Working with Bibliographies

06/07/2017 • 8 minutes to read • ▯ ▯ 👤 👥 👤 +4

In this article

Introduction

The Word object model includes several objects designed for automating the creation of bibliographies. The following table lists the main objects of the Word Bibliography feature. You can use these objects, and additional properties and methods in the Word object model, to add sources to the source lists, cite sources in a document, and manage sources. The objects in the Word model for that you use for managing bibliography sources are shown in the following table.

Object	Description
Source	An individual source, such as a book, journal article, or interview.
Sources	A collection of **Source** objects.
Bibliography	The list of sources cited in the document (the current list) or the list of sources available in the application (in the master list).

Understanding the source XML

Sources are added to the source lists programmatically by using XML strings. Depending on the type of source you want to add, the required XML structure changes. To determine

the XML structure for a source type, you can add the same source type manually, and then view the XML returned. The following steps describe how to do this.

1. On the **References** ribbon, click **Manage Sources**.

2. In the **Source Manager** dialog box, click **New**.

3. In the **Create Source** dialog box, select the type of source to create. For this example, select **Book**.

4. Fill out the source fields, as shown in the following table:

Field	Value
Author	Andrew Dixon
Title	Stylish Bibliographies
Year	2006
City	Chicago
Publisher	Adventure Works Press
Tag name	And01

5. You can view and add information to additional fields by checking **Show All Bibliography Fields**.

6. Click **OK**.

7. Close the **Source Manager** dialog box.

8. Start the Visual Basic Editor (Alt+F11).

9. Display the **Immediate Window** (Ctrl+G).

10. Paste and run the following code. `Sub GetBibliographyXML() Dim strXml As String Dim objSource As Source Set objSource = Application.Bibliography.Sources( _ Application.Bibliography.Sources.Count) Debug.Print objSource.XML End Sub`

After following the previous steps, the Immediate Window contains the following XML code.

VB   📋 Copy

```xml
<b:Source
xmlns:b="http://schemas.microsoft.com/office/word/2004/10/bibliography">
    <b:Tag>And01</b:Tag>
    <b:SourceType>Book</b:SourceType>
    <b:Guid>{6D86D06C-9022-4932-8D4C-84C2B0843381}</b:Guid>
    <b:LCID>0</b:LCID>
    <b:Author>
        <b:Author>
            <b:NameList>
                <b:Person>
                    <b:Last>Dixon</b:Last>
                    <b:First>Andrew</b:First>
                </b:Person>
            </b:NameList>
        </b:Author>
    </b:Author>
    <b:Title>Stylish Bibliographies</b:Title>
    <b:Year>2006</b:Year>
    <b:City>Chicago</b:City>
    <b:Publisher>Adventure Works Press</b:Publisher>
</b:Source>
```

The Guid and LCID elements are optional, but you can provide values for them if you want. The Guid element value should be a valid GUID, which you can generate programmatically outside the Word object model. (See the Visual Studio documentation or the Windows documentation on MSDN for information about programmatically generating ID.) Word generates GUIDs when users add or edit a source. If you do not add a GUID to the XML and a user then edits a source, Word generates a GUID. This enables Word to determine which source is most recent, based on the value of the GUID, and to prompt whether the user wants Word to update the outdated source to maintain continuity between the master list and the current list.

The LCID specifies the language for the source. (See MSDN for valid language identification values.) Word uses the LCID to know how to display a cited source in a document's bibliography. For example, one source may be written in French, one in English, and one in Japanese. From the LCID, Word determines how to display names (for example, Last, First for English), what punctuation to use (for example, using comma in one language and a semicolon in another), and what strings to use (for example, whether to use "et al" or another localized form).

After removing optional elements, you may have a structure similar to the following XML structure. (You can determine which elements are required because they do not have a

corresponding editable field in the **Create Source** dialog box. Omitting one or more required element raises a run-time error.)

```vb
<b:Source
xmlns:b="http://schemas.microsoft.com/office/word/2004/10/bibliography">
    <b:Tag>And01</b:Tag>
    <b:SourceType></b:SourceType>
    <b:Author>
        <b:Author>
            <b:NameList>
                <b:Person>
                    <b:Last></b:Last>
                    <b:First></b:First>
                </b:Person>
            </b:NameList>
        </b:Author>
    </b:Author>
    <b:Title></b:Title>
    <b:Year></b:Year>
    <b:City></b:City>
    <b:Publisher></b:Publisher>
</b:Source>
```

Now that you have the basic structure of the source XML for a book, you can add additional book sources to the master source list and the current source list. You can locate additional elements by checking the **Show All Bibliography Fields** check box.

> ⓘ **Note**
>
> Alternatively, you can obtain the XML from the bibliography source file named "sources.xml" located at `C:\Users\<user>\AppData\Roaming\Microsoft\Bibliography`. This file stores the master source list for a user.

Adding sources to the master source list and the current source list

Adding sources to the master source list is similar to adding sources to the current source list, with the exception that you access the **Sources** collection from different main objects. To add a source to the master source list, you access the **Sources** collection from the **Bibliography** property of the **Application** object. To add a source to the current source

list, access the **Sources** collection from the **Bibliography** property of the **Document** object.

The following example uses the basic structure determined previously to add another book source to the master source list.

VB   ⎘ Copy

```vb
Sub AddBibSource()

    Dim strXml As String

    strXml = "<b:Source xmlns:b=""http://schemas.microsoft.com/" & _
        "office/word/2004/10/bibliography""><b:Tag>Mor01</b:Tag>" & _
        "<b:SourceType>Book</b:SourceType><b:Author><b:Author>" & _
        "<b:NameList><b:Person><b:Last>Hezi</b:Last>" & _
        "<b:First>Mor</b:First></b:Person></b:NameList></b:Author>" & _
        "</b:Author><b:Title>The New Office</b:Title>" & _
        "<b:Year>2006</b:Year><b:City>Seattle</b:City>" & _
        "<b:Publisher>Adventure Works Press</b:Publisher>" & _
        "</b:Source>"

    Application.Bibliography.Sources.Add strXml

End Sub
```

You can change the line `Application.Bibliography.Sources.Add strXml` to `ActiveDocument.Bibliography.Sources.Add strXml`

Inserting a source programmatically into the master source list does not automatically add it to the current source list. However, to add a citation to a document, the source must be listed in the current source list. You can manually copy one or more sources from the master list to the current list by using the **Source Manager** dialog box, or you can programmatically copy one or more sources from the master list to the current list. The following example copies all sources in the master source to the current source. After the sources are added to your current list, you can insert citations for those sources into a document.

VB   ⎘ Copy

```vb
Sub CopyToCurrentList()
    Dim objSource As Source
    Dim strXml As String

    On Error Resume Next
```

```vb
    For Each objSource In Application.Bibliography.Sources
        strXml = objSource.XML
        ActiveDocument.Bibliography.Sources.Add strXml
    Next
End Sub
```

> ⓘ **Note**
>
> The value of the **Tag** property must be unique across sources in the current list. Thus the `On Error Resume Next` line is needed to allow the code to skip over any sources in the master list that have conflicting tag values in the current list. You can modify this code to capture instances when Word cannot copy a source from the master list to the current list.

Sharing your source list

There may be times when you want to share a source list with others in an organization. When you add sources to the master list, Word adds them to a file names "sources.xml" located at `C:\Users\<user>\AppData\Roaming\Microsoft\Bibliography\sources.xml`. You can share this file with others by giving them the file, which users can then load manually from the **Source Manager** dialog box or programmatically through code.

> ⓘ **Note**
>
> When a user loads a source file, this is a one-time-only occurrence and does not change either the existing master list or their current list. They can manually add the items in the shared source file to the current list by using the **Source Manager** dialog box.

You can programmatically load a shared source. The following example shows how to load a shared source file that is located on a share on a local computer.

VB	⧉ Copy

```vb
Sub LoadSharedSource()
    Application.LoadMasterList "\\server\public\sources.xml"
End Sub
```

> ⓘ **Note**
>
> Sharing the source.xml source file shares only sources in the master source list. Sources located in the current source are located in a document's data store. You can access this file by saving a document and opening the resulting DOCX file in a file compression application, such as WinZip. You can find the source file at the path "customXml" with a file name of (or similar to) "item1.xml". If you need to share the sources in a document with other users, you can share this file the same way that you would share the master list source file, as described previously.

Sorting the master source list

You can set the sort order in the **Source Manager** dialog box by using the **BibliographySort** property. The **BibliographySort** property can be a **String** value of "Author", "Tag", "Title", or "Year". This object does not alter the sorting of sources in the document's bibliography. The following example sorts the sources by title.

VB   ⧉ Copy

```vb
Sub SortBibliography()
    Options.BibliographySort = "Title"
End Sub
```

Inserting citations

You can insert a bibliography citation by using the Add method for the Fields collection. The following example inserts a citation at the cursor for the source that you added previously. The text for the field equals the tag value, or the value of the Tag element, which in this case is "Mor01". (See the XML code in the AddBibSource subroutine shown previously for the XML string "<b:Tag>Mor01</b:Tag>".) The value of the Tag element also corresponds to the **Tag** property for a **Source** object.

VB   ⧉ Copy

```vb
Sub InsertBibCitation()
    Selection.Fields.Add Selection.Range, _
        wdFieldCitation, "Mor01"
End Sub
```

Applying a bibliography style

After you insert a bibliography into a document, you can set the bibliography style. Word formats several different styles of bibliographies. You can set the bibliography style by using the **BibliographyStyle** property. This property can be one of the following **String** values:

- APA

- Chicago

- GB7714

- GOST - Name Sort

- GOST - Title Sort

- ISO 690 - First Element Date

- ISO 690 - Numerical Reference

- MLA

- SISTO2

- Turabian

> ⓘ **Note**
>
> These values are included in Word, but new values may be added at any point in the future as new bibliography styles are defined.

The following example sets the default bibliography style to the MLA style.

VB	⧉ Copy

```vb
Sub SetBibliographyStyle()
    Options.BibliographyStyle = "MLA"
End Sub
```

> ⓘ **Note**

You can also define your own documentation style in XML. The directory `C:\Program Files\Microsoft Office\Office15\1033\Bibliography\Style` contains one XSL file for every documentation style on your computer. Open any file for a sample of how to create your own XSLT. Any user can share a custom bibliography style XSL file by placing it into the above folder on their computer.

Inserting a bibliography

As with citations, bibliographies use fields. To insert a bibliography, you need to insert a field with a **wdFieldBibliography** constant specified for the field type. The following code inserts a bibliography into the active document at the cursor. This example assumes that the cursor is located at the end of the document or on a new page.

VB	⧉ Copy

```vb
Sub InsertBibliography()
    Selection.Fields.Add Selection.Range, _
        wdFieldBibliography
End Sub
```

Support and feedback

Have questions or feedback about Office VBA or this documentation? Please see [Office VBA support and feedback](#) for guidance about the ways you can receive support and provide feedback.