

System.Xml.XmlNamespaceManager Class

```
[ILASM]
.class public XmlNamespaceManager extends System.Object
implements System.Collections.IEnumerable

[C#]
public class XmlNamespaceManager: IEnumerable
```

Assembly Info:

- *Name:* System.Xml
- *Public Key:* [00 00 00 00 00 00 00 00 04 00 00 00 00 00 00]
- *Version:* 1.0.x.x
- *Attributes:*
 - CLSCompliantAttribute(true)

Implements:

- **System.Collections.IEnumerable**

Summary

Resolves, adds, and removes namespaces in a collection and provides scope management for these namespaces. This class is used by the **System.Xml.XmlReader** and **System.Xml.XmlTextReader** classes.

Inherits From: System.Object

Library: XML

Thread Safety: All public static members of this type are safe for multithreaded operations. No instance members are guaranteed to be thread safe.

Description

This class stores prefixes and namespaces as unique **System.String** objects.

System.Xml.XmlNamespaceManager assumes all prefixes and namespaces are valid.

If the prefix and namespace already exist within the current scope, they will replace the existing prefix/namespace combination. The same prefix and namespace combination can exist across different scopes.

The following prefix/namespace pairs are added by default to the

1
2

System.Xml.XmlNamespaceManager. They can be determined at any scope.

Prefix	Namespace
xmlns	http://www.w3.org/2000/xmlns/ (the xmlns prefix namespace)
xml	http://www.w3.org/XML/1998/namespace (The XML namespace)
System.String.Empty	System.String.Empty . The empty namespace can be reassigned a different prefix. For example, <code>xmlns=""</code> defines the default namespace to be the empty namespace.

3
4

1 XmlNamespaceManager(System.Xml.Xml 2 NameTable) Constructor

```
3 [ILASM]  
4 public rtspecialname specialname instance void .ctor(class  
5 System.Xml.XmlNameTable nameTable)
```

```
6 [C#]  
7 public XmlNamespaceManager(XmlNameTable nameTable)
```

8 Summary

9 Constructs and initializes a new instance of the
10 **System.Xml.XmlNamespaceManager** class.

11 Parameters

12
13

Parameter	Description
<i>nameTable</i>	The System.Xml.XmlNameTable to use.

14

15 Description

16 [Note: *nameTable* is used to look up prefixes and namespaces.]

17 Exceptions

18
19

Exception	Condition
System.ArgumentNullException	<i>nameTable</i> is null .

20

21

22

1 XmlNamespaceManager.AddNamespace(System.String, System.String) Method

```
3 [ILASM]  
4 .method public hidebysig virtual void AddNamespace(string  
5 prefix, string uri)  
6  
7 [C#]  
8 public virtual void AddNamespace(string prefix, string uri)
```

8 Summary

9 Adds the given namespace to the collection of namespaces maintained
10 by the current instance in
11 **System.Xml.XmlNamespaceManager.NameTable**.

12 Parameters

Parameter	Description
<i>prefix</i>	A System.String specifying the prefix to associate with the namespace being added. Use System.String.Empty to add a default namespace.
<i>uri</i>	A System.String specifying the namespace to add.

16 Description

17 [Note: The same prefix and namespace combination can exist across
18 different scopes.]

19 Behaviors

20 As described above.

21 Default

22 This method assumes all prefixes and namespaces are valid but does
23 not check *prefix* and *uri* for conformance to the W3C specification.

24
25 If the specified prefix/namespace combination already exists within the
26 current scope, their **System.String** instances will be replaced in
27 **System.Xml.XmlNamespaceManager.NameTable** with the new
28 **System.String** instances of *prefix* and *uri*.

29 How and When to Override

30 Override this method to check *prefix* and *uri* for conformance to the
31 W3C specification.

1 **Exceptions**

2

3

Exception	Condition
System.ArgumentException	<i>prefix</i> is "xml" or "xmlns".
System.ArgumentNullException	<i>prefix</i> is null , or <i>uri</i> is null .

4

5

6

1 XmlNamespaceManager.GetEnumerator() 2 Method

```
3 [ILASM]  
4 .method public hidebysig virtual class  
5 System.Collections.IEnumerator GetEnumerator()  
6  
7 [C#]  
8 public virtual IEnumerator GetEnumerator()
```

8 Summary

9 Provides support for iteration over the collection of namespaces
10 maintained by the current instance.

11 Return Value

12

13 A **System.Collections.IEnumerator**.

14 Behaviors

15 As described above.

16 How and When to Override

17 Override this method to customize the behavior of this method in
18 types derived from the **System.Xml.XmlNamespaceManager** class.

19 Usage

20 Use this method to iterate through the collection of namespaces in
21 **System.Xml.XmlNamespaceManager.NameTable**.

22

1 XmlNamespaceManager.HasNamespace(System.String) Method

```
3 [ILASM]
4 .method public hidebysig virtual bool HasNamespace(string
5 prefix)
6 [C#]
7 public virtual bool HasNamespace(string prefix)
```

8 Summary

9 Gets a value indicating whether the supplied prefix has a namespace
10 defined for the current pushed scope.

11 Parameters

12
13

Parameter	Description
<i>prefix</i>	A System.String containing the prefix of the namespace to find.

14
15
16

15 Return Value

17 A **System.Boolean** where **true** indicates there is a namespace
18 defined; otherwise, **false**. If *prefix* is **null**, returns **false**.

19 Behaviors

20 As described above.

21 Default

22 When *prefix* is set to **System.String.Empty**, this method returns **true**
23 to indicate a default empty namespace is defined in the current scope.

24 How and When to Override

25 Override this method to customize the behavior of this method in
26 types derived from the **System.Xml.XmlNamespaceManager** class.

27

1 XmlNamespaceManager.LookupNamespace(2 System.String) Method

```
3 [ILASM]
4 .method public hidebysig virtual string
5 LookupNamespace(string prefix)
6
7 [C#]
8 public virtual string LookupNamespace(string prefix)
```

8 Summary

9 Gets the namespace URI associated with the specified prefix.

10 Parameters

11
12

Parameter	Description
<i>prefix</i>	A System.String containing the prefix whose namespace URI to resolve. To match the default namespace, pass System.String.Empty .

13
14
15

14 Return Value

16 A **System.String** containing the namespace URI for *prefix*, or **null** if
17 there is no mapped namespace.

18 Behaviors

19 As described above.

20 How and When to Override

21 Override this method to customize the behavior of this method in
22 types derived from the **System.Xml.XmlNamespaceManager** class.

23

1 XmlNamespaceManager.LookupPrefix(System.String) Method

```
3 [ILASM]  
4 .method public hidebysig virtual string LookupPrefix(string  
5 uri)  
6 [C#]  
7 public virtual string LookupPrefix(string uri)
```

8 Summary

9 Gets the prefix associated with the given namespace URI.

10 Parameters

11
12

Parameter	Description
uri	A System.String containing the namespace to resolve for the prefix.

13
14
15

14 Return Value

16 A **System.String** containing the matching prefix. If there is no
17 mapped prefix, this method returns **System.String.Empty**. If a **null**
18 value is supplied, then **null** is returned.

19 Behaviors

20 As described above.

21 How and When to Override

22 Override this method to customize the behavior of this method in
23 types derived from the **System.Xml.XmlNamespaceManager** class.

24

1 XmlNamespaceManager.PopScope() 2 Method

```
3 [ILASM]  
4 .method public hidebysig virtual bool PopScope()  
5 [C#]  
6 public virtual bool PopScope()
```

7 Summary

8 Pops a namespace scope off the stack.

9 Return Value

10

11 A **System.Boolean** where **true** indicates a namespace scope was
12 popped off the stack; **false** indicates there were no namespaces
13 scopes to pop.

14 Behaviors

15 As described above.

16 Default

17 This method removes all the namespaces that were added to the
18 collection (by calling
19 **System.Xml.XmlNamespaceManager.AddNamespace**) since the
20 last call to the **System.Xml.XmlNamespaceManager.PopScope**
21 method.

22 How and When to Override

23 Override this method to customize the behavior of this method in
24 types derived from the **System.Xml.XmlNamespaceManager** class.

25

1 XmlNamespaceManager.PushScope() 2 Method

```
3 [ILASM]  
4 .method public hidebysig virtual void PushScope()  
5 [C#]  
6 public virtual void PushScope()
```

7 Summary

8 Pushes a namespace scope onto the stack.

9 Behaviors

10 As described above.

11 Default

12 This method associates the pushed namespace scope with all the
13 namespaces added to the collection (by calling
14 **System.Xml.XmlNamespaceManager.AddNamespace**) since the
15 last call to the **System.Xml.XmlNamespaceManager.PushScope**
16 method.

17 How and When to Override

18 Override this method to customize the behavior of this method in
19 types derived from the **System.Xml.XmlNamespaceManager** class.

20

XmlNamespaceManager.RemoveNamespace(System.String, System.String) Method

```
[ILASM]
.method public hidebysig virtual void
RemoveNamespace(string prefix, string uri)

[C#]
public virtual void RemoveNamespace(string prefix, string
uri)
```

Summary

Removes the specified namespace associated with the specified prefix from the namespace collection maintained by the current instance.

Parameters

Parameter	Description
<i>prefix</i>	A System.String specifying the prefix.
<i>uri</i>	A System.String specifying the namespace to remove.

Behaviors

As described above.

Default

The namespace removed is from the current namespace scope. Namespaces outside the current scope are ignored. If *prefix* or *uri* does not exist in the collection, this method simply returns.

How and When to Override

Override this method to customize the behavior of this method in types derived from the **System.Xml.XmlNamespaceManager** class.

Exceptions

Exception	Condition
System.ArgumentNullException	<i>prefix</i> is null , or <i>uri</i> is null .

1 XmlNamespaceManager.DefaultNamespace 2 e Property

```
3 [ILASM]  
4 .property string DefaultNamespace { public hideby sig  
5 virtual specialname string get_DefaultNamespace() }  
6  
7 [C#]  
8 public virtual string DefaultNamespace { get; }
```

8 Summary

9 Gets the namespace URI for the default namespace.

10 Property Value

11

12 A **System.String** containing the namespace URI for the default
13 namespace, or **System.String.Empty** if there is no default
14 namespace.

15 Description

16 This property is read-only.

17 Behaviors

18 As described above.

19 Default

20 This method is equivalent to calling
21 **System.Xml.XmlNamespaceManager.LookupNamespace**
22 (**System.String.Empty**).

23 How and When to Override

24 Override this property to customize the behavior of this property in
25 types derived from the **System.Xml.XmlNamespaceManager** class.

26

1 XmlNamespaceManager.NameTable 2 Property

```
3 [ILASM]  
4 .property class System.Xml.XmlNameTable NameTable { public  
5 hidebysig specialname instance class  
6 System.Xml.XmlNameTable get_NameTable() }  
  
7 [C#]  
8 public XmlNameTable NameTable { get; }
```

9 Summary

10 Gets the name table used by the current instance to look up prefixes
11 and namespace URIs.

12 Property Value

13

14 The **System.Xml.XmlNameTable** used by the current instance.

15 Description

16 This property is read-only.

17