**XML 架构定义工具 (Xsd.exe)**

**.NET Framework 2.0**

[其他版本](javascript:;)

* [.NET Framework 4](http://msdn.microsoft.com/zh-cn/library/x6c1kb0s%28v=VS.100%29.aspx)
* [Visual Studio 2008](http://msdn.microsoft.com/zh-cn/library/x6c1kb0s%28v=VS.90%29.aspx)

XML 架构定义 (Xsd.exe) 工具从 XDR、XML 和 XSD 文件或者从运行库程序集中的类生成 XML 架构或公共语言运行库类。

[复制](javascript:CodeSnippet_CopyCode('CodeSnippetContainerCode0');" \o "复制到剪贴板。)

xsd file.xdr [/outputdir:directory][/parameters:file.xml]

xsd file.xml [/outputdir:directory] [/parameters:file.xml]

xsd file.xsd {/classes | /dataset} [/element:element]

[/language:language] [/namespace:namespace]

[/outputdir:directory] [URI:uri] [/parameters:file.xml]

xsd {file.dll | file.exe} [/outputdir:directory] [/type:typename [...]][/parameters:file.xml]

参数

|  |  |
| --- | --- |
| **参数** | **说明** |
| file.extension | 指定要转换的输入文件。必须将扩展名指定为下列之一：.xdr、.xml、.xsd、.dll 或 .exe。  如果指定一个 XDR 架构文件（.xdr 扩展名），则 Xsd.exe 将 XDR 架构转换为 XSD 架构。输出文件与 XDR 架构同名，但扩展名为 .xsd。  如果指定一个 XML 文件（.xml 扩展名），则 Xsd.exe 从文件中的数据推导出架构并产生一个 XSD 架构。输出文件与 XML 文件同名，但扩展名为 .xsd。  如果指定一个 XML 架构文件（.xsd 扩展名），则 Xsd.exe 将为对应于 XML 架构的运行库对象生成源代码。  如果指定一个运行库程序集文件（.exe 或 .dll 扩展名），则 Xsd.exe 为该程序集中的一个或多个类型生成架构。可以使用 **/type** 选项来指定为其生成架构的类型。输出架构被命名为 schema0.xsd、schema1.xsd，依此类推。仅当给定类型使用 **XMLRoot** 自定义属性指定命名空间时，Xsd.exe 才生成多个架构。 |

常规选项

|  |  |
| --- | --- |
| **选项** | **说明** |
| **/h**[**elp**] | 显示该工具的命令语法和选项。 |
| **/o**[**utputdir**]**:**directory | 指定输出文件的目录。此参数只能出现一次。默认为当前目录。 |
| **/?** | 显示该工具的命令语法和选项。 |
| **/P[arameters]:** file.xml | 从指定的 .xml 文件读取各种操作模式的选项。缩写形式为“/p:”。有关更多信息，请参见下面的“备注”部分。 |

XSD 文件选项

必须为 xsd 文件仅指定下列选项中的一个。

|  |  |
| --- | --- |
| **选项** | **说明** |
| **/c**[**lasses**] | 生成与指定架构相对应的类。若要将 XML 数据读入对象，请使用 [System.Xml.Serialization.XmlSerializer.Deserializer](http://msdn.microsoft.com/zh-cn/library/system.xml.serialization.xmlserializer.deserialize%28v=VS.80%29.aspx) 方法。 |
| **/d**[**ataset**] | 生成一个从 [DataSet](http://msdn.microsoft.com/zh-cn/library/system.data.dataset%28v=VS.80%29.aspx) 派生的类，该类与指定的架构相对应。若要将 XML 数据读入派生类，请使用 [System.Data.DataSet.ReadXml](http://msdn.microsoft.com/zh-cn/library/system.data.dataset.readxml%28v=VS.80%29.aspx) 方法。 |

还可以为 .xsd 文件指定下列任何选项。

|  |  |
| --- | --- |
| **选项** | **说明** |
| **/e**[**lement**]**:**element | 指定架构中要为其生成代码的元素。默认情况下，键入所有元素。可以多次指定该参数。 |
| **/enableDataBinding** | 在所有生成的类型上实现 [INotifyPropertyChanged](http://msdn.microsoft.com/zh-cn/library/system.componentmodel.inotifypropertychanged%28v=VS.80%29.aspx) 接口以启用数据绑定。缩写形式为“**/edb**”。 |
| **/f**[**ields**] | 生成字段，而不是生成属性。默认情况下生成属性。 |
| **/l**[**anguage**]**:**language | 指定要使用的编程语言。从 **CS**（默认情况下为 C#）、**VB** (Visual Basic)、**JS** (JScript) 或 **VJS** (Visual J#) 中进行选择。还可以为实现 [System.CodeDom.Compiler.CodeDomProvider](http://msdn.microsoft.com/zh-cn/library/system.codedom.compiler.codedomprovider%28v=VS.80%29.aspx) 的类指定完全限定名。 |
| **/n**[**amespace**]**:**namespace | 为生成的类型指定运行库命名空间。默认命名空间为 **Schemas**。 |
| **/nologo** | 取消显示版权标志。 |
| **/order** | 在所有粒子成员上生成显式顺序标识符。 |
| **/o[ut]:** directoryName | 指定要放置文件的输出目录。默认为当前目录。 |
| **/u**[**ri**]**:**uri | 为架构中要为其生成代码的元素指定 URI。该 URI（如果存在）应用于使用 **/element** 选项指定的所有元素。 |

DLL 和 EXE 文件选项

|  |  |
| --- | --- |
| **选项** | **说明** |
| **/t**[**ype**]**:**typename | 指定要为其创建架构的类型的名称。可以指定多个类型参数。如果 typename 不指定一个命名空间，则 Xsd.exe 将程序集中的所有类型与指定类型相匹配。如果 typename 指定一个命名空间，则仅匹配那个类型。如果 typename 以星号字符 (\*) 结尾，则此工具匹配所有以 \* 前的字符串开头的类型。如果省略 **/type** 选项，则 Xsd.exe 为程序集中的所有类型生成架构。 |

备注

下表显示了 Xsd.exe 执行的操作。

XDR 到 XSD

使用精简 XML 数据架构文件生成 XML 架构。XDR 为早期基于 XML 的架构格式。

XML 到 XSD

使用 XML 文件生成 XML 架构。

XSD 到 DataSet

使用 XSD 架构文件生成公共语言运行库 [DataSet](http://msdn.microsoft.com/zh-cn/library/system.data.dataset%28v=VS.80%29.aspx) 类。生成的类为规则 XML 数据提供复杂对象模型。

XSD 到类

使用 XSD 架构文件生成运行库类。生成的类可以和 [System.Xml.Serialization.XmlSerializer](http://msdn.microsoft.com/zh-cn/library/system.xml.serialization.xmlserializer%28v=VS.80%29.aspx) 一起使用，以读取和编写遵循架构的 XML 代码。

类到 XSD

使用运行库程序集文件中的一个或多个类型生成 XML 架构。生成的架构定义了 **System.Xml.Serialization.XmlSerializer** 使用的 XML 格式。

Xsd.exe 只允许操作遵循由万维网联合会 (W3C) 提议的 XML 架构定义 (XSD) 语言的 XML 架构。有关 XML 架构定义提议或 XML 标准的更多信息，请参见 http://w3c.org。

通过 XML 文件设置选项

通过使用 **/parameters** 开关，可指定设置各种选项的单个 XML 文件。可设置的选项取决于您如何使用 XSD.exe 工具。选择包括生成架构、生成代码文件，或生成包含 **DataSet** 功能的代码文件。例如，生成架构时可将 **<assembly>** 元素设置为可执行文件 (.exe) 或类库文件 (.dll) 的名称，但生成代码文件时则不能。下面的 XML 演示如何将 **<generateSchemas>** 元素用于指定的可执行文件：

[复制](javascript:CodeSnippet_CopyCode('CodeSnippetContainerCode1');" \o "复制到剪贴板。)

<!-- This is in a file named GenerateSchemas.xml. -->

<xsd xmlns='http://microsoft.com/dotnet/tools/xsd/'>

<generateSchemas>

<assembly>ConsoleApplication1.exe</assembly>

</generateSchemas>

</xsd>

如果前面的 XML 包含在名为 GenerateSchemas.xml 的文件中，则通过在命令提示处键入下面的内容并按 Enter 使用 **/parameters** 开关：

**xsd /p:GenerateSchemas.xml**

另外，如果为程序集中的单个类型生成架构，则可以使用下面的 XML：

[复制](javascript:CodeSnippet_CopyCode('CodeSnippetContainerCode2');" \o "复制到剪贴板。)

<!-- This is in a file named GenerateSchemaFromType.xml. -->

<xsd xmlns='http://microsoft.com/dotnet/tools/xsd/'>

<generateSchemas>

<type>IDItems</type>

</generateSchemas>

</xsd>

但是若要使用上面的代码，您还必须在命令提示处提供程序集的名称。在命令提示处键入下面的内容（假设 XML 文件名为 GenerateSchemaFromType.xml）：

**xsd /p:GenerateSchemaFromType.xml ConsoleApplication1.exe**

必须为 **<generateSchemas>** 元素仅指定以下选项中的一个。

|  |  |
| --- | --- |
| **元素** | **说明** |
| <assembly> | 指定将从中生成架构的程序集。 |
| <type> | 指定程序集中找到的要为其生成架构的类型。 |
| <xml> | 指定要为其生成架构的 XML 文件。 |
| <xdr> | 指定要为其生成架构的 XDR 文件。 |

若要生成代码文件，请使用 **<generateClasses>** 元素。下面的示例生成一个代码文件。注意，另外还显示了两个属性，它们允许您为生成的文件设置编程语言和命名空间。

[复制](javascript:CodeSnippet_CopyCode('CodeSnippetContainerCode3');" \o "复制到剪贴板。)

<xsd xmlns='http://microsoft.com/dotnet/tools/xsd/'>

<generateClasses language='VB' namespace='Microsoft.Serialization.Examples'/>

</xsd>

<!-- You must supply an .xsd file when typing in the command line.-->

<!-- For example: xsd /p:genClasses mySchema.xsd -->

可为 **<generateClasses>** 元素设置以下选项。

|  |  |
| --- | --- |
| **元素** | **说明** |
| <element> | 指定 .xsd 文件中要为其生成代码的元素。 |
| <schemaImporterExtensions> | 指定派生自 [SchemaImporterExtension](http://msdn.microsoft.com/zh-cn/library/system.xml.serialization.advanced.schemaimporterextension%28v=VS.80%29.aspx) 类的类型。 |

下表显示也可用于 **<generateClasses>** 元素的属性。

|  |  |
| --- | --- |
| **属性** | **说明** |
| language | 指定要使用的编程语言。从 **CS**（默认情况下为 C#）、**VB** (Visual Basic)、**JS** (JScript) 或 **VJS** (Visual J#) 中进行选择。也可指定实现 [CodeDomProvider](http://msdn.microsoft.com/zh-cn/library/system.codedom.compiler.codedomprovider%28v=VS.80%29.aspx) 的类的完全限定名。 |
| namespace | 为生成的代码指定命名空间。命名空间必须符合 CLR 标准（例如，没有空格或反斜杠字符）。 |
| options | 以下值之一：**none**、**properties**（生成属性而不是公共字段）、**order** 或 **enableDataBinding**（请参见前面“XSD 文件选项”一节的 **/order** 和 **/enableDataBinding** 开关）。 |

使用 **<generateDataSets>** 元素还可以控制如何生成 **DataSet** 代码。下面的 XML 指定生成的代码使用 **DataSet** 结构（如 [DataTable](http://msdn.microsoft.com/zh-cn/library/system.data.datatable%28v=VS.80%29.aspx) 类）为指定元素创建 Visual Basic 代码。

<xsd xmlns='http://microsoft.com/dotnet/tools/xsd/'>

<generateDataSet language='VB' namespace='Microsoft.Serialization.Examples'>

<element>IDItems</element>

</generateDataSet>

</xsd>

可为 **<generateDataSet>** 元素设置以下选项。

|  |  |
| --- | --- |
| **元素** | **说明** |
| <element> | 指定 .xsd 文件中要为其生成代码的元素。 |

下表显示可用于 **<generateDataSet>** 元素的属性。

|  |  |
| --- | --- |
| **属性** | **说明** |
| language | 指定要使用的编程语言。从 **CS**（默认情况下为 C#）、**VB** (Visual Basic)、**JS** (JScript) 或 **VJS** (Visual J#) 中进行选择。也可指定实现 **CodeDomProvider** 的类的完全限定名。 |
| namespace | 为生成的代码指定命名空间。命名空间必须符合 CLR 标准（例如，没有空格或反斜杠字符）。 |

有些属性可在顶级 **<xsd>** 元素上设置。这些选项可用于任何子元素（**<generateSchemas>**、**<generateClasses>** 或 **<generateDataSet>**）。下面的 XML 代码在名为“MyOutputDirectory”的输出目录中为名为“IDItems”的元素生成代码。

[复制](javascript:CodeSnippet_CopyCode('CodeSnippetContainerCode4');" \o "复制到剪贴板。)

<xsd xmlns='http://microsoft.com/dotnet/tools/xsd/' output='MyOutputDirectory'>

<generateClasses>

<element>IDItems</element>

</generateClasses>

</xsd>

下表显示也可用于 **<xsd>** 元素的属性。

|  |  |
| --- | --- |
| **属性** | **说明** |
| output | 将放置生成的架构或代码文件的目录的名称。 |
| nologo | 取消显示版权标志。设置为 **true** 或 **false**。 |
| help | 显示该工具的命令语法和选项。设置为 **true** 或 **false**。 |

[复制](javascript:CodeSnippet_CopyCode('CodeSnippetContainerCode5');" \o "复制到剪贴板。)

示例

下面的命令从 myFile.xdr 生成一个 XML 架构并将它保存到当前目录中。

[复制](javascript:CodeSnippet_CopyCode('CodeSnippetContainerCode6');" \o "复制到剪贴板。)

xsd myFile.xdr

下面的命令从 myFile.xml 生成一个 XML 架构并将它保存到指定目录中。

[复制](javascript:CodeSnippet_CopyCode('CodeSnippetContainerCode7');" \o "复制到剪贴板。)

xsd myFile.xml /outputdir:myOutputDir

下面的命令生成一个与 C# 语言中的指定架构相对应的数据集，并在当前目录中将其保存为 XSDSchemaFile.cs。

[复制](javascript:CodeSnippet_CopyCode('CodeSnippetContainerCode8');" \o "复制到剪贴板。)

xsd /dataset /language:CS XSDSchemaFile.xsd

下面的命令为程序集 myAssembly.dll 中的所有类型生成 XML 架构，并在当前目录中将它们保存为 schema0.xsd。

[复制](javascript:CodeSnippet_CopyCode('CodeSnippetContainerCode9');" \o "复制到剪贴板。)

xsd myAssembly.dll

请参见