>
</mxp-wms:WmsLayerList>
...
```

```

    <FeatureLayer id="id10" name="STATECAP" alias="STATECAP"
    volatile="unknown" xmlns="http://www.mapinfo.com/mxp">
      <DataSourceRef ref="id3" />
    </FeatureLayer>
  </mxp-wms:WmsLayer>
  ...

```

每个图层都用 `<mxp-wms:WmsLayer></mxp-wms:WmsLayer>` 元素标记括起来。然后，使用 `<mxp-wms:WmsLayerList></mxp-wms:WmsLayerList>` 元素标记进一步组成图层集合。在我们的示例文件中：

将父图层和子图层一起放置到 `WMSConfig.xml`，则如下所示：

```

  ...
  <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>
...

```

图层元素

编辑 WMSConfig.xml 文件时，需要为每个已配置图层填充多个不同的值。尚未拟定任何指南来说明这些值应在客户端应用程序中如何使用。这些值的特定用法由读取信息的客户端规定。有关如何使用这些值的详细信息，请参阅 OGC WMS 规范 (<http://www.opengeospatial.org/docs/01-068r3.pdf>)。

下面以两个元素 <Name> 和 <Title> 为例。根据 OGC WMS 规范，Name 值用于机器对机器的通信，Title 值则用于与人交互。Name 值是客户端从服务器请求的内容，Title 值则是显示给用户的内容。

以下列表描述了配置文件 <wms:WMSLayer> 部分的每个元素并说明了如何使用：

注： 带有 mxp: 命名空间（前缀）的元素 特定于 MapInfo WMS 实现，不在 OGC 规范中出现。

<Name>

该元素的值通常是指在机器对机器的通信中使用的名称。

示例：<Name>NYSSStreets</Name>

<Title>

该元素的值提供了图层的用户好记的描述。

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

<Abstract>

摘要是对对象的较长叙述性说明。

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

<SRSNameSet>

该元素将 <mxp:SRSName> 元素分组在一起。

<mxp:SRSName>

该元素指定用于特定图层的坐标系。如果该值为空，则图层继承父图层的 <mxp:SRSName> 值。特定图层可包含多个 <mxp:SRSName> 元素。忽略所有重复项。

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

<WmsStyleSet>

该元素用作单个 <WmsStyle> 元素的容器。

<WmsStyle>

每个 <WmsStyle> 都包含 <Name>、<Title> 和 <Abstract> 元素，用于描述在工作空间中创建的特定样式。然后输入样式的详细信息。请确保为元素预加 mxp: namespace 前缀。

示例:

```
<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:imagesize pixel">
      <Pen>mapinfo:pen 2</Pen>
    </mxp:LineStyle>
  </WmsStyle>
</WmsStyleSet>
```

<mxp:FeatureLayer>

该元素标识渲染的图层。每个 <mxp:FeatureLayer> 元素的 id 属性都必须是唯一值。该元素中其它两个属性 - 名称和别名都必须具有各自的值, 但这些值的内容不会对 WMS 产生影响。

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

<mxp:DataSourceRef>

这个空元素具有单一属性 ref, 必须与在配置文件的 <mxp:DataSourceDefinitionSet> 部分指定的图层 (<mxp:TABFileDataSourceDefinition> element) 相匹配。

示例: <mxp:DataSourceRef ref="Streets"/>

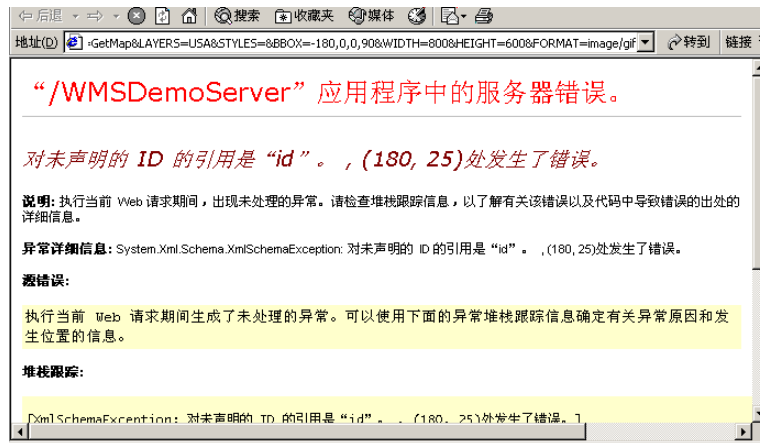
WMS 设置的故障诊断

修复设置

对于所有 web 服务类的应用程序, 配置文件至关重要。一处小的拼写错误都可能导致整个系统停止运行。如果无法连接到刚刚创建的 WMS 服务器, 则可能出现如下所示的典型错误消息。



该错误消息通知您在 WMS 服务器上出现错误，但不解释错误内容也不指出下一步要执行的操作。要获得关于错误的更详细信息，请通过 Internet Explorer 直接连接到服务。该方法允许您查看关于错误的更详细信息。



使用 Internet Explorer 直接连接到服务，可在第 180 行查看 WMSConfig.xml 文件的具体错误。此时，您可以导航到 XML 文件中的这个位置并开始故障诊断。

每次根据需要调整 web.config 或 WMSConfig.xml 文件时，都必须确保已关闭编辑器中的这些文件。如果在服务器尝试访问文件时该文件被另一应用程序占用，则会出现错误。

注： 由于 Windows 的错误，您在每次需要重试 web 服务时都可能需要结束 aspnet_wp.exe（或 Windows 2003 Server 的 w3wp.exe）进程。

修复错误的过程具体如下：

1. 编辑配置文件（web.config 或 wmsconfig.xml）。
2. 关闭文件（或许要重新启动 aspnet_wp.exe 或 w3wp.exe 进程）。
3. 使用 IE 或示例客户端重试 WMS 服务器。

设置的故障诊断

有多个位置可用于验证配置信息是否正确：

Web.config 文件

1. ConfigFile 的值是否正确？
在 <appSettings>|<add> 元素（位于 Web.config 文件）的值属性中，到 wmsconfig.xml 文件的路径必须精确，以确保服务器正常工作。
2. 指定的软件版本号是否正确？
请确保在 <httpHandlers>|<add> 元素（位于 Web.config 文件）的 type(Version) 属性中正确指定版本值（可在步骤 1：验证软件的版本中查看）。
3. 指定的 Public Key Tokens 是否正确？
请确保在 <httpHandlers>|<add> 元素（位于 Web.config 文件）的 type(PublicKeyToken) 属性中正确指定 Public Key Token 值。可在与验证软件版本号的相同位置（开始 > 运行 > Assembly）验证该信息。

即使有任何细致的详细信息不精确，系统也无法正确运行并会抛出异常。每次消除这些详细信息后，系统才能正确运行。

wmsconfig.xml 文件

1. <mxp-wms:OnlineResource> 元素中的值是否正确指定?
该值需要指定正确的 URL, 即, 以 getmap.ashx 结尾。
2. 数据源是否在 <DataSourceDefinitionSet> 元素中正确定义?
每个数据源都需要具有其自身的元素, 该元素包含到每个特定文件的有效路径。请确保文件中的 id 属性值或 <DataSourceName> 元素值不重复。此外, 每个 <FileName> 元素值必须为指向有效文件的有效路径。
3. <WmsLayerList> 中的元素是否正确指定?
 - a. 名称和标题是否唯一?
如果名称或标题重复, 则系统抛出异常。
 - b. <mxp:FeatureLayer> 属性是否唯一?
如果上述任何值出现重复, 则系统抛出错误。
 - c. <DataSourceRef> 元素中的 ref 属性是否正确引用了对应的 <TABFileDataSourceDefinition> 元素?
该属性的每个值都必须与对应 <TABFileDataSourceDefinition> 元素的 id 属性完全匹配, 以确保系统正常运行。

IIS 设置

1. 要使用的目录是否已当作虚拟目录使用?
2. IIS 服务器是否正在运行?
3. 是否已结束 aspnet_wp.exe 或 w3wp.exe 进程?
由于进程几乎立即重新启动, 因此不会直接显示进程已结束, 但内存使用率会显著降低。

访问网络驱动器

ASP.NET 登录的是没有网络文件访问权限的本地机帐户。因此, 必须将运行 WMS 服务器的 ASP.NET 进程配置为作为具有访问网络文件夹权限的用户运行。此外, 必须根据通用命名规则 (UNC) 完全指定网络文件夹, 而不是根据映射驱动器。例如, \\server\data\, 而不是映射驱动器 (如 “Z:\data”)。

配置 ASP.NET 进程用户时需要考虑很多情况和选项。以下链接可访问描述适当解决方案的支持文档。

在 IIS 5.0/5.1 中访问网络驱动器

简单的密码

选择运行 ASP.NET 进程的网络用户, 并确保该用户可访问网络文件夹。该用户具有防止意外或数据意外访问情况发生的限制权限。

以下示例假定用户名和密码分别为 “MYDOMAIN\WMSUser” 和 “WMSUserPassword”。

1. 备份 Web.config 文件。
2. 将下行添加到 web.config 文件的 <system.web> 部分。


```
<identity password="WMSUserPassword" userName="MYDOMAIN\WMSUser"
  impersonate="true" />
```

3. 重新启动 WMS 服务器。

由于命名用户的密码未以加密形式存储，因此使用该方法可能致使其他人访问 web.config 文件并获取该密码。如果需要更安全的方法，请使用下面的“加密密码”部分。

加密密码

有关创建加密密码的指示，可阅读以下 MSDN 文章：

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

可能需要使用以下文章中描述的 hotfix 来启用加密注册表密码的使用：

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

在 IIS 6.0 中访问网络驱动器

上述 IIS5.0/5.1 方法也适用于 IIS 6.0。也可能有用于配置 ASP.NET 进程用户标识和相关安全的更简便方法。例如，可参阅以下文章的“工作进程隔离模式下 IIS 6.0 中用户名和密码的等价设置”部分：

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