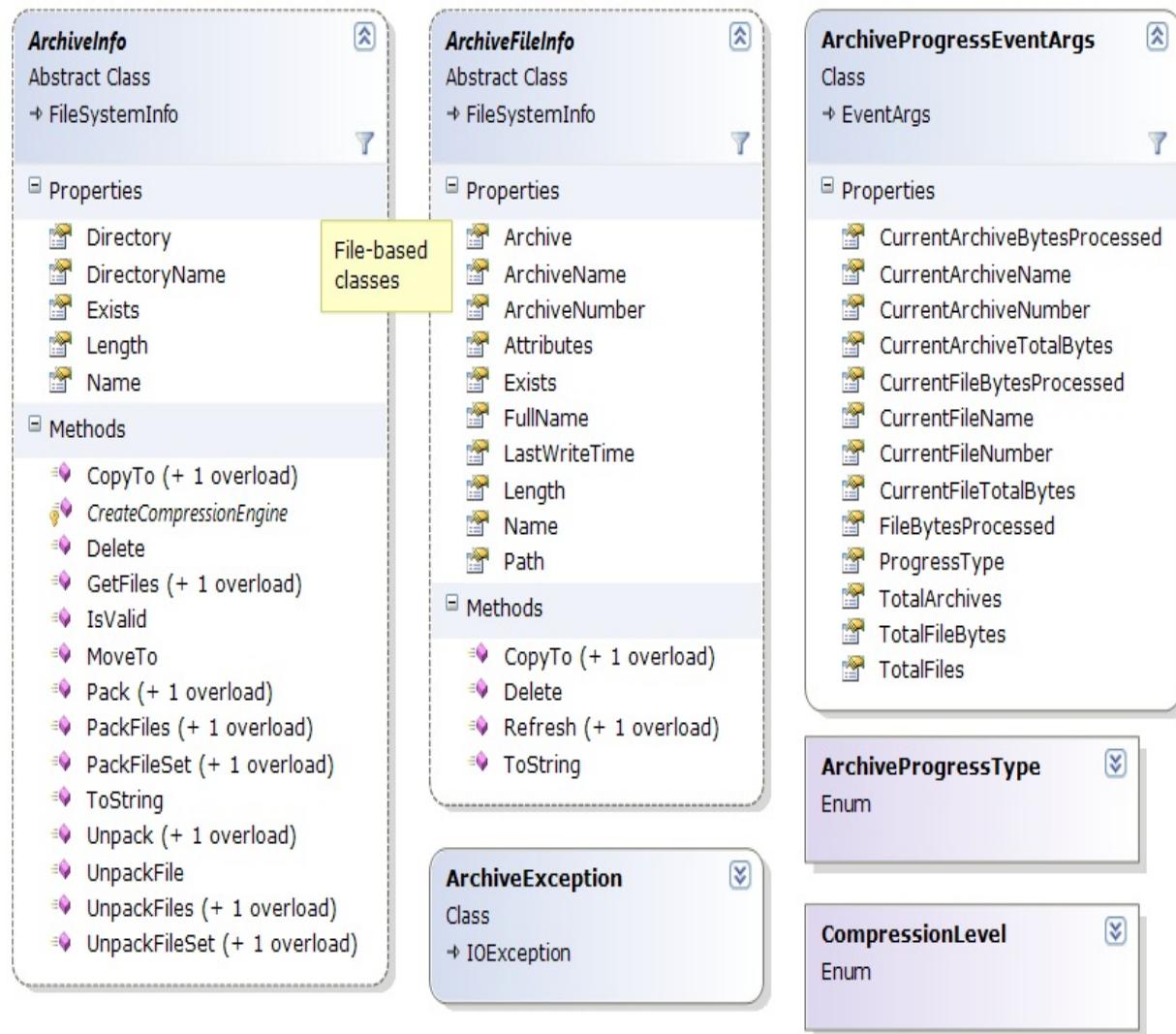
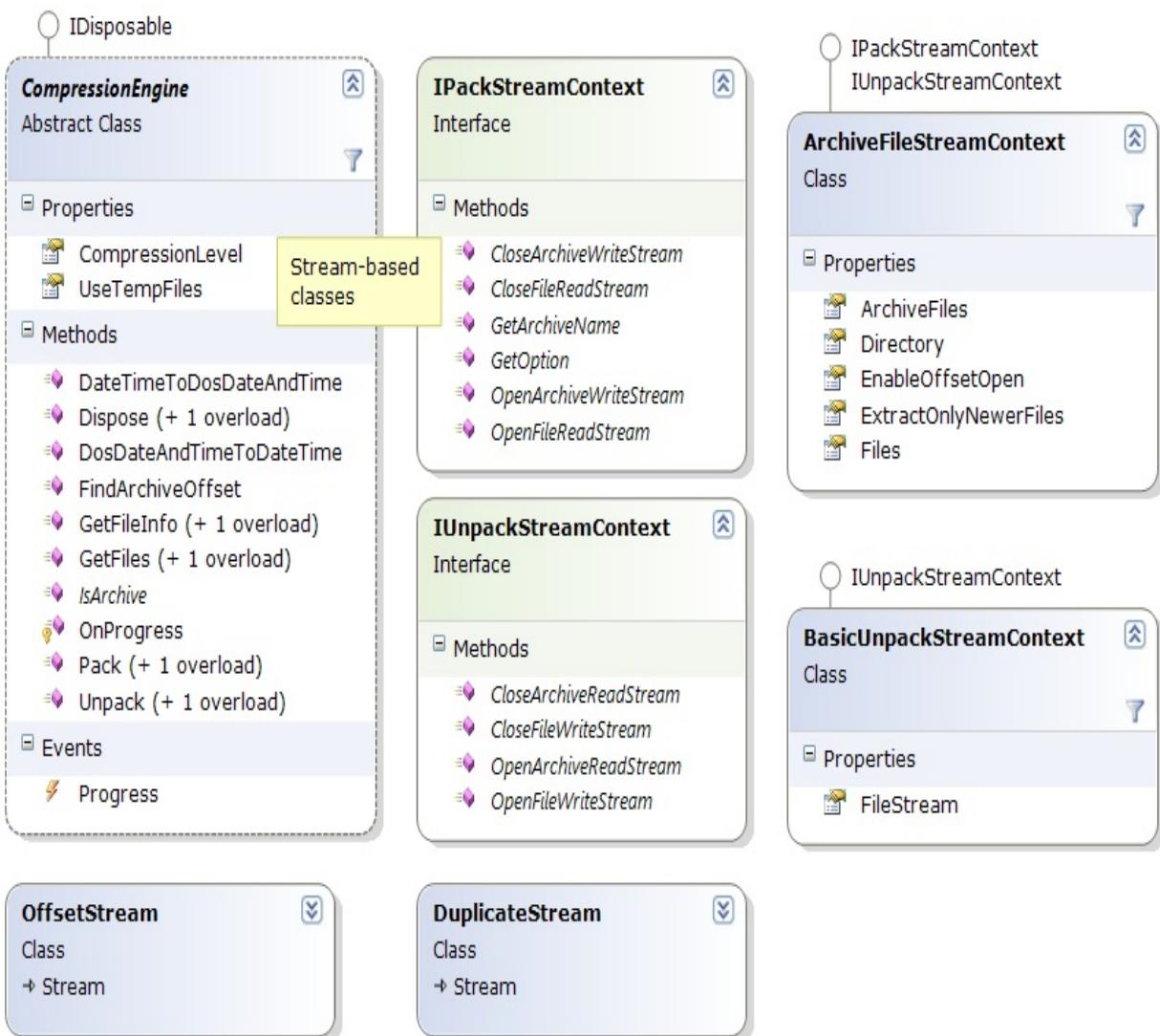


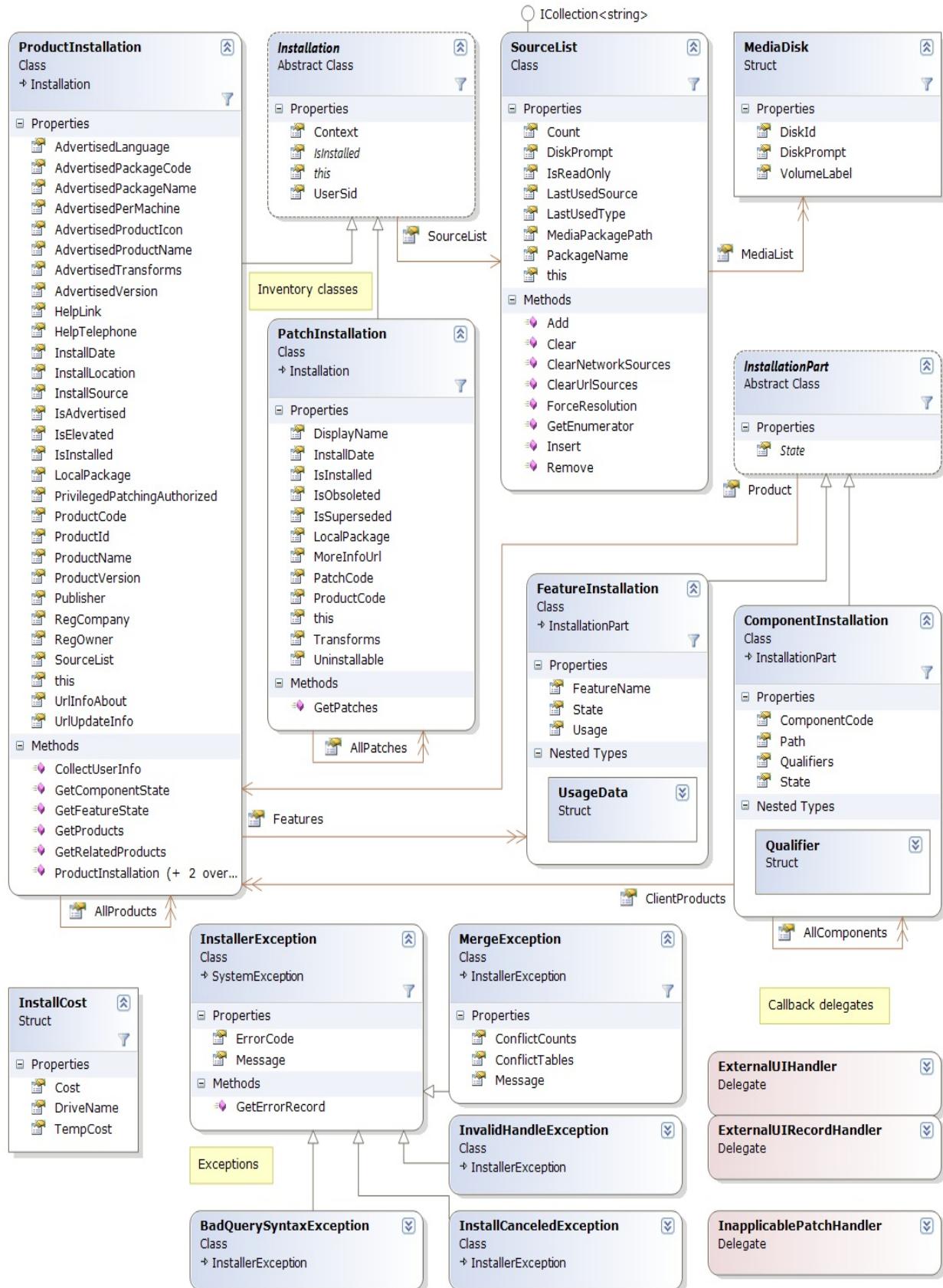
# Microsoft.Deployment.Compression Namespace





## **Microsoft.Deployment.WindowsInstaller Namespace**





<b>Installer</b> Static Class	API enums	Database column enums
<ul style="list-style-type: none"> <li>Properties           <ul style="list-style-type: none"> <li>RebootInitiated</li> <li>RebootRequired</li> <li>Version</li> </ul> </li> <li>Methods           <ul style="list-style-type: none"> <li>AdvertiseProduct</li> <li>AdvertiseScript</li> <li>ApplyMultiplePatches</li> <li>ApplyPatch (+ 1 overload)</li> <li>ConfigureFeature</li> <li>ConfigureProduct</li> <li>DetermineApplicablePatches (+ 1 overload)</li> <li>EnableLog (+ 1 overload)</li> <li>ExtractPatchXmlData</li> <li>GenerateAdvertiseScript (+ 1 overload)</li> <li>GetErrorMessage (+ 3 overloads)</li> <li>GetFileHash</li> <li>GetFileLanguage</li> <li>GetFileVersion</li> <li>GetPatchFileList</li> <li>GetProductInfoFromScript</li> <li>GetShortcutTarget</li> <li>InstallMissingComponent</li> <li>InstallMissingFile</li> <li>InstallProduct</li> <li>NotifySidChange</li> <li>OpenPackage (+ 1 overload)</li> <li>OpenProduct</li> <li>ProcessAdvertiseScript</li> <li>ProvideAssembly</li> <li>ProvideComponent</li> <li>ProvideQualifiedComponent</li> <li>ReinstallFeature</li> <li>ReinstallProduct</li> <li>RemovePatches</li> <li>SetExternalUI (+ 1 overload)</li> <li>SetInternalUI (+ 1 overload)</li> <li>UseFeature</li> <li>VerifyPackage</li> </ul> </li> </ul>	<b>ActionResult</b> Enum  <b>DatabaseOpenMode</b> Enum  <b>InstallLogModes</b> Enum  <b>InstallMessage</b> Enum  <b>InstallMode</b> Enum  <b>InstallRunMode</b> Enum  <b>InstallState</b> Enum  <b>InstallType</b> Enum  <b>InstallUIOptions</b> Enum  <b>MessageResult</b> Enum  <b>ReinstallModes</b> Enum  <b>TransformErrors</b> Enum  <b>TransformValidations</b> Enum  <b>UserContexts</b> Enum  <b>ValidationError</b> Enum  <b>ViewModifyMode</b> Enum	<b>ControlAttributes</b> Enum  <b>CustomActionTypes</b> Enum  <b>ComponentAttributes</b> Enum  <b>DialogAttributes</b> Enum  <b>FeatureAttributes</b> Enum  <b>FileAttributes</b> Enum  <b>IniFileAction</b> Enum  <b>LocatorTypes</b> Enum  <b>RegistryRoot</b> Enum  <b>RemoveFileModes</b> Enum  <b>ServiceAttributes</b> Enum  <b>ServiceControlEvents</b> Enum  <b>TextStyles</b> Enum  <b>UpgradeAttributes</b> Enum
<b>ShortcutTarget</b> Struct		
<ul style="list-style-type: none"> <li>Properties           <ul style="list-style-type: none"> <li>ComponentCode</li> <li>Feature</li> <li>ProductCode</li> </ul> </li> </ul>		

Deployment Tools

# Foundation **Microsoft.Deployment.Compression** Namespace

Deployment Tools Foundation Namespaces ►

## Microsoft.Deployment.Compression

Framework for archive packing and unpacking.

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
namespace Microsoft.Deployment.Compression
```

```
Namespace Microsoft.Deployment.Compression
```

```
namespace Microsoft.Deployment.Compression
```

### Types

All Types

Classes

Interfaces

Enumerations

Icon	Type	Description
	<a href="#">ArchiveException</a>	Base exception class for compression operations. Compression libraries should derive subclass exceptions with more specific error information relevant to the file format.
	<a href="#">ArchiveFileInfo</a>	Abstract object representing a compressed file within an archive provides operations for getting the file properties and unpacking the file.

	<a href="#">ArchiveFileStreamContext</a>	Provides a basic implementation of the archive pack and unpack stream context interfaces, based on a list of archive files, a default directory, and an optional mapping from internal to external file paths
	<a href="#">ArchiveInfo</a>	Abstract object representing a compressed archive on disk; provides access to file-based operations on the archive.
	<a href="#">ArchiveProgressEventArgs</a>	Contains the data reported in an archive progress event.
	<a href="#">ArchiveProgressType</a>	The type of progress event.
	<a href="#">BasicUnpackStreamContext</a>	Stream context used to extract a single file from an archive into a memory stream.
	<a href="#">CargoStream</a>	Wraps a source stream and carries additional items that are disposed when the stream is closed.
	<a href="#">CompressionEngine</a>	Base class for an engine capable of packing and unpacking a particular compressed file format.

 <b>CompressionLevel</b>	Specifies the compression level ranging from minimum compression to maximum compression, or no compression at all.
 <b>DuplicateStream</b>	Duplicates a source stream by maintaining a separate position.
 <b>IPackStreamContext</b>	This interface provides the methods necessary for the <a href="#">CompressionEngine</a> to open and close streams for archives and files. The implementor of this interface can use any kind of logic to determine what kind of streams to open and where.
 <b>IUnpackStreamContext</b>	This interface provides the methods necessary for the <a href="#">CompressionEngine</a> to open and close streams for archives and files. The implementor of this interface can use any kind of logic to determine what kind of streams to open and where
 <b>OffsetStream</b>	Wraps a source stream and offsets all read/write/seek calls by a given value.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

# Deployment Tools FoundationArchiveException Class

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveException

C#

Base exception class for compression operations. Compression libraries should derive subclass exceptions with more specific error information relevant to the file format.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
[SerializableAttribute]
public class ArchiveException : IOException
```

```
<SerializableAttribute>
Public Class ArchiveException
    Inherits IOException
```

```
[SerializableAttribute]
public ref class ArchiveException : public IOException
```

## Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon

Member

Description



ArchiveException(String, Exception)

Creates a new ArchiveException with a specified error message and a reference to the inner exception that is the cause of this exception.

	<a href="#">ArchiveException(String)</a>	Creates a new ArchiveException with a specified error message.
	<a href="#">ArchiveException()</a>	Creates a new ArchiveException.
	<a href="#">ArchiveException(SerializationInfo, StreamingContext)</a>	Initializes a new instance of the ArchiveException class with serialized data.
	<a href="#">Data</a>	Gets a collection of key/value pairs that provide additional, user-defined information about the exception.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetBaseException()</a>	When overridden in a derived class, returns the <a href="#">Exception</a> that is the root cause of one

		or more subsequent exceptions.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetObjectData(SerializationInfo, StreamingContext)</a>	When overridden in a derived class, sets the <a href="#">SerializationInfo</a> with information about the exception.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">GetType()</a>	Gets the runtime type of the current instance.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">HelpLink</a>	Gets or sets a link to the help file associated with this exception.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">HResult</a>	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">InnerException</a>	Gets the <a href="#">Exception</a> instance that caused the current exception.

		(Inherited from <a href="#">Exception</a> .)
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Message</a>	Gets a message that describes the current exception.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">Source</a>	Gets or sets the name of the application or the object that causes the error.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">StackTrace</a>	Gets a string representation of the frames on the call stack at the time the current exception was thrown.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">TargetSite</a>	Gets the method that throws the current exception.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">ToString()</a>	Creates and returns a string representation of the current exception.  (Inherited from <a href="#">Exception</a> .)

## Inheritance Hierarchy

[Object](#)

  └ [Exception](#)

## SystemException

### └ IOException

#### └ ArchiveException

##### └ CabException

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ArchiveException Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveException ►

**ArchiveException()**

C# ▾

### Members

Icon	Member	Description
≡	<a href="#">ArchiveException(String, Exception)</a>	Creates a new ArchiveException with a specified error message and a reference to the inner exception that is the cause of this exception.
≡	<a href="#">ArchiveException(String)</a>	Creates a new ArchiveException with a specified error message.
≡	<a href="#">ArchiveException()</a>	Creates a new ArchiveException.
💡	<a href="#">ArchiveException(SerializationInfo, StreamingContext)</a>	Initializes a new instance of the ArchiveException class with serialized data.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ArchiveException Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveException ►

**ArchiveException()**

C# ▾

Creates a new ArchiveException.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public ArchiveException()
```

```
Public Sub New
```

```
public:  
ArchiveException()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ArchiveException Constructor (info, context)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveException ►  
ArchiveException(SerializationInfo, StreamingContext)

C# ▾

Initializes a new instance of the ArchiveException class with serialized data.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected ArchiveException(  
    SerializationInfo info,  
    StreamingContext context  
)
```

```
Protected Sub New ( _  
    info As SerializationInfo, _  
    context As StreamingContext _  
)
```

```
protected:  
ArchiveException(  
    SerializationInfo^ info,  
    StreamingContext context  
)
```

### Parameters

#### info (SerializationInfo)

The SerializationInfo that holds the serialized object data about the exception being thrown.

#### context (StreamingContext)

The StreamingContext that contains contextual information about the source or destination.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:  
Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ArchiveException Constructor (message)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveException ►  
ArchiveException(String)

C#

Creates a new ArchiveException with a specified error message.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public ArchiveException(  
    string message  
)
```

```
Public Sub New ( _  
    message As String _  
)
```

```
public:  
ArchiveException(  
    String^ message  
)
```

### Parameters

#### message (String)

The message that describes the error.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ArchiveException Constructor (message, innerException)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveException ►

ArchiveException(String, Exception)

C#

Creates a new ArchiveException with a specified error message and a reference to the inner exception that is the cause of this exception.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public ArchiveException(  
    string message,  
    Exception innerException  
)
```

```
Public Sub New ( _  
    message As String, _  
    innerException As Exception _  
)
```

```
public:  
ArchiveException(  
    String^ message,  
    Exception^ innerException  
)
```

### Parameters

#### message (String)

The message that describes the error.

#### innerException (Exception)

The exception that is the cause of the current exception. If the innerException parameter is not a null reference (Nothing in Visual Basic), the current exception is raised in a catch block that handles

the inner exception.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools FoundationArchiveFileInfo Class

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo

C#

Abstract object representing a compressed file within an archive; provides operations for getting the file properties and unpacking the file.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
[SerializableAttribute]
public abstract class ArchiveFileInfo : FileSystemIn
```

```
<SerializableAttribute> _
Public MustInherit Class ArchiveFileInfo _
    Inherits FileSystemInfo
```

```
[SerializableAttribute]
public ref class ArchiveFileInfo abstract : public F
```

## Members

All Members

Constructors

Methods

Properties

Fields

Public

Instance

Declared

Protected

Static

Inherited

Icon	Member	Description
💡	ArchiveFileInfo(ArchiveInfo, String)	Creates a new ArchiveFileInfo object representing a file within an archive at the specified path.
💡	ArchiveFileInfo(String, Int32, FileAttributes, DateTime, Int64)	Creates a new ArchiveFileInfo object with the parameters specified; used by the <code>ReadMetadata</code> method for reading the metadata out of an archive.

	<a href="#">ArchiveFileInfo(SerializationInfo, StreamingContext)</a>	Initializes a new instance of the class with serialized data.
	<a href="#">Archive</a>	Gets or sets the archive that contains this file.
	<a href="#">ArchiveName</a>	Gets the full path of the archive that contains this file.
	<a href="#">ArchiveNumber</a>	Gets the number of the archive that contains this file.
	<a href="#">Attributes</a>	Gets the attributes of the file.
	<a href="#">CopyTo(String)</a>	Extracts the file.
	<a href="#">CopyTo(String, Boolean)</a>	Extracts the file, optionally overwriting an existing file.
	<a href="#">CreateObjRef(Type)</a>	Creates an object that contains information required to generate a proxy object to communicate with a remote object.  (Inherited from <a href="#">MarshalByRefObject</a> )
	<a href="#">CreationTime</a>	Gets or sets the creation time of the <a href="#">FileSystemInfo</a> object.  (Inherited from <a href="#">FileSystemInfo</a> )
	<a href="#">CreationTimeUtc</a>	Gets or sets the creation time, in UTC, of the <a href="#">FileSystemInfo</a> object.  (Inherited from <a href="#">FileSystemInfo</a> )

		universal time (UTC), of the current <a href="#">FileSystemInfo</a> object.
		(Inherited from <a href="#">FileSystemInfo</a> .)
≡	<a href="#">Delete()</a>	Deletes the file. NOT SUPPORTED.  (Overrides <a href="#">FileSystemInfo.Delete()</a> .)
≡	<a href="#">Equals(Object)</a>	Determines whether the specified object is equal to the current <a href="#">Object</a> .
		(Inherited from <a href="#">Object</a> .)
📄	<a href="#">Exists</a>	Checks if the file exists within the current directory.
		(Overrides <a href="#">FileSystemInfo.Exists()</a> .)
📄	<a href="#">Extension</a>	Gets the string representing the extension of the file.
		(Inherited from <a href="#">FileSystemInfo</a> .)
💡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.
		(Inherited from <a href="#">Object</a> .)
📄	<a href="#">FullName</a>	Gets the full path to the file.
		(Overrides <a href="#">FileSystemInfo.FullName()</a> .)
💡	<a href="#">FullPath</a>	Represents the fully qualified path to the directory or file.
		(Inherited from <a href="#">FileSystemInfo</a> .)
≡	<a href="#">GetHashCode()</a>	Serves as a hash function for a <a href="#">Dictionary</a> . <a href="#">GetHashCode()</a> is suitable for use in hash-based algorithms and data structures.
		(Inherited from <a href="#">Object</a> .)
≡	<a href="#">GetLifetimeService()</a>	Retrieves the current lifetime service.

		controls the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">GetObjectData(SerializationInfo, StreamingContext)</a>	Sets the <a href="#">SerializationInfo</a> with information about the archive.  (Overrides <a href="#">FileSystemInfo.GetObjectData(StreamingContext)</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
	<a href="#">InitializeLifetimeService()</a>	Obtains a lifetime service object that controls the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">LastAccessTime</a>	Gets or sets the time the current file was last accessed.  (Inherited from <a href="#">FileSystemInfo</a> .)
	<a href="#">LastAccessTimeUtc</a>	Gets or sets the time, in coordinated universal time (UTC), that the current file was last accessed.  (Inherited from <a href="#">FileSystemInfo</a> .)
	<a href="#">LastWriteTime</a>	Gets the last modification time of the current file.  (Inherited from <a href="#">FileSystemInfo</a> .)
	<a href="#">LastWriteTimeUtc</a>	Gets or sets the time, in coordinated universal time (UTC), when the current file was last written to.  (Inherited from <a href="#">FileSystemInfo</a> .)
	<a href="#">Length</a>	Gets the uncompressed size of the current file.

	<a href="#">MemberwiseClone(Boolean)</a>	Creates a shallow copy of the current <a href="#">MarshalByRefObject</a> object.  (Inherited from <a href="#">MarshalByRefObject</a> )
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">MarshalByRefObject</a> object.  (Inherited from <a href="#">Object</a> .)
	<a href="#">Name</a>	Gets the name of the file.  (Overrides <a href="#">FileSystemInfo.Name</a> )
	<a href="#">OpenRead()</a>	Opens the archive file for reading, without actually extracting the file to disk.
	<a href="#">OpenText()</a>	Opens the archive file reading the file in text encoding without actually extracting the file to disk.
	<a href="#">OriginalPath</a>	The path originally specified by the file, whether relative or absolute.  (Inherited from <a href="#">FileSystemInfo</a> )
	<a href="#">Path</a>	Gets the internal path of the file.
	<a href="#">Refresh()</a>	Refreshes the attributes and other information about the file, by reading information from the archive.
	<a href="#">Refresh(ArchiveFileInfo)</a>	Refreshes the information in this file with new data retrieved from an archive.
	<a href="#">ToString()</a>	Gets the full path to the file.

(Overrides [Object.ToString\(\)](#).)

## Inheritance Hierarchy

[Object](#)

 [MarshalByRefObject](#)

 [FileSystemInfo](#)

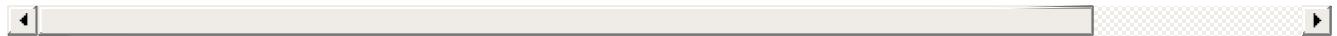
 **ArchiveFileInfo**

 [CabFileInfo](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## ArchiveFileInfo Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ►

ArchiveFileInfo()

C# ▾

### Members

Icon	Member	Description
💡	<a href="#">ArchiveFileInfo(ArchiveInfo, String)</a>	Creates a new ArchiveFileInfo object representing a file within an archive in a specified path.
💡	<a href="#">ArchiveFileInfo(String, Int32, FileAttributes, DateTime, Int64)</a>	Creates a new ArchiveFileInfo object with all parameters specified; used by subclasses when reading the metadata out of an archive.
💡	<a href="#">ArchiveFileInfo(SerializationInfo, StreamingContext)</a>	Initializes a new instance of the ArchiveFileInfo class with serialized data.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ArchiveFileInfo Constructor (archiveInfo, filePath)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveFileInfo ►  
ArchiveFileInfo(ArchiveInfo, String)

C# ▾

Creates a new ArchiveFileInfo object representing a file within an archive in a specified path.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected ArchiveFileInfo(  
    ArchiveInfo archiveInfo,  
    string filePath  
)
```

```
Protected Sub New ( _  
    archiveInfo As ArchiveInfo, _  
    filePath As String _  
)
```

```
protected:  
ArchiveFileInfo(  
    ArchiveInfo^ archiveInfo,  
    String^ filePath  
)
```

### Parameters

#### archiveInfo (ArchiveInfo)

An object representing the archive containing the file.

#### filePath (String)

The path to the file within the archive. Usually, this is a simple file name, but if the archive contains a directory structure this may include the directory.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:  
Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ArchiveFileInfo Constructor (info, context)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveFileInfo ►  
ArchiveFileInfo(SerializationInfo, StreamingContext)

C# ▾

Initializes a new instance of the ArchiveFileInfo class with serialized dat

### Declaration Syntax

C#

Visual Basic

Visual C++

```
protected ArchiveFileInfo(  
    SerializationInfo info,  
    StreamingContext context  
)
```

```
Protected Sub New (  
    info As SerializationInfo, _  
    context As StreamingContext _  
)
```

```
protected:  
ArchiveFileInfo(  
    SerializationInfo^ info,  
    StreamingContext context  
)
```

### Parameters

#### *info* (SerializationInfo)

The SerializationInfo that holds the serialized object data about the exception being thrown.

#### *context* (StreamingContext)

The StreamingContext that contains contextual information about the source or destination.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:  
Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ArchiveFileInfo Constructor (filePath, archiveNumber, attributes, lastWriteTime, length)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ►

ArchiveFileInfo(String, Int32, FileAttributes, DateTime, Int64)

C#

Creates a new ArchiveFileInfo object with all parameters specified; used by subclasses when reading the metadata out of an archive.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
protected ArchiveFileInfo(  
    string filePath,  
    int archiveNumber,  
    FileAttributes attributes,  
    DateTime lastWriteTime,  
    long length  
)
```

```
Protected Sub New ( _  
    filePath As String, _  
    archiveNumber As Integer, _  
    attributes As FileAttributes, _  
    lastWriteTime As DateTime, _  
    length As Long _  
)
```

```
protected:  
ArchiveFileInfo(  
    String^ filePath,  
    int archiveNumber,  
    FileAttributes attributes,  
    DateTime lastWriteTime,  
    long long length
```

)

## - Parameters

### ***filePath* ([String](#))**

The internal path and name of the file in the archive.

### ***archiveNumber* ([Int32](#))**

The archive number where the file starts.

### ***attributes* ([FileAttributes](#))**

The stored attributes of the file.

### ***lastWriteTime* ([DateTime](#))**

The stored last write time of the file.

### ***length* ([Int64](#))**

The uncompressed size of the file.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Archive Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ► Archive

C# ▾

Gets or sets the archive that contains this file.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public ArchiveInfo Archive { get; internal set; }
```

```
Public Property Archive As ArchiveInfo
```

```
public:  
property ArchiveInfo^ Archive {  
    ArchiveInfo^ get ();  
    void set (ArchiveInfo^ value);  
}
```

### Value

The ArchiveInfo instance that retrieved this file information -- this may be null if the ArchiveFileInfo object was returned directly from a stream.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ArchiveName Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ►

### ArchiveName

C# ▾

Gets the full path of the archive that contains this file.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string ArchiveName { get; }
```

```
Public ReadOnly Property ArchiveName As String
```

```
public:  
property String^ ArchiveName {  
    String^ get ();  
}
```

#### Value

The full path of the archive that contains this file.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ArchiveNumber Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ►

### ArchiveNumber

C# ▾

Gets the number of the archive where this file starts.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public int ArchiveNumber { get; }
```

```
Public ReadOnly Property ArchiveNumber As Integer
```

```
public:  
property int ArchiveNumber {  
    int get();  
}
```

#### Value

The number of the archive where this file starts.

#### Remarks

A single archive or the first archive in a chain is numbered 0.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Attributes Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ► Attributes

C# ▾

Gets the attributes of the file.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public FileAttributes Attributes { get; }
```

```
Public ReadOnly Property Attributes As FileAttribute
```

```
public:  
property FileAttributes Attributes {  
    FileAttributes get ();  
}
```

### Value

The attributes of the file as stored in the archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **CopyTo Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ► **CopyTo()**

C# ▾

**Members**

Icon	Member	Description
	<a href="#">CopyTo(String)</a>	Extracts the file.
	<a href="#">CopyTo(String, Boolean)</a>	Extracts the file, optionally overwriting any existing file.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **CopyTo Method (destFileName)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ►

### **CopyTo(String)**

C# ▾

Extracts the file.

#### [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public void CopyTo(  
    string destFileName  
)
```

```
Public Sub CopyTo ( _  
    destFileName As String _  
)
```

```
public:  
void CopyTo(  
    String^ destFileName  
)
```

#### [-] Parameters

##### **destFileName (String)**

The destination path where the file will be extracted.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CopyTo Method (destFileName, overwrite)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveFileInfo ►  
**CopyTo(String, Boolean)**

C# ▾

Extracts the file, optionally overwriting any existing file.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void CopyTo(  
    string destFileName,  
    bool overwrite  
)
```

```
Public Sub CopyTo ( _  
    destFileName As String, _  
    overwrite As Boolean _  
)
```

```
public:  
void CopyTo(  
    String^ destFileName,  
    bool overwrite  
)
```

### Parameters

#### *destFileName* (String)

The destination path where the file will be extracted.

#### *overwrite* (Boolean)

If true, *destFileName* will be overwritten if it exists.

### Exceptions

Exception

Condition

[IOException](#)

*overwrite* is false and *destFileName* exists.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Delete Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ► Delete()

C# ▾

Deletes the file. NOT SUPPORTED.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override void Delete()
```

```
Public Overrides Sub Delete
```

```
public:  
virtual void Delete() override
```

### - Exceptions

Exception	Condition
NotSupportedException	Files cannot be deleted from an existing archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# Exists Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ► Exists

C# ▾

Checks if the file exists within the archive.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public override bool Exists { get; }
```

```
Public Overrides ReadOnly Property Exists As Boolean
```

```
public:  
virtual property bool Exists {  
    bool get () override;  
}
```

## Value

True if the file exists, false otherwise.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## FullName Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ► FullName

C# ▾

Gets the full path to the file.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public override string FullName { get; }
```

```
Public Overrides ReadOnly Property FullName As String
```

```
public:  
virtual property String^ FullName {  
    String^ get () override;  
}
```

### - Value

The full path to the file, including the full path to the archive, the internal path in the archive, and the file name.

### - Remarks

For example, the path "C:\archive.cab\file.txt" refers to file "file.txt" inside the archive "archive.cab".

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetObjectData Method (info, context)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveFileInfo ►  
**GetObjectData(SerializationInfo, StreamingContext)**

C#

Sets the SerializationInfo with information about the archive.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public override void GetObjectData(
    SerializationInfo info,
    StreamingContext context
)
```

```
Public Overrides Sub GetObjectData ( _
    info As SerializationInfo, _
    context As StreamingContext _
)
```

```
public:
virtual void GetObjectData(
    SerializationInfo^ info,
    StreamingContext context
) override
```

[-] **Parameters**

**info (SerializationInfo)**

The SerializationInfo that holds the serialized object data.

**context (StreamingContext)**

The StreamingContext that contains contextual information about the source or destination.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## LastWriteTime Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ►

### LastWriteTime

C# ▾

Gets the last modification time of the file.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public DateTime LastWriteTime { get; }
```

```
Public ReadOnly Property LastWriteTime As DateTime
```

```
public:  
property DateTime LastWriteTime {  
    DateTime get();  
}
```

#### Value

The last modification time of the file as stored in the archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Length Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ► Length

C# ▾

Gets the uncompressed size of the file.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public long Length { get; }
```

```
Public Readonly Property Length As Long
```

```
public:  
property long long Length {  
    long long get ();  
}
```

## Value

The uncompressed size of the file in bytes.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# Name Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ► Name

C# ▾

Gets the name of the file.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public override string Name { get; }
```

```
Public Overrides ReadOnly Property Name As String
```

```
public:  
virtual property String^ Name {  
    String^ get () override;  
}
```

## Value

The name of the file, not including any path.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **OpenRead Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ► **OpenRead()**

C# ▾

Opens the archive file for reading without actually extracting the file to disk.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public Stream OpenRead()
```

```
Public Function OpenRead As Stream
```

```
public:  
Stream^ OpenRead()
```

### [-] Return Value

A stream for reading directly from the packed file. Like any stream this should be closed/disposed as soon as it is no longer needed.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **OpenText Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ► **OpenText()**

C# ▾

Opens the archive file reading text with UTF-8 encoding without actually extracting the file to disk.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public StreamReader OpenText()
```

```
Public Function OpenText As StreamReader
```

```
public:  
StreamReader^ OpenText()
```

### [-] Return Value

A reader for reading text directly from the packed file. Like any reader it should be closed/disposed as soon as it is no longer needed.

### [-] Remarks

To open an archived text file with different encoding, use the [OpenRead](#) method and pass the returned stream to one of the [StreamReader](#) constructor overloads.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# Path Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ► Path

C# ▾

Gets the internal path of the file in the archive.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public string Path { get; }
```

```
Public Readonly Property Path As String
```

```
public:  
property String^ Path {  
    String^ get ();  
}
```

## Value

The internal path of the file in the archive, not including the file name.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Refresh Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ► Refresh()

C# ▾

■ Members

Icon	Member	Description
	<a href="#">Refresh()</a>	Refreshes the attributes and other cached information about the file, by re-reading the information from the archive.
	<a href="#">Refresh(ArchiveFileInfo)</a>	Refreshes the information in this object with new data retrieved from an archive.

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Refresh Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ► Refresh()

C# ▾

Refreshes the attributes and other cached information about the file, by re-reading the information from the archive.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Refresh()
```

```
Public Sub Refresh
```

```
public:  
void Refresh()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Refresh Method (newFileInfo)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ►

## Refresh(ArchiveFileInfo)

C# ▾

Refreshes the information in this object with new data retrieved from an archive.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected virtual void Refresh(  
    ArchiveFileInfo newFileInfo  
)
```

```
Protected Overridable Sub Refresh ( _  
    newFileInfo As ArchiveFileInfo _  
)
```

```
protected:  
virtual void Refresh(  
    ArchiveFileInfo^ newFileInfo  
)
```

### Parameters

#### *newFileInfo* (ArchiveFileInfo)

Fresh instance for the same file just read from the archive.

### Remarks

Subclasses may override this method to refresh subclass fields. However they should always call the base implementation first.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **ToString** Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileInfo ► **ToString()**

C# ▾

Gets the full path to the file.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override string ToString()
```

```
Public Overrides Function ToString As String
```

```
public:  
virtual String^ ToString() override
```

### Return Value

The same as [FullName](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools FoundationArchiveFileStreamContext Class

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileStreamContext

C# ▾

Provides a basic implementation of the archive pack and unpack stream context interfaces, based on a list of archive files, a default directory, and an optional mapping from internal to external file paths.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public class ArchiveFileStreamContext : IPackStreamC  
    IUnpackStreamContext
```

```
Public Class ArchiveFileStreamContext _  
    Implements IPackStreamContext, IUnpackStream
```

```
public ref class ArchiveFileStreamContext : IPackStr  
    IUnpackStreamContext
```

## Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declare

Inherit

Icon	Member	Description
	ArchiveFileStreamContext(String)	Creates a new ArchiveFileStreamContext with a archive file and a default directory or file mapping.
	ArchiveFileStreamContext(String,	Creates a new

	<code>String, IDictionary&lt;String, String&gt;</code>	ArchiveFileStreamContext with a archive file, default directory and mapping from internal to external file paths.
≡	<code>ArchiveFileStreamContext(IList&lt;String&gt;, String, IDictionary&lt;String, String&gt;)</code>	Creates a new ArchiveFileStreamContext with a list of archive files, a default directory and a mapping from internal to external file paths.
	<code>ArchiveFiles</code>	Gets or sets the list of archive files that are created or extracted.
≡	<code>CloseArchiveReadStream(Int32, String, Stream)</code>	Closes a stream where archive was read.
≡	<code>CloseArchiveWriteStream(Int32, String, Stream)</code>	Closes a stream where archive package was written.
≡	<code>CloseFileReadStream(String, Stream)</code>	Closes a stream that has been used to read a file.
≡	<code>CloseFileWriteStream(String, Stream, FileAttributes, DateTime)</code>	Closes a stream where extracted file was written.

	<a href="#">Directory</a>	Gets or sets the default root directory where file are located.
	<a href="#">EnableOffsetOpen</a>	Gets or sets a flag that enables creating or extracting an archive at an offset within an existing file. (This is typically used to open archive-based self-extracting packages.)
	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">ExtractOnlyNewerFiles</a>	Gets or sets a flag that can prevent extracted files from overwriting newer files that already exist.
	<a href="#">Files</a>	Gets or sets the mapping from internal file paths to external file paths.
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed.

		garbage collection.  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">GetArchiveName(Int32)</a>	Gets the name of the archive with a specified number.
≡	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hash algorithms and data structures like a hash table.  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">GetOption(String, Object[])</a>	Gets extended parameter information specific to the compression format being used.
≡	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">OpenArchiveReadStream(Int32, String, CompressionEngine)</a>	Opens the archive stream for reading.
≡	<a href="#">OpenArchiveWriteStream(Int32, String,</a>	Opens a stream for

	<code>Boolean, CompressionEngine)</code>	writing an archive.
≡	<code>OpenFileReadStream(String, FileAttributes, DateTime)</code>	Opens a stream to read file that is to be included in an archive.
≡	<code>OpenFileStream(String, Int64, DateTime)</code>	Opens a stream for writing extracted file bytes.
≡	<code>ToString()</code>	Returns a <code>String</code> that represents the current <code>Object</code> .  (Inherited from <code>Object</code> .)

## [-] Remarks

This class can also handle creating or extracting chained archive packages.

## [-] Inheritance Hierarchy

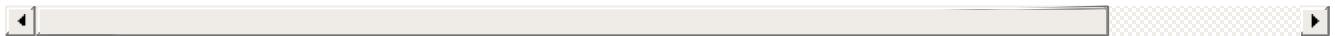
`Object`

└ `ArchiveFileStreamContext`

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



# ArchiveFileStreamContext

## Constructor

Deployment Tools Foundation Namespaces ►  
[Microsoft.Deployment.Compression](#) ► [ArchiveFileStreamContext](#) ►  
**ArchiveFileStreamContext()**

C# ▾

### Members

Icon	Member	Description
	<a href="#">ArchiveFileStreamContext(String)</a>	Creates a new ArchiveFileStreamCont with a archive file and r default directory or file mapping.
	<a href="#">ArchiveFileStreamContext(String, String, IDictionary&lt;String, String&gt;)</a>	Creates a new ArchiveFileStreamCont with a archive file, defa directory and mapping from internal to externa file paths.
	<a href="#">ArchiveFileStreamContext(IList&lt;String&gt;, String, IDictionary&lt;String, String&gt;)</a>	Creates a new ArchiveFileStreamCont with a list of archive file a default directory and mapping from internal t external file paths.

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ArchiveFileStreamContext Constructor (archiveFiles, directory, files)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileStreamContext ►

ArchiveFileStreamContext(IList<String>, String,  
IDictionary<String, String>)

C# ▾

Creates a new ArchiveFileStreamContext with a list of archive files, a default directory and a mapping from internal to external file paths.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public ArchiveFileStreamContext(  
    IList<string> archiveFiles,  
    string directory,  
    IDictionary<string, string> files  
)
```

```
Public Sub New ( _  
    archiveFiles As IList(Of String), _  
    directory As String, _  
    files As IDictionary(Of String, String) _  
)
```

```
public:  
ArchiveFileStreamContext(  
    IList<String>^ archiveFiles,  
    String^ directory,  
    IDictionary<String>^, String^>^ files  
)
```

### Parameters

**archiveFiles (IList<String>)**

A list of paths to archive files that will be created or extracted.

***directory* ([String](#))**

The default root directory where files will be located, optional.

***files* ([IDictionary<String, String>](#))**

A mapping from internal file paths to external file paths, optional.

## ■ Remarks

When creating chained archives, the *archiveFiles* list should include at least enough archives to handle the entire set of input files, based on the maximum archive size that is passed to the [CompressionEngine.Pack\(IPackStreamContext, IEnumerable<String>, Int64\)](#).

If the mapping is not null and a file is not included in the mapping, the file will be skipped.

If the external path in the mapping is a simple file name or relative file path, it will be concatenated onto the default directory, if one was specified.

For more about how the default directory and files mapping are used, see [OpenFileReadStream\(String, FileAttributes, DateTime\)](#) and [OpenFileWriteStream\(String, Int64, DateTime\)](#).

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ArchiveFileStreamContext Constructor (archiveFile)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileStreamContext ►

ArchiveFileStreamContext(String)

C#

Creates a new ArchiveFileStreamContext with a archive file and no default directory or file mapping.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public ArchiveFileStreamContext(  
    string archiveFile  
)
```

```
Public Sub New (  
    archiveFile As String  
)
```

```
public:  
ArchiveFileStreamContext(  
    String^ archiveFile  
)
```

### Parameters

#### archiveFile (String)

The path to a archive file that will be created or extracted.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ArchiveFileStreamContext Constructor (archiveFile, directory, files)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileStreamContext ►

ArchiveFileStreamContext(String, String, IDictionary<String, String>)

C# ▾

Creates a new ArchiveFileStreamContext with a archive file, default directory and mapping from internal to external file paths.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public ArchiveFileStreamContext(  
    string archiveFile,  
    string directory,  
    IDictionary<string, string> files  
)
```

```
Public Sub New ( _  
    archiveFile As String, _  
    directory As String, _  
    files As IDictionary(Of String, String) _  
)
```

```
public:  
ArchiveFileStreamContext(  
    String^ archiveFile,  
    String^ directory,  
    IDictionary<String^, String^>^ files  
)
```

### Parameters

#### archiveFile (String)

The path to a archive file that will be created or extracted.

***directory* ([String](#))**

The default root directory where files will be located, optional.

***files* ([IDictionary<String, String>](#))**

A mapping from internal file paths to external file paths, optional.

## ■ Remarks

If the mapping is not null and a file is not included in the mapping, the file will be skipped.

If the external path in the mapping is a simple file name or relative file path, it will be concatenated onto the default directory, if one was specified.

For more about how the default directory and files mapping are used, see [OpenFileReadStream\(String, FileAttributes, DateTime\)](#) and [OpenFileWriteStream\(String, Int64, DateTime\)](#).

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **ArchiveFiles** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileStreamContext ►

### ArchiveFiles

C# ▾

Gets or sets the list of archive files that are created or extracted.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public IList<string> ArchiveFiles { get; }
```

```
Public ReadOnly Property ArchiveFiles As IList(Of St
```

```
public:  
property IList<String^>^ ArchiveFiles {  
    IList<String^>^ get ();  
}
```

#### Value

The list of archive files that are created or extracted.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## CloseArchiveReadStream Method (archiveNumber, archiveName, stream)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveFileStreamContext ►  
**CloseArchiveReadStream(Int32, String, Stream)**

C# ▾

Closes a stream where an archive was read.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public virtual void CloseArchiveReadStream(  
    int archiveNumber,  
    string archiveName,  
    Stream stream  
)
```

```
Public Overridable Sub CloseArchiveReadStream ( _  
    archiveNumber As Integer, _  
    archiveName As String, _  
    stream As Stream _  
)
```

```
public:  
virtual void CloseArchiveReadStream(  
    int archiveNumber,  
    String^ archiveName,  
    Stream^ stream  
)
```

### - Parameters

#### **archiveNumber (Int32)**

The archive number of the stream to close.

#### **archiveName (String)**

The name of the archive being closed.

***stream* ([Stream](#))**

The stream that was previously returned by [OpenArchiveReadStream\(Int32, String, CompressionEngine\)](#) and is now ready to be closed.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CloseArchiveWriteStream Method (archiveNumber, archiveName, stream)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveFileStreamContext ►  
**CloseArchiveWriteStream(Int32, String, Stream)**

C# ▾

Closes a stream where an archive package was written.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public virtual void CloseArchiveWriteStream(  
    int archiveNumber,  
    string archiveName,  
    Stream stream  
)
```

```
Public Overridable Sub CloseArchiveWriteStream ( _  
    archiveNumber As Integer, _  
    archiveName As String, _  
    stream As Stream _  
)
```

```
public:  
virtual void CloseArchiveWriteStream(  
    int archiveNumber,  
    String^ archiveName,  
    Stream^ stream  
)
```

### Parameters

#### archiveNumber (Int32)

The 0-based index of the archive within the chain.

#### archiveName (String)

The name of the archive that was previously returned by

[GetArchiveName\(Int32\)](#).

***stream* ([Stream](#))**

A stream that was previously returned by [OpenArchiveWriteStream\(Int32, String, Boolean, CompressionEngine\)](#) and is now ready to be closed.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CloseFileReadStream Method (path, stream)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveFileStreamContext ►  
**CloseFileReadStream(String, Stream)**

C# ▾

Closes a stream that has been used to read a file.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public virtual void CloseFileReadStream(
    string path,
    Stream stream
)
```

```
Public Overridable Sub CloseFileReadStream ( _
    path As String, _
    stream As Stream _
)
```

```
public:
virtual void CloseFileReadStream(
    String^ path,
    Stream^ stream
)
```

### Parameters

#### **path (String)**

The path of the file within the archive; the same as the path provided when the stream was opened.

#### **stream (Stream)**

A stream that was previously returned by [OpenFileReadStream\(String, FileAttributes, DateTime\)](#) and is now ready to be closed.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:  
Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CloseFileStream Method (path, stream, attributes, lastWriteTime)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveFileStreamContext ►  
**CloseFileStream(String, Stream, FileAttributes, DateTime)**

C# ▾

Closes a stream where an extracted file was written.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public virtual void CloseFileStream(  
    string path,  
    Stream stream,  
    FileAttributes attributes,  
    DateTime lastWriteTime  
)
```

```
Public Overridable Sub CloseFileStream ( _  
    path As String, _  
    stream As Stream, _  
    attributes As FileAttributes, _  
    lastWriteTime As DateTime _  
)
```

```
public:  
virtual void CloseFileStream(  
    String^ path,  
    Stream^ stream,  
    FileAttributes attributes,  
    DateTime lastWriteTime  
)
```

### Parameters

#### **path (String)**

The path of the file within the archive.

### ***stream* ([Stream](#))**

The stream that was previously returned by [OpenFileWriteStream\(String, Int64, DateTime\)](#) and is now ready to be closed.

### ***attributes* ([FileAttributes](#))**

The attributes of the extracted file.

### ***lastWriteTime* ([DateTime](#))**

The last write time of the file.

## **Remarks**

After closing the extracted file stream, this method applies the date and attributes to that file.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Directory Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileStreamContext ►

## Directory

C# ▾

Gets or sets the default root directory where files are located.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string Directory { get; }
```

```
Public ReadOnly Property Directory As String
```

```
public:  
property String^ Directory {  
    String^ get ();  
}
```

### Value

The default root directory where files are located.

### Remarks

For details about how the default directory is used, see [OpenFileReadStream\(String, FileAttributes, DateTime\)](#) and [OpenFileWriteStream\(String, Int64, DateTime\)](#).

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **EnableOffsetOpen Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileStreamContext ►

**EnableOffsetOpen**

C# ▾

Gets or sets a flag that enables creating or extracting an archive at an offset within an existing file. (This is typically used to open archive-base self-extracting packages.)

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public bool EnableOffsetOpen { get; set; }
```

Public Property EnableOffsetOpen As Boolean

```
public:  
property bool EnableOffsetOpen {  
    bool get ();  
    void set (bool value);  
}
```

[-] **Value**

True to search an existing package file for an archive offset or the end of the file;/ false to always create or open a plain archive file.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ExtractOnlyNewerFiles Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileStreamContext ►

### ExtractOnlyNewerFiles

C# ▾

Gets or sets a flag that can prevent extracted files from overwriting new files that already exist.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool ExtractOnlyNewerFiles { get; set; }
```

```
Public Property ExtractOnlyNewerFiles As Boolean
```

```
public:  
property bool ExtractOnlyNewerFiles {  
    bool get ();  
    void set (bool value);  
}
```

#### Value

True to prevent overwriting newer files that already exist during extraction; false to always extract from the archive regardless of existing files.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Files** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveFileStreamContext ►

## Files

C# ▾

Gets or sets the mapping from internal file paths to external file paths.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public IDictionary<string, string> Files { get; }
```

```
Public ReadOnly Property Files As IDictionary(Of Str
```

```
public:  
property IDictionary<String^, String^>^ Files {  
    IDictionary<String^, String^>^ get ();  
}
```

### Value

A mapping from internal file paths to external file paths.

### Remarks

For details about how the files mapping is used, see [OpenFileReadStream\(String, FileAttributes, DateTime\)](#) and [OpenFileWriteStream\(String, Int64, DateTime\)](#).

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **GetArchiveName Method  
(archiveNumber)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveFileStreamContext ►  
**GetArchiveName(Int32)**

C# ▾

Gets the name of the archive with a specified number.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public virtual string GetArchiveName(  
    int archiveNumber  
)
```

```
Public Overridable Function GetArchiveName ( _  
    archiveNumber As Integer _  
) As String
```

```
public:  
virtual String^ GetArchiveName(  
    int archiveNumber  
)
```

[-] **Parameters**

**archiveNumber (Int32)**

The 0-based index of the archive within the chain.

[-] **Return Value**

The name of the requested archive. May be an empty string for non-chained archives, but may never be null.

[-] **Remarks**

This method returns the file name of the archive from the **archiveFiles**( list with the specified index, or an empty string if the archive number is

outside the bounds of the list. The file name should not include any directory path.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## GetOption Method (optionName, parameters)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveFileStreamContext ►  
**GetOption(String, Object[])**

C# ▾

Gets extended parameter information specific to the compression format being used.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public virtual Object GetOption(  
    string optionName,  
    Object[] parameters  
)
```

```
Public Overridable Function GetOption ( _  
    optionName As String, _  
    parameters As Object() _  
) As Object
```

```
public:  
virtual Object^ GetOption(  
    String^ optionName,  
    array<Object^>^ parameters  
)
```

### Parameters

#### optionName (String)

Name of the option being requested.

#### parameters ( Object [] )

Parameters for the option; for per-file options, the first parameter is typically the internal file path.

## **❑ Return Value**

Option value, or null to use the default behavior.

## **❑ Remarks**

This implementation does not handle any options. Subclasses may override this method to allow for non-default behavior.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## OpenArchiveReadStream Method (archiveNumber, archiveName, compressionEngine)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveFileStreamContext ►  
**OpenArchiveReadStream(Int32, String, CompressionEngine)**

C# ▾

Opens the archive stream for reading.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public virtual Stream OpenArchiveReadStream(  
    int archiveNumber,  
    string archiveName,  
    CompressionEngine compressionEngine  
)
```

```
Public Overridable Function OpenArchiveReadStream (  
    archiveNumber As Integer, _  
    archiveName As String, _  
    compressionEngine As CompressionEngine _  
) As Stream
```

```
public:  
virtual Stream^ OpenArchiveReadStream(  
    int archiveNumber,  
    String^ archiveName,  
    CompressionEngine^ compressionEngine  
)
```

### Parameters

#### **archiveNumber (Int32)**

The zero-based index of the archive to open.

#### **archiveName (String)**

The name of the archive being opened.

### ***compressionEngine* ([CompressionEngine](#))**

Instance of the compression engine doing the operations.

#### **□ Return Value**

A stream from which archive bytes are read, or null to cancel extraction the archive.

#### **□ Remarks**

This method opens the file from the [ArchiveFiles](#) list with the specified index. If the archive number is outside the bounds of the list, this method returns null.

If the [EnableOffsetOpen](#) flag is set, this method will seek to the start of any existing archive in the file, or to the end of the file if the existing file not an archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## OpenArchiveWriteStream Method (archiveNumber, archiveName, truncate, compressionEngine)

Deployment Tools Foundation Namespaces ►  
[Microsoft.Deployment.Compression](#) ► [ArchiveFileStreamContext](#) ►  
**OpenArchiveWriteStream(Int32, String, Boolean,  
CompressionEngine)**

Opens a stream for writing an archive.

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public virtual Stream OpenArchiveWriteStream(  
    int archiveNumber,  
    string archiveName,  
    bool truncate,  
    CompressionEngine compressionEngine  
)
```

```
Public Overridable Function OpenArchiveWriteStream (  
    archiveNumber As Integer, _  
    archiveName As String, _  
    truncate As Boolean, _  
    compressionEngine As CompressionEngine _  
) As Stream
```

```
public:  
virtual Stream^ OpenArchiveWriteStream(  
    int archiveNumber,  
    String^ archiveName,  
    bool truncate,  
    CompressionEngine^ compressionEngine  
)
```

### Parameters

***archiveNumber*** ([Int32](#))

The 0-based index of the archive within the chain.

***archiveName*** ([String](#))

The name of the archive that was returned by [GetArchiveName\(Int32\)](#).

***truncate*** ([Boolean](#))

True if the stream should be truncated when opened (if it already exists); false if an existing stream is being re-opened for writing additional data.

***compressionEngine*** ([CompressionEngine](#))

Instance of the compression engine doing the operations.

**Return Value**

A writable Stream where the compressed archive bytes will be written, or null to cancel the archive creation.

**Remarks**

This method opens the file from the [ArchiveFiles](#) list with the specified index. If the archive number is outside the bounds of the list, this method returns null.

If the [EnableOffsetOpen](#) flag is set, this method will seek to the start of any existing archive in the file, or to the end of the file if the existing file is not an archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## OpenFileReadStream Method (path, attributes, lastWriteTime)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveFileStreamContext ►  
**OpenFileReadStream(String, FileAttributes, DateTime)**

C# ▾

Opens a stream to read a file that is to be included in an archive.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public virtual Stream OpenFileReadStream(  
    string path,  
    out FileAttributes attributes,  
    out DateTime lastWriteTime  
)
```

```
Public Overridable Function OpenFileReadStream ( _  
    path As String, _  
    <OutAttribute> ByRef attributes As FileAttri  
    <OutAttribute> ByRef lastWriteTime As DateTi  
) As Stream
```

```
public:  
virtual Stream^ OpenFileReadStream(  
    String^ path,  
    [OutAttribute] FileAttributes% attributes,  
    [OutAttribute] DateTime% lastWriteTime  
)
```

### Parameters

#### **path (String)**

The path of the file within the archive.

#### **attributes ( FileAttributes )**

The returned attributes of the opened file, to be stored in the archiv

## ***lastWriteTime*** ( [DateTime](#) )

The returned last-modified time of the opened file, to be stored in the archive.

### **Return Value**

A readable Stream where the file bytes will be read from before they are compressed, or null to skip inclusion of the file and continue to the next file.

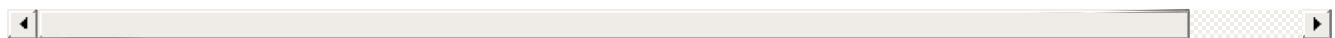
### **Remarks**

This method opens a file using the following logic: If the [Directory](#) and the [Files](#) mapping are both null, the path is treated as relative to the current directory, and that file is opened. If the [Directory](#) is not null but the [Files](#) mapping is null, the path is treated as relative to that directory, and that file is opened. If the [Directory](#) is null but the [Files](#) mapping is not null, the path parameter is used as a key into the mapping, and the resulting value is the file path that is opened, relative to the current directory (or it may be an absolute path). If no mapping exists, the file is skipped. If both the [Directory](#) and the [Files](#) mapping are specified, the path parameter is used as a key into the mapping, and the resulting value is the file path that is opened, relative to the specified directory (or it may be an absolute path). If no mapping exists, the file is skipped.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## OpenFileWriteStream Method (path, fileSize, lastWriteTime)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveFileStreamContext ►  
**OpenFileWriteStream(String, Int64, DateTime)**

C# ▾

Opens a stream for writing extracted file bytes.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public virtual Stream OpenFileWriteStream(  
    string path,  
    long fileSize,  
    DateTime lastWriteTime  
)
```

```
Public Overridable Function OpenFileWriteStream ( _  
    path As String, _  
    fileSize As Long, _  
    lastWriteTime As DateTime _  
) As Stream
```

```
public:  
virtual Stream^ OpenFileWriteStream(  
    String^ path,  
    long long fileSize,  
    DateTime lastWriteTime  
)
```

### - Parameters

#### **path (String)**

The path of the file within the archive.

#### **fileSize (Int64)**

The uncompressed size of the file to be extracted.

## ***LastWriteTime* ([DateTime](#))**

The last write time of the file to be extracted.

### **Return Value**

A stream where extracted file bytes are to be written, or null to skip extraction of the file and continue to the next file.

### **Remarks**

This method opens a file using the following logic: If the [Directory](#) and the [Files](#) mapping are both null, the path is treated as relative to the current directory, and that file is opened. If the [Directory](#) is not null but the [Files](#) mapping is null, the path is treated as relative to that directory, and that file is opened. If the [Directory](#) is null but the [Files](#) mapping is not null, the path parameter is used as a key into the mapping, and the resulting value is the file path that is opened, relative to the current directory (or it may be an absolute path). If no mapping exists, the file is skipped. If both the [Directory](#) and the [Files](#) mapping are specified, the path parameter is used as a key into the mapping, and the resulting value is the file path that is opened, relative to the specified directory (or it may be an absolute path). If no mapping exists, the file is skipped.

If the [ExtractOnlyNewerFiles](#) flag is set, the file is skipped if a file currently exists in the same path with an equal or newer write time.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **ArchiveInfo Class**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveInfo

C# ▾

Abstract object representing a compressed archive on disk; provides access to file-based operations on the archive.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
[SerializableAttribute]  
public abstract class ArchiveInfo : FileSystemInfo
```

```
<SerializableAttribute> _  
Public MustInherit Class ArchiveInfo _  
    Inherits FileSystemInfo
```

```
[SerializableAttribute]  
public ref class ArchiveInfo abstract : public FileS
```

## Members

All Members	Constructors	Methods	Properties	Fields
<input checked="" type="checkbox"/> Public		<input checked="" type="checkbox"/> Instance		<input checked="" type="checkbox"/> Declared
<input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Static		<input checked="" type="checkbox"/> Inherited

Icon	Member	Description
💡	ArchiveInfo(String)	Creates a new ArchiveInfo object representing an arc in a specified path.
💡	ArchiveInfo(SerializationInfo, StreamingContext)	Initializes a new instance of the ArchiveInfo class with serialized data.

	<a href="#">Attributes</a>	Gets or sets the <a href="#">FileAttributes</a> of the current <a href="#">FileSystemInfo</a> .  (Inherited from <a href="#">FileSystemInfo</a> .)
	<a href="#">CopyTo(String)</a>	Copies an existing archive to another location.
	<a href="#">CopyTo(String, Boolean)</a>	Copies an existing archive to another location, optionally overwriting the destination file.
	<a href="#">CreateCompressionEngine()</a>	Creates a compression engine that does the low-level work for the object.
	<a href="#">CreateObjRef(Type)</a>	Creates an object that contains all the relevant information required to generate a proxy used to communicate with a remote object.  (Inherited from <a href="#">MarshalByRefObject</a> )
	<a href="#">CreationTime</a>	Gets or sets the creation time of the current

		<a href="#">FileSystemInfo</a> object (Inherited from <a href="#">FileSystemInfo</a> .)
	<a href="#">CreationTimeUtc</a>	Gets or sets the creation time, in coordinated universal time (UTC), of the current <a href="#">FileSystemInfo</a> object. (Inherited from <a href="#">FileSystemInfo</a> .)
	<a href="#">Delete()</a>	Deletes the archive. (Overrides <a href="#">FileSystemInfo.Delete()</a> .)
	<a href="#">Directory</a>	Gets the directory that contains the archive.
	<a href="#">DirectoryName</a>	Gets the full path of the directory that contains the archive.
	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> . (Inherited from <a href="#">Object</a> .)
	<a href="#">Exists</a>	Checks if the archive exists. (Overrides <a href="#">FileSystemInfo.Exists()</a> .)
	<a href="#">Extension</a>	Gets the string

		representing the extension part of the file path.  (Inherited from <a href="#">FileSystemInfo</a> .)
💡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
📁	<a href="#">FullName</a>	Gets the full path of the directory or file.  (Inherited from <a href="#">FileSystemInfo</a> .)
💡	<a href="#">FullPath</a>	Represents the fully qualified path of the directory or file.  (Inherited from <a href="#">FileSystemInfo</a> .)
📄	<a href="#">GetFiles()</a>	Gets information about the files contained in the archive.
📄	<a href="#">GetFiles(String)</a>	Gets information about the certain files contained in the archive file.
📄	<a href="#">GetHashCode()</a>	Serves as a hash

		function for a particular type. <a href="#">GetHashCode</a> is suitable for use in hashing algorithms and data structures like hash table.
		(Inherited from <a href="#">Object</a> )
≡	<a href="#">GetLifetimeService()</a>	Retrieves the current lifetime service object that controls the life policy for this instance.
		(Inherited from <a href="#">MarshalByRefObject</a> )
≡	<a href="#">GetObjectData(SerializationInfo, StreamingContext)</a>	Sets the <a href="#">SerializationInfo</a> object with the file name and additional exception information.
		(Inherited from <a href="#">FileSystemInfo</a> )
≡	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.
		(Inherited from <a href="#">Object</a> )
≡	<a href="#">InitializeLifetimeService()</a>	Obtains a lifetime service object to control the lifetime policy for this instance.
		(Inherited from <a href="#">MarshalByRefObject</a> )
≡	<a href="#">IsValid()</a>	Checks if the archive contains a valid archive header.

	<a href="#">LastAccessTime</a>	Gets or sets the time when the current file or directory was last accessed.  (Inherited from <a href="#">FileSystemInfo</a> .)
	<a href="#">LastAccessTimeUtc</a>	Gets or sets the time in coordinated universal time (UTC), that the current file or directory was last accessed.  (Inherited from <a href="#">FileSystemInfo</a> .)
	<a href="#">LastWriteTime</a>	Gets or sets the time when the current file or directory was last written to.  (Inherited from <a href="#">FileSystemInfo</a> .)
	<a href="#">LastWriteTimeUtc</a>	Gets or sets the time in coordinated universal time (UTC), when the current file or directory was last written to.  (Inherited from <a href="#">FileSystemInfo</a> .)
	<a href="#">Length</a>	Gets the size of the archive.
	<a href="#">MemberwiseClone(Boolean)</a>	Creates a shallow copy of the current

		<p><code>MarshalByRefObject</code></p> <p>(Inherited from <code>MarshalByRefObject</code>)</p>
💡	<code>MemberwiseClone()</code>	<p>Creates a shallow copy of the current <code>Object</code>.</p> <p>(Inherited from <code>Object</code>)</p>
💡	<code>MoveTo(String)</code>	<p>Moves an existing archive to another location.</p>
📁	<code>Name</code>	<p>Gets the file name of the archive.</p> <p>(Overrides <code>FileSystemInfo.Name</code>)</p>
💡	<code>OpenRead(String)</code>	<p>Opens a file inside the archive for reading without actually extracting the file to disk.</p>
💡	<code>OpenText(String)</code>	<p>Opens a file inside the archive for reading with UTF-8 encoding without actually extracting the file to disk.</p>
💡	<code>OriginalPath</code>	<p>The path originally specified by the user, whether relative or absolute.</p>

		(Inherited from <a href="#">FileSystemInfo</a> .)
≡	<a href="#">Pack(String)</a>	Compresses all files in the specified directory into the archive. Does not include subdirectories.
≡	<a href="#">Pack(String, Boolean, CompressionLevel, EventHandler&lt;ArchiveProgressEventArgs&gt;)</a>	Compresses all files in the specified directory into the archive, optionally including subdirectories.
≡	<a href="#">PackFiles(String, IList&lt;String&gt;, IList&lt;String&gt;)</a>	Compresses files in the archive, specifying the names used to store the files in the archive.
≡	<a href="#">PackFiles(String, IList&lt;String&gt;, IList&lt;String&gt;, CompressionLevel, EventHandler&lt;ArchiveProgressEventArgs&gt;)</a>	Compresses files in the archive, specifying the names used to store the files in the archive.
≡	<a href="#">PackFileSet(String, IDictionary&lt;String, String&gt;)</a>	Compresses files in the archive, specifying the names used to store the files in the archive.
≡	<a href="#">PackFileSet(String, IDictionary&lt;String, String&gt;, CompressionLevel, EventHandler&lt;ArchiveProgressEventArgs&gt;)</a>	Compresses files in the archive, specifying the names used to store the files in the archive.

	<code>Refresh()</code>	Refreshes the state of the object.  (Inherited from <a href="#">FileSystemInfo</a> .)
	<code>ToString()</code>	Gets the full path of the archive.  (Overrides <a href="#">Object.ToString()</a> .)
	<code>Unpack(String)</code>	Extracts all files from the archive to a destination directory.
	<code>Unpack(String, EventHandler&lt;ArchiveProgressEventArgs&gt;)</code>	Extracts all files from the archive to a destination directory, optionally extracting only new files.
	<code>UnpackFile(String, String)</code>	Extracts a single file from the archive.
	<code>UnpackFiles(IList&lt;String&gt;, String, IList&lt;String&gt;)</code>	Extracts multiple files from the archive.
	<code>UnpackFiles(IList&lt;String&gt;, String, IList&lt;String&gt;, EventHandler&lt;ArchiveProgressEventArgs&gt;)</code>	Extracts multiple files from the archive, optionally extracting newer files.

≡	UnpackFileSet(IDictionary<String, String>, String)	Extracts multiple files from the archive.
≡	UnpackFileSet(IDictionary<String, String>, String, EventHandler<ArchiveProgressEventArgs>)	Extracts multiple files from the archive.

## [-] Inheritance Hierarchy

Object  
└ MarshalByRefObject  
  └ FileSystemInfo  
    └ ArchiveInfo  
      └ CabInfo

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## ArchiveInfo Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► ArchiveInfo()

C# ▾

### Members

Icon	Member	Description
	<a href="#">ArchiveInfo(String)</a>	Creates a new ArchiveInfo object representing an archive in a specified path.
	<a href="#">ArchiveInfo(SerializationInfo, StreamingContext)</a>	Initializes a new instance of the ArchiveInfo class with serialized data.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ArchiveInfo Constructor (info, context)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveInfo ►  
ArchiveInfo(SerializationInfo, StreamingContext)

C#

Initializes a new instance of the ArchiveInfo class with serialized data.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
protected ArchiveInfo(  
    SerializationInfo info,  
    StreamingContext context  
)
```

```
Protected Sub New (  
    info As SerializationInfo,  
    context As StreamingContext  
)
```

```
protected:  
ArchiveInfo(  
    SerializationInfo^ info,  
    StreamingContext context  
)
```

### Parameters

#### info (SerializationInfo)

The SerializationInfo that holds the serialized object data about the exception being thrown.

#### context (StreamingContext)

The StreamingContext that contains contextual information about the source or destination.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:  
Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ArchiveInfo Constructor (path)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveInfo ►  
ArchiveInfo(String)

C# ▾

Creates a new ArchiveInfo object representing an archive in a specified path.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected ArchiveInfo(  
    string path  
)
```

```
Protected Sub New ( _  
    path As String _  
)
```

```
protected:  
ArchiveInfo(  
    String^ path  
)
```

### Parameters

#### *path* (String)

The path to the archive. When creating an archive, this file does not necessarily exist yet.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **CopyTo Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► **CopyTo()**

C# ▾

▀ **Members**

Icon	Member	Description
	<a href="#">CopyTo(String)</a>	Copies an existing archive to another location.
	<a href="#">CopyTo(String, Boolean)</a>	Copies an existing archive to another location, optionally overwriting the destination file.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CopyTo Method (destFileName)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► CopyTo(String)

C# ▾

Copies an existing archive to another location.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void CopyTo(  
    string destFileName  
)
```

```
Public Sub CopyTo ( _  
    destFileName As String _  
)
```

```
public:  
void CopyTo(  
    String^ destFileName  
)
```

### Parameters

#### *destFileName* (String)

The destination file path.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CopyTo Method (destFileName, overwrite)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveInfo ► **CopyTo(String, Boolean)**

C# ▾

Copies an existing archive to another location, optionally overwriting the destination file.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void CopyTo(  
    string destFileName,  
    bool overwrite  
)
```

```
Public Sub CopyTo ( _  
    destFileName As String, _  
    overwrite As Boolean _  
)
```

```
public:  
void CopyTo(  
    String^ destFileName,  
    bool overwrite  
)
```

### Parameters

#### **destFileName (String)**

The destination file path.

#### **overwrite (Boolean)**

If true, the destination file will be overwritten if it exists.

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# CreateCompressionEngine

## Method

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveInfo ►  
**CreateCompressionEngine()**

C# ▾

Creates a compression engine that does the low-level work for this object.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected abstract CompressionEngine CreateCompressi
```

```
Protected MustOverride Function CreateCompressionEng
```

```
protected:  
virtual CompressionEngine^ CreateCompressionEngine()
```

### Return Value

A new compression engine instance that matches the specific subclass archive.

### Remarks

Each instance will be [Dispose\(\)](#)d immediately after use.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



C# ▾

Deletes the archive.

## - Declaration Syntax

C#

Visual Basic

Visual C++

```
public override void Delete()
```

```
Public Overrides Sub Delete
```

```
public:  
virtual void Delete() override
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Directory Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchivFileInfo ► Directory

C# ▾

Gets the directory that contains the archive.

## [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public DirectoryInfo Directory { get; }
```

```
Public ReadOnly Property Directory As DirectoryInfo
```

```
public:  
property DirectoryInfo^ Directory {  
    DirectoryInfo^ get ();  
}
```

## [-] Value

A DirectoryInfo object representing the parent directory of the archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## DirectoryName Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► DirectoryName

C# ▾

Gets the full path of the directory that contains the archive.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string DirectoryName { get; }
```

```
Public ReadOnly Property DirectoryName As String
```

```
public:  
property String^ DirectoryName {  
    String^ get ();  
}
```

### - Value

The full path of the directory that contains the archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Exists Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► Exists

C# ▾

Checks if the archive exists.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public override bool Exists { get; }
```

```
Public Overrides ReadOnly Property Exists As Boolean
```

```
public:  
virtual property bool Exists {  
    bool get () override;  
}
```

### Value

True if the archive exists; else false.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetFiles Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► GetFiles()

C# ▾

**Members**

Icon	Member	Description
	<a href="#">GetFiles()</a>	Gets information about the files contained in the archive.
	<a href="#">GetFiles(String)</a>	Gets information about the certain file contained in the archive file.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetFiles Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► GetFiles()

C# ▾

Gets information about the files contained in the archive.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public IList<ArchiveFileInfo> GetFiles()
```

```
Public Function GetFiles As IList(Of ArchiveFileInfo)
```

```
public:  
IList<ArchiveFileInfo^>^ GetFiles()
```

[-] **Return Value**

A list of [ArchiveFileInfo](#) objects, each containing information about a file in the archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **GetFiles Method (searchPattern)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► **GetFiles(String)**

C#

Gets information about the certain files contained in the archive file.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public IList<ArchiveFileInfo> GetFiles(  
    string searchPattern  
)
```

```
Public Function GetFiles ( _  
    searchPattern As String _  
) As IList(Of ArchiveFileInfo)
```

```
public:  
IList<ArchiveFileInfo^>^ GetFiles(  
    String^ searchPattern  
)
```

## Parameters

### **searchPattern (String)**

The search string, such as "\*.txt".

## Return Value

A list of [ArchiveFileInfo](#) objects, each containing information about a file in the archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **IsValid** Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► **IsValid()**

C# ▾

Checks if the archive contains a valid archive header.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool IsValid()
```

```
Public Function IsValid As Boolean
```

```
public:  
bool IsValid()
```

### Return Value

True if the file is a valid archive; false otherwise.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Length Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► Length

C# ▾

Gets the size of the archive.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public long Length { get; }
```

```
Public Readonly Property Length As Long
```

```
public:  
property long long Length {  
    long long get ();  
}
```

### Value

The size of the archive in bytes.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## MoveTo Method (destFileName)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► MoveTo(String)

C# ▾

Moves an existing archive to another location.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void MoveTo(  
    string destFileName  
)
```

```
Public Sub MoveTo ( _  
    destFileName As String _  
)
```

```
public:  
void MoveTo(  
    String^ destFileName  
)
```

### Parameters

#### *destFileName* (String)

The destination file path.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Name Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► Name

C# ▾

Gets the file name of the archive.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public override string Name { get; }
```

```
Public Overrides ReadOnly Property Name As String
```

```
public:  
virtual property String^ Name {  
    String^ get () override;  
}
```

### - Value

The file name of the archive, not including any path.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **OpenRead Method (fileName)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchivFileInfo ►  
**OpenRead(String)**

Opens a file inside the archive for reading without actually extracting the file to disk.

C#

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public Stream OpenRead(  
    string fileName  
)
```

```
Public Function OpenRead ( _  
    fileName As String _  
) As Stream
```

```
public:  
Stream^ OpenRead(  
    String^ fileName  
)
```

## Parameters

### ***fileName* (String)**

The name of the file in the archive. Also includes the internal path to the file, if any. File name matching is case-insensitive.

## Return Value

A stream for reading directly from the packed file. Like any stream this should be closed/disposed as soon as it is no longer needed.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **OpenText Method (fileName)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchivFileInfo ►  
**OpenText(String)**

Opens a file inside the archive for reading text with UTF-8 encoding without actually extracting the file to disk.

C#

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public StreamReader OpenText(  
    string fileName  
)
```

```
Public Function OpenText ( _  
    fileName As String _  
) As StreamReader
```

```
public:  
StreamReader^ OpenText(  
    String^ fileName  
)
```

### Parameters

#### **fileName (String)**

The name of the file in the archive. Also includes the internal path to the file, if any. File name matching is case-insensitive.

### Return Value

A reader for reading text directly from the packed file. Like any reader this should be closed/disposed as soon as it is no longer needed.

### Remarks

To open an archived text file with different encoding, use the [OpenRead\(String\)](#) method and pass the returned stream to one of the

[StreamReader](#) constructor overloads.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

C# ▾

## Members

Icon	Member	Description
	<a href="#">Pack(String)</a>	Compresses all files in a directory into the archive. Does not include subdirectories.
	<a href="#">Pack(String, Boolean, CompressionLevel, EventHandler&lt;ArchiveProgressEventArgs&gt;)</a>	Compresses all files in a directory into the archive, optionally including subdirectories.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Pack Method (sourceDirectory)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchivInfo ► Pack(String)

C# ▾

Compresses all files in a directory into the archive. Does not include subdirectories.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Pack(  
    string sourceDirectory  
)
```

```
Public Sub Pack ( _  
    sourceDirectory As String _  
)
```

```
public:  
void Pack(  
    String^ sourceDirectory  
)
```

## Parameters

### sourceDirectory (String)

The directory containing the files to be included.

## Remarks

Uses maximum compression level.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Pack Method (sourceDirectory, includeSubdirectories, compLevel, progressHandler)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► Pack(String, Boolean, CompressionLevel, EventHandler<ArchiveProgressEventArgs>)

C#

Compresses all files in a directory into the archive, optionally including subdirectories.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Pack(  
    string sourceDirectory,  
    bool includeSubdirectories,  
    CompressionLevel compLevel,  
    EventHandler<ArchiveProgressEventArgs> progressHandler  
)
```

```
Public Sub Pack ( _  
    sourceDirectory As String, _  
    includeSubdirectories As Boolean, _  
    compLevel As CompressionLevel, _  
    progressHandler As EventHandler(Of ArchiveProgressEventArgs)  
)
```

```
public:  
void Pack(  
    String^ sourceDirectory,  
    bool includeSubdirectories,  
    CompressionLevel compLevel,  
    EventHandler<ArchiveProgressEventArgs^>^ progressHandler  
)
```

## **Parameters**

### ***sourceDirectory* ([String](#))**

This parameter may be null, but if specified it is the root directory for any relative paths in *sourceFileNames*.

### ***includeSubdirectories* ([Boolean](#))**

If true, recursively include files in subdirectories.

### ***compLevel* ([CompressionLevel](#))**

The compression level used when creating the archive.

### ***progressHandler* ([EventHandler<ArchiveProgressEventArgs>](#))**

Handler for receiving progress information; this may be null if progress is not desired.

## **Remarks**

The files are stored in the archive using their relative file paths in the directory tree, if supported by the archive file format.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **PackFiles Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► PackFiles()

C# ▾

▀ **Members**

Icon	Member	Description
≡ ♦	PackFiles(String, IList<String>, IList<String>)	Compresses files into the archive, specifying the names used to store the files in the archive.
≡ ♦	PackFiles(String, IList<String>, IList<String>, CompressionLevel, EventHandler<ArchiveProgressEventArgs>)	Compresses files into the archive, specifying the names used to store the files in the archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## PackFiles Method (sourceDirectory, sourceFileNames, fileNames)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveInfo ►  
**PackFiles(String, IList<String>, IList<String>)**

C#

Compresses files into the archive, specifying the names used to store the files in the archive.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public void PackFiles(  
    string sourceDirectory,  
    IList<string> sourceFileNames,  
    IList<string> fileNames  
)
```

```
Public Sub PackFiles ( _  
    sourceDirectory As String, _  
    sourceFileNames As IList(Of String), _  
    fileNames As IList(Of String) _  
)
```

```
public:  
void PackFiles(  
    String^ sourceDirectory,  
    IList<String^>^ sourceFileNames,  
    IList<String^>^ fileNames  
)
```

### - Parameters

#### **sourceDirectory (String)**

This parameter may be null, but if specified it is the root directory for any relative paths in *sourceFileNames*.

### ***sourceFileNames* ([IList<String>](#))**

The list of files to be included in the archive.

### ***fileNames* ([IList<String>](#))**

The names of the files as they are stored in the archive. Each name includes the internal path of the file, if any. This parameter may be null, in which case the files are stored in the archive with their source file names and no path information.

## **Remarks**

Uses maximum compression level.

Duplicate items in the *fileNames* array will cause an [ArchiveException](#).

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## PackFiles Method (sourceDirectory, sourceFileNames, fileNames, compLevel, progressHandler)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchivelInfo ►

**PackFiles(String, IList<String>, IList<String>, CompressionLevel,  
EventHandler<ArchiveProgressEventArgs>)**

C#

Compresses files into the archive, specifying the names used to store the files in the archive.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void PackFiles(  
    string sourceDirectory,  
    IList<string> sourceFileNames,  
    IList<string> fileNames,  
    CompressionLevel compLevel,  
    EventHandler<ArchiveProgressEventArgs> progressHandler  
)
```

```
Public Sub PackFiles ( _  
    sourceDirectory As String, _  
    sourceFileNames As IList(Of String), _  
    fileNames As IList(Of String), _  
    compLevel As CompressionLevel, _  
    progressHandler As EventHandler(Of ArchiveProgressEventArgs)  
)
```

```
public:  
void PackFiles(  
    String^ sourceDirectory,  
    IList<String^>^ sourceFileNames,  
    IList<String^>^ fileNames,  
    CompressionLevel compLevel,
```

```
        EventHandler<ArchiveProgressEventArgs>^ pro  
    )
```

## [-] Parameters

### ***sourceDirectory (String)***

This parameter may be null, but if specified it is the root directory for any relative paths in *sourceFileNames*.

### ***sourceFileNames (IList<String>)***

The list of files to be included in the archive.

### ***fileNames (IList<String>)***

The names of the files as they are stored in the archive. Each name includes the internal path of the file, if any. This parameter may be null, in which case the files are stored in the archive with their source file names and no path information.

### ***compLevel (CompressionLevel)***

The compression level used when creating the archive.

### ***progressHandler (EventHandler<ArchiveProgressEventArgs>)***

Handler for receiving progress information; this may be null if progress is not desired.

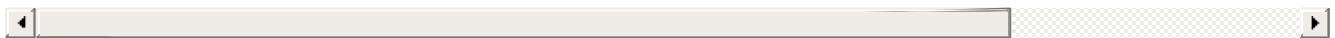
## [-] Remarks

Duplicate items in the *fileNames* array will cause an [ArchiveException](#).

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **PackFileSet Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► PackFileSet()

C# ▾

▀ Members

Icon	Member	Description
≡ ♦	PackFileSet(String, IDictionary<String, String>)	Compresses files into the archive, specifying the names used to store the files in the archive.
≡ ♦	PackFileSet(String, IDictionary<String, String>, CompressionLevel, EventHandler<ArchiveProgressEventArgs>)	Compresses files into the archive, specifying the names used to store the files in the archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## PackFileSet Method (*sourceDirectory*, *fileNames*)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveInfo ►  
**PackFileSet(String, IDictionary<String, String>)**

Compresses files into the archive, specifying the names used to store the files in the archive.

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void PackFileSet(  
    string sourceDirectory,  
    IDictionary<string, string> fileNames  
)
```

```
Public Sub PackFileSet ( _  
    sourceDirectory As String, _  
    fileNames As IDictionary(Of String, String)  
)
```

```
public:  
void PackFileSet(  
    String^ sourceDirectory,  
    IDictionary<String^, String^>^ fileNames  
)
```

### Parameters

#### *sourceDirectory* (String)

This parameter may be null, but if specified it is the root directory for any relative paths in *fileNames*.

#### *fileNames* (IDictionary<String, String>)

A mapping from internal file paths to external file paths.

## Remarks

Uses maximum compression level.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## PackFileSet Method (sourceDirectory, fileNames, compLevel, progressHandler)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchivelInfo ►

**PackFileSet(String, IDictionary<String, String>, CompressionLevel,  
EventHandler<ArchiveProgressEventArgs>)**

C#

Compresses files into the archive, specifying the names used to store the files in the archive.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void PackFileSet(  
    string sourceDirectory,  
    IDictionary<string, string> fileNames,  
    CompressionLevel compLevel,  
    EventHandler<ArchiveProgressEventArgs> progressHandler  
)
```

```
Public Sub PackFileSet ( _  
    sourceDirectory As String, _  
    fileNames As IDictionary(Of String, String), _  
    compLevel As CompressionLevel, _  
    progressHandler As EventHandler(Of ArchivePr  
)
```

```
public:  
void PackFileSet(  
    String^ sourceDirectory,  
    IDictionary<String^, String^>^ fileNames,  
    CompressionLevel compLevel,  
    EventHandler<ArchiveProgressEventArgs^>^ progressHandler  
)
```

## Parameters

### ***sourceDirectory (String)***

This parameter may be null, but if specified it is the root directory for any relative paths in *fileNames*.

### ***fileNames (IDictionary<String, String>)***

A mapping from internal file paths to external file paths.

### ***compLevel (CompressionLevel)***

The compression level used when creating the archive.

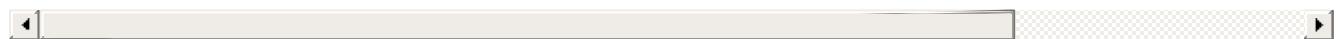
### ***progressHandler (EventHandler<ArchiveProgressEventArgs>)***

Handler for receiving progress information; this may be null if progress is not desired.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



## Deployment Tools Foundation **ToString** Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► **ToString()**

C# ▾

Gets the full path of the archive.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override string ToString()
```

```
Public Overrides Function ToString As String
```

```
public:  
virtual String^ ToString() override
```

### Return Value

The full path of the archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# Unpack Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► Unpack()

C# ▾

## Members

Icon	Member	Description
	<a href="#">Unpack(String)</a>	Extracts all files from an archive to a destination directory.
	<a href="#">Unpack(String, EventHandler&lt;ArchiveProgressEventArgs&gt;)</a>	Extracts all files from an archive to a destination directory, optionally extracting only newer files.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Unpack Method (destDirectory)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► Unpack(String)

C# ▾

Extracts all files from an archive to a destination directory.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Unpack(  
    string destDirectory  
)
```

```
Public Sub Unpack ( _  
    destDirectory As String _  
)
```

```
public:  
void Unpack(  
    String^ destDirectory  
)
```

## Parameters

### *destDirectory* (String)

Directory where the files are to be extracted.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Unpack Method (destDirectory, progressHandler)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveInfo ► **Unpack(String, EventHandler<ArchiveProgressEventArgs>)**

C# ▾

Extracts all files from an archive to a destination directory, optionally extracting only newer files.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Unpack(  
    string destDirectory,  
    EventHandler<ArchiveProgressEventArgs> progressHandler  
)
```

```
Public Sub Unpack ( _  
    destDirectory As String, _  
    progressHandler As EventHandler(Of ArchiveProgressEventArgs)  
)
```

```
public:  
void Unpack(  
    String^ destDirectory,  
    EventHandler<ArchiveProgressEventArgs^>^ progressHandler  
)
```

### Parameters

#### **destDirectory (String)**

Directory where the files are to be extracted.

#### **progressHandler (EventHandler<ArchiveProgressEventArgs>)**

Handler for receiving progress information; this may be null if progress is not desired.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:  
Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



# UnpackFile Method (fileName, destFileName)

Deployment Tools Foundation Namespaces ►  
 Microsoft.Deployment.Compression ► ArchiveInfo ►  
**UnpackFile(String, String)**

Extracts a single file from the archive.

C# ▾

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public void UnpackFile(
    string fileName,
    string destFileName
)
```

```
Public Sub UnpackFile ( _
    fileName As String, _
    destFileName As String _
)
```

```
public:
void UnpackFile(
    String^ fileName,
    String^ destFileName
)
```

## Parameters

### ***fileName* (String)**

The name of the file in the archive. Also includes the internal path to the file, if any. File name matching is case-insensitive.

### ***destFileName* (String)**

The path where the file is to be extracted on disk.

## Remarks

If *destFileName* already exists, it will be overwritten.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## UnpackFiles Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► UnpackFiles()

C# ▾

### Members

Icon	Member	Description
	<code>UnpackFiles(IList&lt;String&gt;, String, IList&lt;String&gt;)</code>	Extracts multiple files from the archive.
	<code>UnpackFiles(IList&lt;String&gt;, String, IList&lt;String&gt;, EventHandler&lt;ArchiveProgressEventArgs&gt;)</code>	Extracts multiple files from the archive, optionally extracting only newer files.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## UnpackFiles Method (fileNames, destDirectory, destFileNames)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveInfo ►  
**UnpackFiles(IList<String>, String, IList<String>)**

C# ▾

Extracts multiple files from the archive.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public void UnpackFiles(  
    IList<string> fileNames,  
    string destDirectory,  
    IList<string> destFileNames  
)
```

```
Public Sub UnpackFiles ( _  
    fileNames As IList(Of String), _  
    destDirectory As String, _  
    destFileNames As IList(Of String) _  
)
```

```
public:  
void UnpackFiles(  
    IList<String^>^ fileNames,  
    String^ destDirectory,  
    IList<String^>^ destFileNames  
)
```

### - Parameters

#### **fileNames (IList<String>)**

The names of the files in the archive. Each name includes the internal path of the file, if any. File name matching is case-insensitive.

### ***destDirectory* ([String](#))**

This parameter may be null, but if specified it is the root directory for any relative paths in *destFileNames*.

### ***destFileNames* ([IList<String>](#))**

The paths where the files are to be extracted on disk. If this parameter is null, the files will be extracted with the names from the archive.

## **Remarks**

If any extracted files already exist on disk, they will be overwritten.

The *destDirectory* and *destFileNames* parameters cannot both be null.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## UnpackFiles Method (fileNames, destDirectory, destFileNames, progressHandler)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveInfo ►  
**UnpackFiles(IList<String>, String, IList<String>, EventHandler<ArchiveProgressEventArgs>)**

C# ▾

Extracts multiple files from the archive, optionally extracting only newer files.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void UnpackFiles(
    IList<string> fileNames,
    string destDirectory,
    IList<string> destFileNames,
    EventHandler<ArchiveProgressEventArgs> program
)
```

```
Public Sub UnpackFiles ( _
    fileNames As IList(Of String), _
    destDirectory As String, _
    destFileNames As IList(Of String), _
    progressHandler As EventHandler(Of ArchivePr
)
```

```
public:
void UnpackFiles(
    IList<String^>^ fileNames,
    String^ destDirectory,
    IList<String^>^ destFileNames,
    EventHandler<ArchiveProgressEventArgs^>^ pro
)
```

### Parameters

### ***fileNames* ([IList<String>](#))**

The names of the files in the archive. Each name includes the internal path of the file, if any. File name matching is case-insensitive.

### ***destDirectory* ([String](#))**

This parameter may be null, but if specified it is the root directory for any relative paths in *destFileNames*.

### ***destFileNames* ([IList<String>](#))**

The paths where the files are to be extracted on disk. If this parameter is null, the files will be extracted with the names from the archive.

### ***progressHandler* ([EventHandler<ArchiveProgressEventArgs>](#))**

Handler for receiving progress information; this may be null if progress is not desired.

## **[-] Remarks**

If any extracted files already exist on disk, they will be overwritten.

The *destDirectory* and *destFileNames* parameters cannot both be null.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

# UnpackFileSet Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ► **UnpackFileSet()**

C# ▾

## Members

Icon	Member	Description
	<code>UnpackFileSet(IDictionary&lt;String, String&gt;, String)</code>	Extracts multiple files from the archive.
	<code>UnpackFileSet(IDictionary&lt;String, String&gt;, String, EventHandler&lt;ArchiveProgressEventArgs&gt;)</code>	Extracts multiple files from the archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## UnpackFileSet Method (*fileNames*, *destDirectory*)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveInfo ►  
**UnpackFileSet(IDictionary<String, String>, String)**

C# ▾

Extracts multiple files from the archive.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void UnpackFileSet(  
    IDictionary<string, string> fileNames,  
    string destDirectory  
)
```

```
Public Sub UnpackFileSet ( _  
    fileNames As IDictionary(Of String, String),  
    destDirectory As String _  
)
```

```
public:  
void UnpackFileSet(  
    IDictionary<String^, String^>^ fileNames,  
    String^ destDirectory  
)
```

### Parameters

#### *fileNames* (**IDictionary<String, String>**)

A mapping from internal file paths to external file paths. Case-sensitivity when matching internal paths depends on the **IDictionary** implementation.

#### *destDirectory* (**String**)

This parameter may be null, but if specified it is the root directory for any relative external paths in *fileNameMap*.

## Remarks

If any extracted files already exist on disk, they will be overwritten.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## UnpackFileSet Method (*fileNames*, *destDirectory*, *progressHandler*)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveInfo ►

**UnpackFileSet(IDictionary<String, String>, String,  
EventHandler<ArchiveProgressEventArgs>)**

C# ▾

Extracts multiple files from the archive.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public void UnpackFileSet(  
    IDictionary<string, string> fileNames,  
    string destDirectory,  
    EventHandler<ArchiveProgressEventArgs> prog  
)
```

```
Public Sub UnpackFileSet ( _  
    fileNames As IDictionary(Of String, String),  
    destDirectory As String, _  
    progressHandler As EventHandler(Of ArchivePr  
)
```

```
public:  
void UnpackFileSet(  
    IDictionary<String^, String^>^ fileNames,  
    String^ destDirectory,  
    EventHandler<ArchiveProgressEventArgs^>^ pro  
)
```

### - Parameters

#### *fileNames* (**IDictionary<String, String>**)

A mapping from internal file paths to external file paths. Case-sensitivity when matching internal paths depends on the IDictionary implementation.

***destDirectory* ([String](#))**

This parameter may be null, but if specified it is the root directory for any relative external paths in *fileNameMap*.

***progressHandler* ([EventHandler<ArchiveProgressEventArgs>](#))**

Handler for receiving progress information; this may be null if progress is not desired.

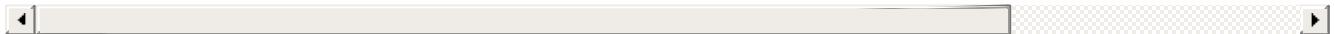
**Remarks**

If any extracted files already exist on disk, they will be overwritten.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools FoundationArchiveProgressEventArgs Class

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveProgressEventArgs

C# ▾

Contains the data reported in an archive progress event.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public class ArchiveProgressEventArgs : EventArgs
```

```
Public Class ArchiveProgressEventArgs _  
    Inherits EventArgs
```

```
public ref class ArchiveProgressEventArgs : public E
```

## Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declare

Inherit

Icon

Icon	Member	Description
	<code>ArchiveProgressEventArgs(ArchiveProgressType, String, Int32, Int32, Int64, Int64, String, Int32, Int32, Int64, Int64, Int64)</code>	Creates a new ArchiveProgressEventArgs object from specified parameters.
	<code>CurrentArchiveBytesProcessed</code>	Gets the number of compressed bytes processed so far during extraction of the archive. Valid extraction methods include ExtractionMethod.ExtractAll, ExtractionMethod.ExtractToDisk, and ExtractionMethod.ExtractToStream.

	<a href="#">CurrentArchiveName</a>	Gets the name of the current archive. This may not necessarily be the name of the archive or the name of the file for all message types.
	<a href="#">CurrentArchiveNumber</a>	Gets the current archive number, where applicable. Returns -1 for a chained set of archives. Valid for all message types.
	<a href="#">CurrentArchiveTotalBytes</a>	Gets the total compressed bytes processed during extraction of the current archive. Valid for extraction messages.
	<a href="#">CurrentFileBytesProcessed</a>	Gets the number of bytes processed so far during compressing or extracting a file. Valid for <a href="#">PartialFile</a> , archive, and message types.
	<a href="#">CurrentFileName</a>	Gets the name of the file being processed. Returns the name of the file for an archive; not the name of the file path. Also returns the name of the file for a partial file.

		internal path component. Valid for <a href="#">StartFile</a> , <a href="#">PartialFile</a> , and <a href="#">FinishFile</a> messages.
	<a href="#">CurrentFileName</a>	Gets the name of the current file being processed. The name starts at index 0, and ends at index <a href="#">TotalFiles-1</a> . Valid for <a href="#">StartFile</a> , <a href="#">PartialFile</a> , and <a href="#">FinishFile</a> messages.
	<a href="#">CurrentFileTotalBytes</a>	Gets the total bytes in the current file. Valid for <a href="#">StartFile</a> , <a href="#">PartialFile</a> , and <a href="#">FinishFile</a> messages.
	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> . (Inherited from <a href="#">Object</a> )
	<a href="#">FileBytesProcessed</a>	Gets the number of uncompressed bytes processed so far for all files. Valid for all message types.
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other finalization operations before it is reclaimed by memory management.

		operations because <a href="#">Object</a> is reclaimed by the garbage collector.
	(Inherited from <a href="#">Object</a> )	
	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular object. Returns the hash code for use in hashing algorithms and data structures like a hash table.
	(Inherited from <a href="#">Object</a> )	
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.
	(Inherited from <a href="#">Object</a> )	
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .
	(Inherited from <a href="#">Object</a> )	
	<a href="#">ProgressType</a>	Gets the type of the message.
	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the <a href="#">Object</a> .
	(Inherited from <a href="#">Object</a> )	
	<a href="#">TotalArchives</a>	Gets the total number of known archives in the chained set. Value message type.

	<a href="#">TotalFileBytes</a>	Gets the total uncompressed bytes to be processed for the specified message type.
	<a href="#">TotalFiles</a>	Gets the total number of files to be processed for all message types.

## Inheritance Hierarchy

[Object](#)

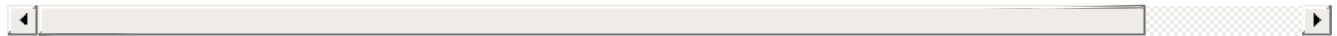
  └ [EventArgs](#)

    └ **ArchiveProgressEventArgs**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## ArchiveProgressEventArgs

### Constructor (progressType, currentFileName, currentFileName, totalFiles, currentFileBytesProcessed, currentFileTotalBytes, currentArchiveName, currentArchiveNumber, totalArchives, currentArchiveBytesProcessed, currentArchiveTotalBytes, fileBytesProcessed, totalFileBytes)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveProgressEventArgs ►

ArchiveProgressEventArgs(ArchiveProgressType, String, Int32, Int32, Int64, Int64, String, Int32, Int32, Int64, Int64, Int64, Int64)

C#

Creates a new ArchiveProgressEventArgs object from specified event parameters.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public ArchiveProgressEventArgs(  
    ArchiveProgressType progressType,  
    string currentFileName,  
    int currentFileName,  
    int totalFiles,  
    long currentFileBytesProcessed,  
    long currentFileTotalBytes,  
    string currentArchiveName,  
    int currentArchiveNumber,  
    int totalArchives,  
    long currentArchiveBytesProcessed,  
    long currentArchiveTotalBytes,  
    long fileBytesProcessed,  
    long totalFileBytes  
)
```

```
Public Sub New ( _  
    progressType As ArchiveProgressType, _  
    currentFileName As String, _  
    currentFileName As Integer, _  
    totalFiles As Integer, _  
    currentFileBytesProcessed As Long, _  
    currentFileTotalBytes As Long, _  
    currentArchiveName As String, _  
    currentArchiveNumber As Integer, _  
    totalArchives As Integer, _  
    currentArchiveBytesProcessed As Long, _  
    currentArchiveTotalBytes As Long, _  
    fileBytesProcessed As Long, _  
    totalFileBytes As Long _  
)
```

```
public:  
ArchiveProgressEventArgs(  
    ArchiveProgressType progressType,  
    String^ currentFileName,  
    int currentFileName,  
    int totalFiles,  
    long long currentFileBytesProcessed,  
    long long currentFileTotalBytes,  
    String^ currentArchiveName,  
    int currentArchiveNumber,  
    int totalArchives,  
    long long currentArchiveBytesProcessed,  
    long long currentArchiveTotalBytes,  
    long long fileBytesProcessed,  
    long long totalFileBytes  
)
```

## ❑ Parameters

**progressType** ([ArchiveProgressType](#))

type of status message

***currentFileName*** ([String](#))

name of the file being processed

***currentFileNumber*** ([Int32](#))

number of the current file being processed

***totalFiles*** ([Int32](#))

total number of files to be processed

***currentFileBytesProcessed*** ([Int64](#))

number of bytes processed so far when compressing or extracting file

***currentFileTotalBytes*** ([Int64](#))

total number of bytes in the current file

***currentArchiveName*** ([String](#))

name of the current Archive

***currentArchiveNumber*** ([Int32](#))

current Archive number, when processing a chained set of Archives

***totalArchives*** ([Int32](#))

total number of Archives in a chained set

***currentArchiveBytesProcessed*** ([Int64](#))

number of compressed bytes processed so far during an extraction

***currentArchiveTotalBytes*** ([Int64](#))

total number of compressed bytes to be processed during an extraction

***fileBytesProcessed*** ([Int64](#))

number of uncompressed file bytes processed so far

***totalFileBytes*** ([Int64](#))

total number of uncompressed file bytes to be processed

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# CurrentArchiveBytesProcessed

## Property

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► ArchiveProgressEventArgs ►  
**CurrentArchiveBytesProcessed**

C# ▾

Gets the number of compressed bytes processed so far during extraction of the current archive. Valid for all extraction messages.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public long CurrentArchiveBytesProcessed { get; }
```

```
Public ReadOnly Property CurrentArchiveBytesProcesse
```

```
public:  
property long long CurrentArchiveBytesProcessed {  
    long long get ();  
}
```

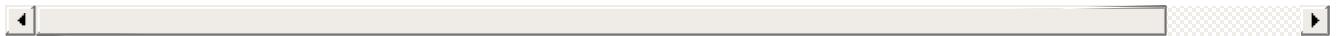
### - Value

The number of compressed bytes processed so far during extraction of the current archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## CurrentArchiveName Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveProgressEventArgs ►

### CurrentArchiveName

C# ▾

Gets the name of the current archive. Not necessarily the name of the archive on disk. Valid for all message types.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string CurrentArchiveName { get; }
```

```
Public ReadOnly Property CurrentArchiveName As String
```

```
public:  
property String^ CurrentArchiveName {  
    String^ get();  
}
```

#### Value

The name of the current archive, or an empty string if no name was specified.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CurrentArchiveNumber Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveProgressEventArgs ►

### CurrentArchiveNumber

C#

Gets the current archive number, when processing a chained set of archives. Valid for all message types.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public int CurrentArchiveNumber { get; }
```

```
Public ReadOnly Property CurrentArchiveNumber As Int
```

```
public:  
property int CurrentArchiveNumber {  
    int get();  
}
```

#### Value

The number of the current archive.

#### Remarks

The first archive is number 0, and the last archive is TotalArchives-1.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



# CurrentArchiveTotalBytes

## Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveProgressEventArgs ►

CurrentArchiveTotalBytes

C#

Gets the total number of compressed bytes to be processed during extraction of the current archive. Valid for all extraction messages.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public long CurrentArchiveTotalBytes { get; }
```

```
Public ReadOnly Property CurrentArchiveTotalBytes As
```

```
public:  
property long long CurrentArchiveTotalBytes {  
    long long get ();  
}
```

### - Value

The total number of compressed bytes to be processed during extraction of the current archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



# CurrentFileBytesProcessed Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveProgressEventArgs ►

**CurrentFileBytesProcessed**

C# ▾

Gets the number of bytes processed so far when compressing or extracting a file. Valid for [StartFile](#), [PartialFile](#), and [FinishFile](#) messages

## - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public long CurrentFileBytesProcessed { get; }
```

```
Public ReadOnly Property CurrentFileBytesProcessed A
```

```
public:  
property long long CurrentFileBytesProcessed {  
    long long get ();  
}
```

## - Value

The number of uncompressed bytes processed so far for the current file or 0 if processing is currently at the stream or archive level.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## CurrentFileName Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveProgressEventArgs ►

### CurrentFileName

C# ▾

Gets the name of the file being processed. (The name of the file within the Archive; not the external file path.) Also includes the internal path of the file, if any. Valid for [StartFile](#), [PartialFile](#), and [FinishFile](#) messages.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string CurrentFileName { get; }
```

```
Public ReadOnly Property CurrentFileName As String
```

```
public:  
property String^ CurrentFileName {  
    String^ get ();  
}
```

#### Value

The name of the file currently being processed, or null if processing is currently at the stream or archive level.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation CurrentFileName Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveProgressEventArgs ►

## CurrentFileName

C# ▾

Gets the number of the current file being processed. The first file is number 0, and the last file is [TotalFiles-1](#). Valid for [StartFile](#), [PartialFile](#), and [FinishFile](#) messages.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public int CurrentFileName { get; }
```

```
Public ReadOnly Property CurrentFileName As Integer
```

```
public:  
property int CurrentFileName {  
    int get();  
}
```

### Value

The number of the file currently being processed, or the most recent file processed if processing is currently at the stream or archive level.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **CurrentFileTotalBytes Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveProgressEventArgs ►

**CurrentFileTotalBytes**

C# ▾

Gets the total number of bytes in the current file. Valid for [StartFile](#), [PartialFile](#), and [FinishFile](#) messages.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public long CurrentFileTotalBytes { get; }
```

```
Public ReadOnly Property CurrentFileTotalBytes As Lo
```

```
public:  
property long long CurrentFileTotalBytes {  
    long long get ();  
}
```

[-] **Value**

The uncompressed size of the current file being processed, or 0 if processing is currently at the stream or archive level.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **FileBytesProcessed** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveProgressEventArgs ►

## FileBytesProcessed

C# ▾

Gets the number of uncompressed bytes processed so far among all files. Valid for all message types.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public long FileBytesProcessed { get; }
```

```
Public ReadOnly Property FileBytesProcessed As Long
```

```
public:  
property long long FileBytesProcessed {  
    long long get ();  
}
```

### Value

The number of uncompressed file bytes processed so far among all files.

### Remarks

When compared to [TotalFileBytes](#), this can be used as a measure of overall progress.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **ProgressType** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveProgressEventArgs ►

## ProgressType

C# ▾

Gets the type of status message.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public ArchiveProgressType ProgressType { get; }
```

```
Public ReadOnly Property ProgressType As ArchiveProg
```

```
public:  
property ArchiveProgressType ProgressType {  
    ArchiveProgressType get ();  
}
```

### Value

A [ArchiveProgressType](#) value indicating what type of progress event occurred.

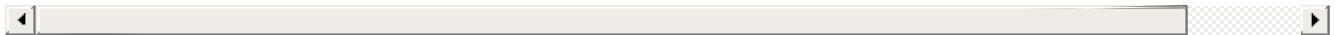
### Remarks

The handler may choose to ignore some types of progress events. For example, if the handler will only list each file as it is compressed/extracted, it can ignore events that are not of type [FinishFil](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **TotalArchives Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveProgressEventArgs ►

**TotalArchives**

C# ▾

Gets the total number of known archives in a chained set. Valid for all message types.

**Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public int TotalArchives { get; }
```

```
Public ReadOnly Property TotalArchives As Integer
```

```
public:  
property int TotalArchives {  
    int get ();  
}
```

**Value**

The total number of known archives in a chained set.

**Remarks**

When using the compression option to auto-split into multiple archives based on data size, this value will not be accurate until the end.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **TotalFileBytes** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveProgressEventArgs ►

## **TotalFileBytes**

C# ▾

Gets the total number of uncompressed file bytes to be processed. Valid for all message types.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public long TotalFileBytes { get; }
```

```
Public ReadOnly Property TotalFileBytes As Long
```

```
public:  
property long long TotalFileBytes {  
    long long get ();  
}
```

### Value

The total number of uncompressed bytes to be processed among all files.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **TotalFiles Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveProgressEventArgs ►

### **TotalFiles**

C#

Gets the total number of files to be processed. Valid for all message types.

#### **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public int TotalFiles { get; }
```

```
Public ReadOnly Property TotalFiles As Integer
```

```
public:  
property int TotalFiles {  
    int get ();  
}
```

#### **Value**

The total number of files to be processed that are known so far.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ArchiveProgressType

### Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► ArchiveProgressType

C# ▾

The type of progress event.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public enum ArchiveProgressType
```

```
Public Enumeration ArchiveProgressType
```

```
public enum class ArchiveProgressType
```

#### Members

Member	Description
<b>StartFile</b>	Status message before beginning the packing or unpacking an individual file.
<b>PartialFile</b>	Status message (possibly reported multiple times) during the process of packing or unpacking a file.
<b>FinishFile</b>	Status message after completion of the packing or unpacking an individual file.
<b>StartArchive</b>	Status message before beginning the packing or unpacking an archive.

<b>PartialArchive</b>	Status message (possibly reported multiple times) during the process of packing or unpacking an archive.
<b>FinishArchive</b>	Status message after completion of the packing or unpacking of an archive.

## **- Remarks**

PACKING EXAMPLE: The following sequence of events might be received when extracting a simple archive file with 2 files.

<b>Message Type</b>	<b>Description</b>
StartArchive	Begin extracting archive
StartFile	Begin extracting first file
PartialFile	Extracting first file
PartialFile	Extracting first file
FinishFile	Finished extracting first file
StartFile	Begin extracting second file
PartialFile	Extracting second file
FinishFile	Finished extracting second file
FinishArchive	Finished extracting archive

UNPACKING EXAMPLE: Packing 3 files into 2 archive chunks, where the second file is continued to the second archive chunk.

<b>Message Type</b>	<b>Description</b>
StartFile	Begin compressing first file

FinishFile	Finished compressing first file
StartFile	Begin compressing second file
PartialFile	Compressing second file
PartialFile	Compressing second file
FinishFile	Finished compressing second file
StartArchive	Begin writing first archive
PartialArchive	Writing first archive
FinishArchive	Finished writing first archive
StartFile	Begin compressing third file
PartialFile	Compressing third file
FinishFile	Finished compressing third file
StartArchive	Begin writing second archive
PartialArchive	Writing second archive
FinishArchive	Finished writing second archive

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# BasicUnpackStreamContext Class

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► BasicUnpackStreamContext

C# ▾

Stream context used to extract a single file from an archive into a memory stream.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public class BasicUnpackStreamContext : IUnpackStrea
```

```
Public Class BasicUnpackStreamContext _  
    Implements IUnpackStreamContext
```

```
public ref class BasicUnpackStreamContext : IUnpackS
```

## Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon	Member	Description
	<a href="#">BasicUnpackStreamContext(Stream)</a>	Creates a new BasicExtractStreamContext that reads from the specified archive stream.
	<a href="#">CloseArchiveReadStream(Int32, String, Stream)</a>	Does *not* close the stream. The archive stream should be managed by the caller.

		code that invokes the archive extraction.
≡	<a href="#">CloseFileWriteStream(String, Stream, FileAttributes, DateTime)</a>	Does *not* close the file stream. The file stream is saved in memory so it can be read later.
≡	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
📄	<a href="#">FileStream</a>	Gets the stream for the extracted file, or null if no file was extracted.
💡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current

		instance.  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">OpenArchiveReadStream(Int32, String, CompressionEngine)</a>	Opens the archive stream for reading. Returns a <a href="#">DuplicateStream</a> instance, so the stream may be virtually opened multiple times.
💡	<a href="#">OpenFileWriteStream(String, Int64, DateTime)</a>	Opens a stream for writing extracted file bytes. The returned stream is a <a href="#">MemoryStream</a> instance, so the file is extracted straight into memory.
💡	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)

## [-] Inheritance Hierarchy

[Object](#)

└ [BasicUnpackStreamContext](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

[◀]

[▶]

Deployment Tools Foundation

## BasicUnpackStreamContext

### Constructor (archiveStream)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► BasicUnpackStreamContext ►  
**BasicUnpackStreamContext(Stream)**

C# ▾

Creates a new BasicExtractStreamContext that reads from the specified archive stream.

#### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public BasicUnpackStreamContext(  
    Stream archiveStream  
)
```

```
Public Sub New (  
    archiveStream As Stream  
)
```

```
public:  
BasicUnpackStreamContext(  
    Stream^ archiveStream  
)
```

#### - Parameters

##### **archiveStream (Stream)**

Archive stream to read from.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CloseArchiveReadStream Method (archiveNumber, archiveName, stream)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► BasicUnpackStreamContext ►  
**CloseArchiveReadStream(Int32, String, Stream)**

C# ▾

Does \*not\* close the stream. The archive stream should be managed by the code that invokes the archive extraction.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void CloseArchiveReadStream(  
    int archiveNumber,  
    string archiveName,  
    Stream stream  
)
```

```
Public Sub CloseArchiveReadStream ( _  
    archiveNumber As Integer, _  
    archiveName As String, _  
    stream As Stream _  
)
```

```
public:  
virtual void CloseArchiveReadStream(  
    int archiveNumber,  
    String^ archiveName,  
    Stream^ stream  
) sealed
```

### Parameters

#### archiveNumber (Int32)

The archive number of the stream to close.

#### archiveName (String)

The name of the archive being closed.

***stream* ([Stream](#))**

The stream being closed.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CloseFileStream Method (path, stream, attributes, lastWriteTime)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► BasicUnpackStreamContext ►  
**CloseFileStream(String, Stream, FileAttributes, DateTime)**

C# ▾

Does \*not\* close the file stream. The file stream is saved in memory so can be read later.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void CloseFileStream(  
    string path,  
    Stream stream,  
    FileAttributes attributes,  
    DateTime lastWriteTime  
)
```

```
Public Sub CloseFileStream ( _  
    path As String, _  
    stream As Stream, _  
    attributes As FileAttributes, _  
    lastWriteTime As DateTime _  
)
```

```
public:  
virtual void CloseFileStream(  
    String^ path,  
    Stream^ stream,  
    FileAttributes attributes,  
    DateTime lastWriteTime  
) sealed
```

### - Parameters

*path* (String)

Path of the file within the archive.

***stream* ([Stream](#))**

The file stream to be closed.

***attributes* ([FileAttributes](#))**

The attributes of the extracted file.

***lastWriteTime* ([DateTime](#))**

The last write time of the file.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## FileStream Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► BasicUnpackStreamContext ►

### FileStream

C#

Gets the stream for the extracted file, or null if no file was extracted.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public Stream FileStream { get; }
```

```
Public ReadOnly Property FileStream As Stream
```

```
public:  
property Stream^ FileStream {  
    Stream^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## OpenArchiveReadStream Method (archiveNumber, archiveName, compressionEngine)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► BasicUnpackStreamContext ►  
**OpenArchiveReadStream(Int32, String, CompressionEngine)**

C# ▾

Opens the archive stream for reading. Returns a `DuplicateStream` instance, so the stream may be virtually opened multiple times.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public Stream OpenArchiveReadStream(  
    int archiveNumber,  
    string archiveName,  
    CompressionEngine compressionEngine  
)
```

```
Public Function OpenArchiveReadStream ( _  
    archiveNumber As Integer, _  
    archiveName As String, _  
    compressionEngine As CompressionEngine _  
) As Stream
```

```
public:  
virtual Stream^ OpenArchiveReadStream(  
    int archiveNumber,  
    String^ archiveName,  
    CompressionEngine^ compressionEngine  
) sealed
```

### Parameters

#### **archiveNumber (Int32)**

The archive number to open (ignored; 0 is assumed).

***archiveName* ([String](#))**

The name of the archive being opened.

***compressionEngine* ([CompressionEngine](#))**

Instance of the compression engine doing the operations.

**Return Value**

A stream from which archive bytes are read.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **OpenFileStream Method**  
**(path, fileSize, lastWriteTime)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► BasicUnpackStreamContext ►  
**OpenFileStream(String, Int64, DateTime)**

C# ▾

Opens a stream for writing extracted file bytes. The returned stream is a `MemoryStream` instance, so the file is extracted straight into memory.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public Stream OpenFileStream(
    string path,
    long fileSize,
    DateTime lastWriteTime
)
```

```
Public Function OpenFileStream ( _
    path As String, _
    fileSize As Long, _
    lastWriteTime As DateTime _ 
) As Stream
```

```
public:
virtual Stream^ OpenFileStream(
    String^ path,
    long long fileSize,
    DateTime lastWriteTime
) sealed
```

[-] **Parameters**

***path* (String)**

Path of the file within the archive.

***fileSize* (Int64)**

The uncompressed size of the file to be extracted.

***lastWriteTime* (DateTime)**

The last write time of the file.

**Return Value**

A stream where extracted file bytes are to be written.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **CargoStream Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► **CargoStream**

C#

Wraps a source stream and carries additional items that are disposed when the stream is closed.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public class CargoStream : Stream
```

```
Public Class CargoStream
    Inherits Stream
```

```
public ref class CargoStream : public Stream
```

### Members

All Members	Constructors	Methods	Properties	
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static		<input checked="" type="checkbox"/> Declare <input checked="" type="checkbox"/> Inherit

Icon	Member	Description
≡	<a href="#">CargoStream(Stream, IDisposable[])</a>	Creates a new a cargo stream.
≡	<a href="#">BeginRead(Byte[], Int32, Int32, AsyncCallback, Object)</a>	HostProtectionAttribute. Begins an asynchronous read operation.  (Inherited from <a href="#">Stream</a> .)
≡	<a href="#">BeginWrite(Byte[], Int32, Int32, AsyncCallback, Object)</a>	HostProtectionAttribute. Begins an asynchronous write operation.

		(Inherited from <a href="#">Stream</a> .)
	<a href="#">CanRead</a>	Gets a value indicating whether the source stream supports reading.  (Overrides <a href="#">Stream.CanRead</a> .)
	<a href="#">CanSeek</a>	Gets a value indicating whether the source stream supports seeking.  (Overrides <a href="#">Stream.CanSeek</a> .)
	<a href="#">CanTimeout</a>	Gets a value that determines whether the current stream can time out.  (Inherited from <a href="#">Stream</a> .)
	<a href="#">CanWrite</a>	Gets a value indicating whether the source stream supports writing.  (Overrides <a href="#">Stream.CanWrite</a> .)
	<a href="#">Cargo</a>	Gets the list of additional items that are disposed when the stream is closed. The order of the list is the order in which the items are disposed. The contents can be modified any time.
	<a href="#">Close()</a>	Closes the source stream and also closes the additional objects that are carried.  (Overrides <a href="#">Stream.Close</a> .)
	<a href="#">CreateObjRef(Type)</a>	Creates an object that contains all the relevant information required to generate a proxy used to communicate with a remote object.

		(Inherited from <a href="#">MarshalByRefObject</a> .)
💡	<a href="#">CreateWaitHandle()</a>	<b>Obsolete.</b> Allocates a <a href="#">WaitHandle</a> object.  (Inherited from <a href="#">Stream</a> .)
💡	<a href="#">Dispose()</a>	Releases all resources used by the <a href="#">Stream</a> .  (Inherited from <a href="#">Stream</a> .)
💡	<a href="#">Dispose(Boolean)</a>	Releases the unmanaged resource used by the <a href="#">Stream</a> and optionally releases the managed resources.  (Inherited from <a href="#">Stream</a> .)
💡	<a href="#">EndRead(IAsyncResult)</a>	Waits for the pending asynchronous read to complete.  (Inherited from <a href="#">Stream</a> .)
💡	<a href="#">EndWrite(IAsyncResult)</a>	Ends an asynchronous write operation.  (Inherited from <a href="#">Stream</a> .)
💡	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
💡		

	<a href="#">Flush()</a>	Flushes the source stream.  (Overrides <a href="#">Stream.Flush()</a> .)
≡	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like hash table.  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">GetLifetimeService()</a>	Retrieves the current lifetime service object that controls the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject</a> .)
≡	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">InitializeLifetimeService()</a>	Obtains a lifetime service object to control the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">Length</a>	Gets the length of the source stream.  (Overrides <a href="#">Stream.Length</a> .)
💡	<a href="#">MemberwiseClone(Boolean)</a>	Creates a shallow copy of the current <a href="#">MarshalByRefObject</a> object  (Inherited from <a href="#">MarshalByRefObject</a> .)
💡	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the

		current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Position</a>	Gets or sets the position of the source stream.  (Overrides <a href="#">Stream.Position</a> .)
	<a href="#">Read(Byte[], Int32, Int32)</a>	Reads from the source stream.  (Overrides <a href="#">Stream.Read(Byte[], Int32, Int32)</a> .)
	<a href="#">ReadByte()</a>	Reads a byte from the stream and advances the position within the stream by one byte, or returns -1 if at the end of the stream.  (Inherited from <a href="#">Stream</a> .)
	<a href="#">ReadTimeout</a>	Gets or sets a value that determine how long the stream will attempt to read before timing out.  (Inherited from <a href="#">Stream</a> .)
	<a href="#">Seek(Int64, SeekOrigin)</a>	Changes the position of the source stream.  (Overrides <a href="#">Stream.Seek(Int64, SeekOrigin)</a> .)
	<a href="#">SetLength(Int64)</a>	Sets the length of the source stream.  (Overrides <a href="#">Stream.SetLength(Int64)</a> .)
	<a href="#">Source</a>	Gets the source stream of the cargo stream.

	<a href="#">ToString()</a>	Returns a <code>String</code> that represents the current <code>Object</code> .  (Inherited from <code>Object</code> .)
	<a href="#">Write(Byte[], Int32, Int32)</a>	Writes to the source stream.  (Overrides <code>Stream.Write(Byte[], Int32, Int32)</code> .)
	<a href="#">WriteByte(Byte)</a>	Writes a byte to the current position in the stream and advances the position within the stream by one byte.  (Inherited from <code>Stream</code> .)
	<a href="#">WriteTimeout</a>	Gets or sets a value that determines how long the stream will attempt to write before timing out.  (Inherited from <code>Stream</code> .)

## Inheritance Hierarchy

```

Object
└ MarshalByRefObject
  └ Stream
    └ CargoStream

```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# CargoStream Constructor (source, cargo)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► CargoStream ►  
**CargoStream(Stream, IDisposable[])**

C# ▾

Creates a new a cargo stream.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public CargoStream(  
    Stream source,  
    params IDisposable[] cargo  
)
```

```
Public Sub New ( _  
    source As Stream, _  
    ParamArray cargo As IDisposable() _  
)
```

```
public:  
CargoStream(  
    Stream^ source,  
    ... array<IDisposable^>^ cargo  
)
```

## Parameters

### source (Stream)

source of the stream

### cargo ( IDisposable [] )

List of additional items that are disposed when the stream is closed  
The order of the list is the order in which the items are disposed.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CanRead Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CargoStream ► CanRead

C#

Gets a value indicating whether the source stream supports reading.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public override bool CanRead { get; }
```

```
Public Overrides ReadOnly Property CanRead As Boolean
```

```
public:  
virtual property bool CanRead {  
    bool get () override;  
}
```

### Value

true if the stream supports reading; otherwise, false.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CanSeek Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CargoStream ► CanSeek

C#

Gets a value indicating whether the source stream supports seeking.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public override bool CanSeek { get; }
```

```
Public Overrides ReadOnly Property CanSeek As Boolean
```

```
public:  
virtual property bool CanSeek {  
    bool get () override;  
}
```

### Value

true if the stream supports seeking; otherwise, false.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CanWrite Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CargoStream ► CanWrite

C# ▾

Gets a value indicating whether the source stream supports writing.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override bool CanWrite { get; }
```

```
Public Overrides ReadOnly Property CanWrite As Boolean
```

```
public:  
virtual property bool CanWrite {  
    bool get () override;  
}
```

### Value

true if the stream supports writing; otherwise, false.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Cargo Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CargoStream ► Cargo

C# ▾

Gets the list of additional items that are disposed when the stream is closed. The order of the list is the order in which the items are disposed. The contents can be modified any time.

▀ Declaration Syntax

C#

Visual Basic

Visual C++

```
public IList<IDisposable> Cargo { get; }
```

```
Public ReadOnly Property Cargo As IList(Of IDisposable)
```

```
public:  
property IList<IDisposable^>^ Cargo {  
    IList<IDisposable^>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

C# ▾

Closes the source stream and also closes the additional objects that are carried.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public override void Close()
```

```
Public Overrides Sub Close
```

```
public:  
    virtual void Close() override
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

C# ▾

Flushes the source stream.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public override void Flush()
```

```
Public Overrides Sub Flush
```

```
public:  
virtual void Flush() override
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# Length Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CargoStream ► Length

C# ▾

Gets the length of the source stream.

## - Declaration Syntax

C#

Visual Basic

Visual C++

```
public override long Length { get; }
```

```
Public Overrides ReadOnly Property Length As Long
```

```
public:  
virtual property long long Length {  
    long long get () override;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# Position Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CargoStream ► Position

C# ▾

Gets or sets the position of the source stream.

## - Declaration Syntax

C#

Visual Basic

Visual C++

```
public override long Position { get; set; }
```

```
Public Overrides Property Position As Long
```

```
public:  
virtual property long long Position {  
    long long get () override;  
    void set (long long value) override;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Read Method (buffer, offset, count)

Deployment Tools Foundation Namespaces ► Microsoft.Deployment.Compression ► CargoStream ► **Read(Byte[], Int32, Int32)**

Reads from the source stream.

C# ▾

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public override int Read(
    byte[] buffer,
    int offset,
    int count
)
```

```
Public Overrides Function Read ( _
    buffer As Byte(), _
    offset As Integer, _
    count As Integer _)
) As Integer
```

```
public:
virtual int Read(
    array<unsigned char>^ buffer,
    int offset,
    int count
) override
```

## Parameters

### **buffer ( Byte [] )**

An array of bytes. When this method returns, the buffer contains the specified byte array with the values between offset and (offset + count - 1) replaced by the bytes read from the source.

***offset* ([Int32](#))**

The zero-based byte offset in buffer at which to begin storing the data read from the stream.

***count* ([Int32](#))**

The maximum number of bytes to be read from the stream.

**[-] Return Value**

The total number of bytes read into the buffer. This can be less than the number of bytes requested if that many bytes are not currently available or zero (0) if the end of the stream has been reached.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Seek Method (offset, origin)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CargoStream ► Seek(Int64, SeekOrigin)

C# ▾

Changes the position of the source stream.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public override long Seek(  
    long offset,  
    SeekOrigin origin  
)
```

```
Public Overrides Function Seek ( _  
    offset As Long, _  
    origin As SeekOrigin _  
) As Long
```

```
public:  
virtual long long Seek(  
    long long offset,  
    SeekOrigin origin  
) override
```

## Parameters

### offset (Int64)

A byte offset relative to the origin parameter.

### origin (SeekOrigin)

A value of type SeekOrigin indicating the reference point used to obtain the new position.

## Return Value

The new position within the stream.

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:  
Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## SetLength Method (value)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CargoStream ►

### SetLength(Int64)

C# ▾

Sets the length of the source stream.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override void SetLength(  
    long value  
)
```

```
Public Overrides Sub SetLength ( _  
    value As Long _  
)
```

```
public:  
virtual void SetLength(  
    long long value  
) override
```

#### Parameters

##### **value (Int64)**

The desired length of the stream in bytes.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Source Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CargoStream ► Source

C# ▾

Gets the source stream of the cargo stream.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public Stream Source { get; }
```

```
Public ReadOnly Property Source As Stream
```

```
public:  
property Stream^ Source {  
    Stream^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Write Method (buffer, offset, count)

Deployment Tools Foundation Namespaces ► Microsoft.Deployment.Compression ► CargoStream ► Write(Byte[], Int32, Int32)

Writes to the source stream.

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public override void Write(  
    byte[] buffer,  
    int offset,  
    int count  
)
```

```
Public Overrides Sub Write ( _  
    buffer As Byte(), _  
    offset As Integer, _  
    count As Integer _  
)
```

```
public:  
virtual void Write(  
    array<unsigned char>^ buffer,  
    int offset,  
    int count  
) override
```

### Parameters

#### buffer (Byte [])

An array of bytes. This method copies count bytes from buffer to the stream.

#### offset (Int32)

The zero-based byte offset in buffer at which to begin copying bytes to the stream.

***count* ([Int32](#))**

The number of bytes to be written to the stream.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **CompressionEngine Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► **CompressionEngine**

C# ▾

Base class for an engine capable of packing and unpacking a particular compressed file format.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public abstract class CompressionEngine : IDisposable
```

```
Public MustInherit Class CompressionEngine _
    Implements IDisposable
```

```
public ref class CompressionEngine abstract : IDisposable
```

## Members

All Members	Constructors	Methods	Properties	Events
<input checked="" type="checkbox"/> Public		<input checked="" type="checkbox"/> Instance		<input checked="" type="checkbox"/> Declared
<input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Static		<input checked="" type="checkbox"/> Inherited

Icon	Member	Description
💡	<a href="#">CompressionEngine()</a>	Creates a new instance of the compression engine base class.
📝	<a href="#">CompressionLevel</a>	Compression level to use when compressing files.
⌚️	<a href="#">DateTimeToDosDateAndTime(DateTime, Int16, Int16)</a>	Comprision utility function for converting

		a DateTime structure to old-style date and time values.
≡	<a href="#">Dispose()</a>	Disposes of resources allocated by the compression engine.
≡	<a href="#">Dispose(Boolean)</a>	Disposes of resources allocated by the compression engine.
≡ S	<a href="#">DosDateAndTimeToDateTime(Int16, Int16, DateTime)</a>	Compression utility function for converting old-style date and time values to a DateTime structure.
≡	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">Finalize()</a>	Disposes the compression engine.  (Overrides <a href="#">Object.Finalize()</a> .)
≡	<a href="#">FindArchiveOffset(Stream)</a>	Gets the offset of an archive that is positioned 0 or more bytes from the start of

		the Stream.
≡	<a href="#">GetFileInfo(Stream)</a>	Gets information about all files in an archive stream.
≡	<a href="#">GetFileInfo(IUnpackStreamContext, Predicate&lt;String&gt;)</a>	Gets information about files in an archive or archive chain.
≡	<a href="#">GetFiles(Stream)</a>	Gets the list of files in an archive Stream.
≡	<a href="#">GetFiles(IUnpackStreamContext, Predicate&lt;String&gt;)</a>	Gets the list of files in an archive or archive chain.
≡	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)

≡	<a href="#">IsArchive(Stream)</a>	Checks whether a Stream begins with a header that indicates it is a valid archive.
💡	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">OnProgress(ArchiveProgressEventArgs)</a>	Called by subclasses to distribute a packing or unpacking progress event to listeners.
≡	<a href="#">Pack(IPackStreamContext, IEnumerable&lt;String&gt;)</a>	Creates an archive.
≡	<a href="#">Pack(IPackStreamContext, IEnumerable&lt;String&gt;, Int64)</a>	Creates an archive or chain of archives.
⚡	<a href="#">Progress</a>	Occurs when the compression engine reports progress in packing or unpacking an archive.
≡	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">Unpack(Stream, String)</a>	Reads a single file from

		an archive stream.
➤	<a href="#">Unpack(IUnpackStreamContext, Predicate&lt;String&gt;)</a>	Extracts files from an archive or archive chain.
📁	<a href="#">UseTempFiles</a>	Gets or sets a flag indicating whether temporary files are created and used during compression.

## [-] Inheritance Hierarchy

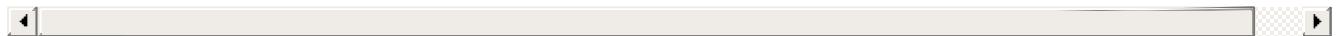
[Object](#)

- └ **CompressionEngine**
  - └ [CabEngine](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

# CompressionEngine Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CompressionEngine ►

**CompressionEngine()**

C#

Creates a new instance of the compression engine base class.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected CompressionEngine()
```

```
Protected Sub New
```

```
protected:  
CompressionEngine()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **CompressionLevel Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CompressionEngine ►

**CompressionLevel**

C# ▾

Compression level to use when compressing files.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public CompressionLevel CompressionLevel { get; set;
```

```
Public Property CompressionLevel As CompressionLevel
```

```
public:  
property CompressionLevel CompressionLevel {  
    CompressionLevel get ();  
    void set (CompressionLevel value);  
}
```

[-] **Value**

A compression level ranging from minimum to maximum compression, (no compression).

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **DateTimeToDosDateAndTime**

## Method (**dateTime**, **dosDate**, **dosTime**)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CompressionEngine ►

**DateTimeToDosDateAndTime(DateTime, Int16, Int16)**

C#

Compression utility function for converting a DateTime structure to old-style date and time values.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static void DateTimeToDosDateAndTime(
    DateTime dateTime,
    out short dosDate,
    out short dosTime
)
```

```
Public Shared Sub DateTimeToDosDateAndTime ( _
    dateTime As DateTime, _
    <OutAttribute> ByRef dosDate As Short, _
    <OutAttribute> ByRef dosTime As Short _
)
```

```
public:
static void DateTimeToDosDateAndTime(
    DateTime dateTime,
    [OutAttribute] short% dosDate,
    [OutAttribute] short% dosTime
)
```

### - Parameters

**dateTime** ([DateTime](#))

**dosDate** ([Int16](#))

**dosTime** ([Int16](#))

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:  
Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Dispose Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CompressionEngine ►

### Dispose()

C# ▾

#### ▀ Members

Icon	Member	Description
≡	<a href="#">Dispose()</a>	Disposes of resources allocated by the compression engine.
💡	<a href="#">Dispose(Boolean)</a>	Disposes of resources allocated by the compression engine.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Dispose Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CompressionEngine ►

### Dispose()

C#

Disposes of resources allocated by the compression engine.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Dispose()
```

```
Public Sub Dispose
```

```
public:  
virtual void Dispose() sealed
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Dispose Method (disposing)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CompressionEngine ►

## Dispose(Boolean)

C# ▾

Disposes of resources allocated by the compression engine.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected virtual void Dispose(  
    bool disposing  
)
```

```
Protected Overridable Sub Dispose ( _  
    disposing As Boolean _  
)
```

```
protected:  
virtual void Dispose(  
    bool disposing  
)
```

### Parameters

#### disposing (Boolean)

If true, the method has been called directly or indirectly by a user's code, so managed and unmanaged resources will be disposed. If false, the method has been called by the runtime from inside the finalizer, and only unmanaged resources will be disposed.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **DosDateAndTimeToDateTime**

## Method (dosDate, dosTime, dateTime)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CompressionEngine ►

**DosDateAndTimeToDateTime(Int16, Int16, DateTime)**

C# ▾

Compression utility function for converting old-style date and time values to a DateTime structure.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static void DosDateAndTimeToDateTime(
    short dosDate,
    short dosTime,
    out DateTime dateTime
)
```

```
Public Shared Sub DosDateAndTimeToDateTime ( _
    dosDate As Short, _
    dosTime As Short, _
    <OutAttribute> ByRef dateTime As DateTime _
)
```

```
public:
static void DosDateAndTimeToDateTime(
    short dosDate,
    short dosTime,
    [OutAttribute] DateTime% dateTime
)
```

### Parameters

**dosDate (Int16)**

**dosTime (Int16)**

**dateTime ( DateTime )**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:  
Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

C# ▾

Disposes the compression engine.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
protected override void Finalize()
```

```
Protected Overrides Sub Finalize
```

```
protected:  
virtual void Finalize() override
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## FindArchiveOffset Method (*stream*)

Deployment Tools Foundation Namespaces ►  
 Microsoft.Deployment.Compression ► CompressionEngine ►  
**FindArchiveOffset(Stream)**

C# ▾

Gets the offset of an archive that is positioned 0 or more bytes from the start of the Stream.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public virtual long FindArchiveOffset(
    Stream stream
)
```

```
Public Overridable Function FindArchiveOffset ( _
    stream As Stream _ 
) As Long
```

```
public:
virtual long long FindArchiveOffset(
    Stream^ stream
)
```

### [-] Parameters

#### *stream* (**Stream**)

A stream for reading the archive.

### [-] Return Value

The offset in bytes of the archive, or -1 if no archive is found in the Stream.

### [-] Remarks

The archive must begin on a 4-byte boundary.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:  
Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetFileInfo Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CompressionEngine ►

**GetFileInfo()**

C# ▾

▀ **Members**

Icon	Member	Description
≡	<a href="#">GetFileInfo(Stream)</a>	Gets information about all files in an archive stream.
≡	<a href="#">GetFileInfo(IUnpackStreamContext, Predicate&lt;String&gt;)</a>	Gets information about files in an archive or archive chain.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetFileInfo Method**  
**(streamContext, fileFilter)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► CompressionEngine ►  
GetFileInfo(IUnpackStreamContext, Predicate<String>)

C#

Gets information about files in an archive or archive chain.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public abstract IList<ArchiveFileInfo> GetFileInfo(  
    IUnpackStreamContext streamContext,  
    Predicate<string> fileFilter  
)
```

```
Public MustOverride Function GetFileInfo ( _  
    streamContext As IUnpackStreamContext, _  
    fileFilter As Predicate(Of String) _  
) As IList(Of ArchiveFileInfo)
```

```
public:  
virtual IList<ArchiveFileInfo^>^ GetFileInfo(  
    IUnpackStreamContext^ streamContext,  
    Predicate<String^>^ fileFilter  
) abstract
```

[-] **Parameters**

**streamContext (IUnpackStreamContext)**

A context interface to handle opening and closing of archive and file streams.

**fileFilter (Predicate<String>)**

A predicate that can determine which files to process, optional.

[-] **Return Value**

Information about files in the archive stream.

## **Remarks**

The *fileFilter* predicate takes an internal file path and returns true to include the file or false to exclude it.

## **Exceptions**

Exception	Condition
<a href="#">ArchiveException</a>	The archive provided by the stream context is not valid.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **GetFileInfo Method (stream)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CompressionEngine ►

## **GetFileInfo(Stream)**

C# ▾

Gets information about all files in an archive stream.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public IList<ArchiveFileInfo> GetFileInfo(  
    Stream stream  
)
```

```
Public Function GetFileInfo ( _  
    stream As Stream _  
) As IList(Of ArchiveFileInfo)
```

```
public:  
IList<ArchiveFileInfo^>^ GetFileInfo(  
    Stream^ stream  
)
```

### [-] Parameters

#### **stream (Stream)**

A stream for reading the archive.

### [-] Return Value

Information about all files in the archive stream.

### [-] Exceptions

Exception	Condition
ArchiveException	The stream is not a valid archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:  
Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetFiles Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CompressionEngine ►

**GetFiles()**

C# ▾

▀ **Members**

Icon	Member	Description
≡	<a href="#">GetFiles(Stream)</a>	Gets the list of files in an archive Stream.
≡	<a href="#">GetFiles(IUnpackStreamContext, Predicate&lt;String&gt;)</a>	Gets the list of files in an archive or archive chain.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetFiles Method (streamContext, fileFilter)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► CompressionEngine ►  
**GetFiles(IUnpackStreamContext, Predicate<String>)**

C# ▾

Gets the list of files in an archive or archive chain.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public IList<string> GetFiles(  
    IUnpackStreamContext streamContext,  
    Predicate<string> fileFilter  
)
```

```
Public Function GetFiles ( _  
    streamContext As IUnpackStreamContext, _  
    fileFilter As Predicate(Of String) _  
) As IList(Of String)
```

```
public:  
IList<String^>^ GetFiles(  
    IUnpackStreamContext^ streamContext,  
    Predicate<String^>^ fileFilter  
)
```

[-] **Parameters**

**streamContext (IUnpackStreamContext)**

A context interface to handle opening and closing of archive and file streams.

**fileFilter (Predicate<String>)**

A predicate that can determine which files to process, optional.

[-] **Return Value**

An array containing the names of all files contained in the archive or archive chain.

## **Remarks**

The *fileFilter* predicate takes an internal file path and returns true to include the file or false to exclude it.

## **Exceptions**

Exception	Condition
ArchiveException	The archive provided by the stream context is not valid.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **GetFiles Method (stream)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CompressionEngine ►

## **GetFiles(Stream)**

C# ▾

Gets the list of files in an archive Stream.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public IList<string> GetFiles(  
    Stream stream  
)
```

```
Public Function GetFiles ( _  
    stream As Stream _  
) As IList(of String)
```

```
public:  
IList<String^>^ GetFiles(  
    Stream^ stream  
)
```

### [-] Parameters

#### **stream (Stream)**

A stream for reading the archive.

### [-] Return Value

A list of the paths of all files contained in the archive.

### [-] Exceptions

Exception	Condition
ArchiveException	The stream is not a valid archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:  
Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation IsArchive Method (stream)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CompressionEngine ►

### IsArchive(Stream)

C#

Checks whether a Stream begins with a header that indicates it is a valid archive.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public abstract bool IsArchive(  
    Stream stream  
)
```

```
Public MustOverride Function IsArchive ( _  
    stream As Stream _  
) As Boolean
```

```
public:  
virtual bool IsArchive(  
    Stream^ stream  
) abstract
```

#### Parameters

##### stream (Stream)

Stream for reading the archive file.

#### Return Value

True if the stream is a valid archive (with no offset); false otherwise.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **OnProgress Method (e)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CompressionEngine ►

**OnProgress(ArchiveEventArgs)**

C# ▾

Called by subclasses to distribute a packing or unpacking progress event to listeners.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected void OnProgress(  
    ArchiveEventArgs e  
)
```

```
Protected Sub OnProgress ( _  
    e As ArchiveEventArgs _  
)
```

```
protected:  
void OnProgress(  
    ArchiveEventArgs^ e  
)
```

### Parameters

**e (ArchiveEventArgs)**

Event details.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Pack Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CompressionEngine ► Pack()

C# ▾

### Members

Icon	Member	Description
	Pack(IPackStreamContext, IEnumerable<String>)	Creates an archive.
	Pack(IPackStreamContext, IEnumerable<String>, Int64)	Creates an archive or chain of archives.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Pack Method (streamContext, files)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► CompressionEngine ►  
**Pack(IPackStreamContext, IEnumerable<String>)**

C# ▾

Creates an archive.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Pack(  
    IPackStreamContext streamContext,  
    IEnumerable<string> files  
)
```

```
Public Sub Pack ( _  
    streamContext As IPackStreamContext, _  
    files As IEnumerable(Of String) _  
)
```

```
public:  
void Pack(  
    IPackStreamContext^ streamContext,  
    IEnumerable<String^>^ files  
)
```

### Parameters

#### streamContext (IPackStreamContext)

A context interface to handle opening and closing of archive and file streams.

#### files (IEnumerable<String>)

The paths of the files in the archive (not external file paths).

### Remarks

The stream context implementation may provide a mapping from the file paths within the archive to the external file paths.

## **Exceptions**

Exception	Condition
ArchiveException	The archive could not be created.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Pack Method (streamContext, files, maxArchiveSize)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► CompressionEngine ►  
**Pack(IPackStreamContext, IEnumerable<String>, Int64)**

C# ▾

Creates an archive or chain of archives.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public abstract void Pack(  
    IPackStreamContext streamContext,  
    IEnumerable<string> files,  
    long maxArchiveSize  
)
```

```
Public MustOverride Sub Pack ( _  
    streamContext As IPackStreamContext, _  
    files As IEnumerable(Of String), _  
    maxArchiveSize As Long _  
)
```

```
public:  
virtual void Pack(  
    IPackStreamContext^ streamContext,  
    IEnumerable<String^>^ files,  
    long long maxArchiveSize  
) abstract
```

[-] **Parameters**

**streamContext (IPackStreamContext)**

A context interface to handle opening and closing of archive and file streams.

**files (IEnumerable<String>)**

The paths of the files in the archive (not external file paths).

### ***maxArchiveSize (Int64)***

The maximum number of bytes for one archive before the contents are chained to the next archive, or zero for unlimited archive size.

## **Remarks**

The stream context implementation may provide a mapping from the file paths within the archive to the external file paths.

## **Exceptions**

Exception	Condition
ArchiveException	The archive could not be created.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# Progress Event

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CompressionEngine ►

## Progress

Occurs when the compression engine reports progress in packing or unpacking an archive.

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public event EventHandler<ArchiveProgressEventArgs>
```

```
Public Event Progress As EventHandler(Of ArchiveProg
```

```
public:  
    event EventHandler<ArchiveProgressEventArgs^>^ Prog  
        void add (EventHandler<ArchiveProgressEventArgs^>  
        void remove (EventHandler<ArchiveProgressEventArgs^>  
}
```

### See Also

[ArchiveProgressType](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

# Unpack Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CompressionEngine ►

## Unpack()

C# ▾

### Members

Icon	Member	Description
≡	<a href="#">Unpack(Stream, String)</a>	Reads a single file from an archive stream.
≡	<a href="#">Unpack(IUnpackStreamContext, Predicate&lt;String&gt;)</a>	Extracts files from an archive or archive chain.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Unpack Method (streamContext, fileFilter)

Deployment Tools Foundation Namespaces ►  
 Microsoft.Deployment.Compression ► CompressionEngine ►  
**Unpack(IUnpackStreamContext, Predicate<String>)**

C# ▾

Extracts files from an archive or archive chain.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public abstract void Unpack(
    IUnpackStreamContext streamContext,
    Predicate<string> fileFilter
)
```

```
Public MustOverride Sub Unpack ( _
    streamContext As IUnpackStreamContext, _
    fileFilter As Predicate(Of String) _
)
```

```
public:
virtual void Unpack(
    IUnpackStreamContext^ streamContext,
    Predicate<String^>^ fileFilter
) abstract
```

## Parameters

### **streamContext (IUnpackStreamContext)**

A context interface to handle opening and closing of archive and file streams.

### **fileFilter (Predicate<String>)**

An optional predicate that can determine which files to process.

## Remarks

The *fileFilter* predicate takes an internal file path and returns true to include the file or false to exclude it.

## Exceptions

Exception	Condition
ArchiveException	The archive provided by the stream context is not valid.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Unpack Method (stream, path)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CompressionEngine ►

## Unpack(Stream, String)

C# ▾

Reads a single file from an archive stream.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public Stream Unpack(  
    Stream stream,  
    string path  
)
```

```
Public Function Unpack ( _  
    stream As Stream, _  
    path As String _  
) As Stream
```

```
public:  
Stream^ Unpack(  
    Stream^ stream,  
    String^ path  
)
```

### Parameters

#### ***stream* (Stream)**

A stream for reading the archive.

#### ***path* (String)**

The path of the file within the archive (not the external file path).

### Return Value

A stream for reading the extracted file, or null if the file does not exist in the archive.

## **Remarks**

The entire extracted file is cached in memory, so this method requires enough free memory to hold the file.

## **Exceptions**

Exception	Condition
<a href="#">ArchiveException</a>	The stream is not a valid archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **UseTempFiles** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► CompressionEngine ►

### UseTempFiles

C# ▾

Gets or sets a flag indicating whether temporary files are created and used during compression.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool UseTempFiles { get; set; }
```

Public Property UseTempFiles As Boolean

```
public:  
property bool UseTempFiles {  
    bool get ();  
    void set (bool value);  
}
```

#### Value

True if temporary files are used; false if compression is done entirely in-memory.

#### Remarks

The value of this property is true by default. Using temporary files can greatly reduce the memory requirement of compression, especially when compressing large archives. However, setting this property to false may yield slightly better performance when creating small archives. Or it may be necessary if the process does not have sufficient privileges to create temporary files.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **CompressionLevel Enumeration**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► **CompressionLevel**

C# ▾

Specifies the compression level ranging from minimum compression to maximum compression, or no compression at all.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public enum CompressionLevel
```

```
Public Enumeration CompressionLevel
```

```
public enum class CompressionLevel
```

[-] **Members**

Member	Description
<b>None</b>	Do not compress files, only store.
<b>Min</b>	Minimum compression; fastest.
<b>Normal</b>	A compromise between speed and compression efficiency.
<b>Max</b>	Maximum compression; slowest.

[-] **Remarks**

Although only four values are enumerated, any integral value between

**Min** and **Max** can also be used.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation DuplicateStream Class

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► DuplicateStream

C#

Duplicates a source stream by maintaining a separate position.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public class DuplicateStream : Stream
```

```
Public Class DuplicateStream _  
    Inherits Stream
```

```
public ref class DuplicateStream : public Stream
```

## Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon

Member

Description



DuplicateStream(Stream)

Creates a new duplicate of a stream.



BeginRead(Byte[], Int32, Int32, AsyncCallback, Object)

HostProtectionAttribute.  
Begins an asynchronous read operation.  
(Inherited from Stream.)



BeginWrite(Byte[], Int32, Int32, AsyncCallback, Object)

HostProtectionAttribute.  
Begins an asynchronous write operation.

		(Inherited from <a href="#">Stream</a> .)
	<a href="#">CanRead</a>	Gets a value indicating whether the source stream supports reading.  (Overrides <a href="#">Stream.CanRead</a> .)
	<a href="#">CanSeek</a>	Gets a value indicating whether the source stream supports seeking.  (Overrides <a href="#">Stream.CanSeek</a> .)
	<a href="#">CanTimeout</a>	Gets a value that determines whether the current stream can time out.  (Inherited from <a href="#">Stream</a> .)
	<a href="#">CanWrite</a>	Gets a value indicating whether the source stream supports writing.  (Overrides <a href="#">Stream.CanWrite</a> .)
	<a href="#">Close()</a>	Closes the underlying stream, effectively closing ALL duplicates.  (Overrides <a href="#">Stream.Close</a> .)
	<a href="#">CreateObjRef(Type)</a>	Creates an object that contains all the relevant information required to generate a proxy used to communicate with a remote object.  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">CreateWaitHandle()</a>	<b>Obsolete.</b> Allocates a <a href="#">WaitHandle</a> object.  (Inherited from <a href="#">Stream</a> .)
	<a href="#">Dispose()</a>	Releases all resources used by the <a href="#">Stream</a> .

		(Inherited from <a href="#">Stream</a> .)
💡	<a href="#">Dispose(Boolean)</a>	Releases the unmanaged resource used by the <a href="#">Stream</a> and optionally releases the managed resources.
		(Inherited from <a href="#">Stream</a> .)
💡	<a href="#">EndRead(IAsyncResult)</a>	Waits for the pending asynchronous read to complete.
		(Inherited from <a href="#">Stream</a> .)
💡	<a href="#">EndWrite(IAsyncResult)</a>	Ends an asynchronous write operation.
		(Inherited from <a href="#">Stream</a> .)
💡	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .
		(Inherited from <a href="#">Object</a> .)
💡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.
		(Inherited from <a href="#">Object</a> .)
💡	<a href="#">Flush()</a>	Flushes the source stream.
		(Overrides <a href="#">Stream.Flush()</a> .)
💡	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like hash table.
		(Inherited from <a href="#">Object</a> .)

 	<a href="#">GetLifetimeService()</a>	Retrieves the current lifetime service object that controls the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject</a> .)
 	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
 	<a href="#">InitializeLifetimeService()</a>	Obtains a lifetime service object to control the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject</a> .)
 	<a href="#">Length</a>	Gets the length of the source stream.  (Overrides <a href="#">Stream.Length</a> .)
 	<a href="#">MemberwiseClone(Boolean)</a>	Creates a shallow copy of the current <a href="#">MarshalByRefObject</a> object  (Inherited from <a href="#">MarshalByRefObject</a> .)
 	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
 	<a href="#">OriginalStream(Stream)</a>	Retrieves the original stream from a possible duplicate stream.
 	<a href="#">Position</a>	Gets or sets the position of the current stream, ignoring the position of the source stream.

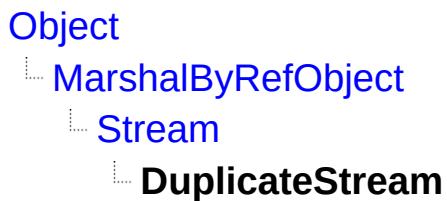
		(Overrides <a href="#">Stream.Position</a> .)
	<a href="#">Read(Byte[], Int32, Int32)</a>	Reads from the source stream while maintaining a separate position and not impacting the source stream's position.  (Overrides <a href="#">Stream.Read(Byte[], Int32, Int32)</a> .)
	<a href="#">ReadByte()</a>	Reads a byte from the stream and advances the position within the stream by one byte, or returns -1 if at the end of the stream.  (Inherited from <a href="#">Stream</a> .)
	<a href="#">ReadTimeout</a>	Gets or sets a value that determine how long the stream will attempt to read before timing out.  (Inherited from <a href="#">Stream</a> .)
	<a href="#">Seek(Int64, SeekOrigin)</a>	Changes the position of this stream without impacting the source stream's position.  (Overrides <a href="#">Stream.Seek(Int64, SeekOrigin)</a> .)
	<a href="#">SetLength(Int64)</a>	Sets the length of the source stream.  (Overrides <a href="#">Stream.SetLength(Int64)</a> .)
	<a href="#">Source</a>	Gets the original stream that was used to create the duplicate.
	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the

		current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Write(Byte[], Int32, Int32)</a>	Writes to the source stream while maintaining a separate position and not impacting the source stream's position.  (Overrides <a href="#">Stream.Write(Byte[], Int32, Int32)</a> .)
	<a href="#">WriteByte(Byte)</a>	Writes a byte to the current position in the stream and advances the position within the stream by one byte.  (Inherited from <a href="#">Stream</a> .)
	<a href="#">WriteTimeout</a>	Gets or sets a value that determine how long the stream will attempt to write before timing out.  (Inherited from <a href="#">Stream</a> .)

## Remarks

WARNING: duplicate streams are not thread-safe with respect to each other or the original stream. If multiple threads use duplicate copies of the same stream, they must synchronize for any operations.

## Inheritance Hierarchy




---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# DuplicateStream Constructor (source)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► DuplicateStream ►

**DuplicateStream(Stream)**

C# ▾

Creates a new duplicate of a stream.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public DuplicateStream(  
    Stream source  
)
```

```
Public Sub New ( _  
    source As Stream _  
)
```

```
public:  
DuplicateStream(  
    Stream^ source  
)
```

## Parameters

**source (Stream)**

source of the duplicate

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CanRead Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► DuplicateStream ► CanRead

C#

Gets a value indicating whether the source stream supports reading.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public override bool CanRead { get; }
```

```
Public Overrides ReadOnly Property CanRead As Boolean
```

```
public:  
virtual property bool CanRead {  
    bool get () override;  
}
```

### Value

true if the stream supports reading; otherwise, false.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CanSeek Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► DuplicateStream ► CanSeek

C#

Gets a value indicating whether the source stream supports seeking.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public override bool CanSeek { get; }
```

```
Public Overrides ReadOnly Property CanSeek As Boolean
```

```
public:  
virtual property bool CanSeek {  
    bool get () override;  
}
```

### Value

true if the stream supports seeking; otherwise, false.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CanWrite Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► DuplicateStream ► CanWrite

C# ▾

Gets a value indicating whether the source stream supports writing.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override bool CanWrite { get; }
```

```
Public Overrides ReadOnly Property CanWrite As Boolean
```

```
public:  
virtual property bool CanWrite {  
    bool get () override;  
}
```

### Value

true if the stream supports writing; otherwise, false.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Close Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► DuplicateStream ► Close()

C# ▾

Closes the underlying stream, effectively closing ALL duplicates.

## - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override void Close()
```

```
Public Overrides Sub Close
```

```
public:  
virtual void Close() override
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Flush Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► DuplicateStream ► Flush()

C# ▾

Flushes the source stream.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public override void Flush()
```

```
Public Overrides Sub Flush
```

```
public:  
virtual void Flush() override
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools FoundationLength Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► DuplicateStream ► Length

C# ▾

Gets the length of the source stream.

## - Declaration Syntax

C#

Visual Basic

Visual C++

```
public override long Length { get; }
```

```
Public Overrides ReadOnly Property Length As Long
```

```
public:  
virtual property long long Length {  
    long long get () override;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **OriginalStream Method (stream)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► DuplicateStream ►

**OriginalStream(Stream)**

C# ▾

Retrieves the original stream from a possible duplicate stream.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static Stream OriginalStream(  
    Stream stream  
)
```

```
Public Shared Function OriginalStream ( _  
    stream As Stream _  
) As Stream
```

```
public:  
static Stream^ OriginalStream(  
    Stream^ stream  
)
```

### Parameters

#### **stream (Stream)**

Possible duplicate stream.

### Return Value

If the stream is a DuplicateStream, returns the duplicate's source; otherwise returns the same stream.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Position Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► DuplicateStream ► Position

C#

Gets or sets the position of the current stream, ignoring the position of the source stream.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public override long Position { get; set; }
```

```
Public Overrides Property Position As Long
```

```
public:  
virtual property long long Position {  
    long long get () override;  
    void set (long long value) override;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Read Method (buffer, offset, count)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► DuplicateStream ►

**Read(Byte[], Int32, Int32)**

C#

Reads from the source stream while maintaining a separate position and not impacting the source stream's position.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public override int Read(
    byte[] buffer,
    int offset,
    int count
)
```

```
Public Overrides Function Read ( _
    buffer As Byte(), _
    offset As Integer, _
    count As Integer _
) As Integer
```

```
public:
virtual int Read(
    array<unsigned char>^ buffer,
    int offset,
    int count
) override
```

## Parameters

### buffer ( Byte [] )

An array of bytes. When this method returns, the buffer contains the specified byte array with the values between offset and (offset + count - 1) replaced by the bytes read from the current source.

***offset* ([Int32](#))**

The zero-based byte offset in buffer at which to begin storing the data read from the current stream.

***count* ([Int32](#))**

The maximum number of bytes to be read from the current stream.

**[-] Return Value**

The total number of bytes read into the buffer. This can be less than the number of bytes requested if that many bytes are not currently available or zero (0) if the end of the stream has been reached.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Seek Method (offset, origin)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► DuplicateStream ► Seek(Int64, SeekOrigin)

C# ▾

Changes the position of this stream without impacting the source stream's position.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override long Seek(  
    long offset,  
    SeekOrigin origin  
)
```

```
Public Overrides Function Seek ( _  
    offset As Long, _  
    origin As SeekOrigin _  
) As Long
```

```
public:  
virtual long long Seek(  
    long long offset,  
    SeekOrigin origin  
) override
```

### Parameters

#### *offset* (Int64)

A byte offset relative to the *origin* parameter.

#### *origin* (SeekOrigin)

A value of type SeekOrigin indicating the reference point used to obtain the new position.

### Return Value

The new position within the current stream.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## SetLength Method (value)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► DuplicateStream ►

### SetLength(Int64)

C# ▾

Sets the length of the source stream.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override void SetLength(  
    long value  
)
```

```
Public Overrides Sub SetLength ( _  
    value As Long _  
)
```

```
public:  
virtual void SetLength(  
    long long value  
) override
```

#### Parameters

##### **value (Int64)**

The desired length of the stream in bytes.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Source Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► DuplicateStream ► Source

C# ▾

Gets the original stream that was used to create the duplicate.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public Stream Source { get; }
```

```
Public ReadOnly Property Source As Stream
```

```
public:  
property Stream^ Source {  
    Stream^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Write Method (buffer, offset, count)

[Deployment Tools Foundation Namespaces](#) ►

[Microsoft.Deployment.Compression](#) ► [DuplicateStream](#) ►

[Write\(Byte\[\], Int32, Int32\)](#)

C#

Writes to the source stream while maintaining a separate position and not impacting the source stream's position.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public override void Write(
    byte[] buffer,
    int offset,
    int count
)
```

```
Public Overrides Sub Write ( _
    buffer As Byte(), _
    offset As Integer, _
    count As Integer _
)
```

```
public:
virtual void Write(
    array<unsigned char>^ buffer,
    int offset,
    int count
) override
```

### Parameters

#### **buffer ( Byte [] )**

An array of bytes. This method copies count bytes from buffer to the current stream.

***offset* ([Int32](#))**

The zero-based byte offset in buffer at which to begin copying bytes to the current stream.

***count* ([Int32](#))**

The number of bytes to be written to the current stream.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation IPackStreamContext Interface

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► IPackStreamContext

C# ▾

This interface provides the methods necessary for the [CompressionEngine](#) to open and close streams for archives and files. The implementor of this interface can use any kind of logic to determine what kind of streams to open and where.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public interface IPackStreamContext
```

```
Public Interface IPackStreamContext
```

```
public interface class IPackStreamContext
```

## Members

All Members

Methods

Public

Instance

Declared

Protected

Static

Inherited

Icon	Member	Description
≡	<a href="#">CloseArchiveWriteStream(Int32, String, Stream)</a>	Closes a stream where an archive package was written.
≡	<a href="#">CloseFileReadStream(String, Stream)</a>	Closes a stream that has been used to read a file.
≡	<a href="#">GetArchiveName(Int32)</a>	Gets the name of the archive with a specified number.

	<code>getOption(String, Object[])</code>	Gets extended parameter information specific to the compression format being used.
	<code>openArchiveWriteStream(Int32, String, Boolean, CompressionEngine)</code>	Opens a stream for writing an archive package.
	<code>openFileReadStream(String, FileAttributes, DateTime)</code>	Opens a stream to read a file that is to be included in an archive.

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CloseArchiveWriteStream Method (archiveNumber, archiveName, stream)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► IPackStreamContext ►  
**CloseArchiveWriteStream(Int32, String, Stream)**

C# ▾

Closes a stream where an archive package was written.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
void CloseArchiveWriteStream(  
    int archiveNumber,  
    string archiveName,  
    Stream stream  
)
```

```
Sub CloseArchiveWriteStream ( _  
    archiveNumber As Integer, _  
    archiveName As String, _  
    stream As Stream _  
)
```

```
void CloseArchiveWriteStream(  
    int archiveNumber,  
    String^ archiveName,  
    Stream^ stream  
)
```

### Parameters

#### archiveNumber (Int32)

The 0-based index of the archive within the chain.

#### archiveName (String)

The name of the archive that was previously returned by [GetArchiveName\(Int32\)](#).

### ***stream* ([Stream](#))**

A stream that was previously returned by [OpenArchiveWriteStream\(Int32, String, Boolean, CompressionEngine\)](#) and is now ready to be closed.

### **Remarks**

If there is another archive package in the chain, then after this stream is closed a new stream will be opened.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CloseFileStream Method (path, stream)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► IPackStreamContext ►  
**CloseFileStream(String, Stream)**

C# ▾

Closes a stream that has been used to read a file.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
void CloseFileStream(  
    string path,  
    Stream stream  
)
```

```
Sub CloseFileStream ( _  
    path As String, _  
    stream As Stream _  
)
```

```
void CloseFileStream(  
    String^ path,  
    Stream^ stream  
)
```

### Parameters

#### *path* (String)

The path of the file within the archive; the same as the path provided when the stream was opened.

#### *stream* (Stream)

A stream that was previously returned by [OpenFileReadStream\(String, FileAttributes, DateTime\)](#) and is now ready to be closed.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:  
Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetArchiveName Method  
(archiveNumber)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► IPackStreamContext ►  
**GetArchiveName(Int32)**

C# ▾

Gets the name of the archive with a specified number.

▀ Declaration Syntax

C#

Visual Basic

Visual C++

```
string GetArchiveName(  
    int archiveNumber  
)
```

```
Function GetArchiveName ( _  
    archiveNumber As Integer _  
) As String
```

```
String^ GetArchiveName(  
    int archiveNumber  
)
```

▀ Parameters

**archiveNumber (Int32)**

The 0-based index of the archive within the chain.

▀ Return Value

The name of the requested archive. May be an empty string for non-chained archives, but may never be null.

▀ Remarks

The archive name is the name stored within the archive, used for identification of the archive especially among archive chains. That name is often, but not necessarily the same as the filename of the archive

package.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## GetOption Method (optionName, parameters)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► IPackStreamContext ►

**GetOption(String, Object[])**

C# ▾

Gets extended parameter information specific to the compression format being used.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
Object GetOption(  
    string optionName,  
    Object[] parameters  
)
```

```
Function GetOption ( _  
    optionName As String, _  
    parameters As Object() _  
) As Object
```

```
Object^ GetOption(  
    String^ optionName,  
    array<Object^>^ parameters  
)
```

### - Parameters

#### ***optionName* (String)**

Name of the option being requested.

#### ***parameters* ( Object [] )**

Parameters for the option; for per-file options, the first parameter is typically the internal file path.

### - Return Value

Option value, or null to use the default behavior.

## **Remarks**

This method provides a way to set uncommon options during packaging or a way to handle aspects of compression formats not supported by the base library.

For example, this may be used by the zip compression library to specify different compression methods/levels on a per-file basis.

The available option names, parameters, and expected return values should be documented by each compression library.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## OpenArchiveWriteStream Method (archiveNumber, archiveName, truncate, compressionEngine)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► IPackStreamContext ►  
**OpenArchiveWriteStream(Int32, String, Boolean,  
CompressionEngine)**

Opens a stream for writing an archive package.

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
Stream OpenArchiveWriteStream(  
    int archiveNumber,  
    string archiveName,  
    bool truncate,  
    CompressionEngine compressionEngine  
)
```

```
Function OpenArchiveWriteStream ( _  
    archiveNumber As Integer, _  
    archiveName As String, _  
    truncate As Boolean, _  
    compressionEngine As CompressionEngine _  
) As Stream
```

```
Stream^ OpenArchiveWriteStream(  
    int archiveNumber,  
    String^ archiveName,  
    bool truncate,  
    CompressionEngine^ compressionEngine  
)
```

### Parameters

***archiveNumber*** ([Int32](#))

The 0-based index of the archive within the chain.

***archiveName*** ([String](#))

The name of the archive that was returned by [GetArchiveName\(Int32\)](#).

***truncate*** ([Boolean](#))

True if the stream should be truncated when opened (if it already exists); false if an existing stream is being re-opened for writing additional data.

***compressionEngine*** ([CompressionEngine](#))

Instance of the compression engine doing the operations.

**▀ Return Value**

A writable Stream where the compressed archive bytes will be written, or null to cancel the archive creation.

**▀ Remarks**

If this method returns null, the archive engine will throw a [FileNotFoundException](#).

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## OpenFileReadStream Method (path, attributes, lastWriteTime)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► IPackStreamContext ►  
**OpenFileReadStream(String, FileAttributes, DateTime)**

C# ▾

Opens a stream to read a file that is to be included in an archive.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
Stream OpenFileReadStream(  
    string path,  
    out FileAttributes attributes,  
    out DateTime lastWriteTime  
)
```

```
Function OpenFileReadStream ( _  
    path As String, _  
    <OutAttribute> ByRef attributes As FileAttri  
    <OutAttribute> ByRef lastWriteTime As DateTi  
) As Stream
```

```
Stream^ OpenFileReadStream(  
    String^ path,  
    [OutAttribute] FileAttributes% attributes,  
    [OutAttribute] DateTime% lastWriteTime  
)
```

### Parameters

#### *path* (String)

The path of the file within the archive. This is often, but not necessarily, the same as the relative path of the file outside the archive.

#### *attributes* ( FileAttributes )

Returned attributes of the opened file, to be stored in the archive.

#### ***lastWriteTime* ( [DateTime](#) )**

Returned last-modified time of the opened file, to be stored in the archive.

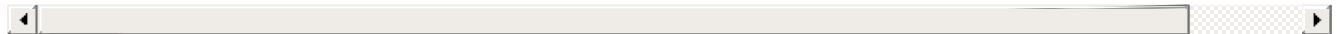
### **Return Value**

A readable Stream where the file bytes will be read from before they are compressed, or null to skip inclusion of the file and continue to the next file.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation **IUnpackStreamContext** Interface

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► IUnpackStreamContext

C# ▾

This interface provides the methods necessary for the [CompressionEngine](#) to open and close streams for archives and files. The implementor of this interface can use any kind of logic to determine what kind of streams to open and where

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public interface IUnpackStreamContext
```

```
Public Interface IUnpackStreamContext
```

```
public interface class IUnpackStreamContext
```

## Members

All Members

Methods

Public

Instance

Declared

Protected

Static

Inherited

Icon	Member	Description
≡	<a href="#">CloseArchiveReadStream(Int32, String, Stream)</a>	Closes a stream where an archive package was read.
≡	<a href="#">CloseFileWriteStream(String, Stream, FileAttributes, DateTime)</a>	Closes a stream where an extracted file was written.
≡	<a href="#">OpenArchiveReadStream(Int32, String, CompressionEngine)</a>	Opens the archive stream for reading.

	<code>OpenFileWriteStream(String, Int64, DateTime)</code>	Opens a stream for writing extracted file bytes.
---	---	--

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CloseArchiveReadStream Method (archiveNumber, archiveName, stream)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► IUnpackStreamContext ►

**CloseArchiveReadStream(Int32, String, Stream)**

C# ▾

Closes a stream where an archive package was read.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
void CloseArchiveReadStream(  
    int archiveNumber,  
    string archiveName,  
    Stream stream  
)
```

```
Sub CloseArchiveReadStream ( _  
    archiveNumber As Integer, _  
    archiveName As String, _  
    stream As Stream _  
)
```

```
void CloseArchiveReadStream(  
    int archiveNumber,  
    String^ archiveName,  
    Stream^ stream  
)
```

### Parameters

#### **archiveNumber (Int32)**

The archive number of the stream to close.

#### **archiveName (String)**

The name of the archive being closed.

#### **stream (Stream)**

The stream that was previously returned by  
[OpenArchiveReadStream\(Int32, String, CompressionEngine\)](#) and is now ready to be closed.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CloseFileStream Method (path, stream, attributes, lastWriteTime)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► IUnpackStreamContext ►

**CloseFileStream(String, Stream, FileAttributes, DateTime)**

C#

Closes a stream where an extracted file was written.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
void CloseFileStream(  
    string path,  
    Stream stream,  
    FileAttributes attributes,  
    DateTime lastWriteTime  
)
```

```
Sub CloseFileStream ( _  
    path As String, _  
    stream As Stream, _  
    attributes As FileAttributes, _  
    lastWriteTime As DateTime _  
)
```

```
void CloseFileStream(  
    String^ path,  
    Stream^ stream,  
    FileAttributes attributes,  
    DateTime lastWriteTime  
)
```

### Parameters

#### ***path* (String)**

The path of the file within the archive.

### ***stream* ([Stream](#))**

The stream that was previously returned by [OpenFileWriteStream\(String, Int64, DateTime\)](#) and is now ready to be closed.

### ***attributes* ([FileAttributes](#))**

The attributes of the extracted file.

### ***lastWriteTime* ([DateTime](#))**

The last write time of the file.

## **[-] Remarks**

The implementor may wish to apply the attributes and date to the newly extracted file.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## OpenArchiveReadStream Method (archiveNumber, archiveName, compressionEngine)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► IUnpackStreamContext ►

**OpenArchiveReadStream(Int32, String, CompressionEngine)**

C# ▾

Opens the archive stream for reading.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
Stream OpenArchiveReadStream(  
    int archiveNumber,  
    string archiveName,  
    CompressionEngine compressionEngine  
)
```

```
Function OpenArchiveReadStream ( _  
    archiveNumber As Integer, _  
    archiveName As String, _  
    compressionEngine As CompressionEngine _  
) As Stream
```

```
Stream^ OpenArchiveReadStream(  
    int archiveNumber,  
    String^ archiveName,  
    CompressionEngine^ compressionEngine  
)
```

### Parameters

#### archiveNumber (Int32)

The zero-based index of the archive to open.

#### archiveName (String)

The name of the archive being opened.

## ***compressionEngine* ([CompressionEngine](#))**

Instance of the compression engine doing the operations.

### **▀ Return Value**

A stream from which archive bytes are read, or null to cancel extraction the archive.

### **▀ Remarks**

When the first archive in a chain is opened, the name is not yet known, so the provided value will be an empty string. When opening further archives, the provided value is the next-archive name stored in the previous archive. This name is often, but not necessarily, the same as the filename of the archive package to be opened.

If this method returns null, the archive engine will throw a `FileNotFoundException`.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **OpenFileWriteStream Method  
(path, fileSize, lastWriteTime)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► IUnpackStreamContext ►  
**OpenFileWriteStream(String, Int64, DateTime)**

C# ▾

Opens a stream for writing extracted file bytes.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
Stream OpenFileWriteStream(  
    string path,  
    long fileSize,  
    DateTime lastWriteTime  
)
```

```
Function OpenFileWriteStream ( _  
    path As String, _  
    fileSize As Long, _  
    lastWriteTime As DateTime _  
) As Stream
```

```
Stream^ OpenFileWriteStream(  
    String^ path,  
    long long fileSize,  
    DateTime lastWriteTime  
)
```

[-] **Parameters**

***path* (String)**

The path of the file within the archive. This is often, but not necessarily, the same as the relative path of the file outside the archive.

***fileSize* (Int64)**

The uncompressed size of the file to be extracted.

#### ***lastWriteTime* ([DateTime](#))**

The last write time of the file to be extracted.

#### **□ Return Value**

A stream where extracted file bytes are to be written, or null to skip extraction of the file and continue to the next file.

#### **□ Remarks**

The implementor may use the path, size and date information to dynamically decide whether or not the file should be extracted.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **OffsetStream Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► **OffsetStream**

C#

Wraps a source stream and offsets all read/write/seek calls by a given value.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public class OffsetStream : Stream
```

```
Public Class OffsetStream _  
    Inherits Stream
```

```
public ref class OffsetStream : public Stream
```

## Members

All Members	Constructors	Methods	Properties	
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static		<input checked="" type="checkbox"/> Declare <input checked="" type="checkbox"/> Inherit

Icon	Member	Description
≡	<a href="#">OffsetStream(Stream, Int64)</a>	Creates a new <b>OffsetStream</b> instance from a source stream and using a specified offset.
≡	<a href="#">BeginRead(Byte[], Int32, Int32, AsyncCallback, Object)</a>	HostProtectionAttribute. Begins an asynchronous read operation.  (Inherited from <a href="#">Stream</a> .)
≡	<a href="#">BeginWrite(Byte[], Int32,</a>	HostProtectionAttribute.

	<code>Int32, AsyncCallback, Object)</code>	Begins an asynchronous write operation.  (Inherited from <a href="#">Stream</a> .)
	<a href="#">CanRead</a>	Gets a value indicating whether the source stream supports reading.  (Overrides <a href="#">Stream.CanRead</a> .)
	<a href="#">CanSeek</a>	Gets a value indicating whether the source stream supports seeking.  (Overrides <a href="#">Stream.CanSeek</a> .)
	<a href="#">CanTimeout</a>	Gets a value that determines whether the current stream can time out.  (Inherited from <a href="#">Stream</a> .)
	<a href="#">CanWrite</a>	Gets a value indicating whether the source stream supports writing.  (Overrides <a href="#">Stream.CanWrite</a> .)
	<a href="#">Close()</a>	Closes the underlying stream.  (Overrides <a href="#">Stream.Close</a> .)
	<a href="#">CreateObjRef(Type)</a>	Creates an object that contains all the relevant information required to generate a proxy used to communicate with a remote object.  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">CreateWaitHandle()</a>	<b>Obsolete.</b> Allocates a <a href="#">WaitHandle</a> object.  (Inherited from <a href="#">Stream</a> .)
	<a href="#">Dispose()</a>	Releases all resources used by the

		<a href="#">Stream</a> .
		(Inherited from <a href="#">Stream</a> .)
💡	<a href="#">Dispose(Boolean)</a>	Releases the unmanaged resource used by the <a href="#">Stream</a> and optionally releases the managed resources.  (Inherited from <a href="#">Stream</a> .)
💡	<a href="#">EndRead(IAsyncResult)</a>	Waits for the pending asynchronous read to complete.  (Inherited from <a href="#">Stream</a> .)
💡	<a href="#">EndWrite(IAsyncResult)</a>	Ends an asynchronous write operation.  (Inherited from <a href="#">Stream</a> .)
💡	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">Flush()</a>	Flushes the source stream.  (Overrides <a href="#">Stream.Flush()</a> .)
💡	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like hash table.

		(Inherited from <a href="#">Object</a> .)
	<a href="#">GetLifetimeService()</a>	Retrieves the current lifetime service object that controls the lifetime policy for this instance.
		(Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.
		(Inherited from <a href="#">Object</a> .)
	<a href="#">InitializeLifetimeService()</a>	Obtains a lifetime service object to control the lifetime policy for this instance.
		(Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">Length</a>	Gets the effective length of the stream, which is equal to the length of the source stream minus the offset.
		(Overrides <a href="#">Stream.Length</a> .)
	<a href="#">MemberwiseClone(Boolean)</a>	Creates a shallow copy of the current <a href="#">MarshalByRefObject</a> object
		(Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .
		(Inherited from <a href="#">Object</a> .)
	<a href="#">Offset</a>	Gets the number of bytes to offset all calls before redirecting to the underlying stream.

	<a href="#">Position</a>	Gets or sets the effective position of the stream, which is equal to the position of the source stream minus the offset.  (Overrides <a href="#">Stream.Position</a> .)
	<a href="#">Read(Byte[], Int32, Int32)</a>	Reads a sequence of bytes from the source stream and advances the position within the stream by the number of bytes read.  (Overrides <a href="#">Stream.Read(Byte[], Int32, Int32)</a> .)
	<a href="#">ReadByte()</a>	Reads a byte from the stream and advances the position within the source stream by one byte, or returns -1 if at the end of the stream.  (Overrides <a href="#">Stream.ReadByte</a> .)
	<a href="#">ReadTimeout</a>	Gets or sets a value that determines how long the stream will attempt to read before timing out.  (Inherited from <a href="#">Stream</a> .)
	<a href="#">Seek(Int64, SeekOrigin)</a>	Sets the position within the current stream, which is equal to the position within the source stream minus the offset.  (Overrides <a href="#">Stream.Seek(Int64, SeekOrigin)</a> .)
	<a href="#">SetLength(Int64)</a>	Sets the effective length of the stream, which is equal to the length of the source stream minus the

		offset.  (Overrides <a href="#">Stream.SetLength(Int64)</a> .)
	<a href="#">Source</a>	Gets the underlying stream that this <a href="#">OffsetStream</a> calls into.
	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Write(Byte[], Int32, Int32)</a>	Writes a sequence of bytes to the source stream and advances the current position within this stream by the number of bytes written.  (Overrides <a href="#">Stream.Write(Byte[], Int32, Int32)</a> .)
	<a href="#">WriteByte(Byte)</a>	Writes a byte to the current position in the source stream and advances the position within the stream by one byte.  (Overrides <a href="#">Stream.WriteByte(Byte)</a> )
	<a href="#">WriteTimeout</a>	Gets or sets a value that determine how long the stream will attempt to write before timing out.  (Inherited from <a href="#">Stream</a> .)

## Remarks

This class is used to trick archive and packing or unpacking process into reading or writing at an offset into a file, primarily for self-extracting packages.

## Inheritance Hierarchy

Object

└ MarshalByRefObject

  └ Stream

    └ **OffsetStream**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## OffsetStream Constructor (source, offset)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► OffsetStream ►  
OffsetStream(Stream, Int64)

C# ▾

Creates a new OffsetStream instance from a source stream and using a specified offset.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public OffsetStream(  
    Stream source,  
    long offset  
)
```

```
Public Sub New ( _  
    source As Stream, _  
    offset As Long _  
)
```

```
public:  
OffsetStream(  
    Stream^ source,  
    long long offset  
)
```

### Parameters

#### source (Stream)

Underlying stream for which all calls will be offset.

#### offset (Int64)

Positive or negative number of bytes to offset.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CanRead Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► OffsetStream ► CanRead

C#

Gets a value indicating whether the source stream supports reading.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public override bool CanRead { get; }
```

```
Public Overrides ReadOnly Property CanRead As Boolean
```

```
public:  
virtual property bool CanRead {  
    bool get () override;  
}
```

### Value

true if the stream supports reading; otherwise, false.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CanSeek Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► OffsetStream ► CanSeek

C#

Gets a value indicating whether the source stream supports seeking.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public override bool CanSeek { get; }
```

```
Public Overrides ReadOnly Property CanSeek As Boolean
```

```
public:  
virtual property bool CanSeek {  
    bool get () override;  
}
```

### Value

true if the stream supports seeking; otherwise, false.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CanWrite Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► OffsetStream ► CanWrite

C# ▾

Gets a value indicating whether the source stream supports writing.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override bool CanWrite { get; }
```

```
Public Overrides ReadOnly Property CanWrite As Boolean
```

```
public:  
virtual property bool Canwrite {  
    bool get () override;  
}
```

### Value

true if the stream supports writing; otherwise, false.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Close Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► OffsetStream ► Close()

C# ▾

Closes the underlying stream.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public override void Close()
```

```
Public Overrides Sub Close
```

```
public:  
virtual void Close() override
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Flush Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► OffsetStream ► **Flush()**

C# ▾

Flushes the source stream.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public override void Flush()
```

```
Public Overrides Sub Flush
```

```
public:  
virtual void Flush() override
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools FoundationLength Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► OffsetStream ► Length

C#

Gets the effective length of the stream, which is equal to the length of the source stream minus the offset.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public override long Length { get; }
```

```
Public Overrides ReadOnly Property Length As Long
```

```
public:  
virtual property long long Length {  
    long long get () override;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Offset Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► OffsetStream ► Offset

C# ▾

Gets the number of bytes to offset all calls before redirecting to the underlying stream.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public long Offset { get; }
```

```
Public Readonly Property Offset As Long
```

```
public:  
property long long Offset {  
    long long get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Position** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► OffsetStream ► Position

C# ▾

Gets or sets the effective position of the stream, which is equal to the position of the source stream minus the offset.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override long Position { get; set; }
```

```
Public Overrides Property Position As Long
```

```
public:  
virtual property long long Position {  
    long long get () override;  
    void set (long long value) override;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Read Method (buffer, offset, count)

Deployment Tools Foundation Namespaces ► Microsoft.Deployment.Compression ► OffsetStream ► Read(Byte[], Int32, Int32)

C# ▾

Reads a sequence of bytes from the source stream and advances the position within the stream by the number of bytes read.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override int Read(
    byte[] buffer,
    int offset,
    int count
)
```

```
Public Overrides Function Read ( _
    buffer As Byte(), _
    offset As Integer, _
    count As Integer _
) As Integer
```

```
public:
virtual int Read(
    array<unsigned char>^ buffer,
    int offset,
    int count
) override
```

### Parameters

#### **buffer ( Byte [] )**

An array of bytes. When this method returns, the buffer contains the specified byte array with the values between offset and (offset + count - 1) replaced by the bytes read from the current source.

***offset* ([Int32](#))**

The zero-based byte offset in buffer at which to begin storing the data read from the current stream.

***count* ([Int32](#))**

The maximum number of bytes to be read from the current stream.

**[-] Return Value**

The total number of bytes read into the buffer. This can be less than the number of bytes requested if that many bytes are not currently available or zero (0) if the end of the stream has been reached.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **ReadByte Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► OffsetStream ► ReadByte()

C# ▾

Reads a byte from the stream and advances the position within the source stream by one byte, or returns -1 if at the end of the stream.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public override int ReadByte()
```

```
Public Overrides Function ReadByte As Integer
```

```
public:  
virtual int ReadByte() override
```

[-] **Return Value**

The unsigned byte cast to an Int32, or -1 if at the end of the stream.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Seek Method (offset, origin)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► OffsetStream ► Seek(Int64, SeekOrigin)

C# ▾

Sets the position within the current stream, which is equal to the position within the source stream minus the offset.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override long Seek(  
    long offset,  
    SeekOrigin origin  
)
```

```
Public Overrides Function Seek ( _  
    offset As Long,  
    origin As SeekOrigin _  
) As Long
```

```
public:  
virtual long long Seek(  
    long long offset,  
    SeekOrigin origin  
) override
```

### Parameters

#### offset (Int64)

A byte offset relative to the origin parameter.

#### origin (SeekOrigin)

A value of type SeekOrigin indicating the reference point used to obtain the new position.

### Return Value

The new position within the current stream.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **SetLength Method (value)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► OffsetStream ►

### **SetLength(Int64)**

C# ▾

Sets the effective length of the stream, which is equal to the length of the source stream minus the offset.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override void SetLength(  
    long value  
)
```

```
Public Overrides Sub SetLength ( _  
    value As Long _  
)
```

```
public:  
virtual void SetLength(  
    long long value  
) override
```

#### Parameters

##### **value (Int64)**

The desired length of the current stream in bytes.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Source Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► OffsetStream ► Source

C# ▾

Gets the underlying stream that this OffsetStream calls into.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public Stream Source { get; }
```

```
Public ReadOnly Property Source As Stream
```

```
public:  
property Stream^ Source {  
    Stream^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Write Method (buffer, offset, count)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression ► OffsetStream ► Write(Byte[], Int32, Int32)

C#

Writes a sequence of bytes to the source stream and advances the current position within this stream by the number of bytes written.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public override void Write(  
    byte[] buffer,  
    int offset,  
    int count  
)
```

```
Public Overrides Sub Write ( _  
    buffer As Byte(), _  
    offset As Integer, _  
    count As Integer _  
)
```

```
public:  
virtual void Write(  
    array<unsigned char>^ buffer,  
    int offset,  
    int count  
) override
```

### Parameters

#### buffer ( Byte [] )

An array of bytes. This method copies count bytes from buffer to the current stream.

***offset* ([Int32](#))**

The zero-based byte offset in buffer at which to begin copying bytes to the current stream.

***count* ([Int32](#))**

The number of bytes to be written to the current stream.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module: Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation WriteByte Method (value)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression ► OffsetStream ►

### WriteByte(Byte)

C#

Writes a byte to the current position in the source stream and advances the position within the stream by one byte.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public override void WriteByte(  
    byte value  
)
```

```
Public Overrides Sub WriteByte ( _  
    value As Byte _  
)
```

```
public:  
virtual void WriteByte(  
    unsigned char value  
) override
```

#### Parameters

##### value (Byte)

The byte to write to the stream.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression (Module:

Microsoft.Deployment.Compression) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools

## Foundation **Microsoft.Deployment.Compression.Cab** Namespace

Deployment Tools Foundation Namespaces ►

**Microsoft.Deployment.Compression.Cab**

C# ▾

Implements cabinet archive packing and unpacking.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
namespace Microsoft.Deployment.Compression.Cab
```

```
Namespace Microsoft.Deployment.Compression.Cab
```

```
namespace Microsoft.Deployment.Compression.Cab
```

### [-] Types

All Types

Classes

Icon	Type	Description
	<a href="#">CabEngine</a>	Engine capable of packing and unpacking archives in the cabinet format.
	<a href="#">CabException</a>	Exception class for cabinet operations.
	<a href="#">CabFileInfo</a>	Object representing a compressed file within a cabinet package; provides operations for getting the file properties and extracting the file.



## CabInfo

Object representing a cabinet file on disk; provides access to file-based operations on the cabinet file.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

# Deployment Tools Foundation **CabEngine Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabEngine

C#

Engine capable of packing and unpacking archives in the cabinet forma

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public class CabEngine : CompressionEngine
```

```
Public Class CabEngine _  
    Inherits CompressionEngine
```

```
public ref class CabEngine : public CompressionEngin
```

## Members

All Members

Constructors

Methods

Properties

Events

Public

Instance

Declared

Protected

Static

Inherited

Icon	Member	Description
	<a href="#">CabEngine()</a>	Creates a new instance
	<a href="#">CompressionLevel</a>	Compression level to u (Inherited from <a href="#">Compre</a> )
	<a href="#">Dispose(Boolean)</a>	Disposes of resources (Overrides <a href="#">Compressio</a> )
	<a href="#">Dispose()</a>	Disposes of resources engine. (Inherited from <a href="#">Compre</a> )

≡	<a href="#">Equals(Object)</a>	Determines whether the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">Finalize()</a>	Disposes the compressed stream.  (Inherited from <a href="#">CompressedStreamBase</a> .)
≡	<a href="#">FindArchiveOffset(Stream)</a>	Gets the offset of an archive file's bytes from the start of the stream.  (Inherited from <a href="#">CompressedStreamBase</a> .)
≡	<a href="#">GetFileInfo(IUnpackStreamContext, Predicate&lt;String&gt;)</a>	Gets information about files in the archive chain.  (Overrides <a href="#">CompressionEngine.GetFileInfo(IUnpackStreamContext, Predicate&lt;String&gt;)</a> .)
≡	<a href="#">GetFileInfo(Stream)</a>	Gets information about files in the archive.  (Inherited from <a href="#">CompressedStreamBase</a> .)
≡	<a href="#">GetFiles(Stream)</a>	Gets the list of files in a stream.  (Inherited from <a href="#">CompressedStreamBase</a> .)
≡	<a href="#">GetFiles(IUnpackStreamContext, Predicate&lt;String&gt;)</a>	Gets the list of files in a stream.  (Inherited from <a href="#">CompressedStreamBase</a> .)
≡	<a href="#">GetHashCode()</a>	Serves as a hash function. <a href="#">GetHashCode()</a> is suitable for use in hash tables and data structures like <a href="#">Dictionary&lt;TKey, TValue&gt;</a> .  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">IsArchive(Stream)</a>	Checks whether a Stream indicates it is a valid archive.

		(Overrides <a href="#">CompressionEngine</a> )
💡	<a href="#">MemberwiseClone()</a>	Creates a shallow copy.  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">OnProgress(ArchiveProgressEventArgs)</a>	Called by subclasses to report unpacking progress events.  (Inherited from <a href="#">CompressionEngine</a> )
💡	<a href="#">Pack(IPackStreamContext, IEnumerable&lt;String&gt;, Int64)</a>	Creates a cabinet or chm file.  (Overrides <a href="#">CompressionEngine.Pack(IEnumerable&lt;String&gt;, Int64)</a> )
💡	<a href="#">Pack(IPackStreamContext, IEnumerable&lt;String&gt;)</a>	Creates an archive.  (Inherited from <a href="#">CompressionEngine</a> )
⚡	<a href="#">Progress</a>	Occurs when the compression engine begins packing or unpacking files.  (Inherited from <a href="#">CompressionEngine</a> )
💡	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents this instance.  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">Unpack(IUnpackStreamContext, Predicate&lt;String&gt;)</a>	Extracts files from a cabinet or chm file.  (Overrides <a href="#">CompressionEngine.Unpack(IUnpackStreamContext, Predicate&lt;String&gt;)</a> )
💡	<a href="#">Unpack(Stream, String)</a>	Reads a single file from a stream.  (Inherited from <a href="#">CompressionEngine</a> )
📅	<a href="#">UseTempFiles</a>	Gets or sets a flag indicating whether temporary files are created and used during compression.  (Inherited from <a href="#">CompressionEngine</a> )

---

## **[-] Inheritance Hierarchy**

Object

└ CompressionEngine

  └ CabEngine

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## CabEngine Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabEngine ►

### CabEngine()

C# ▾

Creates a new instance of the cabinet engine.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public CabEngine()
```

```
Public Sub New
```

```
public:  
CabEngine()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Dispose Method (disposing)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabEngine ►

## Dispose(Boolean)

C# ▾

Disposes of resources allocated by the cabinet engine.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected override void Dispose(  
    bool disposing  
)
```

```
Protected Overrides Sub Dispose ( _  
    disposing As Boolean _  
)
```

```
protected:  
virtual void Dispose(  
    bool disposing  
) override
```

### Parameters

#### disposing (Boolean)

If true, the method has been called directly or indirectly by a user's code, so managed and unmanaged resources will be disposed. If false, the method has been called by the runtime from inside the finalizer, and only unmanaged resources will be disposed.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetFileInfo Method**  
**(streamContext, fileFilter)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression.Cab ► CabEngine ►  
GetFileInfo(IUnpackStreamContext, Predicate<String>)

C# ▾

Gets information about files in a cabinet or cabinet chain.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public override IList<ArchiveFileInfo> GetFileInfo(  
    IUnpackStreamContext streamContext,  
    Predicate<string> fileFilter  
)
```

```
Public Overrides Function GetFileInfo ( _  
    streamContext As IUnpackStreamContext, _  
    fileFilter As Predicate(Of String) _  
) As IList(Of ArchiveFileInfo)
```

```
public:  
virtual IList<ArchiveFileInfo^>^ GetFileInfo(  
    IUnpackStreamContext^ streamContext,  
    Predicate<String^>^ fileFilter  
) override
```

[-] **Parameters**

**streamContext (IUnpackStreamContext)**

A context interface to handle opening and closing of cabinet and file streams.

**fileFilter (Predicate<String>)**

A predicate that can determine which files to process, optional.

[-] **Return Value**

Information about files in the cabinet stream.

## **Remarks**

The *fileFilter* predicate takes an internal file path and returns true to include the file or false to exclude it.

## **Exceptions**

Exception	Condition
<a href="#">ArchiveException</a>	The cabinet provided by the stream context is not valid.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation IsArchive Method (stream)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabEngine ►

### IsArchive(Stream)

C# ▾

Checks whether a Stream begins with a header that indicates it is a valid cabinet file.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override bool IsArchive(  
    Stream stream  
)
```

```
Public Overrides Function IsArchive ( _  
    stream As Stream _  
) As Boolean
```

```
public:  
virtual bool IsArchive(  
    Stream^ stream  
) override
```

#### Parameters

##### stream (Stream)

Stream for reading the cabinet file.

#### Return Value

True if the stream is a valid cabinet file (with no offset); false otherwise.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Pack Method (streamContext, files, maxArchiveSize)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabEngine ►

**Pack(IPackStreamContext, IEnumerable<String>, Int64)**

C# ▾

Creates a cabinet or chain of cabinets.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public override void Pack(  
    IPackStreamContext streamContext,  
    IEnumerable<string> files,  
    long maxArchiveSize  
)
```

```
Public Overrides Sub Pack ( _  
    streamContext As IPackStreamContext, _  
    files As IEnumerable(Of String), _  
    maxArchiveSize As Long _  
)
```

```
public:  
virtual void Pack(  
    IPackStreamContext^ streamContext,  
    IEnumerable<String^>^ files,  
    long long maxArchiveSize  
) override
```

### [-] Parameters

#### **streamContext (IPackStreamContext)**

A context interface to handle opening and closing of cabinet and file streams.

#### **files (IEnumerable<String>)**

The paths of the files in the archive (not external file paths).

### **maxArchiveSize (Int64)**

The maximum number of bytes for one cabinet before the contents are chained to the next cabinet, or zero for unlimited cabinet size.

## **Remarks**

The stream context implementation may provide a mapping from the file paths within the cabinet to the external file paths.

Smaller folder sizes can make it more efficient to extract individual files out of large cabinet packages.

## **Exceptions**

Exception	Condition
<a href="#">ArchiveException</a>	The cabinet could not be created.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

# Unpack Method (streamContext, fileFilter)

Deployment Tools Foundation Namespaces ►  
 Microsoft.Deployment.Compression.Cab ► CabEngine ►  
**Unpack(IUnpackStreamContext, Predicate<String>)**

C# ▾

Extracts files from a cabinet or cabinet chain.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public override void Unpack(
    IUnpackStreamContext streamContext,
    Predicate<string> fileFilter
)
```

```
Public Overrides Sub Unpack ( _
    streamContext As IUnpackStreamContext, _
    fileFilter As Predicate(Of String) _
)
```

```
public:
virtual void Unpack(
    IUnpackStreamContext^ streamContext,
    Predicate<String^>^ fileFilter
) override
```

## Parameters

### **streamContext (IUnpackStreamContext)**

A context interface to handle opening and closing of cabinet and file streams.

### **fileFilter (Predicate<String>)**

An optional predicate that can determine which files to process.

## Remarks

The *fileFilter* predicate takes an internal file path and returns true to include the file or false to exclude it.

## **Exceptions**

Exception	Condition
ArchiveException	The cabinet provided by the stream context is not valid.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **CabException Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabException

C# ▾

Exception class for cabinet operations.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
[SerializableAttribute]
public class CabException : ArchiveException
```

```
<SerializableAttribute> _
Public Class CabException _
    Inherits ArchiveException
```

```
[SerializableAttribute]
public ref class CabException : public ArchiveExcept
```

## Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

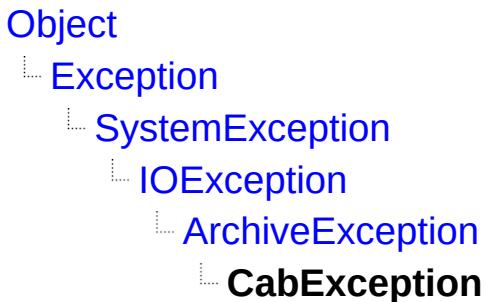
Icon	Member	Description
≡	<code>CabException(String, Exception)</code>	Creates a new CabException with the specified error message and a reference to the inner exception that is the cause of this exception.
≡	<code>CabException(String)</code>	Creates a new CabException with the specified error message.

	 <a href="#">CabException()</a>	Creates a new CabException.
	 <a href="#">CabException(SerializationInfo, StreamingContext)</a>	Initializes a new instance of the CabException class with seriali
	<a href="#">Data</a>	Gets a collection of key/value p provide additional, user-defined information about the exception  (Inherited from <a href="#">Exception</a> .)
	 <a href="#">Equals(Object)</a>	Determines whether the specific equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Error</a>	Gets the FCI or FDI cabinet eng number.
	<a href="#">ErrorCode</a>	Gets the Win32 error code.
	 <a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to f resources and perform other cle operations before the <a href="#">Object</a> is by garbage collection.  (Inherited from <a href="#">Object</a> .)
	 <a href="#">GetBaseException()</a>	When overridden in a derived c returns the <a href="#">Exception</a> that is the of one or more subsequent exc  (Inherited from <a href="#">Exception</a> .)
		

	<a href="#">GetHashCode()</a>	Serves as a hash function for a type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures that require a hash table.
		(Inherited from <a href="#">Object</a> .)
≡	<a href="#">GetObjectData(SerializationInfo, StreamingContext)</a>	Sets the <code>SerializationInfo</code> with information about the exception.
		(Overrides <a href="#">Exception.GetObjectData(SerializationInfo, StreamingContext)</a> .)
≡	<a href="#">GetType()</a>	Gets the runtime type of the current instance.
		(Inherited from <a href="#">Exception</a> .)
🔗	<a href="#">HelpLink</a>	Gets or sets a link to the help file associated with this exception.
		(Inherited from <a href="#">Exception</a> .)
📋	<a href="#">HResult</a>	Gets or sets HRESULT, a code-based value that is assigned to a specific exception.
		(Inherited from <a href="#">Exception</a> .)
📋	<a href="#">InnerException</a>	Gets the <a href="#">Exception</a> instance that caused the current exception.
		(Inherited from <a href="#">Exception</a> .)
💡	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current object.
		(Inherited from <a href="#">Object</a> .)
📋	<a href="#">Message</a>	Gets a message that describes the exception.

		(Inherited from <a href="#">Exception</a> .)
	<a href="#">Source</a>	Gets or sets the name of the application or the object that causes the error.
		(Inherited from <a href="#">Exception</a> .)
	<a href="#">StackTrace</a>	Gets a string representation of the stack trace on the call stack at the time the exception was thrown.
		(Inherited from <a href="#">Exception</a> .)
	<a href="#">TargetSite</a>	Gets the method that throws the exception.
		(Inherited from <a href="#">Exception</a> .)
	<a href="#">ToString()</a>	Creates and returns a string representation of the current exception.
		(Inherited from <a href="#">Exception</a> .)

## Inheritance Hierarchy




---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## CabException Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabException ►

**CabException()**

C# ▾

### ▀ Members

Icon	Member	Description
≡ ♦	<a href="#">CabException(String, Exception)</a>	Creates a new CabException with a specified error message and a reference to the inner exception that is the cause of this exception.
≡ ♦	<a href="#">CabException(String)</a>	Creates a new CabException with a specified error message.
≡ ♦	<a href="#">CabException()</a>	Creates a new CabException.
💡 ♦	<a href="#">CabException(SerializationInfo, StreamingContext)</a>	Initializes a new instance of the CabException class with serialized data.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CabException Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabException ►

**CabException()**

C# ▾

Creates a new CabException.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public CabException()
```

```
Public Sub New
```

```
public:  
CabException()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CabException Constructor (info, context)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression.Cab ► CabException ►  
**CabException(SerializationInfo, StreamingContext)**

C# ▾

Initializes a new instance of the CabException class with serialized data

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected CabException(  
    SerializationInfo info,  
    StreamingContext context  
)
```

```
Protected Sub New (  
    info As SerializationInfo, _  
    context As StreamingContext _  
)
```

```
protected:  
CabException(  
    SerializationInfo^ info,  
    StreamingContext context  
)
```

### Parameters

#### *info* (SerializationInfo)

The SerializationInfo that holds the serialized object data about the exception being thrown.

#### *context* (StreamingContext)

The StreamingContext that contains contextual information about the source or destination.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CabException Constructor (message)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression.Cab ► CabException ►  
**CabException(String)**

C#

Creates a new CabException with a specified error message.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public CabException(  
    string message  
)
```

```
Public Sub New ( _  
    message As String _  
)
```

```
public:  
CabException(  
    String^ message  
)
```

### - Parameters

#### **message (String)**

The message that describes the error.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CabException Constructor (message, innerException)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabException ►

**CabException(String, Exception)**

C# ▾

Creates a new CabException with a specified error message and a reference to the inner exception that is the cause of this exception.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public CabException(  
    string message,  
    Exception innerException  
)
```

```
Public Sub New ( _  
    message As String, _  
    innerException As Exception _  
)
```

```
public:  
CabException(  
    String^ message,  
    Exception^ innerException  
)
```

### Parameters

#### **message (String)**

The message that describes the error.

#### **innerException (Exception)**

The exception that is the cause of the current exception. If the innerException parameter is not a null reference (Nothing in Visual Basic), the current exception is raised in a catch block that handles

the inner exception.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Error Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabException ► Error

C# ▾

Gets the FCI or FDI cabinet engine error number.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public int Error { get; }
```

```
Public Readonly Property Error As Integer
```

```
public:  
property int Error {  
    int get ();  
}
```

### Value

A cabinet engine error number, or 0 if the exception was not related to a cabinet engine error number.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **ErrorCode** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabException ►

### ErrorCode

C# ▾

Gets the Win32 error code.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public int ErrorCode { get; }
```

```
Public ReadOnly Property ErrorCode As Integer
```

```
public:  
property int ErrorCode {  
    int get();  
}
```

#### Value

A Win32 error code, or 0 if the exception was not related to a Win32 error.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetObjectData Method (info, context)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression.Cab ► CabException ►  
**GetObjectData(SerializationInfo, StreamingContext)**

C#

Sets the SerializationInfo with information about the exception.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public override void GetObjectData(
    SerializationInfo info,
    StreamingContext context
)
```

```
Public Overrides Sub GetObjectData ( _
    info As SerializationInfo, _
    context As StreamingContext _
)
```

```
public:
virtual void GetObjectData(
    SerializationInfo^ info,
    StreamingContext context
) override
```

[-] **Parameters**

**info (SerializationInfo)**

The SerializationInfo that holds the serialized object data about the exception being thrown.

**context (StreamingContext)**

The StreamingContext that contains contextual information about the source or destination.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **CabFileInfo Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabFileInfo

C# ▾

Object representing a compressed file within a cabinet package; provides operations for getting the file properties and extracting the file.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
[SerializableAttribute]
public class CabFileInfo : ArchiveFileInfo
```

```
<SerializableAttribute> _
Public Class CabFileInfo
    Inherits ArchiveFileInfo
```

```
[SerializableAttribute]
public ref class CabFileInfo : public ArchiveFileInfo
```

### Members

All Members

Constructors

Methods

Properties

Fields

Public

Instance

Declared

Protected

Static

Inherited

Icon	Member	Description
💡	CabFileInfo(CabInfo, String)	Creates a new CabinetFileInfo representing a file within a cab path.
💡	CabFileInfo(SerializationInfo, StreamingContext)	Initializes a new instance of the class with serialized data.
📝		

	<b>Archive</b>	Gets or sets the archive that contains this file.  (Inherited from <a href="#">ArchiveFileInfo</a> .)
	<b>ArchiveName</b>	Gets the full path of the archive that contains this file.  (Inherited from <a href="#">ArchiveFileInfo</a> .)
	<b>ArchiveNumber</b>	Gets the number of the archive that contains this file.  (Inherited from <a href="#">ArchiveFileInfo</a> .)
	<b>Attributes</b>	Gets the attributes of the file.  (Inherited from <a href="#">ArchiveFileInfo</a> .)
	<b>Cabinet</b>	Gets or sets the cabinet that contains this file.  (Inherited from <a href="#">ArchiveFileInfo</a> .)
	<b>CabinetFolderNumber</b>	Gets the number of the folder containing this file.
	<b>CabinetName</b>	Gets the full path of the cabinet that contains this file.
	<b>CopyTo(String)</b>	Extracts the file.  (Inherited from <a href="#">ArchiveFileInfo</a> .)
	<b>CopyTo(String, Boolean)</b>	Extracts the file, optionally overwriting an existing file.  (Inherited from <a href="#">ArchiveFileInfo</a> .)
	<b>CreateObjRef(Type)</b>	Creates an object that contains information required to generate a proxy to communicate with a remote object.

		(Inherited from <a href="#">MarshalByRefObject</a> )
	<a href="#">CreationTime</a>	Gets or sets the creation time of the current <a href="#">FileSystemInfo</a> object.
		(Inherited from <a href="#">FileSystemInfo</a> )
	<a href="#">CreationTimeUtc</a>	Gets or sets the creation time, in universal time (UTC), of the current <a href="#">FileSystemInfo</a> object.
		(Inherited from <a href="#">FileSystemInfo</a> )
	<a href="#">Delete()</a>	Deletes the file. NOT SUPPORTED.
		(Inherited from <a href="#">ArchiveFileInfo</a> )
	<a href="#">Equals(Object)</a>	Determines whether the specified object is equal to the current <a href="#">Object</a> .
		(Inherited from <a href="#">Object</a> )
	<a href="#">Exists</a>	Checks if the file exists within the current directory.
		(Inherited from <a href="#">ArchiveFileInfo</a> )
	<a href="#">Extension</a>	Gets the string representing the extension of the file.
		(Inherited from <a href="#">FileSystemInfo</a> )
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.
		(Inherited from <a href="#">Object</a> )
	<a href="#">FullName</a>	Gets the full path to the file.
		(Inherited from <a href="#">ArchiveFileInfo</a> )
	<a href="#">FullPath</a>	Represents the fully qualified path to the directory or file.
		(Inherited from <a href="#">FileSystemInfo</a> )

	<a href="#">GetHashCode()</a>	Serves as a hash function for a <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures such as <a href="#">Dictionary</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetLifetimeService()</a>	Retrieves the current lifetime service object that controls the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">GetObjectData(SerializationInfo, StreamingContext)</a>	Sets the <a href="#">SerializationInfo</a> with information required for the archive.  (Overrides <a href="#">ArchiveFileInfo.GetObjectData(StreamingContext)</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
	<a href="#">InitializeLifetimeService()</a>	Obtains a lifetime service object that controls the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">LastAccessTime</a>	Gets or sets the time the current file was last accessed.  (Inherited from <a href="#">FileSystemInfo</a> .)
	<a href="#">LastAccessTimeUtc</a>	Gets or sets the time, in coordinated universal time (UTC), that the current file was last accessed.  (Inherited from <a href="#">FileSystemInfo</a> .)
	<a href="#">LastWriteTime</a>	Gets the last modification time of the current file.  (Inherited from <a href="#">ArchiveFileInfo</a> .)
	<a href="#">LastWriteTimeUtc</a>	Gets or sets the time, in coordinated universal time (UTC), that the current file was last written.  (Inherited from <a href="#">ArchiveFileInfo</a> .)

		time (UTC), when the current file was last written to.
		(Inherited from <a href="#">FileSystemInfo</a> .)
	<a href="#">Length</a>	Gets the uncompressed size of the file.
		(Inherited from <a href="#">ArchiveFileInfo</a> .)
	<a href="#">MemberwiseClone(Boolean)</a>	Creates a shallow copy of the current <a href="#">MarshalByRefObject</a> object.
		(Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .
		(Inherited from <a href="#">Object</a> .)
	<a href="#">Name</a>	Gets the name of the file.
		(Inherited from <a href="#">ArchiveFileInfo</a> .)
	<a href="#">OpenRead()</a>	Opens the archive file for reading, without actually extracting the file to disk.
		(Inherited from <a href="#">ArchiveFileInfo</a> .)
	<a href="#">OpenText()</a>	Opens the archive file reading the file's contents in the specified encoding without actually extracting the file to disk.
		(Inherited from <a href="#">ArchiveFileInfo</a> .)
	<a href="#">OriginalPath</a>	The path originally specified by the file, whether relative or absolute.
		(Inherited from <a href="#">FileSystemInfo</a> .)
	<a href="#">Path</a>	Gets the internal path of the file.
		(Inherited from <a href="#">ArchiveFileInfo</a> .)
	<a href="#">Refresh(ArchiveFileInfo)</a>	Refreshes the information in this file with new data retrieved from an archive.

		(Overrides <a href="#">ArchiveFileInfo.Refresh(ArchiveFile)</a> )
≡	<a href="#">Refresh()</a>	Refreshes the attributes and other information about the file, by reading information from the archive.
≡	<a href="#">ToString()</a>	(Inherited from <a href="#">ArchiveFileInfo</a> ) Gets the full path to the file. (Inherited from <a href="#">ArchiveFileInfo</a> )

## [-] Inheritance Hierarchy

Object

- └ MarshalByRefObject
- └ FileSystemInfo
- └ ArchiveFileInfo
- └ **CabFileInfo**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## CabFileInfo Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabFileInfo ►

**CabFileInfo()**

C# ▾

### ▀ Members

Icon	Member	Description
✳	<a href="#">CabFileInfo(CabInfo, String)</a>	Creates a new CabinetFileInfo object representing a file within a cabinet in a specified path.
✳	<a href="#">CabFileInfo(SerializationInfo, StreamingContext)</a>	Initializes a new instance of the CabinetFileInfo class with serialize data.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CabFileInfo Constructor (cabinetInfo, filePath)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression.Cab ► CabFileInfo ►  
CabFileInfo(CabInfo, String)

C# ▾

Creates a new CabinetFileInfo object representing a file within a cabinet in a specified path.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public CabFileInfo(  
    CabInfo cabinetInfo,  
    string filePath  
)
```

```
Public Sub New ( _  
    cabinetInfo As CabInfo, _  
    filePath As String _  
)
```

```
public:  
CabFileInfo(  
    CabInfo^ cabinetInfo,  
    String^ filePath  
)
```

### - Parameters

#### **cabinetInfo (CabInfo)**

An object representing the cabinet containing the file.

#### **filePath (String)**

The path to the file within the cabinet. Usually, this is a simple file name, but if the cabinet contains a directory structure this may include the directory.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CabFileInfo Constructor (info, context)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression.Cab ► CabFileInfo ►  
CabFileInfo(SerializationInfo, StreamingContext)

C# ▾

Initializes a new instance of the CabinetFileInfo class with serialized dat

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected CabinetFileInfo(  
    SerializationInfo info,  
    StreamingContext context  
)
```

```
Protected Sub New (  
    info As SerializationInfo,  
    context As StreamingContext  
)
```

```
protected:  
CabinetFileInfo(  
    SerializationInfo^ info,  
    StreamingContext context  
)
```

### Parameters

#### info (SerializationInfo)

The SerializationInfo that holds the serialized object data about the exception being thrown.

#### context (StreamingContext)

The StreamingContext that contains contextual information about the source or destination.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Cabinet Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabFileInfo ► Cabinet

C# ▾

Gets or sets the cabinet that contains this file.

[-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public CabInfo Cabinet { get; }
```

```
Public ReadOnly Property Cabinet As CabInfo
```

```
public:  
property CabInfo^ Cabinet {  
    CabInfo^ get ();  
}
```

[-] Value

The CabinetInfo instance that retrieved this file information -- this may be null if the CabinetFileInfo object was returned directly from a stream.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **CabinetFolderNumber Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabFileInfo ►

**CabinetFolderNumber**

C# ▾

Gets the number of the folder containing this file.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public int CabinetFolderNumber { get; }
```

```
Public ReadOnly Property CabinetFolderNumber As Inte
```

```
public:  
property int CabinetFolderNumber {  
    int get ();  
}
```

[-] **Value**

The number of the cabinet folder containing this file.

[-] **Remarks**

A single folder or the first folder of a cabinet (or chain of cabinets) is numbered 0.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation CabinetName Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabFileInfo ►

## CabinetName

C# ▾

Gets the full path of the cabinet that contains this file.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string CabinetName { get; }
```

```
Public ReadOnly Property CabinetName As String
```

```
public:  
property String^ CabinetName {  
    String^ get ();  
}
```

### Value

The full path of the cabinet that contains this file.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## GetObjectData Method (info, context)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression.Cab ► CabFileInfo ►  
GetObjectData(SerializationInfo, StreamingContext)

C#

Sets the SerializationInfo with information about the archive.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public override void GetObjectData(  
    SerializationInfo info,  
    StreamingContext context  
)
```

```
Public Overrides Sub GetObjectData ( _  
    info As SerializationInfo, _  
    context As StreamingContext _  
)
```

```
public:  
virtual void GetObjectData(  
    SerializationInfo^ info,  
    StreamingContext context  
) override
```

### Parameters

#### info (SerializationInfo)

The SerializationInfo that holds the serialized object data.

#### context (StreamingContext)

The StreamingContext that contains contextual information about the source or destination.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module:  
Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Refresh Method (newFileInfo)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabFileInfo ►

### Refresh(ArchiveFileInfo)

C# ▾

Refreshes the information in this object with new data retrieved from an archive.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected override void Refresh(  
    ArchiveFileInfo newFileInfo  
)
```

```
Protected Overrides Sub Refresh ( _  
    newFileInfo As ArchiveFileInfo _  
)
```

```
protected:  
virtual void Refresh(  
    ArchiveFileInfo^ newFileInfo  
) override
```

#### Parameters

##### *newFileInfo* (ArchiveFileInfo)

Fresh instance for the same file just read from the archive.

#### Remarks

This implementation refreshes the CabinetFolderNumber.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **CabInfo Class**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression.Cab ► **CabInfo**

C# ▾

Object representing a cabinet file on disk; provides access to file-based operations on the cabinet file.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
[SerializableAttribute]  
public class CabInfo : ArchiveInfo
```

```
<SerializableAttribute> _  
Public Class CabInfo _  
    Inherits ArchiveInfo
```

```
[SerializableAttribute]  
public ref class CabInfo : public ArchiveInfo
```

### Members

All Members

Constructors

Methods

Properties

Fields

Public

Instance

Declared

Protected

Static

Inherited

Icon	Member	Description
≡	CabInfo(String)	Creates a new CabInfo object representing a cabinet file path.
💡	CabInfo(SerializationInfo, StreamingContext)	Initializes a new instance of the CabinetInfo class with the specified serialization information and streaming context.
📅		

	<b>Attributes</b>	Gets or sets the <a href="#">File</a> current <a href="#">FileSystem</a> . (Inherited from <a href="#">File</a> )
≡	<a href="#">CopyTo(String)</a>	Copies an existing archive to a location. (Inherited from <a href="#">ArchiveInfo</a> )
≡	<a href="#">CopyTo(String, Boolean)</a>	Copies an existing archive to a location, optionally creating the destination file. (Inherited from <a href="#">ArchiveInfo</a> )
💡	<a href="#">CreateCompressionEngine()</a>	Creates a compression engine for the low-level work function. (Overrides <a href="#">ArchiveInfo.CreateCompressionEngine()</a> )
≡	<a href="#">CreateObjRef(Type)</a>	Creates an object that contains relevant information for a proxy used to connect to a remote object. (Inherited from <a href="#">MarshalByObject</a> )
📅	<a href="#">CreationTime</a>	Gets or sets the creation time of the current <a href="#">FileSystem</a> . (Inherited from <a href="#">File</a> )
📅	<a href="#">CreationTimeUtc</a>	Gets or sets the creation time in coordinated universal time of the current <a href="#">FileSystem</a> . (Inherited from <a href="#">File</a> )
≡	<a href="#">Delete()</a>	Deletes the archive. (Inherited from <a href="#">ArchiveInfo</a> )

	<a href="#">Directory</a>	Gets the directory the archive.
		(Inherited from <a href="#">Archive</a> )
	<a href="#">DirectoryName</a>	Gets the full path of the folder that contains the archive.
		(Inherited from <a href="#">Archive</a> )
	<a href="#">Equals(Object)</a>	Determines whether the current instance is equal to the current object.
		(Inherited from <a href="#">Object</a> )
	<a href="#">Exists</a>	Checks if the archive exists.
		(Inherited from <a href="#">Archive</a> )
	<a href="#">Extension</a>	Gets the string representation of the file extension part of the file.
		(Inherited from <a href="#">File</a> )
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to free resources and perform operations before the object is garbage collected.
		(Inherited from <a href="#">Object</a> )
	<a href="#">FullName</a>	Gets the full path of the archive.
		(Inherited from <a href="#">File</a> )
	<a href="#">FullPath</a>	Represents the fully qualified directory or file.
		(Inherited from <a href="#">File</a> )
	<a href="#">GetFiles()</a>	Gets information about the files in the archive.

	<a href="#">GetFiles(String)</a>	Gets information about the files contained in the archive.
	<a href="#">GetHashCode()</a>	Serves as a hash function for this type. <a href="#">GetHashCode</a> is particularly useful for hashing algorithms like a hash table.  (Inherited from <a href="#">Object</a> )
	<a href="#">GetLifetimeService()</a>	Retrieves the current lifetime service object that controls this instance.  (Inherited from <a href="#">MarshalByRefObject</a> )
	<a href="#">GetObjectData(SerializationInfo, StreamingContext)</a>	Sets the <a href="#">SerializationInfo</a> object with the file name and additional information.  (Inherited from <a href="#">FileBase</a> )
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> )
	<a href="#">InitializeLifetimeService()</a>	Obtains a lifetime service object based on the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject</a> )
	<a href="#">IsValid()</a>	Checks if the archive header is valid.  (Inherited from <a href="#">ArchiveBase</a> )
	<a href="#">LastAccessTime</a>	Gets or sets the time when the directory was last accessed.  (Inherited from <a href="#">FileBase</a> )
	<a href="#">LastAccessTimeUtc</a>	Gets or sets the time when the directory was last accessed in UTC time.  (Inherited from <a href="#">FileBase</a> )

		universal time (UTC) or directory was last modified.
	(Inherited from <a href="#">File</a> )	
 <a href="#">LastWriteTime</a>	Gets or sets the time the file or directory was last written.	Gets or sets the time the file or directory was last written.
	(Inherited from <a href="#">File</a> )	
 <a href="#">LastWriteTimeUtc</a>	Gets or sets the time the file or directory was last written in universal time (UTC).	Gets or sets the time the file or directory was last written in universal time (UTC).
	(Inherited from <a href="#">File</a> )	
 <a href="#">Length</a>	Gets the size of the file.	Gets the size of the file.
	(Inherited from <a href="#">Archivable</a> )	
 <a href="#">MemberwiseClone(Boolean)</a>	Creates a shallow copy of the current <a href="#">Object</a> using the specified <a href="#">MarshalByRefObject</a> flag.	Creates a shallow copy of the current <a href="#">Object</a> using the specified <a href="#">MarshalByRefObject</a> flag.
	(Inherited from <a href="#">MarshalByRefObject</a> )	
 <a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .	Creates a shallow copy of the current <a href="#">Object</a> .
	(Inherited from <a href="#">Object</a> )	
 <a href="#">MoveTo(String)</a>	Moves an existing archive to a new location.	Moves an existing archive to a new location.
	(Inherited from <a href="#">Archivable</a> )	
 <a href="#">Name</a>	Gets the file name of the archive.	Gets the file name of the archive.
	(Inherited from <a href="#">Archivable</a> )	
 <a href="#">OpenRead(String)</a>	Opens a file inside the archive without actually extracting it.	Opens a file inside the archive without actually extracting it.
	(Inherited from <a href="#">Archivable</a> )	
 <a href="#">OpenText(String)</a>	Opens a file inside the archive.	Opens a file inside the archive.

		text with UTF-8 encoding. extracting the file to (Inherited from <a href="#">Archive</a> )
💡	<a href="#">OriginalPath</a>	The path originally specified for the file, whether relative or absolute. (Inherited from <a href="#">File</a> )
💡	<a href="#">Pack(String)</a>	Compresses all files in the archive. Does not include subdirectories. (Inherited from <a href="#">Archive</a> )
💡	<a href="#">Pack(String, Boolean, CompressionLevel, EventHandler&lt;ArchiveProgressEventArgs&gt;)</a>	Compresses all files in the archive, optionally including subdirectories. (Inherited from <a href="#">Archive</a> )
💡	<a href="#">PackFiles(String, IList&lt;String&gt;, IList&lt;String&gt;)</a>	Compresses files in the archive, specifying the names of the files in the archive. (Inherited from <a href="#">Archive</a> )
💡	<a href="#">PackFiles(String, IList&lt;String&gt;, IList&lt;String&gt;, CompressionLevel, EventHandler&lt;ArchiveProgressEventArgs&gt;)</a>	Compresses files in the archive, specifying the names of the files in the archive. (Inherited from <a href="#">Archive</a> )
💡	<a href="#">PackFileSet(String, IDictionary&lt;String, String&gt;)</a>	Compresses files in the archive, specifying the names of the files in the archive. (Inherited from <a href="#">Archive</a> )
💡	<a href="#">PackFileSet(String, IDictionary&lt;String, String&gt;, CompressionLevel, EventHandler&lt;ArchiveProgressEventArgs&gt;)</a>	Compresses files in the archive, specifying the names of the files in the archive.

		(Inherited from <a href="#">Archive</a> )
≡	<a href="#">Refresh()</a>	Refreshes the state (Inherited from <a href="#">FileSource</a> )
≡	<a href="#">ToString()</a>	Gets the full path of (Inherited from <a href="#">Archive</a> )
≡	<a href="#">Unpack(String)</a>	Extracts all files from destination directory (Inherited from <a href="#">Archive</a> )
≡	<a href="#">Unpack(String, EventHandler&lt;ArchiveProgressEventArgs&gt;)</a>	Extracts all files from destination directory, only newer files. (Inherited from <a href="#">Archive</a> )
≡	<a href="#">UnpackFile(String, String)</a>	Extracts a single file (Inherited from <a href="#">Archive</a> )
≡	<a href="#">UnpackFiles(IList&lt;String&gt;, String, IList&lt;String&gt;)</a>	Extracts multiple files (Inherited from <a href="#">Archive</a> )
≡	<a href="#">UnpackFiles(IList&lt;String&gt;, String, IList&lt;String&gt;, EventHandler&lt;ArchiveProgressEventArgs&gt;)</a>	Extracts multiple files, optionally extracting (Inherited from <a href="#">Archive</a> )
≡	<a href="#">UnpackFileSet(IDictionary&lt;String, String&gt;, String)</a>	Extracts multiple files (Inherited from <a href="#">Archive</a> )
≡	<a href="#">UnpackFileSet(IDictionary&lt;String, String&gt;, String, EventHandler&lt;ArchiveProgressEventArgs&gt;)</a>	Extracts multiple files (Inherited from <a href="#">Archive</a> )

## ■ Remarks

Generally, the methods on this class are much easier to use than the

stream-based interfaces provided by the [CabEngine](#) class.

## **[-] Inheritance Hierarchy**

Object

- └ MarshalByRefObject
- └ FileSystemInfo
- └ ArchiveInfo
- └ **CabInfo**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **CabInfo Constructor**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabInfo ► CabInfo()

C# ▾

**Members**

Icon	Member	Description
	<a href="#">CabInfo(String)</a>	Creates a new CabinetInfo object representing a cabinet file in a specified path.
	<a href="#">CabInfo(SerializationInfo, StreamingContext)</a>	Initializes a new instance of the CabinetInfo class with serialized data.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

## CabInfo Constructor (info, context)

Deployment Tools Foundation Namespaces ►  
 Microsoft.Deployment.Compression.Cab ► CabInfo ►  
**CabInfo(SerializationInfo, StreamingContext)**

C# ▾

Initializes a new instance of the CabinetInfo class with serialized data.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected CabInfo(  
    SerializationInfo info,  
    StreamingContext context  
)
```

```
Protected Sub New (  
    info As SerializationInfo, _  
    context As StreamingContext _  
)
```

```
protected:  
CabInfo(  
    SerializationInfo^ info,  
    StreamingContext context  
)
```

### Parameters

#### *info* (SerializationInfo)

The SerializationInfo that holds the serialized object data about the exception being thrown.

#### *context* (StreamingContext)

The StreamingContext that contains contextual information about the source or destination.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation CabInfo Constructor (path)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabInfo ► CabInfo(String)

C# ▾

Creates a new CabinetInfo object representing a cabinet file in a specified path.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public CabInfo(  
    string path  
)
```

```
Public Sub New (  
    path As String  
)
```

```
public:  
CabInfo(  
    String^ path  
)
```

## Parameters

### *path* (String)

The path to the cabinet file. When creating a cabinet file, this file does not necessarily exist yet.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

## Method

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression.Cab ► CabInfo ►  
**CreateCompressionEngine()**

C# ▾

Creates a compression engine that does the low-level work for this object.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected override CompressionEngine CreateCompressi
```

```
Protected Overrides Function CreateCompressionEngine
```

```
protected:  
virtual CompressionEngine^ CreateCompressionEngine()
```

### Return Value

A new [CabEngine](#) instance.

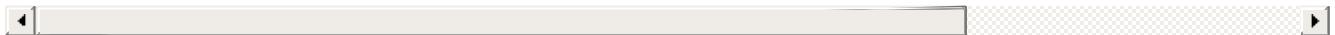
### Remarks

Each instance will be [Dispose\(\)](#)d immediately after use.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **GetFiles Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabInfo ► GetFiles()

C# ▾

▀ **Members**

Icon	Member	Description
	<a href="#">GetFiles()</a>	Gets information about the files contained in the archive.
	<a href="#">GetFiles(String)</a>	Gets information about the certain file contained in the archive file.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetFiles Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Compression.Cab ► CabInfo ► GetFiles()

C# ▾

Gets information about the files contained in the archive.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public IList<CabFileInfo> GetFiles()
```

```
Public Function GetFiles As IList(Of CabFileInfo)
```

```
public:  
IList<CabFileInfo^>^ GetFiles()
```

[-] **Return Value**

A list of [CabFileInfo](#) objects, each containing information about a file in the archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **GetFiles Method (searchPattern)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Compression.Cab ► CabInfo ►  
**GetFiles(String)**

C#

Gets information about the certain files contained in the archive file.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public IList<CabFileInfo> GetFiles(  
    string searchPattern  
)
```

```
Public Function GetFiles ( _  
    searchPattern As String _  
) As IList(of CabFileInfo)
```

```
public:  
IList<CabFileInfo^>^ GetFiles(  
    String^ searchPattern  
)
```

## Parameters

### **searchPattern (String)**

The search string, such as "\*.txt".

## Return Value

A list of [CabFileInfo](#) objects, each containing information about a file in the archive.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Compression.Cab (Module: Microsoft.Deployment.Compression.Cab) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# Microsoft.Deployment.Resources Namespace

Deployment Tools Foundation Namespaces ►  
**Microsoft.Deployment.Resources**

C# ▾

Classes for reading and writing resource data in executable files.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
namespace Microsoft.Deployment.Resources
```

```
Namespace Microsoft.Deployment.Resources
```

```
namespace Microsoft.Deployment.Resources
```

## Types

All Types

Classes

Enumerations

Icon	Type	Description
	<a href="#">Resource</a>	Represents a Win32 resource which can be loaded from and saved to a PE file.
	<a href="#">ResourceCollection</a>	Allows reading and editing of resource data in a Win32 PE file.
	<a href="#">ResourceType</a>	Represents either a standard integer resource type or a custom resource type name.
	<a href="#">VersionBuildTypes</a>	Identifies build types of a versioned

		file.
	<b>VersionFileSubtype</b>	Identifies the sub-type of a versioned file.
	<b>VersionFileType</b>	Identifies the type of a versioned file.
	<b>VersionResource</b>	A subclass of Resource which provides specific methods for manipulating the resource data.
	<b>VersionStringTable</b>	Represents a string table of a file version resource.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

# Deployment Tools Foundation **Resource Class**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Resources ► **Resource**

C#

Represents a Win32 resource which can be loaded from and saved to a PE file.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public class Resource
```

```
Public Class Resource
```

```
public ref class Resource
```

## Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon

Icon	Member	Description
	<a href="#">Resource(ResourceType, String, Int32)</a>	Creates a new Resource object without any data. The data can be later loaded from a file.
	<a href="#">Resource(ResourceType, String, Int32, Byte[])</a>	Creates a new Resource object with data. The data can be later saved to a file.
	<a href="#">Data</a>	Gets or sets the raw data of the resource.

	<a href="#">Equals(Object)</a>	Tests if type, name, and locale of this Resource object match another Resource object.  (Overrides <a href="#">Object.Equals(Object)</a> .)
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetHashCode()</a>	Gets a hash code for this Resource object.  (Overrides <a href="#">Object.GetHashCode()</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
	<a href="#">Load(String)</a>	Loads the resource data from a file. The file is searched for a resource with matching type, name, and locale.
	<a href="#">Locale</a>	Gets or sets the locale of the resource.
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Name</a>	Gets or sets the name of the resource. For a numeric resource identifier, the decimal number is

		prefixed with a "#".
	<a href="#"> ResourceType</a>	Gets or sets the type of the resource. This may be one of the ResourceType constants or a user-defined type name.
	<a href="#"> Save(String)</a>	Saves the resource to a file. Any existing resource data with matching type, name, and locale is overwritten.
	<a href="#"> ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)

## Inheritance Hierarchy

```
Object
  ↘ Resource
    ↘ VersionResource
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

C# ▾

## Members

Icon	Member	Description
	<code>Resource(ResourceType, String, Int32)</code>	Creates a new Resource object without any data. The data can be later loaded from a file.
	<code>Resource(ResourceType, String, Int32, Byte[])</code>	Creates a new Resource object with data. The data can be later saved to a file.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Resource Constructor (type, name, locale)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Resources ► Resource ►  
**Resource(ResourceType, String, Int32)**

C# ▾

Creates a new Resource object without any data. The data can be later loaded from a file.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public Resource(  
    ResourceType type,  
    string name,  
    int locale  
)
```

```
Public Sub New ( _  
    type As ResourceType, _  
    name As String, _  
    locale As Integer _  
)
```

```
public:  
Resource(  
    ResourceType^ type,  
    String^ name,  
    int locale  
)
```

[-] **Parameters**

**type (ResourceType)**

Type of the resource; may be one of the ResourceType constants or a user-defined type.

***name*** ([String](#))

Name of the resource. For a numeric resource identifier, prefix the decimal number with a "#".

***locale*** ([Int32](#))

Locale of the resource

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

# Resource Constructor (type, name, locale, data)

Deployment Tools Foundation Namespaces ►  
 Microsoft.Deployment.Resources ► Resource ►  
**Resource(ResourceType, String, Int32, Byte[])**

C# ▾

Creates a new Resource object with data. The data can be later saved to a file.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public Resource(
    ResourceType type,
    string name,
    int locale,
    byte[] data
)
```

```
Public Sub New ( _
    type As ResourceType, _
    name As String, _
    locale As Integer, _
    data As Byte() _
)
```

```
public:
Resource(
    ResourceType^ type,
    String^ name,
    int locale,
    array<unsigned char>^ data
)
```

## Parameters

**type (ResourceType)**

Type of the resource; may be one of the ResourceType constants or a user-defined type.

***name* ([String](#))**

Name of the resource. For a numeric resource identifier, prefix the decimal number with a "#".

***locale* ([Int32](#))**

Locale of the resource

***data* ([Byte \[\]](#))**

Raw resource data

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# Data Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► Resource ► Data

C# ▾

Gets or sets the raw data of the resource.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public virtual byte[] Data { get; set; }
```

```
Public Overridable Property Data As Byte()
```

```
public:  
virtual property array<unsigned char>^ Data {  
    array<unsigned char>^ get ();  
    void set (array<unsigned char>^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Equals Method (obj)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► Resource ► Equals(Object)

C#

Tests if type, name, and locale of this Resource object match another Resource object.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public override bool Equals(  
    Object obj  
)
```

```
Public Overrides Function Equals ( _  
    obj As Object _  
) As Boolean
```

```
public:  
virtual bool Equals(  
    Object^ obj  
) override
```

### Parameters

#### *obj* (Object)

Resource object to be compared

### Return Value

True if the objects represent the same resource; false otherwise.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetHashCode Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► Resource ► **GetHashCode()**

C# ▾

Gets a hash code for this Resource object.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public override int GetHashCode()
```

```
Public Overrides Function GetHashCode As Integer
```

```
public:  
virtual int GetHashCode() override
```

[-] **Return Value**

Hash code generated from the resource type, name, and locale.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Load Method (file)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► Resource ► Load(String)

C# ▾

Loads the resource data from a file. The file is searched for a resource with matching type, name, and locale.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Load(  
    string file  
)
```

```
Public Sub Load ( _  
    file As String _  
)
```

```
public:  
void Load(  
    String^ file  
)
```

### Parameters

#### **file (String)**

Win32 PE file containing the resource

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Locale Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► Resource ► Locale

C# ▾

Gets or sets the locale of the resource.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public int Locale { get; set; }
```

```
Public Property Locale As Integer
```

```
public:  
property int Locale {  
    int get ();  
    void set (int value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Name Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► Resource ► Name

C# ▾

Gets or sets the name of the resource. For a numeric resource identifier the decimal number is prefixed with a "#".

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string Name { get; set; }
```

```
Public Property Name As String
```

```
public:  
property String^ Name {  
    String^ get ();  
    void set (String^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation ResourceType Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► Resource ► ResourceType

C# ▾

Gets or sets the type of the resource. This may be one of the ResourceType constants or a user-defined type name.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public ResourceType ResourceType { get; set; }
```

```
Public Property ResourceType As ResourceType
```

```
public:  
property ResourceType^ ResourceType {  
    ResourceType^ get ();  
    void set (ResourceType^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Save Method (file)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► Resource ► **Save(String)**

C#

Saves the resource to a file. Any existing resource data with matching type, name, and locale is overwritten.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Save(  
    string file  
)
```

```
Public Sub Save ( _  
    file As String _  
)
```

```
public:  
void Save(  
    String^ file  
)
```

### Parameters

#### **file (String)**

Win32 PE file to contain the resource

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **ResourceCollection Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► **ResourceCollection**

C#

Allows reading and editing of resource data in a Win32 PE file.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public class ResourceCollection : ICollection<Resour  
IEnumerable<Resource>, IEnumerable
```

```
Public Class ResourceCollection _  
    Implements ICollection(Of Resource), IEnumerable  
    IEnumerable
```

```
public ref class ResourceCollection : ICollection<Re  
IEnumerable<Resource^>, IEnumerable
```

## Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declare

Inherit

Icon	Member	Description
	<a href="#">ResourceCollection()</a>	Creates a new, empty ResourceCollection.
	<a href="#">Add(Resource)</a>	Adds a new item to the collection.
	<a href="#">Clear()</a>	Removes all resources from the collection.

	 <a href="#">Contains(Resource)</a>	Tests if the collection contains an item.
	 <a href="#">CopyTo(Resource[], Int32)</a>	Copies the collection into an array.
	<a href="#">Count</a>	Gets the number of resources in the collection.
	 <a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
	 <a href="#">Find(String)</a>	Locates all resources in a file, including all resource types and languages. For each located resource, a <a href="#">Resource</a> instance (or subclass) is added to the collection.
	 <a href="#">Find(String, ResourceType)</a>	Locates all resources in a file of a given type, including all languages. For each located resource, a <a href="#">Resource</a> instance (or subclass) is added to the collection.

 <a href="#">Find(String, ResourceType, String)</a>	Locates all resources in a file of a given type and language. For each located resource, a <a href="#">Resource</a> instance (or subclass) is added to the collection.
 <a href="#">GetEnumerator()</a>	Gets an enumerator over all resources in the collection.
 <a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.  (Inherited from <a href="#">Object</a> .)
 <a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance  (Inherited from <a href="#">Object</a> .)
 <a href="#">IndexOf(Resource)</a>	Gets the index of an item in the collection.
 <a href="#">Insert(Int32, Resource)</a>	Inserts a item into the collection.
 <a href="#">Item[Int32]</a>	Gets or sets the element at the specified index.
 <a href="#">Load(String)</a>	For all resources in the collection, loads their data from a resource file.
	

	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">Remove(Resource)</a>	Removes an item to the collection.
≡	<a href="#">Save(String)</a>	For all resources in the collection, saves their data to a resource file.
≡	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)

## [-] Remarks

To use this class:

1. Create a new [ResourceCollection](#)
2. Locate resources for the collection by calling one of the [Find\(String\)](#) methods
3. Load data of one or more [Resources](#) from a file by calling the [Load\(String\)](#) method of the [Resource](#) class, or load them all at once (more efficient) with the [Load\(String\)](#) method of the [ResourceCollection](#).
4. Read and/or edit data of the individual [Resource](#) objects using the methods on that class.
5. Save data of one or more [Resources](#) to a file by calling the [Save\(String\)](#) method of the [Resource](#) class, or save them all at once (more efficient) with the [Save\(String\)](#) method of the [ResourceCollection](#).

## [-] Inheritance Hierarchy

[Object](#)

└ [ResourceCollection](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **ResourceCollection Constructor**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceCollection ►

## ResourceCollection()

C# ▾

Creates a new, empty ResourceCollection.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public ResourceCollection()
```

```
Public Sub New
```

```
public:  
ResourceCollection()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Add Method (item)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceCollection ►

### Add(Resource)

C# ▾

Adds a new item to the collection.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Add(  
    Resource item  
)
```

```
Public Sub Add ( _  
    item As Resource _  
)
```

```
public:  
virtual void Add(  
    Resource^ item  
) sealed
```

#### Parameters

##### *item* (Resource)

The Resource to add.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

C# ▾

Removes all resources from the collection.

## - Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Clear()
```

```
Public Sub Clear
```

```
public:  
virtual void Clear() sealed
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Contains Method (item)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceCollection ►

**Contains(Resource)**

C# ▾

Tests if the collection contains an item.

▀ **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public bool Contains(  
    Resource item  
)
```

```
Public Function Contains ( _  
    item As Resource _  
) As Boolean
```

```
public:  
virtual bool Contains(  
    Resource^ item  
) sealed
```

▀ **Parameters**

***item* (Resource)**

The Resource to search for.

▀ **Return Value**

true if the item is found; false otherwise

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CopyTo Method (array, arrayIndex)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Resources ► ResourceCollection ►  
**CopyTo(Resource[], Int32)**

C# ▾

Copies the collection into an array.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void CopyTo(  
    Resource[] array,  
    int arrayIndex  
)
```

```
Public Sub CopyTo ( _  
    array As Resource(), _  
    arrayIndex As Integer _  
)
```

```
public:  
virtual void CopyTo(  
    array<Resource^>^ array,  
    int arrayIndex  
) sealed
```

### Parameters

#### array ( Resource [] )

The array to copy into.

#### arrayIndex (Int32)

The starting index in the destination array.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Count Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceCollection ► Count

C# ▾

Gets the number of resources in the collection.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public int Count { get; }
```

```
Public ReadOnly Property Count As Integer
```

```
public:  
virtual property int Count {  
    int get () sealed;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

▾

## Members

Icon	Member	Description
	<a href="#">Find(String)</a>	Locates all resources in a file, including all resource types and languages. For each located resource, a <a href="#">Resource</a> instance (or subclass) is added to the collection.
	<a href="#">Find(String, ResourceType)</a>	Locates all resources in a file of a given type, including all languages. For each located resource, a <a href="#">Resource</a> instance (or subclass) is added to the collection.
	<a href="#">Find(String, ResourceType, String)</a>	Locates all resources in a file of a given type and language. For each located resource, a <a href="#">Resource</a> instance (or subclass) is added to the collection.

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Find Method (resFile)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceCollection ►

### Find(String)

C# ▾

Locates all resources in a file, including all resource types and languages. For each located resource, a [Resource](#) instance (or subclass) is added to the collection.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Find(  
    string resFile  
)
```

```
Public Sub Find ( _  
    resFile As String _  
)
```

```
public:  
void Find(  
    String^ resFile  
)
```

### Parameters

#### resFile ([String](#))

The file to be searched for resources.

### Exceptions

Exception

Condition

IOException

resources could not be read from the file

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Find Method (resFile, type)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceCollection ►

**Find(String, ResourceType)**

C# ▾

Locates all resources in a file of a given type, including all languages. For each located resource, a [Resource](#) instance (or subclass) is added to the collection.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Find(  
    string resFile,  
    ResourceType type  
)
```

```
Public Sub Find ( _  
    resFile As String, _  
    type As ResourceType _  
)
```

```
public:  
void Find(  
    String^ resFile,  
    ResourceType^ type  
)
```

### Parameters

#### **resFile (String)**

The file to be searched for resources.

#### **type (ResourceType)**

The type of resource to search for; may be one of the ResourceType constants or a user-defined type.

### Exceptions

Exception	Condition
IOException	resources could not be read from the file

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Find Method (resFile, type, name)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceCollection ►

**Find(String, ResourceType, String)**

C#

Locates all resources in a file of a given type and language. For each located resource, a [Resource](#) instance (or subclass) is added to the collection.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Find(  
    string resFile,  
    ResourceType type,  
    string name  
)
```

```
Public Sub Find ( _  
    resFile As String, _  
    type As ResourceType, _  
    name As String _  
)
```

```
public:  
void Find(  
    String^ resFile,  
    ResourceType^ type,  
    String^ name  
)
```

### Parameters

**resFile (String)**

The file to be searched for resources.

**type (ResourceType)**

The type of resource to search for; may be one of the ResourceType

constants or a user-defined type.

***name* ([String](#))**

The name of the resource to search for.

**Exceptions**

Exception	Condition
<a href="#">IOException</a>	resources could not be read from the file

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetEnumerator Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceCollection ►

### **GetEnumerator()**

C# ▾

Gets an enumerator over all resources in the collection.

#### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public IEnum<Resource> GetEnumerator()
```

```
Public Function GetEnumerator As IEnum<Resource>
```

```
public:  
virtual IEnum<Resource>^ GetEnumerator() seal
```

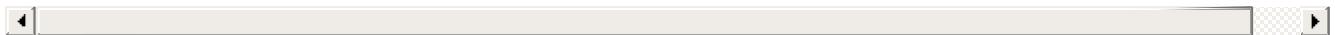
#### [-] Return Value

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)



## Deployment Tools Foundation **IndexOf Method (item)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceCollection ►

### IndexOf(Resource)

C# ▾

Gets the index of an item in the collection.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public int IndexOf(  
    Resource item  
)
```

```
Public Function IndexOf ( _  
    item As Resource _  
) As Integer
```

```
public:  
int IndexOf(  
    Resource^ item  
)
```

#### Parameters

##### *item* (Resource)

The Resource to search for.

#### Return Value

The index of the item, or -1 if not found.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Insert Method (index, item)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceCollection ►

### **Insert(Int32, Resource)**

C# ▾

Inserts a item into the collection.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Insert(  
    int index,  
    Resource item  
)
```

```
Public Sub Insert ( _  
    index As Integer, _  
    item As Resource _  
)
```

```
public:  
void Insert(  
    int index,  
    Resource^ item  
)
```

#### Parameters

##### ***index* (Int32)**

The insertion index.

##### ***item* (Resource)**

The Resource to insert.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Item Property (index)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceCollection ► Item[Int32]

C# ▾

Gets or sets the element at the specified index.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public Resource this[  
    int index  
] { get; set; }
```

```
Public Default Property Item ( _  
    index As Integer _  
) As Resource
```

```
public:  
property Resource^ default[int index] {  
    Resource^ get (int index);  
    void set (int index, Resource^ value);  
}
```

### - Parameters

*index* (Int32)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Load Method (file)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceCollection ►

### **Load(String)**

C#

For all resources in the collection, loads their data from a resource file.

#### [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Load(  
    string file  
)
```

```
Public Sub Load ( _  
    file As String _  
)
```

```
public:  
void Load(  
    String^ file  
)
```

#### [-] Parameters

##### **file (String)**

The file from which resources are loaded.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Remove Method (item)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceCollection ►

### Remove(Resource)

C# ▾

Removes an item to the collection.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public bool Remove(  
    Resource item  
)
```

```
Public Function Remove ( _  
    item As Resource _  
) As Boolean
```

```
public:  
virtual bool Remove(  
    Resource^ item  
) sealed
```

#### Parameters

##### *item* (Resource)

The Resource to remove.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Save Method (file)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceCollection ►

### Save(String)

C#

For all resources in the collection, saves their data to a resource file.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Save(  
    string file  
)
```

```
Public Sub Save ( _  
    file As String _  
)
```

```
public:  
void Save(  
    String^ file  
)
```

#### Parameters

##### file (String)

The file to which resources are saved.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation ResourceType Class

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType

C# ▾

Represents either a standard integer resource type or a custom resource type name.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public class ResourceType
```

```
Public Class ResourceType
```

```
public ref class ResourceType
```

## Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon

Icon	Member	Description
	<a href="#">ResourceType(String)</a>	Creates a new resource type from a string resource name.
	<a href="#">ResourceType(Int32)</a>	Creates a new integer resource type.
	<a href="#">Accelerator</a>	
	<a href="#">AniCursor</a>	
	<a href="#">Anilcon</a>	

 <a href="#">S</a>	<a href="#">Bitmap</a>	
 <a href="#">S</a>	<a href="#">Cursor</a>	
 <a href="#">S</a>	<a href="#">Dialog</a>	
 <a href="#">S</a>	<a href="#">DialogInclude</a>	
 <a href="#">S</a>	<a href="#">Equals(Object)</a>	Tests whether one resource type equals another object.  (Overrides <a href="#">Object.Equals(Object)</a> .)
 <a href="#">S</a>	<a href="#">Equals(ResourceType)</a>	Tests whether one resource type equals another.
 <a href="#">S</a>	<a href="#">Explicit(ResourceType)</a>	Explicitly converts a ResourceType to a string.
 <a href="#">S</a>	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
 <a href="#">S</a>	<a href="#">Font</a>	
 <a href="#">S</a>	<a href="#">FontDir</a>	
 <a href="#">S</a>	<a href="#">GetHashCode()</a>	Gets a hash code suitable for using the resource type as a dictionary key  (Overrides <a href="#">Object.GetHashCode()</a> .)
 <a href="#">S</a>	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance  (Inherited from <a href="#">Object</a> .)
 <a href="#">S</a>	<a href="#">GroupCursor</a>	
 <a href="#">S</a>	<a href="#">GroupIcon</a>	

 <a href="#">Html</a>		
 <a href="#">Icon</a>		
 <a href="#">Implicit(String)</a>		Implicitly converts a string to a ResourceType.
 <a href="#">IntegerValue</a>		Gets the integer value of the resource type, or -1 if the resource type is not an integer.
 <a href="#">IsInteger</a>		Gets a flag indicating whether the resource type is an integer type.
 <a href="#">Manifest</a>		
 <a href="#">MemberwiseClone()</a>		Creates a shallow copy of the current Object.  (Inherited from <a href="#">Object</a> .)
 <a href="#">Menu</a>		
 <a href="#">MessageTable</a>		
 <a href="#">None</a>		
 <a href="#">PlugPlay</a>		
 <a href="#">RCData</a>		
 <a href="#">String</a>		
 <a href="#">ToString()</a>		Gets a string representation of the resource type.  (Overrides <a href="#">Object.ToString()</a> .)
 <a href="#">Version</a>		
 <a href="#">Vxd</a>		

---

## **[-] Inheritance Hierarchy**

Object

└ ResourceType

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# ResourceType Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► ResourceType()

C# ▾

## Members

Icon	Member	Description
≡	<a href="#">ResourceType(String)</a>	Creates a new resource type from a string resource name.
≡	<a href="#">ResourceType(Int32)</a>	Creates a new integer resource type

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ResourceType Constructor (resourceType)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Resources ► ResourceType ►  
ResourceType(Int32)

C# ▾

Creates a new integer resource type.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public ResourceType(  
    int resourceType  
)
```

```
Public Sub New ( _  
    resourceType As Integer _  
)
```

```
public:  
ResourceType(  
    int resourceType  
)
```

### Parameters

#### resourceType (Int32)

Integer value of a well-known resource type.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **ResourceType Constructor  
(resourceType)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Resources ► ResourceType ►  
**ResourceType(String)**

C#

Creates a new resource type from a string resource name.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public ResourceType(  
    string resourceType  
)
```

```
Public Sub New ( _  
    resourceType As String _  
)
```

```
public:  
ResourceType(  
    String^ resourceType  
)
```

[-] **Parameters**

**resourceType (String)**

String resource name, or an integer resource type prefixed by a #.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Accelerator Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► Accelerator

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static ResourceType Accelerator { get; }
```

```
Public Shared ReadOnly Property Accelerator As Resou
```

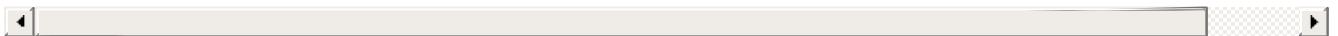
```
public:  
static property ResourceType^ Accelerator {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## AniCursor Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► AniCursor

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static ResourceType AniCursor { get; }
```

```
Public Shared ReadOnly Property AniCursor As Resourc
```

```
public:  
static property ResourceType^ AniCursor {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## Anilcon Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► Anilcon

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static ResourceType AniIcon { get; }
```

```
Public Shared ReadOnly Property AniIcon As ResourceType
```

```
public:  
static property ResourceType^ AniIcon {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Bitmap Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► Bitmap

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static ResourceType Bitmap { get; }
```

```
Public Shared ReadOnly Property Bitmap As ResourceType
```

```
public:  
static property ResourceType^ Bitmap {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Cursor Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► Cursor

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static ResourceType Cursor { get; }
```

```
Public Shared ReadOnly Property Cursor As ResourceType
```

```
public:  
static property ResourceType^ Cursor {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Dialog Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► Dialog

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static ResourceType Dialog { get; }
```

```
Public Shared ReadOnly Property Dialog As ResourceType
```

```
public:  
static property ResourceType^ Dialog {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## DialogInclude Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► DialogInclude

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static ResourceType DialogInclude { get; }
```

```
Public Shared ReadOnly Property DialogInclude As Res
```

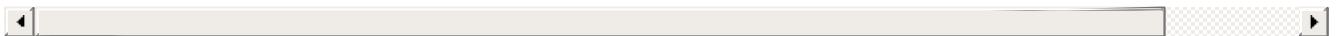
```
public:  
static property ResourceType^ DialogInclude {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)



C# ▾

## Members

Icon	Member	Description
	<a href="#">Equals(Object)</a>	Tests whether one resource type equals another object.  (Overrides <a href="#">Object.Equals(Object)</a> .)
	<a href="#">Equals(ResourceType)</a>	Tests whether one resource type equals another.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Equals Method (otherType)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Resources ► ResourceType ►  
**Equals(ResourceType)**

C# ▾

Tests whether one resource type equals another.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public bool Equals(  
    ResourceType otherType  
)
```

```
Public Function Equals ( _  
    otherType As ResourceType _  
) As Boolean
```

```
public:  
bool Equals(  
    ResourceType^ otherType  
)
```

## Parameters

### *otherType* (ResourceType)

Other resource type.

## Return Value

True if equal, else false.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Equals Method (obj)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► Equals(Object)

C# ▾

Tests whether one resource type equals another object.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override bool Equals(  
    Object obj  
)
```

```
Public Overrides Function Equals ( _  
    obj As Object _  
) As Boolean
```

```
public:  
virtual bool Equals(  
    Object^ obj  
) override
```

### Parameters

#### *obj* (Object)

Other object.

### Return Value

True if equal, else false.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Explicit Operator**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Resources ► ResourceType ►  
**Explicit(ResourceType)**

C# ▾

Explicitly converts a ResourceType to a string.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static explicit operator string (
    ResourceType resourceType
)
```

```
Public Shared Narrowing Operator CType ( _
    resourceType As ResourceType _
) As String
```

```
static explicit operator String^ (
    ResourceType^ resourceType
)
```

## Parameters

### *resourceType* (ResourceType)

ResourceType object to convert.

## Return Value

The resource type string.

## Remarks

Unlike [ToString\(\)](#), this conversion does not return the common name of well-known integer resource types. Therefore, the returned string is suitable for passing directly to Win32 resource APIs that accept resource type strings.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Font Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► Font

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static ResourceType Font { get; }
```

```
Public Shared ReadOnly Property Font As ResourceType
```

```
public:  
static property ResourceType^ Font {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **FontDir Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► FontDir

C# ▾

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public static ResourceType FontDir { get; }
```

```
Public Shared ReadOnly Property FontDir As ResourceType
```

```
public:  
static property ResourceType^ FontDir {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetHashCode Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► **GetHashCode()**

C#

Gets a hash code suitable for using the resource type as a dictionary key.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public override int GetHashCode()
```

```
Public Overrides Function GetHashCode As Integer
```

```
public:  
virtual int GetHashCode() override
```

[-] **Return Value**

Hash code based on the resource type string.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GroupCursor Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► GroupCursor

C# ▾

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public static ResourceType GroupCursor { get; }
```

```
Public Shared ReadOnly Property GroupCursor As Resou
```

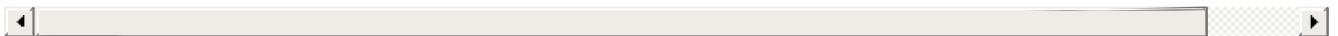
```
public:  
static property ResourceType^ GroupCursor {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **GroupIcon Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► GroupIcon

C# ▾

Declaration Syntax

C#

Visual Basic

Visual C++

```
public static ResourceType GroupIcon { get; }
```

```
Public Shared ReadOnly Property GroupIcon As Resourc
```

```
public:  
static property ResourceType^ GroupIcon {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **Html Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► Html

C# ▾

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public static ResourceType Html { get; }
```

```
Public Shared ReadOnly Property Html As ResourceType
```

```
public:  
static property ResourceType^ Html {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Icon Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► Icon

C# ▾

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public static ResourceType Icon { get; }
```

```
Public Shared ReadOnly Property Icon As ResourceType
```

```
public:  
static property ResourceType^ Icon {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools FoundationImplicit Operator

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► Implicit(String)

C# ▾

Implicitly converts a string to a ResourceType.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static implicit operator ResourceType ( _  
    string resourceType  
)
```

```
Public Shared Widening Operator CType ( _  
    resourceType As String _  
) As ResourceType
```

```
static implicit operator ResourceType^ ( _  
    String^ resourceType  
)
```

### Parameters

#### ***resourceType* (String)**

String resource type to convert.

### Return Value

ResourceType object.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## IntegerValue Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► IntegerValue

C# ▾

Gets the integer value of the resource type, or -1 if the resource type is not an integer.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public int IntegerValue { get; }
```

```
Public ReadOnly Property IntegerValue As Integer
```

```
public:  
property int IntegerValue {  
    int get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## IsInteger Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► IsInteger

C# ▾

Gets a flag indicating whether the resource type is an integer type.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public bool IsInteger { get; }
```

```
Public Readonly Property IsInteger As Boolean
```

```
public:  
property bool IsInteger {  
    bool get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# Manifest Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► Manifest

C# ▾

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public static ResourceType Manifest { get; }
```

```
Public Shared ReadOnly Property Manifest As Resource
```

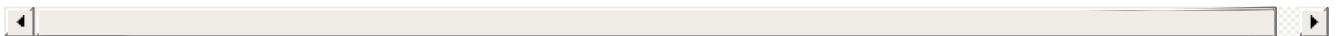
```
public:  
static property ResourceType^ Manifest {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## Menu Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► Menu

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static ResourceType Menu { get; }
```

```
Public Shared ReadOnly Property Menu As ResourceType
```

```
public:  
static property ResourceType^ Menu {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## MessageTable Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► MessageTable

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static ResourceType MessageTable { get; }
```

```
Public Shared ReadOnly Property MessageTable As Reso
```

```
public:  
static property ResourceType^ MessageTable {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## None Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► None

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static ResourceType None { get; }
```

```
Public Shared ReadOnly Property None As ResourceType
```

```
public:  
static property ResourceType^ None {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## PlugPlay Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► PlugPlay

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static ResourceType PlugPlay { get; }
```

```
Public Shared ReadOnly Property PlugPlay As Resource
```

```
public:  
static property ResourceType^ PlugPlay {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **RCData Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► RCData

C# ▾

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public static ResourceType RCData { get; }
```

```
Public Shared ReadOnly Property RCData As ResourceType
```

```
public:  
static property ResourceType^ RCData {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **String Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► String

C# ▾

[-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public static ResourceType String { get; }
```

```
Public Shared ReadOnly Property String As ResourceType
```

```
public:  
static property ResourceType^ String {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **ToString** Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► **ToString()**

C# ▾

Gets a string representation of the resource type.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public override string ToString()
```

```
Public Overrides Function ToString As String
```

```
public:  
virtual String^ ToString() override
```

### Return Value

The custom resource name, or the name of a well-known resource type

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Version Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► Version

C# ▾

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public static ResourceType Version { get; }
```

```
Public Shared ReadOnly Property Version As ResourceType
```

```
public:  
static property ResourceType^ Version {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Vxd Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► ResourceType ► Vxd

C# ▾

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public static ResourceType Vxd { get; }
```

```
Public Shared ReadOnly Property Vxd As ResourceType
```

```
public:  
static property ResourceType^ Vxd {  
    ResourceType^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **VersionBuildTypes** Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► **VersionBuildTypes**

C# ▾

Identifies build types of a versioned file.

## - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
[FlagsAttribute]
public enum VersionBuildTypes
```

```
<FlagsAttribute> _
Public Enumeration VersionBuildTypes
```

```
[FlagsAttribute]
public enum class VersionBuildTypes
```

## - Members

Member	Description
<b>None</b>	
<b>Debug</b>	
<b>Prerelease</b>	
<b>Patched</b>	
<b>PrivateBuild</b>	
<b>InfoInferred</b>	
<b>SpecialBuild</b>	

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **VersionFileSubtype** Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionFileSubtype

C# ▾

Identifies the sub-type of a versioned file.

## - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public enum VersionFileSubtype
```

```
Public Enumeration VersionFileSubtype
```

```
public enum class VersionFileSubtype
```

## - Members

Member	Description
Unknown	
PrinterDriver	
KeyboardDriver	
LanguageDriver	
DisplayDriver	
MouseDriver	
NetworkDriver	
SystemDriver	
InstallableDriver	
SoundDriver	
CommDriver	
InputMethodDriver	
VersionedPrinterDriver	

**RasterFont**

**VectorFont**

**TrueTypeFont**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **VersionFileType Enumeration**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Resources ► VersionFileType

C# ▾

Identifies the type of a versioned file.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public enum VersionFileType
```

```
Public Enumeration VersionFileType
```

```
public enum class VersionFileType
```

[-] **Members**

Member	Description
<b>Unknown</b>	
<b>Application</b>	
<b>Dll</b>	
<b>Driver</b>	
<b>Font</b>	
<b>VirtualDevice</b>	
<b>StaticLibrary</b>	

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation VersionResource Class

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionResource

C#

A subclass of Resource which provides specific methods for manipulating the resource data.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public sealed class VersionResource : Resource, ICol
    I Enumerable<VersionStringTable>, I Enumerable
```

```
Public NotInheritable Class VersionResource _
    Inherits Resource _
    Implements ICollection(Of VersionStringTable)
    I Enumerable
```

```
public ref class VersionResource sealed : public Res
    I Collection<VersionStringTable^>, I Enumerable
```

## Members

All Members	Constructors	Methods	Properties
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static	<input checked="" type="checkbox"/> Declared <input checked="" type="checkbox"/> Inherited
<b>Icon</b> <b>Member</b> <b>Description</b>			
	<a href="#">VersionResource(String, Int32)</a>	Creates a new VersionResource object without any data. The data can be later loaded from a file.	
	<a href="#">VersionResource(String, Int32, Byte[])</a>	Creates a new VersionResource object with data. The data can be	

		later saved to a file.
	<a href="#">Add(Int32)</a>	Adds a new version string table for a locale.
	<a href="#">BuildTypes</a>	Gets or sets a bitmask that specifies the build types of the file.
	<a href="#">Clear()</a>	Removes all string tables from the version resource.
	<a href="#">Contains(Int32)</a>	Checks if a version string table exists for a given locale.
	<a href="#">CopyTo(VersionStringTable[], Int32)</a>	Copies the version string tables to an array, starting at a particular array index.
	<a href="#">Count</a>	Gets the number string tables in the version resource.
	<a href="#">Data</a>	Gets or sets the raw data of the resource. The data is in the format of the VS_VERSIONINFO structure.  (Overrides <a href="#">Resource.Data</a> .)
	<a href="#">Equals(Object)</a>	Tests if type, name, and locale of this Resource object match another

		Resource object.  (Inherited from <a href="#">Resource</a> .)
	<a href="#">FileSubtype</a>	Gets or sets the specific type of the file.
	<a href="#">FileType</a>	Gets or sets the general type of the file.
	<a href="#">FileVersion</a>	Gets or sets the binary locale-independent file version of the version resource.
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetEnumerator()</a>	Gets an enumerator that can iterate over the version string tables in the collection.
	<a href="#">GetHashCode()</a>	Gets a hash code for this Resource object.  (Inherited from <a href="#">Resource</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)

	<a href="#">Item[Int32]</a>	Gets the string table for a specific locale, or null if there is no table for that locale.
	<a href="#">Load(String)</a>	Loads the resource data from a file. The file is searched for a resource with matching type, name, and locale.  (Inherited from <a href="#">Resource</a> .)
	<a href="#">Locale</a>	Gets or sets the locale of the resource.  (Inherited from <a href="#">Resource</a> .)
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Name</a>	Gets or sets the name of the resource. For a numeric resource identifier, the decimal number is prefixed with a "#".  (Inherited from <a href="#">Resource</a> .)
	<a href="#">ProductVersion</a>	Gets or sets the binary locale-independent product version of the version resource.
	<a href="#">Remove(Int32)</a>	Removes a version string table for a locale.
	<a href="#">ResourceType</a>	Gets or sets the type of the resource. This may be one of the

		ResourceType constants or a user defined type name.  (Inherited from <a href="#">Resource</a> .)
≡	<a href="#">Save(String)</a>	Saves the resource to a file. Any existing resource data with matching type, name, and locale is overwritten.  (Inherited from <a href="#">Resource</a> .)
📅	<a href="#">Timestamp</a>	Gets or sets the binary creation date and time.
≡	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)

## Remarks

The resource is of type [Version](#) (RT\_VERSION).

## Inheritance Hierarchy

```
Object
  ↳ Resource
    ↳ VersionResource
```

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **VersionResource Constructor**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionResource ►

**VersionResource()**

C# ▾

▀ **Members**

Icon	Member	Description
≡	<a href="#">VersionResource(String, Int32)</a>	Creates a new VersionResource object without any data. The data can be later loaded from a file.
≡	<a href="#">VersionResource(String, Int32, Byte[])</a>	Creates a new VersionResource object with data. The data can be later saved to a file.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation VersionResource Constructor (name, locale)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Resources ► VersionResource ►  
**VersionResource(String, Int32)**

C# ▾

Creates a new VersionResource object without any data. The data can be later loaded from a file.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public VersionResource(  
    string name,  
    int locale  
)
```

```
Public Sub New ( _  
    name As String, _  
    locale As Integer _  
)
```

```
public:  
VersionResource(  
    String^ name,  
    int locale  
)
```

## Parameters

### *name* (String)

Name of the resource. For a numeric resource identifier, prefix the decimal number with a "#".

### *locale* (Int32)

Locale of the resource

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **VersionResource Constructor  
(name, locale, data)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Resources ► VersionResource ►  
**VersionResource(String, Int32, Byte[])**

C# ▾

Creates a new VersionResource object with data. The data can be later saved to a file.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public VersionResource(  
    string name,  
    int locale,  
    byte[] data  
)
```

```
Public Sub New ( _  
    name As String, _  
    locale As Integer, _  
    data As Byte() _  
)
```

```
public:  
VersionResource(  
    String^ name,  
    int locale,  
    array<unsigned char>^ data  
)
```

[-] **Parameters**

**name (String)**

Name of the resource. For a numeric resource identifier, prefix the decimal number with a "#".

***locale*** ([Int32](#))

Locale of the resource

***data*** ([Byte \[\]](#))

Raw resource data

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Add Method (locale)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionResource ► **Add(Int32)**

C# ▾

Adds a new version string table for a locale.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public VersionStringTable Add(  
    int locale  
)
```

```
Public Function Add ( _  
    locale As Integer _  
) As VersionStringTable
```

```
public:  
VersionStringTable^ Add(  
    int locale  
)
```

### Parameters

#### *locale* (Int32)

Locale of the table

### Return Value

The new string table, or the existing table if the locale already existed.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## BuildTypes Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionResource ► BuildTypes

C# ▾

Gets or sets a bitmask that specifies the build types of the file.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public VersionBuildTypes BuildTypes { get; set; }
```

```
Public Property BuildTypes As VersionBuildTypes
```

```
public:  
property VersionBuildTypes BuildTypes {  
    VersionBuildTypes get ();  
    void set (VersionBuildTypes value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

C# ▾

Removes all string tables from the version resource.

## - Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Clear()
```

```
Public Sub Clear
```

```
public:  
virtual void Clear() sealed
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Contains Method (locale)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionResource ►

## Contains(Int32)

C# ▾

Checks if a version string table exists for a given locale.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool Contains(  
    int locale  
)
```

```
Public Function Contains ( _  
    locale As Integer _  
) As Boolean
```

```
public:  
bool Contains(  
    int locale  
)
```

### Parameters

#### *locale* (Int32)

Locale to search for

### Return Value

True if a string table was found for the locale; false otherwise.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CopyTo Method (array, arrayIndex)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.Resources ► VersionResource ►  
**CopyTo(VersionStringTable[], Int32)**

C# ▾ Copies the version string tables to an array, starting at a particular array index.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void CopyTo(  
    VersionStringTable[] array,  
    int arrayIndex  
)
```

```
Public Sub CopyTo ( _  
    array As VersionStringTable(), _  
    arrayIndex As Integer _  
)
```

```
public:  
virtual void CopyTo(  
    array<VersionStringTable^>^ array,  
    int arrayIndex  
) sealed
```

### Parameters

#### array ( [VersionStringTable \[\]](#) )

The one-dimensional Array that is the destination of the elements copied from the collection. The Array must have zero-based indexing.

#### arrayIndex ( [Int32](#) )

The zero-based index in array at which copying begins.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module: Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Count Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionResource ► Count

C# ▾

Gets the number string tables in the version resource.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public int Count { get; }
```

```
Public Readonly Property Count As Integer
```

```
public:  
virtual property int Count {  
    int get () sealed;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Data Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionResource ► Data

C# ▾

Gets or sets the raw data of the resource. The data is in the format of the VS\_VERSIONINFO structure.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override byte[] Data { get; set; }
```

```
Public Overrides Property Data As Byte()
```

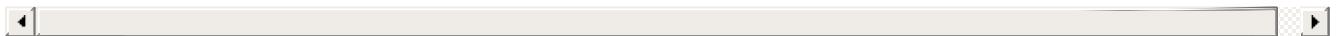
```
public:  
virtual property array<unsigned char>^ Data {  
    array<unsigned char>^ get () override;  
    void set (array<unsigned char>^ value) override;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation **FileSubtype** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionResource ► **FileSubtype**

C# ▾

Gets or sets the specific type of the file.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public VersionFileSubtype FileSubtype { get; set; }
```

```
Public Property FileSubtype As VersionFileSubtype
```

```
public:  
property VersionFileSubtype FileSubtype {  
    VersionFileSubtype get ();  
    void set (VersionFileSubtype value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **FileType** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionResource ► **FileType**

C# ▾

Gets or sets the general type of the file.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public VersionFileType FileType { get; set; }
```

```
Public Property FileType As VersionFileType
```

```
public:  
property VersionFileType FileType {  
    VersionFileType get ();  
    void set (VersionFileType value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# FileVersion Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionResource ► FileVersion

C# ▾

Gets or sets the binary locale-independent file version of the version resource.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public Version FileVersion { get; set; }
```

```
Public Property FileVersion As Version
```

```
public:  
property Version^ FileVersion {  
    Version^ get ();  
    void set (Version^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetEnumerator Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionResource ►

### GetEnumerator()

C# ▾

Gets an enumerator that can iterate over the version string tables in the collection.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public IEnum<VersionStringTable> GetEnumerator
```

```
Public Function GetEnumerator As IEnum<of Vers
```

```
public:  
virtual IEnum<VersionStringTable^>^ GetEnumerator
```

#### Return Value

An enumerator that returns [VersionStringTable](#) objects.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation Item Property (locale)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionResource ► Item[Int32]

C#

Gets the string table for a specific locale, or null if there is no table for that locale.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public VersionStringTable this[  
    int locale  
] { get; }
```

```
Public ReadOnly Default Property Item ( _  
    locale As Integer _  
) As VersionStringTable
```

```
public:  
property VersionStringTable^ default[int locale] {  
    VersionStringTable^ get (int locale);  
}
```

## Parameters

*locale* (Int32)

## See Also

Add(Int32)

Remove(Int32)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ProductVersion Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionResource ►

### ProductVersion

C#

Gets or sets the binary locale-independent product version of the version resource.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public Version ProductVersion { get; set; }
```

Public Property ProductVersion As Version

```
public:  
property Version^ ProductVersion {  
    Version^ get ();  
    void set (Version^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Remove Method (locale)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionResource ►

### Remove(Int32)

C# ▾

Removes a version string table for a locale.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Remove(  
    int locale  
)
```

```
Public Sub Remove ( _  
    locale As Integer _  
)
```

```
public:  
void Remove(  
    int locale  
)
```

#### Parameters

##### *locale* (Int32)

Locale of the table

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Timestamp Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionResource ► **Timestamp**

C# ▾

Gets or sets the binary creation date and time.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public DateTime Timestamp { get; set; }
```

```
Public Property Timestamp As DateTime
```

```
public:  
property DateTime Timestamp {  
    DateTime get ();  
    void set (DateTime value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation VersionStringTable Class

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionStringTable

C# ▾

Represents a string table of a file version resource.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public sealed class VersionStringTable : IDictionary<KeyValuePair<string, string>>, I1  
I2Enumerable
```

```
Public NotInheritable Class VersionStringTable _  
    Implements IDictionary(Of String, String),  
    ICollection(Of KeyValuePair(Of String, Strin  
I1Enumerable
```

```
public ref class VersionStringTable sealed : IDictio  
I2Collection<KeyValuePair<String^, String^>>,  
I3Enumerable
```

## Members

All Members	Methods	Properties	
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static	<input checked="" type="checkbox"/> Declared <input checked="" type="checkbox"/> Inherited

Icon	Member	Description
	<a href="#">Clear()</a>	Removes all strings from the string table.
	<a href="#">Count</a>	Gets the number of strings in the table.

  	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
  	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
 	<a href="#">GetEnumerator()</a>	Gets an enumeration over all strings in the table.
 	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.  (Inherited from <a href="#">Object</a> .)
 	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance  (Inherited from <a href="#">Object</a> .)
 	<a href="#">Item[String]</a>	Gets or sets a string value.
 	<a href="#">Keys</a>	Gets a collection of all the names of the strings in the table.
 	<a href="#">Locale</a>	Gets the locale (LCID) of the string table.

	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Values</a>	Gets a collection of all the values in the table.

## Inheritance Hierarchy

[Object](#)

 **VersionStringTable**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)



C# ▾

Removes all strings from the string table.

## - Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Clear()
```

```
Public Sub Clear
```

```
public:  
virtual void Clear() sealed
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Count Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionStringTable ► Count

C# ▾

Gets the number of strings in the table.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public int Count { get; }
```

```
Public ReadOnly Property Count As Integer
```

```
public:  
virtual property int Count {  
    int get () sealed;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetEnumerator Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionStringTable ►

**GetEnumerator()**

C# ▾

Gets an enumeration over all strings in the table.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public IEnumarator<KeyValuePair<string, string>> GetEnumerator();
```

```
Public Function GetEnumerator As IEnumarator(Of KeyValuePair<String, String>) End Function
```

```
public:  
virtual IEnumarator<KeyValuePair<String^, String^>>^ GetEnumerator();
```

[-] **Return Value**

Enumeration of string name and value pairs

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation Item Property (key)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionStringTable ► Item[String]

C# ▾

Gets or sets a string value.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string this[  
    string key  
] { get; set; }
```

```
Public Default Property Item ( _  
    key As String _  
) As String
```

```
public:  
virtual property String^ default[String^ key] {  
    String^ get (String^ key) sealed;  
    void set (String^ key, String^ value) sealed  
}
```

## Parameters

### key (String)

Name of the string.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Keys Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionStringTable ► Keys

C# ▾

Gets a collection of all the names of the strings in the table.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public ICollection<string> Keys { get; }
```

```
Public ReadOnly Property Keys As ICollection(Of Stri
```

```
public:  
virtual property ICollection<String^>^ Keys {  
    ICollection<String^>^ get () sealed;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Locale Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionStringTable ► **Locale**

C# ▾

Gets the locale (LCID) of the string table.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public int Locale { get; set; }
```

```
Public Property Locale As Integer
```

```
public:  
property int Locale {  
    int get ();  
    void set (int value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Values Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.Resources ► VersionStringTable ► Values

C# ▾

Gets a collection of all the values in the table.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public ICollection<string> Values { get; }
```

```
Public ReadOnly Property Values As ICollection(Of St
```

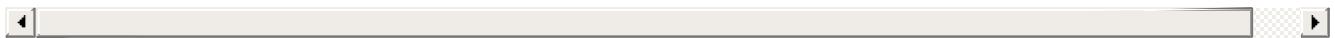
```
public:  
virtual property ICollection<String^>^ Values {  
    ICollection<String^>^ get () sealed;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.Resources (Module:

Microsoft.Deployment.Resources) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools

# Foundation Microsoft.Deployment.WindowsInstaller Namespace

Deployment Tools Foundation Namespaces ►

## Microsoft.Deployment.WindowsInstaller

C# ▾

Complete class library for the Windows Installer APIs.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
namespace Microsoft.Deployment.WindowsInstaller
```

```
Namespace Microsoft.Deployment.WindowsInstaller
```

```
namespace Microsoft.Deployment.WindowsInstaller
```

### - Types

All Types	Classes	Structures	Interfaces	Enumerations	Delegates
-----------	---------	------------	------------	--------------	-----------

Icon	Type	Description
	<a href="#">ActionResult</a>	Specifies a return status value for actions.
	<a href="#">BadQuerySyntaxException</a>	A bad SQL query string was passed to <a href="#">OpenView(String, Object[])</a> or <a href="#">Execute(String, Object[])</a> .
	<a href="#">ColumnCollection</a>	Collection of column information to a <a href="#">TableInfo</a> or <a href="#">View</a> .

	<b>ColumnInfo</b>	Defines a single column of a table in the installer database.
	<b>ComponentAttributes</b>	Available values for the Attribute column of the Component table.
	<b>ComponentInfo</b>	Provides access to information about a component within the context of an installation session.
	<b>ComponentInfoCollection</b>	Accessor for information about all components within the context of an installation session.
	<b>ComponentInstallation</b>	Represents an instance of a component.
	<b>ControlAttributes</b>	Defines flags for the Attribute column of the Control table.
	<b>CustomActionAttribute</b>	Marks a method as a custom point.
	<b>CustomActionData</b>	Contains a collection of key-value pairs suitable for passing between and deferred/rollback/commit actions.

	<b>CustomActionTypes</b>	Defines flags for the Type column in the CustomAction table.
	<b>Database</b>	Accesses a Windows Installer database.
	<b>DatabaseOpenMode</b>	Specifies the open mode for a database.
	<b>DialogAttributes</b>	Defines flags for the Attribute column in the Dialog table.
	<b>ExternalUIHandler</b>	Defines a callback function that the installer calls for progress notifications and error messages.
	<b>ExternalUIRecordHandler</b>	[MSI 3.1] Defines a callback function that the installer calls for record-based progress notification and error messages.
	<b>FeatureAttributes</b>	Available values for the Attribute column of the Feature table.
	<b>FeatureInfo</b>	Provides access to information about a feature within the context of a specific installation session.
	<b>FeatureInfoCollection</b>	Accessor for information about features within the context of an installation session.

		session.
	<b>FeatureInstallation</b>	Represents an instance of a feature installed on an installed product.
	<b>FileAttributes</b>	Available values for the Attribute column of the File table.
	<b>IEmbeddedUI</b>	[MSI 4.5] Interface for an embedded external user interface for an MSI package.
	<b>InapplicablePatchHandler</b>	Receives an exception from <a href="#">DetermineApplicablePatches(String[], InapplicablePatchHandler, String, UserContexts)</a> indicating the reason a particular patch is not applicable to a product.
	<b>IniFileAction</b>	Defines values for the Action column of the IniFile and RemoveIniFile tables.
	<b>Installation</b>	Subclasses of this abstract class represent a unique instance of a registered product or patch in a session.
	<b>InstallationPart</b>	Subclasses of this abstract class represent an instance of a registered feature or component.

	<b>InstallCanceledException</b>	User Canceled the installation.
	<b>InstallCost</b>	Represents a per-drive disk space cost for an installation.
	<b>Installer</b>	Provides static methods for installing products and patching products.
	<b>InstallerException</b>	Base class for Windows Installer exceptions.
	<b>InstallerHandle</b>	Base class for Windows Installer types (Database, View, Record, SummaryInfo).
	<b>InstallLogModes</b>	Log modes available for <a href="#">EnableLog(InstallLogModes)</a> , <a href="#">SetExternalUI(ExternalUIHandler, InstallLogModes)</a> .
	<b>InstallMessage</b>	Type of message to be processed by <a href="#">Message(InstallMessage, Record, ExternalUIHandler, or ExternalUIRecordHandler)</a> .
	<b>InstallMode</b>	Specifies the install mode for <a href="#">ProvideComponent(String, String, InstallMode)</a> or <a href="#">ProvideFile(String, String, InstallMode)</a> .

		ProvideQualifiedComponent( String, InstallMode, String).
	<b>InstallRunMode</b>	Specifies the run mode for <a href="#">GetMode(InstallRunMode)</a> .
	<b>InstallState</b>	Installed state of a Component Feature.
	<b>InstallType</b>	Specifies the type of installation <a href="#">ApplyPatch(String, String, Ins String)</a> .
	<b>InstallUIOptions</b>	Level of the installation user interface specified with <a href="#">SetInternalUI(InstallUIOptions)</a> .
	<b>InvalidHandleException</b>	A method was called on an invalid installer handle. The handle may have been already closed.
	<b>LocatorTypes</b>	Defines values for the Type column in the CompLocator, IniLocator, RegLocator tables.
	<b>MediaDisk</b>	Represents a media disk source product or a patch.

	<b>MergeException</b>	A failure occurred when executing <a href="#">Merge(Database, String)</a> . The exception may contain details about the conflict.
	<b>MessageResult</b>	Specifies a return status value for message handlers. These values are returned by <a href="#">Message(InstallMessageRecord)</a> , <a href="#">ExternalUIHandler</a> , <a href="#">InstallMessageProcessor</a> , <a href="#">ProcessMessage(InstallMessageRecord)</a> , <a href="#">MessageBoxButtons</a> , <a href="#">MessageBoxIcon</a> , <a href="#">MessageBoxDefaultButton</a> ).
	<b>PatchInstallation</b>	The Patch object represents a unique instance of a patch that has been registered or applied.
	<b>PatchStates</b>	Specifies the different patch states returned by <a href="#">GetPatches(String, String, String, UserContexts, PatchStates)</a> .
	<b>ProductInstallation</b>	Represents a unique instance of a product that is either advertised or unknown.
	<b>ComponentInstallation.Qualifier</b>	Holds data about a component's qualifiers.
	<b>Record</b>	The Record object is a container holding and transferring a variable number of values. Fields with

		record are numerically indexed and contain strings, integers, streams, and null values. Record fields are starting with 1. Field 0 is a special field.
	<b>RegistryRoot</b>	Defines values for the Root column of the Registry, RemoveRegistry, and RegLocator tables.
	<b>ReinstallModes</b>	Specifies the reinstall mode for ReinstallFeature(String, String, ReinstallModes) or ReinstallProduct(String, ReinstallModes).
	<b>RemoveFileModes</b>	Defines values for the InstallMode column of the RemoveFile table.
	<b>ServiceAttributes</b>	Defines values for the ServiceName, StartType, and ErrorControl columns of the ServiceInstall table.
	<b>ServiceControlEvents</b>	Defines values for the EventCode column of the ServiceControl table.
	<b>Session</b>	The Session object controls the installation process. It opens the database, which contains the tables and data.

	<b>ShortcutTarget</b>	Holds information about the target of a shortcut file.
	<b>SourceList</b>	A list of sources for an installer or patch.
	<b>SourceMediaList</b>	A list of source media for an installer product or patch.
	<b>SummaryInfo</b>	Provides access to summary information of a Windows Installer database.
	<b>TableCollection</b>	Contains information about all tables in a Windows Installer database.
	<b>TableInfo</b>	Defines a table in an installation database.
	<b>TextStyles</b>	Defines values for the StyleBitMask of the TextStyle table.
	<b>Transaction</b>	[MSI 4.5] Handle to a multi-session transaction.
	<b>TransactionAttributes</b>	Attributes for <a href="#">Transaction</a> methods.
		

	<b>TransformErrors</b>	Transform error conditions available in CreateTransformSummaryInfo(String, TransformErrors, TransformValidations) or ApplyTransform(String, TransformErrors).
⌚	<b>TransformValidations</b>	Transform validation flags available in CreateTransformSummaryInfo(String, TransformErrors, TransformValidations).
⌚	<b>UpgradeAttributes</b>	Defines values for the Attributes of the Upgrade table.
◊	<b>FeatureInstallation.UsageData</b>	Holds data about the usage of features.
⌚	<b>UserContexts</b>	Specifies the installation contexts for ProductInstallations, PatchInstallations, and DetermineApplicablePatchSets(String[], InapplicablePatchHandler, String, UserContexts)
⌚	<b>ValidationError</b>	Defines the type of error encountered by the Validate(Record), ValidateNew(Record), or ValidateFields(Record) methods of the View class.
◊	<b>ValidationErrorInfo</b>	Contains specific information about the error encountered by the View class.

		Validate(Record), ValidateNewRecord or ValidateFields(Record) methods in the View class.
	<b>View</b>	A View represents a result set returned when processing a query using the <code>OpenView(String, Object[])</code> method in the Database. Before any data can be transferred, the query must be executed using the <code>Execute(Record)</code> method, passing to it all replaceable parameters designated within the SQL query.
	<b>ViewModifyMode</b>	Specifies the modify mode for the <code>Modify(ViewModifyMode, Record)</code> method.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)



# Deployment Tools Foundation **ActionResult Enumeration**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► **ActionResult**

C# ▾

Specifies a return status value for custom actions.

## - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public enum ActionResult
```

```
Public Enumeration ActionResult
```

```
public enum class ActionResult
```

## - Members

Member	Description
<b>Success</b>	Action completed successfully.
<b>SkipRemainingActions</b>	Skip remaining actions, not an error.
<b>UserExit</b>	User terminated prematurely.
<b>Failure</b>	Unrecoverable error or unhandled exception occurred.
<b>NotExecuted</b>	Action not executed.

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)  
Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **BadQuerySyntaxException** Class

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ►

## BadQuerySyntaxException

C# ▾

A bad SQL query string was passed to `OpenView(String, Object[])` or `Execute(String, Object[])`.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
[SerializableAttribute]
public class BadQuerySyntaxException : InstallerExce
```

```
<SerializableAttribute> _
Public Class BadQuerySyntaxException _
    Inherits InstallerException
```

```
[SerializableAttribute]
public ref class BadQuerySyntaxException : public In
```

### Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon

Member

Description



`BadQuerySyntaxException(String, Exception)`

Creates a new `BadQuerySyntaxException` object with a specified error message and a reference to the inner exception that is the cause of this exception.

	 <a href="#">BadQuerySyntaxException(String)</a>	Creates a new <code>BadQuerySyntaxException</code> with a specified error message.
	 <a href="#">BadQuerySyntaxException()</a>	Creates a new <code>BadQuerySyntaxException</code> .
	 <a href="#">BadQuerySyntaxException(SerializationInfo, StreamingContext)</a>	Initializes a new instance of the <code>BadQuerySyntaxException</code> class with serialized state information and a streaming context.
	<a href="#">Data</a>	Gets a collection of key/value pairs that contain additional, user-defined information about the exception.
		(Inherited from <a href="#">Exception</a> )
	<a href="#">Equals(Object)</a>	Determines whether the specified <code>Object</code> is equal to the current <code>Object</code> .
		(Inherited from <a href="#">Object</a> )
	<a href="#">ErrorCode</a>	Gets the system error code that resulted in this exception, or 0 if no code is applicable.
		(Inherited from <a href="#">InstallerException</a> )

💡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.
💡	<a href="#">GetBaseException()</a>	When overridden in a derived class, returns the <a href="#">Exception</a> that is the cause of one or more subsequent exceptions.
💡	<a href="#">GetErrorRecord()</a>	Gets extended information about the error, or returns null if no further information is available.
💡	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is typically implemented for use in hashing algorithms and data structures like a hash table.
💡	<a href="#">GetObjectData(SerializationInfo, StreamingContext)</a>	Sets the <a href="#">SerializationInfo</a> object with information about the exception.
💡	<a href="#">GetType()</a>	Gets the runtime type of the current instance.

		the current instance.
		(Inherited from <a href="#">Exception</a> )
	<a href="#">HelpLink</a>	Gets or sets a link to the help file associated with this exception.
		(Inherited from <a href="#">Exception</a> )
	<a href="#">HResult</a>	Gets or sets HRES coded numerical value that is assigned to a specific exception.
		(Inherited from <a href="#">Exception</a> )
	<a href="#">InnerException</a>	Gets the <a href="#">Exception</a> instance that caused this current exception.
		(Inherited from <a href="#">Exception</a> )
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .
		(Inherited from <a href="#">Object</a> )
	<a href="#">Message</a>	Gets a message that describes the exception. This message may be a detailed formatted string or data if it was available.
		(Inherited from <a href="#">InstallerException</a> )
	<a href="#">Source</a>	Gets or sets the name of the application or thread that causes the error.
		(Inherited from <a href="#">Exception</a> )

	<b>StackTrace</b>	Gets a string representation of the frames on the call stack at the time the current exception was thrown.
		(Inherited from <a href="#">Exception</a> )
	<b>TargetSite</b>	Gets the method that throws the current exception.
		(Inherited from <a href="#">Exception</a> )
	<b>ToString()</b>	Creates and returns a string representation of the current exception.
		(Inherited from <a href="#">Exception</a> )

## Inheritance Hierarchy

```

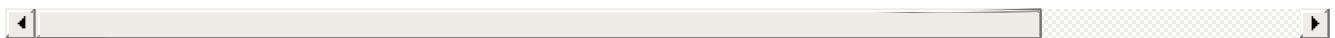
Object
└ Exception
  └ SystemException
    └ InstallerException
      └ BadQuerySyntaxException

```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

# BadQuerySyntaxException

## Constructor

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► BadQuerySyntaxException  
► **BadQuerySyntaxException()**

C# ▾

### ▀ Members

Icon	Member	Description
✖	<a href="#">BadQuerySyntaxException(String, Exception)</a>	Creates a new BadQuerySyntaxException with a specified error message and a reference to the inner exception that is the cause of this exception.
✖	<a href="#">BadQuerySyntaxException(String)</a>	Creates a new BadQuerySyntaxException with a specified error message.
✖	<a href="#">BadQuerySyntaxException()</a>	Creates a new BadQuerySyntaxException.
✖	<a href="#">BadQuerySyntaxException(SerializationInfo, StreamingContext)</a>	Initializes a new instance of the BadQuerySyntaxException class with serialized state information.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## BadQuerySyntaxException Constructor

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► BadQuerySyntaxException  
► **BadQuerySyntaxException()**

Creates a new BadQuerySyntaxException.

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public BadQuerySyntaxException()
```

```
Public Sub New
```

```
public:  
BadQuerySyntaxException()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## BadQuerySyntaxException Constructor (info, context)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► BadQuerySyntaxException  
► **BadQuerySyntaxException(SerializationInfo, StreamingContext)**

C# ▾

Initializes a new instance of the BadQuerySyntaxException class with serialized data.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected BadQuerySyntaxException(  
    SerializationInfo info,  
    StreamingContext context  
)
```

```
Protected Sub New ( _  
    info As SerializationInfo, _  
    context As StreamingContext _  
)
```

```
protected:  
BadQuerySyntaxException(  
    SerializationInfo^ info,  
    StreamingContext context  
)
```

### [-] Parameters

#### **info (SerializationInfo)**

The SerializationInfo that holds the serialized object data about the exception being thrown.

#### **context (StreamingContext)**

The StreamingContext that contains contextual information about the source or destination.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# BadQuerySyntaxException

## Constructor (msg)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► BadQuerySyntaxException  
► **BadQuerySyntaxException(String)**

C# Creates a new BadQuerySyntaxException with a specified error message.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public BadQuerySyntaxException(  
    string msg  
)
```

```
Public Sub New ( _  
    msg As String _  
)
```

```
public:  
BadQuerySyntaxException(  
    String^ msg  
)
```

### Parameters

#### **msg (String)**

The message that describes the error.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **BadQuerySyntaxException**

## Constructor (msg, innerException)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► BadQuerySyntaxException

► **BadQuerySyntaxException(String, Exception)**

C# ▾

Creates a new BadQuerySyntaxException with a specified error message and a reference to the inner exception that is the cause of this exception.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public BadQuerySyntaxException(  
    string msg,  
    Exception innerException  
)
```

```
Public Sub New ( _  
    msg As String, _  
    innerException As Exception _  
)
```

```
public:  
BadQuerySyntaxException(  
    String^ msg,  
    Exception^ innerException  
)
```

### Parameters

#### *msg* (**String**)

The message that describes the error.

#### *innerException* (**Exception**)

The exception that is the cause of the current exception. If the innerException parameter is not a null reference (Nothing in Visual Basic), the current exception is raised in a catch block that handles

the inner exception.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **ColumnCollection Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► **ColumnCollection**

C#

Collection of column information related to a [TableInfo](#) or [View](#).

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public sealed class ColumnCollection : ICollection<C  
    I Enumerable<ColumnInfo>, I Enumerable
```

```
Public NotInheritable Class ColumnCollection _  
    Implements ICollection(Of ColumnInfo), IEnum  
    I Enumerable
```

```
public ref class ColumnCollection sealed : ICollecti  
    I Enumerable<ColumnInfo^>, I Enumerable
```

## Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declare

Inherite

Icon	Member	Description
	<a href="#">ColumnCollection(Collection&lt;ColumnInfo&gt;)</a>	Creates a new ColumnCollection based on a specific list of columns.
	<a href="#">Add(ColumnInfo)</a>	Not supported because the collection is read-

		only.
≡	<a href="#">Clear()</a>	Not supported because the collection is read-only.
≡	<a href="#">Contains(String)</a>	Checks if a column with a given name exists in the collection.
≡	<a href="#">CopyToColumnInfo[], Int32)</a>	Copies the columns from this collection into an array.
⊕	<a href="#">Count</a>	Gets the number of columns in the collection.
≡	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations

		before the <a href="#">Object</a> is reclaimed by garbage collection.
		(Inherited from <a href="#">Object</a> .)
	<a href="#">FormatString</a>	Gets a string suitable for printing all the values of a record containing these columns.
	<a href="#">GetEnumerator()</a>	Gets an enumerator over the columns in the collection.
	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
	<a href="#">IndexOf(String)</a>	Gets the index of a column within the collection.

	<a href="#">IsReadOnly</a>	Gets a boolean value indicating whether the collection is read-only. A ColumnCollection is read-only if it is associated with a <a href="#">View</a> or a read-only <a href="#">Database</a> .
	<a href="#">Item[Int32]</a>	Gets information about a specific column in the collection.
	<a href="#">Item[String]</a>	Gets information about a specific column in the collection.
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from

Object.)

## [-] Inheritance Hierarchy

Object

└ ColumnCollection

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## ColumnCollection Constructor (columns)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► ColumnCollection ►  
**ColumnCollection(ICollection<ColumnInfo>)**

C# ▾

Creates a new ColumnCollection based on a specified list of columns.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public ColumnCollection(  
    ICollection<ColumnInfo> columns  
)
```

```
Public Sub New ( _  
    columns As ICollection(Of ColumnInfo) _  
)
```

```
public:  
ColumnCollection(  
    ICollection<ColumnInfo>^ columns  
)
```

### Parameters

**columns (ICollection<ColumnInfo>)**

columns to be added to the new collection

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Add Method (item)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnCollection ►

### Add(ColumnInfo)

C# ▾

Not supported because the collection is read-only.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Add(  
    ColumnInfo item  
)
```

```
Public Sub Add ( _  
    item As ColumnInfo _  
)
```

```
public:  
virtual void Add(  
    ColumnInfo^ item  
) sealed
```

### Parameters

#### *item* (ColumnInfo)

information about the column being added

### Exceptions

Exception	Condition
InvalidOperationException	the collection is read-only

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

C# ▾

Not supported because the collection is read-only.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Clear()
```

```
Public Sub Clear
```

```
public:  
virtual void Clear() sealed
```

## Exceptions

Exception	Condition
InvalidOperationException	the collection is read-only

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Contains Method (columnName)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnCollection ►

## Contains(String)

C# ▾

Checks if a column with a given name exists in the collection.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool Contains(  
    string columnName  
)
```

```
Public Function Contains ( _  
    columnName As String _  
) As Boolean
```

```
public:  
bool Contains(  
    String^ columnName  
)
```

### Parameters

#### columnName (String)

case-sensitive name of the column to look for

### Return Value

true if the column exists in the collection, false otherwise

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CopyTo Method (array, arrayIndex)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► ColumnCollection ►  
**CopyTo(ColumnInfo[], Int32)**

C# ▾

Copies the columns from this collection into an array.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void CopyTo(  
    ColumnInfo[] array,  
    int arrayIndex  
)
```

```
Public Sub CopyTo ( _  
    array As ColumnInfo(), _  
    arrayIndex As Integer _  
)
```

```
public:  
virtual void CopyTo(  
    array<ColumnInfo^>^ array,  
    int arrayIndex  
) sealed
```

### Parameters

#### array (ColumnInfo [])

destination array to be filled

#### arrayIndex (Int32)

offset into the destination array where copying begins

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Count Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnCollection ► Count

C# ▾

Gets the number of columns in the collection.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public int Count { get; }
```

```
Public ReadOnly Property Count As Integer
```

```
public:  
virtual property int Count {  
    int get () sealed;  
}
```

### - Value

number of columns in the collection

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## FormatString Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnCollection ►

### FormatString

C# ▾

Gets a string suitable for printing all the values of a record containing these columns.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string FormatString { get; }
```

```
Public ReadOnly Property FormatString As String
```

```
public:  
property String^ FormatString {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetEnumerator Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnCollection ►

**GetEnumerator()**

C# ▾

Gets an enumerator over the columns in the collection.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public IEnum<ColumnInfo> GetEnumerator()
```

```
Public Function GetEnumerator As IEnum<ColumnInfo>
```

```
public:  
virtual IEnum<ColumnInfo^>^ GetEnumerator() se
```

[-] **Return Value**

An enumerator of `ColumnInfo` objects.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## IndexOf Method (columnName)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnCollection ►

### IndexOf(String)

C# ▾

Gets the index of a column within the collection.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public int IndexOf(  
    string columnName  
)
```

```
Public Function IndexOf ( _  
    columnName As String _  
) As Integer
```

```
public:  
int IndexOf(  
    String^ columnName  
)
```

#### Parameters

##### columnName (String)

case-sensitive name of the column to look for

#### Return Value

0-based index of the column, or -1 if not found

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation IsReadOnly Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnCollection ►

## IsReadOnly

C# ▾

Gets a boolean value indicating whether the collection is read-only. A ColumnCollection is read-only if it is associated with a [View](#) or a read-only [Database](#).

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool IsReadOnly { get; }
```

```
Public ReadOnly Property IsReadOnly As Boolean
```

```
public:  
virtual property bool IsReadOnly {  
    bool get () sealed;  
}
```

### Value

read-only status of the collection

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Item Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnCollection ► Item

C# ▾

**Members**

Icon	Member	Description
	<a href="#">Item[Int32]</a>	Gets information about a specific column in the collection.
	<a href="#">Item[String]</a>	Gets information about a specific column in the collection.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Item Property (columnIndex)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnCollection ►

## Item[Int32]

C# ▾

Gets information about a specific column in the collection.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public ColumnInfo this[  
    int columnIndex  
] { get; }
```

```
Public ReadOnly Default Property Item ( _  
    columnIndex As Integer _  
) As ColumnInfo
```

```
public:  
property ColumnInfo^ default[int columnIndex] {  
    ColumnInfo^ get (int columnIndex);  
}
```

### Parameters

#### columnIndex (Int32)

1-based index into the column collection

### Exceptions

#### Exception

#### Condition

ArgumentOutOfRangeException	columnIndex is less than 1 or greater than the number of columns in the collection
-----------------------------	--

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Item Property (columnName)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnCollection ►

## Item[String]

C#

Gets information about a specific column in the collection.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public ColumnInfo this[  
    string columnName  
] { get; }
```

```
Public ReadOnly Default Property Item ( _  
    columnName As String _  
) As ColumnInfo
```

```
public:  
property ColumnInfo^ default[String^ columnName] {  
    ColumnInfo^ get (String^ columnName);  
}
```

### Parameters

#### columnName (String)

case-sensitive name of a column collection

### Exceptions

Exception	Condition
ArgumentOutOfRangeException	columnName does not exist in the collection

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **ColumnInfo Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► **ColumnInfo**

C#

Defines a single column of a table in an installer database.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public class ColumnInfo
```

```
Public Class ColumnInfo
```

```
public ref class ColumnInfo
```

## Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon	Member	Description
	<a href="#">ColumnInfo(String, String)</a>	Creates a new ColumnInfo object from a column definition.
	<a href="#">ColumnInfo(String, Type, Int32, Boolean)</a>	Creates a new ColumnInfo object from a list of parameters.
	<a href="#">ColumnInfo(String, Type, Int32, Boolean, Boolean, Boolean)</a>	Creates a new ColumnInfo object from a list of parameters.

	<code>ColumnDefinitionString</code>	Gets a short string defining the type and size of the column.
	<code>DBType</code>	Gets the type of the column as a <code>System.Data.DbType</code> . This is one of the following: <code>Int16</code> , <code>Int32</code> , <code>String</code> , or <code>Binary</code>
	<code>Equals(Object)</code>	Determines whether the specified <code>Object</code> is equal to the current <code>Object</code> .  (Inherited from <code>Object</code> .)
	<code>Finalize()</code>	Allows an <code>Object</code> to attempt to free resources and perform other cleanup operations before the <code>Object</code> is reclaimed by garbage collection.  (Inherited from <code>Object</code> .)
	<code>GetHashCode()</code>	Serves as a hash function for a particular type. <code>GetHashCode()</code> is suitable for use in hashing algorithms and data structures like a hash table.  (Inherited from <code>Object</code> .)
	<code>GetType()</code>	Gets the <code>Type</code> of the current instance  (Inherited from <code>Object</code> .)
	<code>IsLocalizable</code>	Gets a value indicating whether the column is a string column that is localizable.
	<code>IsRequired</code>	Gets a value indicating whether the column must be non-null when

		inserting a record.
	<a href="#">IsTemporary</a>	Gets a value indicating whether the column is temporary. Temporary columns are not persisted when the database is saved to disk.
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Name</a>	Gets the name of the column.
	<a href="#">Size</a>	Gets the size of the column.
	<a href="#">SqlCreateString</a>	Gets an SQL fragment that can be used to create this column within a CREATE TABLE statement.
	<a href="#">ToString()</a>	Gets the name of the column.  (Overrides <a href="#">Object.ToString()</a> .)
	<a href="#">Type</a>	Gets the type of the column as a System.Type. This is one of the following: Int16, Int32, String, or Stream

## Remarks

Once created, a ColumnInfo object is immutable.

## Inheritance Hierarchy

[Object](#)

└ [ColumnInfo](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ColumnInfo Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnInfo ►

**ColumnInfo()**

C# ▾

▀ Members

Icon	Member	Description
≡	<a href="#">ColumnInfo(String, String)</a>	Creates a new ColumnInfo object from a column definition.
≡	<a href="#">ColumnInfo(String, Type, Int32, Boolean)</a>	Creates a new ColumnInfo object from a list of parameters.
≡	<a href="#">ColumnInfo(String, Type, Int32, Boolean, Boolean, Boolean)</a>	Creates a new ColumnInfo object from a list of parameters.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ColumnInfo Constructor (name, columnDefinition)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnInfo ►

**ColumnInfo(String, String)**

C#

Creates a new ColumnInfo object from a column definition.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public ColumnInfo(  
    string name,  
    string columnDefinition  
)
```

```
Public Sub New ( _  
    name As String, _  
    columnDefinition As String _  
)
```

```
public:  
ColumnInfo(  
    String^ name,  
    String^ columnDefinition  
)
```

### Parameters

**name (String)**

name of the column

**columnDefinition (String)**

column definition string

### See Also

[ColumnDefinitionString](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **ColumnInfo Constructor (name, type, size,isRequired)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnInfo ►

**ColumnInfo(String, Type, Int32, Boolean)**

C#

Creates a new ColumnInfo object from a list of parameters.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public ColumnInfo(  
    string name,  
    Type type,  
    int size,  
    boolisRequired  
)
```

```
Public Sub New ( _  
    name As String, _  
    type As Type, _  
    size As Integer, _  
    isRequired As Boolean _  
)
```

```
public:  
ColumnInfo(  
    String^ name,  
    Type^ type,  
    int size,  
    boolisRequired  
)
```

[-] **Parameters**

**name (String)**

name of the column

***type* ([Type](#))**

type of the column; must be one of the following: Int16, Int32, String or Stream

***size* ([Int32](#))**

the maximum number of characters for String columns; ignored for other column types

***isRequired* ([Boolean](#))**

true if the column is required to have a non-null value

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ColumnInfo Constructor (name, type, size, isRequired, isTemporary, isLocalizable)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnInfo ►

ColumnInfo(String, Type, Int32, Boolean, Boolean, Boolean)

C#

Creates a new ColumnInfo object from a list of parameters.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public ColumnInfo(  
    string name,  
    Type type,  
    int size,  
    boolisRequired,  
    boolisTemporary,  
    boolisLocalizable  
)
```

```
Public Sub New ( _  
    name As String, _  
    type As Type, _  
    size As Integer, _  
    isRequired As Boolean, _  
    isTemporary As Boolean, _  
    isLocalizable As Boolean _  
)
```

```
public:  
ColumnInfo(  
    String^ name,  
    Type^ type,  
    int size,  
    bool.isRequired,
```

```
        bool isTemporary,  
        bool isLocalizable  
    )
```

## ■ Parameters

### ***name* ([String](#))**

name of the column

### ***type* ([Type](#))**

type of the column; must be one of the following: Int16, Int32, String or Stream

### ***size* ([Int32](#))**

the maximum number of characters for String columns; ignored for other column types

### ***isRequired* ([Boolean](#))**

true if the column is required to have a non-null value

### ***isTemporary* ([Boolean](#))**

true to if the column is only in-memory and not persisted with the database

### ***isLocalizable* ([Boolean](#))**

for String columns, indicates the column is localizable; ignored for other column types

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **ColumnDefinitionString** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnInfo ►

## ColumnDefinitionString

C# ▾

Gets a short string defining the type and size of the column.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string ColumnDefinitionString { get; }
```

```
Public ReadOnly Property ColumnDefinitionString As S
```

```
public:  
property String^ ColumnDefinitionString {  
    String^ get ();  
}
```

### Value

The definition string consists of a single letter representing the data type followed by the width of the column (in characters when applicable, bytes otherwise). A width of zero designates an unbounded width (for example long text fields and streams). An uppercase letter indicates that null values are allowed in the column.

### Remarks

s? - String, variable length (?=1-255) s0 - String, variable length i2 - Short integer i4 - Long integer v0 - Binary Stream g? - Temporary string (?=0-255) j? - Temporary integer (?=0,1,2,4) l? - Localizable string, variable length (?=1-255) l0 - Localizable string, variable length

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

[◀]

[▶]

## Deployment Tools Foundation **DBType Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnInfo ► DBType

C# ▾

Gets the type of the column as a System.Data.DbType. This is one of the following: Int16, Int32, String, or Binary

### [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public DbType DBType { get; }
```

```
Public ReadOnly Property DBType As DbType
```

```
public:  
property DbType DBType {  
    DbType get ();  
}
```

### [-] Value

equivalent DbType of the column

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation IsLocalizable Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnInfo ►

### IsLocalizable

C# ▾

Gets a value indicating whether the column is a string column that is localizable.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool IsLocalizable { get; }
```

```
Public ReadOnly Property IsLocalizable As Boolean
```

```
public:  
property bool IsLocalizable {  
    bool get ();  
}
```

#### Value

localizable status of the column

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation IsRequired Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnInfo ► IsRequired

C# ▾

Gets a value indicating whether the column must be non-null when inserting a record.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool IsRequired { get; }
```

```
Public ReadOnly Property IsRequired As Boolean
```

```
public:  
property bool IsRequired {  
    bool get ();  
}
```

### Value

required status of the column

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation IsTemporary Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnInfo ► IsTemporary

C# ▾

Gets a value indicating whether the column is temporary. Temporary columns are not persisted when the database is saved to disk.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool IsTemporary { get; }
```

```
Public ReadOnly Property IsTemporary As Boolean
```

```
public:  
property bool IsTemporary {  
    bool get ();  
}
```

### Value

temporary status of the column

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Name Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnInfo ► Name

C# ▾

Gets the name of the column.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public string Name { get; }
```

```
Public Readonly Property Name As String
```

```
public:  
property String^ Name {  
    String^ get ();  
}
```

### - Value

name of the column

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Size Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnInfo ► **Size**

C# ▾

Gets the size of the column.

## [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public int Size { get; }
```

```
Public Readonly Property Size As Integer
```

```
public:  
property int Size {  
    int get ();  
}
```

## [-] Value

The size of integer columns this is either 2 or 4. For string columns this the maximum recommended length of the string, or 0 for unlimited length. For stream columns, 0 is returned.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **SqlCreateString** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnInfo ►

## SqlCreateString

C# ▾

Gets an SQL fragment that can be used to create this column within a CREATE TABLE statement.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string SqlCreateString { get; }
```

```
Public ReadOnly Property SqlCreateString As String
```

```
public:  
property String^ SqlCreateString {  
    String^ get ();  
}
```

### Value

SQL fragment to be used for creating the column

### Remarks

Examples:

- LONG
- SHORT TEMPORARY
- CHAR(0) LOCALIZABLE
- CHAR(72) NOT NULL LOCALIZABLE
- OBJECT

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **ToString** Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnInfo ► **ToString()**

C# ▾

Gets the name of the column.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public override string ToString()
```

```
Public Overrides Function ToString As String
```

```
public:  
virtual String^ ToString() override
```

### Return Value

Name of the column.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Type Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ColumnInfo ► Type

C# ▾

Gets the type of the column as a System.Type. This is one of the following: Int16, Int32, String, or Stream

## [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public Type Type { get; }
```

```
Public ReadOnly Property Type As Type
```

```
public:  
property Type^ Type {  
    Type^ get ();  
}
```

## [-] Value

type of the column

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **ComponentAttributes**

## Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ComponentAttributes

C# ▾

Available values for the Attributes column of the Component table.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
[FlagsAttribute]
public enum ComponentAttributes
```

```
<FlagsAttribute> _
Public Enumeration ComponentAttributes
```

```
[FlagsAttribute]
public enum class ComponentAttributes
```

### Members

Member	Description
<b>None</b>	Local only - Component cannot be run from source.
<b>SourceOnly</b>	Component can only be run from source.
<b>Optional</b>	Component can run locally or from source.
<b>RegistryKeyPath</b>	If this bit is set, the value in the KeyPath column is used as a key into the Registry table.

<b>SharedDllRefCount</b>	If this bit is set, the installer increments the reference count in the shared DLL registry of the component's key file. If this bit is not set, the installer increments the reference count only if the reference count already exists.
<b>Permanent</b>	If this bit is set, the installer does not remove the component during an uninstall. The installer registers an extra system client for the component in the Windows Installer registry settings.
<b>OdbcDataSource</b>	If this bit is set, the value in the KeyPath column is a key into the ODBCDataSource table.
<b>Transitive</b>	If this bit is set, the installer reevaluates the value of the statement in the Condition column upon a reinstall. If the value was previously False and has changed to true the installer installs the component. If the value was previously true and has changed to false, the installer removes the component even if the component has other products as clients.
<b>NeverOverwrite</b>	If this bit is set, the installer does not install or reinstall the component if a key path file or a key path registry entry for the component already exists. The application

	does register itself as a client of the component.
<b>SixtyFourBit</b>	Set this bit to mark this as a 64-bit component. This attribute facilitates the installation of packages that include both 32-bit and 64-bit components. If this bit is not set, the component is registered as a 32-bit component.
<b>DisableRegistryReflection</b>	Set this bit to disable registry reflection or all existing and new registry keys affected by this component.
<b>UninstallOnSupersedence</b>	[MSI 4.5] Set this bit for a component in a patch package to prevent leaving orphan components on the computer.
<b>Shared</b>	[MSI 4.5] If a component is marked with this attribute value in at least one package installed on the system, the installer treats the component as marked in all packages. If a package that shares the marked component is uninstalled, Windows Installer 4.5 can continue to share the highest version of the component on the system, even if that highest version was installed by the package that is being uninstalled.

Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **ComponentInfo** Class

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ComponentInfo

C# ▾

Provides access to information about a component within the context of an installation session.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public class ComponentInfo
```

```
Public Class ComponentInfo
```

```
public ref class ComponentInfo
```

## Members

All Members	Methods	Properties	Events
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static	<input checked="" type="checkbox"/> Declared <input checked="" type="checkbox"/> Inherited

Icon	Member	Description
	<a href="#">CurrentState</a>	Gets the current install state of the designated Component.
	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> . (Inherited from <a href="#">Object</a> .)
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.

		(Inherited from <a href="#">Object</a> .)
	<a href="#">GetCost(InstallState)</a>	Gets disk space per drive required to install a component.
	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.
		(Inherited from <a href="#">Object</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance
		(Inherited from <a href="#">Object</a> .)
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .
		(Inherited from <a href="#">Object</a> .)
	<a href="#">Name</a>	Gets the name of the component (primary key in the Component table)
	<a href="#">RequestState</a>	Gets or sets the action state of the designated Component.
	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .
		(Inherited from <a href="#">Object</a> .)

## Inheritance Hierarchy

[Object](#)

[ComponentInfo](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **CurrentState** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ComponentInfo ►

## CurrentState

C# ▾

Gets the current install state of the designated Component.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public InstallState CurrentState { get; }
```

```
Public ReadOnly Property CurrentState As InstallStat
```

```
public:  
property InstallState CurrentState {  
    InstallState get();  
}
```

### Remarks

Win32 MSI API: [MsiGetComponentState](#)

### Exceptions

Exception	Condition
InvalidOperationException	the Session handle is invalid
ArgumentException	an unknown Component was requested

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetCost Method (installState)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ComponentInfo ►

### **GetCost(InstallState)**

C# ▾

Gets disk space per drive required to install a component.

#### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public IList<InstallCost> GetCost(
    InstallState installState
)
```

```
Public Function GetCost ( _
    installState As InstallState _ 
) As IList(Of InstallCost)
```

```
public:
IList<InstallCost>^ GetCost(
    InstallState installState
)
```

#### [-] Parameters

##### ***installState* (InstallState)**

Requested component state

#### [-] Return Value

A list of InstallCost structures, specifying the cost for each drive for the component

#### [-] Remarks

Win32 MSI API: [MsiEnumComponentCosts](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Name Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ComponentInfo ► Name

C#

Gets the name of the component (primary key in the Component table).

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public string Name { get; }
```

```
Public Readonly Property Name As String
```

```
public:  
property String^ Name {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation RequestState Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ComponentInfo ►

## RequestState

C# ▾

Gets or sets the action state of the designated Component.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public InstallState RequestState { get; set; }
```

```
Public Property RequestState As InstallState
```

```
public:  
property InstallState RequestState {  
    InstallState get ();  
    void set (InstallState value);  
}
```

### - Remarks

Win32 MSI APIs: [MsiGetComponentState](#), [MsiSetComponentState](#)

### - Exceptions

Exception	Condition
InvalidOperationException	the Session handle is invalid
ArgumentException	an unknown Component was requested
InstallCanceledException	the user exited the installation

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **ComponentInfoCollection Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ComponentInfoCollection

C#

Accessor for information about components within the context of an installation session.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public sealed class ComponentInfoCollection : IColle  
    IEnumerable<ComponentInfo>, IEnumerable
```

```
Public NotInheritable Class ComponentInfoCollection  
    Implements ICollection(Of ComponentInfo), IE  
    IEnumerable
```

```
public ref class ComponentInfoCollection sealed : IC  
    IEnumerble<ComponentInfo^>, IEnumerable
```

## Members

All Members

Methods

Properties

Public

Protected

Instance

Static



Declare



Inherit

Icon	Member	Description
	<a href="#">Contains(String)</a>	Checks if the collection contains a component.
	<a href="#">CopyTo(ComponentInfo[], Int32)</a>	Copies the features into an array.
	<a href="#">Count</a>	Gets the number of components

		defined for the product.
	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> (Inherited from <a href="#">Object</a> .)
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a> .)
	<a href="#">GetEnumerator()</a>	Enumerates the components in the collection.
	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithm and data structures like a hash table (Inherited from <a href="#">Object</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance. (Inherited from <a href="#">Object</a> .)
	<a href="#">Item[String]</a>	Gets information about a component within the context of an installation session.
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> . (Inherited from <a href="#">Object</a> .)

	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
---	----------------------------	--

## Inheritance Hierarchy

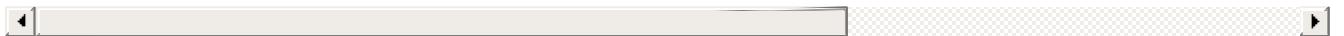
[Object](#)

 **ComponentInfoCollection**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation Contains Method (component)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ComponentInfoCollection ►

## Contains(String)

C# ▾

Checks if the collection contains a component.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public bool Contains(  
    string component  
)
```

```
Public Function Contains ( _  
    component As String _  
) As Boolean
```

```
public:  
bool Contains(  
    String^ component  
)
```

### Parameters

#### component (String)

name of the component

### Return Value

true if the component is in the collection, else false

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CopyTo Method (array, arrayIndex)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► ComponentInfoCollection ►  
**CopyTo(ComponentInfo[], Int32)**

C# ▾

Copies the features into an array.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void CopyTo(  
    ComponentInfo[] array,  
    int arrayIndex  
)
```

```
Public Sub CopyTo ( _  
    array As ComponentInfo(), _  
    arrayIndex As Integer _  
)
```

```
public:  
virtual void CopyTo(  
    array<ComponentInfo^>^ array,  
    int arrayIndex  
) sealed
```

### Parameters

#### array ( ComponentInfo [] )

array that receives the features

#### arrayIndex (Int32)

offset into the array

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Count Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ComponentInfoCollection ►

**Count**

C# ▾

Gets the number of components defined for the product.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public int Count { get; }
```

```
Public ReadOnly Property Count As Integer
```

```
public:  
virtual property int Count {  
    int get () sealed;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetEnumerator Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ComponentInfoCollection ►

**GetEnumerator()**

C# ▾

Enumerates the components in the collection.

**Declaration Syntax**

C#

Visual Basic

Visual C++

```
public IEnumrator<ComponentInfo> GetEnumerator()
```

```
Public Function GetEnumerator As IEnumrator(Of Comp
```

```
public:  
virtual IEnumrator<ComponentInfo^>^ GetEnumerator()
```

**Return Value**

an enumerator over all features in the collection

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation **Item Property (component)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ComponentInfoCollection ►

**Item[String]**

C# ▾

Gets information about a component within the context of an installation session.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public ComponentInfo this[  
    string component  
] { get; }
```

```
Public ReadOnly Default Property Item ( _  
    component As String _  
) As ComponentInfo
```

```
public:  
property ComponentInfo^ default[String^ component] {  
    ComponentInfo^ get (String^ component);  
}
```

## Parameters

**component (String)**

name of the component

## Return Value

component object

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **ComponentInstallation Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ComponentInstallation

C# ▾

Represents an instance of a registered component.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public class ComponentInstallation : InstallationPart
```

```
Public Class ComponentInstallation _  
    Inherits InstallationPart
```

```
public ref class ComponentInstallation : public Inst
```

## Members

All Members	Constructors	Methods	Properties
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static	<input checked="" type="checkbox"/> Declared <input checked="" type="checkbox"/> Inherited

Icon	Member	Description
≡	<a href="#">ComponentInstallation(String)</a>	Creates a new ComponentInstallation, automatically detecting the product that the component is a part of.
≡	<a href="#">ComponentInstallation(String, String)</a>	Creates a new ComponentInstallation for a component installed by a specific product.

 <a href="#">AllComponents</a>	Gets the set of installed components for all products.
 <a href="#">ClientProducts</a>	Gets all client products of a specified component.
 <a href="#">ComponentCode</a>	Gets the component code (GUID) of the component.
 <a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
 <a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
 <a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.  (Inherited from <a href="#">Object</a> .)
 <a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
 <a href="#">MemberwiseClone()</a>	Creates a shallow copy of the

		current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Path</a>	Gets the full path to an installed component. If the key path for the component is a registry key then the registry key is returned.
	<a href="#">Product</a>	Gets the product that this item is a part of.  (Inherited from <a href="#">InstallationPart</a> .)
	<a href="#">Qualifiers</a>	Gets the set of registered qualifiers for the component.
	<a href="#">State</a>	Gets the installed state of a component.  (Overrides <a href="#">InstallationPart.State</a> .)
	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)

## Inheritance Hierarchy




---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)  
Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

# ComponentInstallation

## Constructor

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► ComponentInstallation ►  
**ComponentInstallation()**

C# ▾

### Members

Icon	Member	Description
	<a href="#">ComponentInstallation(String)</a>	Creates a new ComponentInstallation, automatically detecting the product that the component is a part of.
	<a href="#">ComponentInstallation(String, String)</a>	Creates a new ComponentInstallation for a component installed by a specific product.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# ComponentInstallation

## Constructor (componentCode)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ComponentInstallation ►

### ComponentInstallation(String)

C#

Creates a new ComponentInstallation, automatically detecting the product that the component is a part of.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public ComponentInstallation(  
    string componentCode  
)
```

```
Public Sub New ( _  
    componentCode As String _  
)
```

```
public:  
ComponentInstallation(  
    String^ componentCode  
)
```

#### Parameters

**componentCode (String)**  
component GUID

#### Remarks

Win32 MSI API: [MsiGetProductCode](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# ComponentInstallation

## Constructor (componentCode, productCode)

Deployment Tools Foundation Namespaces ►  
 Microsoft.Deployment.WindowsInstaller ► ComponentInstallation ►  
**ComponentInstallation(String, String)**

C# ▾

Creates a new ComponentInstallation for a component installed by a specific product.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public ComponentInstallation(
    string componentCode,
    string productCode
)
```

```
Public Sub New ( _
    componentCode As String, _
    productCode As String _
)
```

```
public:
ComponentInstallation(
    String^ componentCode,
    String^ productCode
)
```

### Parameters

#### **componentCode (String)**

component GUID

#### **productCode (String)**

ProductCode GUID

Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **AllComponents** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ComponentInstallation ►

## AllComponents

C# ▾

Gets the set of installed components for all products.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static IEnumerable<ComponentInstallation> All
```

```
Public Shared ReadOnly Property AllComponents As IEnumerable<
```

```
public:  
static property IEnumerable<ComponentInstallation>^  
    IEnumerable<ComponentInstallation>^ get ();  
}
```

### Remarks

Win32 MSI API: [MsiEnumComponents](#)

### Exceptions

Exception	Condition
InstallerException	The installer configuration data is corrupt

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation ClientProducts Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ComponentInstallation ►

## ClientProducts

C# ▾

Gets all client products of a specified component.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public IEnumerable<ProductInstallation> ClientProduc
```

```
Public ReadOnly Property ClientProducts As IEnumerable<
```

```
public:  
property IEnumerable<ProductInstallation^>^ ClientPr  
    IEnumerable<ProductInstallation^>^ get ();  
}
```

### Return Value

enumeration over all client products of the component

### Remarks

Because clients are not ordered, any new component has an arbitrary index. This means that the property may return clients in any order.

Win32 MSI API: [MsiEnumClients](#)

### Exceptions

Exception	Condition
InstallerException	The installer configuration data is corrupt

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## ComponentCode Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ComponentInstallation ►

### ComponentCode

C# ▾

Gets the component code (GUID) of the component.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string ComponentCode { get; }
```

```
Public ReadOnly Property ComponentCode As String
```

```
public:  
property String^ ComponentCode {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Path Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ComponentInstallation ►

## Path

C# ▾

Gets the full path to an installed component. If the key path for the component is a registry key then the registry key is returned.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string Path { get; }
```

```
Public ReadOnly Property Path As String
```

```
public:  
property String^ Path {  
    String^ get ();  
}
```

### Return Value

The file or registry keypath to the component, or null if the component is not available.

### Remarks

If the component is a registry key, the registry roots are represented numerically. For example, a registry path of "HKEY\_CURRENT\_USER\SOFTWARE\Microsoft" would be returned as "01:\SOFTWARE\Microsoft". The registry roots returned are defined as follows: HKEY\_CLASSES\_ROOT=00, HKEY\_CURRENT\_USER=01, HKEY\_LOCAL\_MACHINE=02, HKEY\_USERS=03, HKEY\_PERFORMANCE\_DATA=04

Win32 MSI APIs: [MsiGetComponentPath](#), [MsiLocateComponent](#)

### Exceptions

Exception	Condition
ArgumentException	An unknown product or component was specified
InstallerException	The installer configuration data is corrupt

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Qualifiers Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ComponentInstallation ►

## Qualifiers

C# ▾

Gets the set of registered qualifiers for the component.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public IEnumerable<ComponentInstallation.Qualifier>
```

```
Public ReadOnly Property Qualifiers As IEnumerable(0
```

```
public:  
property IEnumerable<ComponentInstallation.Qualifier  
    IEnumerable<ComponentInstallation.Qualifier>  
}
```

### Return Value

Enumeration of the qualifiers for the component.

### Remarks

Because qualifiers are not ordered, any new qualifier has an arbitrary index, meaning the function can return qualifiers in any order.

Win32 MSI API: [MsiEnumComponentQualifiers](#)

### Exceptions

Exception	Condition
InstallerException	The installer configuration data is corrupt

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation State Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ComponentInstallation ►

## State

C# ▾

Gets the installed state of a component.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override InstallState State { get; }
```

```
Public Overrides ReadOnly Property State As Installs
```

```
public:  
virtual property InstallState State {  
    InstallState get () override;  
}
```

### Return Value

the installed state of the component, or `InstallState.Unknown` if this component is not part of a product

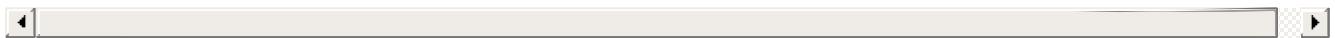
### Remarks

Win32 MSI API: [MsiGetComponentPath](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **ComponentInstallation.Qualifier**  
**Structure**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► ComponentInstallation ► **ComponentInstallation.Qualifier**

C# ▾

Holds data about a component qualifier.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public struct Qualifier
```

```
Public Structure Qualifier
```

```
public value class Qualifier
```

[-] **Members**

All Members	Methods	Properties	Events
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static	<input checked="" type="checkbox"/> Declared <input checked="" type="checkbox"/> Inherited

Icon	Member	Description
	<a href="#">Data</a>	Gets the qualifier data.
	<a href="#">Equals(Object)</a>	Indicates whether this instance and a specified object are equal. (Inherited from <a href="#">ValueType</a> .)
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is

		reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
 	<a href="#">GetHashCode()</a>	Returns the hash code for this instance.  (Inherited from <a href="#">ValueType</a> .)
 	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance  (Inherited from <a href="#">Object</a> .)
 	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
 	<a href="#">QualifierCode</a>	Gets the qualifier code.
 	<a href="#">ToString()</a>	Returns the fully qualified type name of this instance.  (Inherited from <a href="#">ValueType</a> .)

## Remarks

Win32 MSI API: [MsiEnumComponentQualifiers](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)  
Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# Data Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ►

ComponentInstallation.Qualifier ► Data

C# ▾

Gets the qualifier data.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public string Data { get; }
```

```
Public ReadOnly Property Data As String
```

```
public:  
property String^ Data {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **QualifierCode Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ►

ComponentInstallation.Qualifier ► QualifierCode

C# ▾

Gets the qualifier code.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public string QualifierCode { get; }
```

```
Public ReadOnly Property QualifierCode As String
```

```
public:  
property String^ QualifierCode {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation ControlAttributes Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ControlAttributes

C#

Defines flags for the Attributes column of the Control table.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
[FlagsAttribute]
public enum ControlAttributes
```

```
<FlagsAttribute> _
Public Enumeration ControlAttributes
```

```
[FlagsAttribute]
public enum class ControlAttributes
```

## Members

Member	Description
<b>Visible</b>	If this bit is set, the control is visible on the dialog box.
<b>Enabled</b>	specifies if the given control is enabled or disabled. Most controls appear gray when disabled.
<b>Sunken</b>	If this bit is set, the control is displayed with a sunken, three dimensional look.
<b>Indirect</b>	The Indirect control attribute specifies whether the value displayed or changed

	by this control is referenced indirectly.
<b>Integer</b>	If this bit is set on a control, the associate property specified in the Property column of the Control table is an integer.
<b>RightToLeftReadingOrder</b>	If this bit is set the text in the control is displayed in a right-to-left reading order.
<b>RightAligned</b>	If this style bit is set, text in the control is aligned to the right.
<b>LeftScroll</b>	If this bit is set, the scroll bar is located on the left side of the control, otherwise it is on the right.
<b>Bidirectional</b>	This is a combination of the RightToLeftReadingOrder, RightAligned, and LeftScroll attributes.
<b>Transparent</b>	If this bit is set on a text control, the control is displayed transparently with the background showing through the control where there are no characters.
<b>NoPrefix</b>	If this bit is set on a text control, the occurrence of the character "&" in a text string is displayed as itself.

<b>NoWrap</b>	If this bit is set the text in the control is displayed on a single line.
<b>FormatSize</b>	If this bit is set for a text control, the control will automatically attempt to format the displayed text as a number representing a count of bytes.
<b>UsersLanguage</b>	If this bit is set, fonts are created using the user's default UI code page. Otherwise it is created using the database code page.
<b>Multiline</b>	If this bit is set on an Edit control, the installer creates a multiple line edit control with a vertical scroll bar.
<b>PasswordInput</b>	This attribute creates an edit control for entering passwords. The control displays each character as an asterisk (*) as they are typed into the control.
<b>Progress95</b>	If this bit is set on a ProgressBar control, the bar is drawn as a series of small rectangles in Microsoft Windows 95-style. Otherwise it is drawn as a single continuous rectangle.
<b>RemovableVolume</b>	If this bit is set, the control shows removable volumes.

<b>FixedVolume</b>	If this bit is set, the control shows fixed internal hard drives.
<b>RemoteVolume</b>	If this bit is set, the control shows remote volumes.
<b>CdromVolume</b>	If this bit is set, the control shows CD-ROM volumes.
<b>RamDiskVolume</b>	If this bit is set, the control shows RAM disk volumes.
<b>FloppyVolume</b>	If this bit is set, the control shows floppy volumes.
<b>ShowRollbackCost</b>	Specifies whether or not the rollback backup files are included in the costs displayed by the VolumeCostList control.
<b>Sorted</b>	If this bit is set, the items listed in the control are displayed in a specified order. Otherwise, items are displayed in alphabetical order.
<b>ComboList</b>	If this bit is set on a combo box, the edit field is replaced by a static text field. This prevents a user from entering a new value.

	and requires the user to choose only one of the predefined values.
<b>PushLike</b>	If this bit is set on a check box or a radio button group, the button is drawn with the appearance of a push button, but its logic stays the same.
<b>Bitmap</b>	If this bit is set, the text in the control is replaced by a bitmap image. The Text column in the Control table is a foreign key into the Binary table.
<b>Icon</b>	If this bit is set, text is replaced by an icon image and the Text column in the Control table is a foreign key into the Binary table
<b>FixedSize</b>	If this bit is set, the picture is cropped or centered in the control without changing its shape or size.
<b>IconSize16</b>	Specifies which size of the icon image to load. If none of the bits are set, the first image is loaded.
<b>IconSize32</b>	Specifies which size of the icon image to load. If none of the bits are set, the first image is loaded.

<b>IconSize48</b>	Specifies which size of the icon image to load. If none of the bits are set, the first image is loaded.
<b>ElevationShield</b>	If this bit is set, and the installation is not yet running with elevated privileges, the control is created with a UAC icon.
<b>HasBorder</b>	If this bit is set, the RadioButtonGroup has text and a border displayed around it.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **CustomActionAttribute** Class

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► **CustomActionAttribute**

C# ▾

Marks a method as a custom action entry point.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
[SerializableAttribute]  
public sealed class CustomActionAttribute : Attribut
```

```
<SerializableAttribute> _  
Public NotInheritable Class CustomActionAttribute _  
    Inherits Attribute
```

```
[SerializableAttribute]  
public ref class CustomActionAttribute sealed : publ
```

## Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon	Member	Description
=	<a href="#">CustomActionAttribute()</a>	Marks a method as a custom action entry point.
=	<a href="#">CustomActionAttribute(String)</a>	Marks a method as a custom action entry point.
=	<a href="#">Equals(Object)</a>	Returns a value that indicates

		whether this instance is equal to a specified object.  (Inherited from <a href="#">Attribute</a> .)
💡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">GetHashCode()</a>	Returns the hash code for this instance.  (Inherited from <a href="#">Attribute</a> .)
💡	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">IsDefaultAttribute()</a>	When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class.  (Inherited from <a href="#">Attribute</a> .)
💡	<a href="#">Match(<a href="#">Object</a>)</a>	When overridden in a derived class, returns a value that indicates whether this instance equals a specified object.  (Inherited from <a href="#">Attribute</a> .)
💡	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
📝	<a href="#">Name</a>	Gets or sets the name of the

		custom action entrypoint. A null value defaults to the name of the method.
≡	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
⊕	<a href="#">TypeId</a>	When implemented in a derived class, gets a unique identifier for this <a href="#">Attribute</a> .  (Inherited from <a href="#">Attribute</a> .)

## Remarks

A custom action method must be defined as public and static, take a single [Session](#) object as a parameter, and return an [ActionResult](#) enumeration value.

## Inheritance Hierarchy

[Object](#)

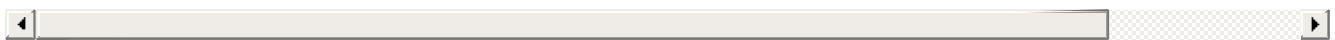
└ [Attribute](#)

  └ **CustomActionAttribute**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

# CustomActionAttribute Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionAttribute ►

CustomActionAttribute()

C# ▾

## Members

Icon	Member	Description
	<a href="#">CustomActionAttribute()</a>	Marks a method as a custom action entry point.
	<a href="#">CustomActionAttribute(String)</a>	Marks a method as a custom action entry point.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# CustomActionAttribute Constructor

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► CustomActionAttribute ►  
**CustomActionAttribute()**

Marks a method as a custom action entry point.

C# ▾

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public CustomActionAttribute()
```

```
Public Sub New
```

```
public:  
CustomActionAttribute()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# CustomActionAttribute

## Constructor (name)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionAttribute ►

### CustomActionAttribute(String)

C# ▾

Marks a method as a custom action entry point.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public CustomActionAttribute(  
    string name  
)
```

```
Public Sub New ( _  
    name As String _  
)
```

```
public:  
CustomActionAttribute(  
    String^ name  
)
```

#### Parameters

##### ***name* (String)**

Name of the function to be exported, defaults to the name of this method

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Name Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionAttribute ►

### Name

C# ▾

Gets or sets the name of the custom action entrypoint. A null value defaults to the name of the method.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string Name { get; }
```

```
Public ReadOnly Property Name As String
```

```
public:  
property String^ Name {  
    String^ get ();  
}
```

#### Value

name of the custom action entrypoint, or null if none was specified

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **CustomActionData Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► **CustomActionData**

C# ▾

Contains a collection of key-value pairs suitable for passing between immediate and deferred/rollback/commit custom actions.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public sealed class CustomActionData : IDictionary<string, string>, ICollection<KeyValuePair<string, string>>, IEnumerable
```

```
Public NotInheritable Class CustomActionData _  
    Implements IDictionary(Of String, String),  
    ICollection(Of KeyValuePair(Of String, String)),  
    IEnumerable
```

```
public ref class CustomActionData sealed : IDictionary<String^, String^>,  
    ICollection<KeyValuePair<String^, String^>>,  
    IEnumerable
```

### Members

All Members	Constructors	Methods	Properties	Fields
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static		<input checked="" type="checkbox"/> Declared <input checked="" type="checkbox"/> Inherited

Icon	Member	Description
≡	<a href="#">CustomActionData()</a>	Creates a new empty custom action data object.
≡	<a href="#">CustomActionData(String)</a>	Reconstructs a custom action data object from data that was previously persisted in a string.

	<a href="#">Add(String, String)</a>	Adds a key and value to the data collection.
	<a href="#">Add(KeyValuePair&lt;String, String&gt;)</a>	Adds an item with key and value to the data collection.
	<a href="#">AddObject&lt;T&gt;(String, T)</a>	Adds a value to the data collection, using XML serialization to persist the object as a string.
	<a href="#">Clear()</a>	Removes all items from the data
	<a href="#">Contains(KeyValuePair&lt;String, String&gt;)</a>	Determines whether the data contains a specified item.
	<a href="#">ContainsKey(String)</a>	Determines whether the data contains an item with the specified key.
	<a href="#">CopyTo(KeyValuePair&lt;String, String&gt;[], Int32)</a>	Copies the data to an array, starting at a particular array index.
	<a href="#">Count</a>	Gets the number of items in the data.
		

	<code>Equals(Object)</code>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
💡	<code>Finalize()</code>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
✳️	<code>GetEnumerator()</code>	Returns an enumerator that iterates through the collection.
✳️	<code>GetHashCode()</code>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.  (Inherited from <a href="#">Object</a> .)
✳️	<code>GetObject&lt;T&gt;(String)</code>	Gets a value from the data collection, using XML serialization to load the object from a string.
✳️	<code>GetType()</code>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
📝	<code>IsReadOnly</code>	Gets a value indicating whether the data is read-only.
📝		

	<code>Item[String]</code>	Gets or sets a data value with a specified key.
	<code>Keys</code>	Gets a collection object containing all the keys of the data.
	<code>MemberwiseClone()</code>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<code>PropertyName</code>	"CustomActionData" literal property name.
	<code>Remove(String)</code>	Removes the item with the specified key from the data.
	<code>Remove(KeyValuePair&lt;String, String&gt;)</code>	Removes an item from the data.
	<code>ToString()</code>	Gets a string representation of the data suitable for persisting in a property.  (Overrides <a href="#">Object.ToString()</a> .)
	<code>TryGetValue(String, String)</code>	Gets the value with the specified key.
	<code>Values</code>	Gets a collection containing all the values of the data.

## **Remarks**

Call the [ToString\(\)](#) method to get a string suitable for storing in a property and reconstructing the custom action data later.

## **Inheritance Hierarchy**

Object

└ CustomActionData

## **See Also**

[CustomActionData](#)

[DoAction\(String, CustomActionData\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **CustomActionData Constructor**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ►

**CustomActionData()**

C# ▾

▀ **Members**

Icon	Member	Description
≡	<a href="#">CustomActionData()</a>	Creates a new empty custom action data object.
≡	<a href="#">CustomActionData(String)</a>	Reconstructs a custom action data object from data that was previously persisted in a string.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# CustomActionData Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ►

## CustomActionData()

C# ▾

Creates a new empty custom action data object.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public CustomActionData()
```

```
Public Sub New
```

```
public:  
CustomActionData()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CustomActionData Constructor (keyValueList)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► CustomActionData ►  
CustomActionData(String)

C# ▾

Reconstructs a custom action data object from data that was previously persisted in a string.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public CustomActionData(  
    string keyValueList  
)
```

```
Public Sub New (  
    keyValueList As String  
)
```

```
public:  
CustomActionData(  
    String^ keyValueList  
)
```

### - Parameters

#### keyValueList (String)

Previous output from [ToString\(\)](#).

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Add Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ► Add()

C# ▾

### Members

Icon	Member	Description
≡	Add(String, String)	Adds a key and value to the data collection.
≡	Add(KeyValuePair<String, String>)	Adds an item with key and value to the data collection.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Add Method (item)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ►

### Add(KeyValuePair<String, String>)

C# ▾

Adds an item with key and value to the data collection.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Add(  
    KeyValuePair<string, string> item  
)
```

```
Public Sub Add ( _  
    item As KeyValuePair(Of String, String) _  
)
```

```
public:  
virtual void Add(  
    KeyValuePair<String^, String^> item  
) sealed
```

#### Parameters

##### *item* (KeyValuePair<String, String>)

Case-sensitive data key, with a data value that may be null.

#### Exceptions

Exception	Condition
ArgumentException	the key does not consist solely of letters, numbers, and the period, underscore, and space characters.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Add Method (key, value)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ►

### Add(String, String)

C# ▾

Adds a key and value to the data collection.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Add(  
    string key,  
    string value  
)
```

```
Public Sub Add ( _  
    key As String, _  
    value As String _  
)
```

```
public:  
virtual void Add(  
    String^ key,  
    String^ value  
) sealed
```

#### Parameters

##### **key (String)**

Case-sensitive data key.

##### **value (String)**

Data value (may be null).

#### Exceptions

Exception	Condition
ArgumentException	the key does not consist solely of letters,

numbers, and the period, underscore, and space characters.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## AddObject<T> Method (key, value)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ►

**AddObject<T>(String, T)**

C# ▾

Adds a value to the data collection, using XML serialization to persist the object as a string.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void AddObject<T>(  
    string key,  
    T value  
)
```

```
Public Sub AddObject(Of T) ( _  
    key As String, _  
    value As T _  
)
```

```
public:  
generic<typename T>  
void AddObject(  
    String^ key,  
    T value  
)
```

### - Generic Template Parameters

*T*

### - Parameters

**key (String)**

Case-sensitive data key.

***value (T)***

Data value (may be null).

**Exceptions**

Exception	Condition
ArgumentException	the key does not consist solely of letters, numbers, and the period, underscore, and space characters.
NotSupportedException	The value type does not support XML serialization.
InvalidOperationException	The value could not be serialized.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Clear Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ►

### Clear()

C# ▾

Removes all items from the data.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Clear()
```

```
Public Sub Clear
```

```
public:  
virtual void Clear() sealed
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Contains Method (item)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ►

### Contains(KeyValuePair<String, String>)

C# ▾

Determines whether the data contains a specified item.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool Contains(  
    KeyValuePair<string, string> item  
)
```

```
Public Function Contains ( _  
    item As KeyValuePair(Of String, String) _  
) As Boolean
```

```
public:  
virtual bool Contains(  
    KeyValuePair<String^, String^> item  
) sealed
```

#### Parameters

##### *item* (KeyValuePair<String, String>)

The data item to locate.

#### Return Value

true if the data contains the item; otherwise, false

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation ContainsKey Method (key)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ►

## ContainsKey(String)

C#

Determines whether the data contains an item with the specified key.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public bool ContainsKey(  
    string key  
)
```

```
Public Function ContainsKey ( _  
    key As String _  
) As Boolean
```

```
public:  
virtual bool ContainsKey(  
    String^ key  
) sealed
```

### Parameters

#### key (String)

Case-sensitive data key.

### Return Value

true if the data contains an item with the key; otherwise, false

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CopyTo Method (array, arrayIndex)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► CustomActionData ►  
**CopyTo(KeyValuePair<String, String>[][], Int32)**

C# ▾

Copies the data to an array, starting at a particular array index.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void CopyTo(  
    KeyValuePair<string, string>[] array,  
    int arrayIndex  
)
```

```
Public Sub CopyTo ( _  
    array As KeyValuePair(Of String, String)(),  
    arrayIndex As Integer _  
)
```

```
public:  
virtual void CopyTo(  
    array<KeyValuePair<String^, String^>>^ array  
    int arrayIndex  
) sealed
```

### Parameters

**array ( KeyValuePair<String, String>[] )**

Destination array.

**arrayIndex (Int32)**

Index in the array at which copying begins.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Count Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ► **Count**

C# ▾

Gets the number of items in the data.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public int Count { get; }
```

```
Public Readonly Property Count As Integer
```

```
public:  
virtual property int Count {  
    int get () sealed;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetEnumerator Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ►

**GetEnumerator()**

C# ▾

Returns an enumerator that iterates through the collection.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public IEnum<KeyValuePair<string, string>> GetEnumerator();
```

```
Public Function GetEnumerator As IEnum<of KeyValuePair<String, String>>
```

```
public:  
virtual IEnum<KeyValuePair<String^, String^>>^
```

[-] **Return Value**

An enumerator that can be used to iterate through the collection.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



## Deployment Tools Foundation **GetObject<T> Method (key)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ►

### GetObject<T>(String)

C# ▾

Gets a value from the data collection, using XML serialization to load the object from a string.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public T GetObject<T>(  
    string key  
)
```

```
Public Function GetObject(Of T) ( _  
    key As String _  
) As T
```

```
public:  
generic<typename T>  
T GetObject(  
    String^ key  
)
```

#### Generic Template Parameters

T

#### Parameters

##### key (String)

Case-sensitive data key.

#### Exceptions

Exception

Condition

InvalidOperationException

The value could not be deserialized.

---

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)  
Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation IsReadOnly Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ►

## IsReadOnly

C# ▾

Gets a value indicating whether the data is read-only.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public bool IsReadOnly { get; }
```

```
Public ReadOnly Property IsReadOnly As Boolean
```

```
public:  
virtual property bool IsReadOnly {  
    bool get () sealed;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Item Property (key)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ►

## Item[String]

C# ▾

Gets or sets a data value with a specified key.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string this[  
    string key  
] { get; set; }
```

```
Public Default Property Item ( _  
    key As String _  
) As String
```

```
public:  
virtual property String^ default[String^ key] {  
    String^ get (String^ key) sealed;  
    void set (String^ key, String^ value) sealed  
}
```

### Parameters

#### key (String)

Case-sensitive data key.

### Exceptions

Exception	Condition
ArgumentException	the key does not consist solely of letters, numbers, and the period, underscore, and space characters.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Keys Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ► Keys

C# ▾

Gets a collection object containing all the keys of the data.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public ICollection<string> Keys { get; }
```

```
Public ReadOnly Property Keys As ICollection(Of Stri
```

```
public:  
virtual property ICollection<String^>^ Keys {  
    ICollection<String^>^ get () sealed;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## PropertyName Field

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ►

### PropertyName

C# ▾

"CustomActionData" literal property name.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public const string PropertyName
```

```
Public Const PropertyName As String
```

```
public:  
literal String^ PropertyName
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Remove Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ►

### Remove()

C# ▾

#### Members

Icon	Member	Description
≡	<a href="#">Remove(String)</a>	Removes the item with the specified key from the data.
≡	<a href="#">Remove(KeyValuePair&lt;String, String&gt;)</a>	Removes an item from the data.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Remove Method (item)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ►

**Remove(KeyValuePair<String, String>)**

C# ▾

Removes an item from the data.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public bool Remove(  
    KeyValuePair<string, string> item  
)
```

```
Public Function Remove ( _  
    item As KeyValuePair(Of String, String) _  
) As Boolean
```

```
public:  
virtual bool Remove(  
    KeyValuePair<String^, String^> item  
) sealed
```

### Parameters

**item (KeyValuePair<String, String>)**

The item to remove.

### Return Value

true if the item was successfully removed from the data; false if the item was not found

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Remove Method (key)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ►

### Remove(String)

C# ▾

Removes the item with the specified key from the data.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool Remove(  
    string key  
)
```

```
Public Function Remove ( _  
    key As String _  
) As Boolean
```

```
public:  
virtual bool Remove(  
    String^ key  
) sealed
```

#### Parameters

##### key (String)

Case-sensitive data key.

#### Return Value

true if the item was successfully removed from the data; false if an item with the specified key was not found

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **ToString** Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ►

### **ToString()**

C# ▾

Gets a string representation of the data suitable for persisting in a property.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override string ToString()
```

```
Public Overrides Function ToString As String
```

```
public:  
virtual String^ ToString() override
```

#### Return Value

Data string in the form "Key1=Value1;Key2=Value2"

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation TryGetValue Method (key, value)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ►

## TryGetValue(String, String)

C# ▾

Gets the value with the specified key.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public bool TryGetValue(  
    string key,  
    out string value  
)
```

```
Public Function TryGetValue ( _  
    key As String, _  
    <OutAttribute> ByRef value As String _  
) As Boolean
```

```
public:  
virtual bool TryGetValue(  
    String^ key,  
    [OutAttribute] String^% value  
) sealed
```

### Parameters

#### key (String)

Case-sensitive data key.

#### value ( String )

Value associated with the specified key, or null if an item with the specified key was not found

### Return Value

true if the data contains an item with the specified key; otherwise, false.

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Values Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► CustomActionData ►

## Values

C# ▾

Gets a collection containing all the values of the data.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public ICollection<string> Values { get; }
```

```
Public ReadOnly Property Values As ICollection(Of St
```

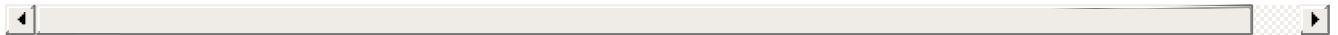
```
public:  
virtual property ICollection<String^>^ Values {  
    ICollection<String^>^ get () sealed;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation **CustomActionTypes Enumeration**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► **CustomActionTypes**

C# ▾

Defines flags for the Type column of the CustomAction table.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
[FlagsAttribute]
public enum CustomActionTypes
```

```
<FlagsAttribute> _
Public Enumeration CustomActionTypes
```

```
[FlagsAttribute]
public enum class CustomActionTypes
```

## Members

Member	Description
<b>None</b>	Unspecified custom action type.
<b>Dll</b>	Target = entry point name
<b>Exe</b>	Target = command line args
<b>TextData</b>	Target = text string to be formatted and set into property
<b>JScript</b>	Target = entry point name, null if none to

	call
<b>VBScript</b>	Target = entry point name, null if none to call
<b>Install</b>	Target = property list for nested engine initialization
<b>SourceFile</b>	Source = File.File, file part of installation
<b>Directory</b>	Source = Directory.Directory, folder containing existing file
<b>Property</b>	Source = Property.Property, full path to executable
<b>Continue</b>	Ignore action return status, continue running
<b>Async</b>	Run asynchronously
<b>FirstSequence</b>	Skip if UI sequence already run
<b>OncePerProcess</b>	Skip if UI sequence already run in same process

<b>ClientRepeat</b>	Run on client only if UI already run on client
<b>InScript</b>	Queue for execution within script
<b>Rollback</b>	In conjunction with InScript: queue in Rollback script
<b>Commit</b>	In conjunction with InScript: run Commit ops from script on success
<b>NoImpersonate</b>	No impersonation, run in system context
<b>TSAware</b>	Impersonate for per-machine installs on TS machines
<b>SixtyFourBitScript</b>	Script requires 64bit process
<b>HideTarget</b>	Don't record the contents of the Target field in the log file
<b>PatchUninstall</b>	The custom action runs only when a patch is being uninstalled

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Database Class

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database

C# ▾

Accesses a Windows Installer database.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public class Database : InstallerHandle
```

```
Public Class Database  
    Inherits InstallerHandle
```

```
public ref class Database : public InstallerHandle
```

## Members

All Members	Constructors	Methods	Properties
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static	<input checked="" type="checkbox"/> Declared <input checked="" type="checkbox"/> Inherited

Icon	Member	Description
≡	Database(String)	Opens an existing database in read-only mode.
≡	Database(String, String)	Opens an existing database or another database as context.
≡	Database(String, DatabaseOpenMode)	Opens an existing database or creates a new one.

	<code>Database(IntPtr, Boolean, String, DatabaseOpenMode)</code>	Creates a new database MSI handle.
	<code>ApplyTransform(String)</code>	Apply a transform to the database, suppressing any error specified by the transform, and returning summary information.
	<code>ApplyTransform(String, TransformErrors)</code>	Apply a transform to the database, specifying error conditions to suppress.
	<code>Close()</code>	Closes the handle. After closing the handle, further methods that require it will throw an <code>InvalidOperationException</code> .  (Inherited from <code>Installer</code> )
	<code>CodePage</code>	Gets or sets the code page for the database.
	<code>Commit()</code>	Finalizes the persisting database. All persisting changes written to the writeable database and no temporary columns are written.
	<code>CountRows(String)</code>	Gets the count of all rows in the table.
	<code>CountRows(String, String)</code>	Gets the count of all rows in the table.

		table that satisfy a given condition.
≡	<a href="#">CreateObjRef(Type)</a>	Creates an object that contains the relevant information to generate a proxy used to communicate with a remote object.  (Inherited from <a href="#">MarshalByRefObject</a> .)
≡	<a href="#">CreateRecord(Int32)</a>	Creates a new record at the requested number.
≡	<a href="#">CreateTransformSummaryInfo(Database, String, TransformErrors, TransformValidations)</a>	Creates and populates summary information for an existing transform file, setting the properties with the reference ProductCode and ProductVersion.
≡	<a href="#">DeleteOnClose(String)</a>	Schedules a file or directory for deletion after the database is closed.
≡	<a href="#">Dispose(Boolean)</a>	Closes the database handle. After closing a handle, further calls may throw <a href="#">InvalidOperationException</a> .  (Overrides <a href="#">InstallerHandle.Dispose(Boolean)</a> )
≡	<a href="#">Dispose()</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .

		(Inherited from <a href="#">Installer</a> )
≡	<a href="#">Equals(Object)</a>	Tests whether this handle is equal to another handle. Two handle objects are equal if their types are the same and their native integer handles are the same.
		(Inherited from <a href="#">Installer</a> )
≡	<a href="#">Execute(String, Object[])</a>	Executes the query specified by the SQL string. The query must be a SELECT statement.
≡	<a href="#">Execute(String, Record)</a>	Executes the query specified by the SQL string. The query must be a SELECT statement.
≡	<a href="#">ExecuteIntegerQuery(String, Object[])</a>	Executes the specified SELECT query and returns results as integers.
≡	<a href="#">ExecuteIntegerQuery(String, Record)</a>	Executes the specified SELECT query and returns results as integers.
≡	<a href="#">ExecuteQuery(String, Object[])</a>	Executes the specified SELECT query and returns results.
≡	<a href="#">ExecuteQuery(String, Record)</a>	Executes the specified

		SELECT query and re results.
≡	<a href="#">ExecuteScalar(String, Object[])</a>	Executes the specified SELECT query and re single result.
≡	<a href="#">ExecuteScalar(String, Record)</a>	Executes the specified SELECT query and re single result.
≡	<a href="#">ExecuteStringQuery(String, Object[])</a>	Executes the specified SELECT query and re results as strings.
≡	<a href="#">ExecuteStringQuery(String, Record)</a>	Executes the specified SELECT query and re results as strings.
≡	<a href="#">Export(String, String)</a>	Copies the structure a a specified table to a t file.
≡	<a href="#">ExportAll(String)</a>	Exports all database t streams, and summary to archive files.
	<a href="#">FilePath</a>	Gets the file path the I was originally opened

		if not known.
💡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attach resources and perform cleanup operations before the <a href="#">Object</a> is reclaimed by the collection.  (Inherited from <a href="#">Object</a> )
💡 S	<a href="#">FromHandle(IntPtr, Boolean)</a>	Creates a new <a href="#">Database</a> object from an integer database handle.
💡	<a href="#">GenerateTransform(Database, String)</a>	Creates a transform that is applied to the object during results in the reference.
💡	<a href="#">GetHashCode()</a>	Gets a hash value for the object.  (Inherited from <a href="#">InstalledObject</a> )
💡	<a href="#">GetLifetimeService()</a>	Retrieves the current lifetime service object that controls the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject</a> )
💡	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> )
📁	<a href="#">Handle</a>	Gets the native integer handle.  (Inherited from <a href="#">InstalledObject</a> )

	<a href="#">Import(String)</a>	Imports a database table from a text archive file, dropping existing table.
	<a href="#">ImportAll(String)</a>	Imports all database tables, streams, and summary from archive files.
	<a href="#">InitializeLifetimeService()</a>	Obtains a lifetime service to control the lifetime policy of the instance.  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">IsClosed</a>	Checks if the handle is closed. When closed, method throws an exception. The handle object may throw <a href="#">InvalidHandleException</a> .  (Inherited from <a href="#">Installer</a> .)
	<a href="#">IsColumnPersistent(String, String)</a>	Checks whether a table has a persistent column with the specified name.
	<a href="#">IsReadOnly</a>	Gets a boolean value indicating whether this database is in read-only mode.
	<a href="#">IsTablePersistent(String)</a>	Checks whether a table is persistent in the database.
		

	<code>IsTransformValid(String)</code>	Checks whether a transform is valid for this Database, based on its validation data and the specified string.
≡	<code>IsTransformValid(SummaryInfo)</code>	Checks whether a transform is valid for this Database, based on its SummaryInfo data.
💡	<code>MemberwiseClone(Boolean)</code>	Creates a shallow copy of the current <a href="#">MarshalByRefObject</a> object.  (Inherited from <a href="#">MarshalByRefObject</a> .)
💡	<code>MemberwiseClone()</code>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> )
≡	<code>Merge(Database, String)</code>	Merges another database into this database.
≡	<code>Merge(Database)</code>	Merges another database into this database.
📊	<code>OpenMode</code>	Gets the open mode for this database.
≡	<code>OpenView(String, Object[])</code>	Gets a <a href="#">View</a> object representing the query specified by the string.

	<a href="#">SummaryInfo</a>	Gets the SummaryInfo for this database that can examine and modify parts of the summary information.
	<a href="#">Tables</a>	Gets the collection of tables in the Database.
	<a href="#">ToString()</a>	Returns the file path of the database, or the handle if the file path was not specified.  (Overrides <a href="#">Object.ToString()</a> )
	<a href="#">ViewTransform(String)</a>	Apply a transform to the recording the changes in the "_TransformView" table.

## Remarks

The [Commit\(\)](#) method must be called before the Database is closed to write out all persistent changes. If the Commit method is not called, the installer performs an implicit rollback upon object destruction.

The client can use the following procedure for data access:

1. Obtain a Database object using one of the Database constructors.
2. Initiate a query using a SQL string by calling the [OpenView\(String, Object\[\]\)](#) method of the Database.
3. Set query parameters in a [Record](#) and execute the database query by calling the [Execute\(Record\)](#) method of the [View](#). This produces a result that can be fetched or updated.
4. Call the [Fetch\(\)](#) method of the View repeatedly to return Records.

5. Update database rows of a Record object obtained by the Fetch method using one of the [Modify\(ViewModifyMode, Record\)](#) method of the View.
6. Release the query and any unfetched records by calling the [Close\(\)](#) method of the View.
7. Persist any database updates by calling the Commit method of the Database.

## Inheritance Hierarchy

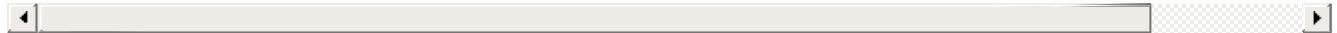
[Object](#)



---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

# Database Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ► Database()

C# ▾

## Members

Icon	Member	Description
≡	Database(String)	Opens an existing database in read-only mode.
≡	Database(String, String)	Opens an existing database with another database as output.
≡	Database(String, DatabaseOpenMode)	Opens an existing database or creates a new one.
💡	Database(IntPtr, Boolean, String, DatabaseOpenMode)	Creates a new database from an MS handle.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Database Constructor (handle, ownsHandle, filePath, openMode)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ►

**Database(IntPtr, Boolean, String, DatabaseOpenMode)**

C# ▾

Creates a new database from an MSI handle.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected Database(  
    IntPtr handle,  
    bool ownsHandle,  
    string filePath,  
    DatabaseOpenMode openMode  
)
```

```
Protected Sub New ( _  
    handle As IntPtr, _  
    ownsHandle As Boolean, _  
    filePath As String, _  
    openMode As DatabaseOpenMode _  
)
```

```
protected:  
Database(  
    IntPtr handle,  
    bool ownsHandle,  
    String^ filePath,  
    DatabaseOpenMode openMode  
)
```

### Parameters

#### handle (IntPtr)

Native MSI database handle.

***ownsHandle* ([Boolean](#))**

True if the handle should be closed when the database object is disposed

***filePath* ([String](#))**

Path of the database file, if known

***openMode* ([DatabaseOpenMode](#))**

Mode the handle was originally opened in

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Database Constructor (filePath)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ►  
**Database(String)**

C# ▾

Opens an existing database in read-only mode.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public Database(  
    string filePath  
)
```

```
Public Sub New (  
    filePath As String  
)
```

```
public:  
Database(  
    String^ filePath  
)
```

### Parameters

#### filePath (String)

Path to the database file.

### Remarks

Because this constructor initiates database access, it cannot be used with a running installation.

The Database object should be [Close\(\)](#)d after use. It is best that the handle be closed manually as soon as it is no longer needed, as leaving lots of unused handles open can degrade performance.

Win32 MSI API: [MsiOpenDatabase](#)

## **Exceptions**

<b>Exception</b>	<b>Condition</b>
<a href="#">InstallerException</a>	the database could not be created/opened

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Database Constructor (filePath, mode)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ►  
**Database(String, DatabaseOpenMode)**

C# ▾

Opens an existing database or creates a new one.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public Database(  
    string filePath,  
    DatabaseOpenMode mode  
)
```

```
Public Sub New (  
    filePath As String,  
    mode As DatabaseOpenMode _  
)
```

```
public:  
Database(  
    String^ filePath,  
    DatabaseOpenMode mode  
)
```

### Parameters

#### filePath (String)

Path to the database file. If an empty string is supplied, a temporary database is created that is not persisted.

#### mode (DatabaseOpenMode)

Open mode for the database

### Remarks

Because this constructor initiates database access, it cannot be used with a running installation.

The database object should be [Close\(\)](#)d after use. The finalizer will close the handle if it is still open, however due to the nondeterministic nature of finalization it is best that the handle be closed manually as soon as it is no longer needed, as leaving lots of unused handles open can degrade performance.

A database opened in [CreateDirect](#) or [Direct](#) mode will be automatically committed when it is closed. However a database opened in [Create](#) or [Transact](#) mode must have the [Commit\(\)](#) method called before it is closed otherwise no changes will be persisted.

Win32 MSI API: [MsiOpenDatabase](#)

## Exceptions

Exception	Condition
<a href="#">InstallerException</a>	the database could not be created/opened

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Database Constructor (filePath, outputPath)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ►  
**Database(String, String)**

C# ▾

Opens an existing database with another database as output.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public Database(  
    string filePath,  
    string outputPath  
)
```

```
Public Sub New ( _  
    filePath As String, _  
    outputPath As String _  
)
```

```
public:  
Database(  
    String^ filePath,  
    String^ outputPath  
)
```

### Parameters

#### *filePath* (String)

Path to the database to be read.

#### *outputPath* (String)

Open mode for the database

### Return Value

Database object representing the created or opened database

## **Remarks**

When a database is opened as the output of another database, the summary information stream of the output database is actually a read-only mirror of the original database and thus cannot be changed. Additionally, it is not persisted with the database. To create or modify the summary information for the output database it must be closed and re-opened.

The Database object should be [Close\(\)](#) after use. It is best that the handle be closed manually as soon as it is no longer needed, as leaving lots of unused handles open can degrade performance.

The database is opened in [CreateDirect](#) mode, and will be automatically committed when it is closed.

Win32 MSI API: [MsiOpenDatabase](#)

## **Exceptions**

Exception	Condition
<a href="#">InstallerException</a>	the database could not be created/opened

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ApplyTransform Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ►

**ApplyTransform()**

C# ▾

### ▀ Members

Icon	Member	Description
≡	<a href="#">ApplyTransform(String)</a>	Apply a transform to the database, suppressing any error conditions specified by the transform's summary information.
≡	<a href="#">ApplyTransform(String, TransformErrors)</a>	Apply a transform to the database, specifying error conditions to suppress.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ApplyTransform Method (transformFile)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ►  
**ApplyTransform(String)**

C# ▾

Apply a transform to the database, suppressing any error conditions specified by the transform's summary information.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void ApplyTransform(  
    string transformFile  
)
```

```
Public Sub ApplyTransform (  
    transformFile As String  
)
```

```
public:  
void ApplyTransform(  
    String^ transformFile  
)
```

### Parameters

#### **transformFile (String)**

Path to the transform file

### Remarks

Win32 MSI API: [MsiDatabaseApplyTransform](#)

### Exceptions

Exception	Condition

InstallerException	the transform could not be applied
InvalidHandleException	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ApplyTransform Method (transformFile, errorConditionsToSuppress)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ►

**ApplyTransform(String, TransformErrors)**

C#

Apply a transform to the database, specifying error conditions to suppress.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void ApplyTransform(  
    string transformFile,  
    TransformErrors errorConditionsToSuppress  
)
```

```
Public Sub ApplyTransform ( _  
    transformFile As String, _  
    errorConditionsToSuppress As TransformErrors  
)
```

```
public:  
void ApplyTransform(  
    String^ transformFile,  
    TransformErrors errorConditionsToSuppress  
)
```

### Parameters

**transformFile (String)**

Path to the transform file

**errorConditionsToSuppress (TransformErrors)**

Error conditions that are to be suppressed

### Remarks

Win32 MSI API: [MsiDatabaseApplyTransform](#)

## **Exceptions**

<b>Exception</b>	<b>Condition</b>
<a href="#">InstallerException</a>	the transform could not be applied
<a href="#">InvalidHandleException</a>	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **CodePage Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ► **CodePage**

C# ▾

Gets or sets the code page of the Database.

## [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public int CodePage { get; set; }
```

```
Public Property CodePage As Integer
```

```
public:  
property int CodePage {  
    int get ();  
    void set (int value);  
}
```

## [-] Remarks

Getting or setting the code page is a slow operation because it involves an export or import of the codepage data to/from a temporary file.

## [-] Exceptions

Exception	Condition
IOException	error exporting/importing the codepage data
InvalidHandleException	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Commit Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ► Commit()

C# ▾

Finalizes the persistent form of the database. All persistent data is written to the writeable database, and no temporary columns or rows are written.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Commit()
```

```
Public Sub Commit
```

```
public:  
void Commit()
```

### - Remarks

For a database open in [ReadOnly](#) mode, this method has no effect.

For a database open in [CreateDirect](#) or [Direct](#) mode, it is not necessary to call this method because the database will be automatically committed when it is closed. However this method may be called at any time to persist the current state of tables loaded into memory.

For a database open in [Create](#) or [Transact](#) mode, no changes will be persisted until this method is called. If the database object is closed without calling this method, the database file remains unmodified.

Win32 MSI API: [MsiDatabaseCommit](#)

### - Exceptions

Exception	Condition
<a href="#">InvalidOperationException</a>	the Database handle is invalid

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **CountRows Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ► **CountRows()**

C# ▾

▀ **Members**

Icon	Member	Description
	<a href="#">CountRows(String)</a>	Gets the count of all rows in the table.
	<a href="#">CountRows(String, String)</a>	Gets the count of all rows in the table that satisfy a given condition.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation CountRows Method (table)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ►  
**CountRows(String)**

C# ▾

Gets the count of all rows in the table.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public int CountRows(  
    string table  
)
```

```
Public Function CountRows ( _  
    table As String _  
) As Integer
```

```
public:  
int CountRows(  
    String^ table  
)
```

## Parameters

### table (String)

Name of the table whose rows are to be counted

## Return Value

The count of all rows in the table

## Exceptions

Exception	Condition
InstallerException	the View could not be executed
InvalidHandleException	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation CountRows Method (table, where)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ►

### CountRows(String, String)

C#

Gets the count of all rows in the table that satisfy a given condition.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public int CountRows(  
    string table,  
    string where  
)
```

```
Public Function CountRows ( _  
    table As String, _  
    where As String _  
) As Integer
```

```
public:  
int CountRows(  
    String^ table,  
    String^ where  
)
```

#### Parameters

##### **table (String)**

Name of the table whose rows are to be counted

##### **where (String)**

Conditional expression, such as could be placed on the end of a SQL WHERE clause

#### Return Value

The count of all rows in the table satisfying the condition

## Exceptions

Exception	Condition
BadQuerySyntaxException	the SQL WHERE syntax is invalid
InstallerException	the View could not be executed
InvalidHandleException	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **CreateRecord Method**  
**(fieldCount)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ►  
**CreateRecord(Int32)**

Creates a new record object with the requested number of fields.

C# ▾

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public Record CreateRecord(  
    int fieldCount  
)
```

```
Public Function CreateRecord ( _  
    fieldCount As Integer _  
) As Record
```

```
public:  
Record^ CreateRecord(  
    int fieldCount  
)
```

[-] **Parameters**

***fieldCount* (Int32)**

Required number of fields, which may be 0. The maximum number of fields in a record is limited to 65535.

[-] **Return Value**

A new record object that can be used with the database.

[-] **Remarks**

This method is equivalent to directly calling the **Record** constructor in all cases outside of a custom action context. When in a custom action

session, this method allows creation of a record that can work with a database other than the session database.

The Record object should be [Close\(\)](#)d after use. It is best that the handle be closed manually as soon as it is no longer needed, as leaving lots of unused handles open can degrade performance.

Win32 MSI API: [MsiCreateRecord](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CreateTransformSummaryInfo Method (referenceDatabase, transformFile, errors, validations)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ►  
**CreateTransformSummaryInfo(Database, String, TransformErrors, TransformValidations)**

C# ▾

Creates and populates the summary information stream of an existing transform file, and fills in the properties with the base and reference ProductCode and ProductVersion.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void CreateTransformSummaryInfo(  
    Database referenceDatabase,  
    string transformFile,  
    TransformErrors errors,  
    TransformValidations validations  
)
```

```
Public Sub CreateTransformSummaryInfo ( _  
    referenceDatabase As Database, _  
    transformFile As String, _  
    errors As TransformErrors, _  
    validations As TransformValidations _  
)
```

```
public:  
void CreateTransformSummaryInfo(  
    Database^ referenceDatabase,  
    String^ transformFile,  
    TransformErrors errors,  
    TransformValidations validations  
)
```

## [-] Parameters

### ***referenceDatabase* ([Database](#))**

Database that does not include the changes

### ***transformFile* ([String](#))**

Name of the generated transform file

### ***errors* ([TransformErrors](#))**

Error conditions that should be suppressed when the transform is applied

### ***validations* ([TransformValidations](#))**

Defines which properties should be validated to verify that this transform can be applied to a database.

## [-] Remarks

Win32 MSI API: [MsiCreateTransformSummaryInfo](#)

## [-] Exceptions

Exception	Condition
<a href="#">InstallerException</a>	the transform summary info could not be generated
<a href="#">InvalidHandleException</a>	a Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation DeleteOnClose Method (path)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ►

### DeleteOnClose(String)

C#

Schedules a file or directory for deletion after the database handle is closed.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void DeleteOnClose(  
    string path  
)
```

```
Public Sub DeleteOnClose ( _  
    path As String _  
)
```

```
public:  
void DeleteOnClose(  
    String^ path  
)
```

#### Parameters

##### *path* (String)

File or directory path to be deleted. All files and subdirectories under a directory are deleted.

#### Remarks

Once an item is scheduled, it cannot be unscheduled.

The items cannot be deleted if the Database object is auto-disposed by the garbage collector; the handle must be explicitly closed.

Files which are read-only or otherwise locked cannot be deleted, but the operation will not cause an exception to be thrown.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Dispose Method (disposing)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ►

## Dispose(Boolean)

C# ▾

Closes the database handle. After closing a handle, further method calls may throw [InvalidOperationException](#).

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected override void Dispose(  
    bool disposing  
)
```

```
Protected Overrides Sub Dispose ( _  
    disposing As Boolean _  
)
```

```
protected:  
virtual void Dispose(  
    bool disposing  
) override
```

### Parameters

#### *disposing* (Boolean)

If true, the method has been called directly or indirectly by a user's code, so managed and unmanaged resources will be disposed. If false, only unmanaged resources will be disposed.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Execute Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ► Execute()

C# ▾

### Members

Icon	Member	Description
	<a href="#">Execute(String, Object[])</a>	Executes the query specified by a SQL string. The query may not be a SELECT statement.
	<a href="#">Execute(String, Record)</a>	Executes the query specified by a SQL string. The query may not be a SELECT statement.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Execute Method (sql, record)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ►

### Execute(String, Record)

C#

Executes the query specified by a SQL string. The query may not be a SELECT statement.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Execute(  
    string sql,  
    Record record  
)
```

```
Public Sub Execute ( _  
    sql As String, _  
    record As Record _  
)
```

```
public:  
void Execute(  
    String^ sql,  
    Record^ record  
)
```

#### Parameters

##### sql (String)

SQL query string

##### record (Record)

Optional Record object containing the values that replace the parameter tokens (?) in the SQL query.

#### Remarks

Win32 MSI APIs: [MsiDatabaseOpenView](#), [MsiViewExecute](#)

## [-] Exceptions

Exception	Condition
<a href="#">BadQuerySyntaxException</a>	the SQL syntax is invalid
<a href="#">InstallerException</a>	the View could not be executed
<a href="#">InvalidHandleException</a>	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Execute Method (sqlFormat, args)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ►

### Execute(String, Object[])

C#

Executes the query specified by a SQL string. The query may not be a SELECT statement.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Execute(  
    string sqlFormat,  
    params Object[] args  
)
```

```
Public Sub Execute ( _  
    sqlFormat As String, _  
    ParamArray args As Object() _  
)
```

```
public:  
void Execute(  
    String^ sqlFormat,  
    ... array<Object^>^ args  
)
```

#### Parameters

##### sqlFormat (String)

SQL query string, which may contain format items

##### args ( Object [] )

Zero or more objects to format

#### Remarks

The *sqlFormat* parameter is formatted using **Format(String, Object[])**.

Win32 MSI APIs: [MsiDatabaseOpenView](#), [MsiViewExecute](#)

## [-] Exceptions

Exception	Condition
<a href="#">BadQuerySyntaxException</a>	the SQL syntax is invalid
<a href="#">InstallerException</a>	the View could not be executed
<a href="#">InvalidHandleException</a>	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ExecuteIntegerQuery Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ►

**ExecuteIntegerQuery()**

C# ▾

▀ Members

Icon	Member	Description
≡	<a href="#">ExecuteIntegerQuery(String, Object[])</a>	Executes the specified SQL SELECT query and returns all results as integers.
≡	<a href="#">ExecuteIntegerQuery(String, Record)</a>	Executes the specified SQL SELECT query and returns all results as integers.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ExecuteIntegerQuery Method (sql, record)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ►  
**ExecuteIntegerQuery(String, Record)**

C# ▾

Executes the specified SQL SELECT query and returns all results as integers.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public IList<int> ExecuteIntegerQuery(  
    string sql,  
    Record record  
)
```

```
Public Function ExecuteIntegerQuery ( _  
    sql As String, _  
    record As Record _  
) As IList(Of Integer)
```

```
public:  
IList<int>^ ExecuteIntegerQuery(  
    String^ sql,  
    Record^ record  
)
```

### Parameters

#### sql (String)

SQL SELECT query string

#### record (Record)

Optional Record object containing the values that replace the parameter tokens (?) in the SQL query.

## **▀ Return Value**

All results combined into an array

## **▀ Remarks**

Multiple rows columns will be collapsed into a single one-dimensional list.

Win32 MSI APIs: [MsiDatabaseOpenView](#), [MsiViewExecute](#),  
[MsiViewFetch](#)

## **▀ Exceptions**

Exception	Condition
<a href="#">BadQuerySyntaxException</a>	the SQL syntax is invalid
<a href="#">InstallerException</a>	the View could not be executed
<a href="#">InvalidHandleException</a>	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## ExecuteIntegerQuery Method (sqlFormat, args)

Deployment Tools Foundation Namespaces ►  
 Microsoft.Deployment.WindowsInstaller ► Database ►  
**ExecuteIntegerQuery(String, Object[])**

C#

Executes the specified SQL SELECT query and returns all results as integers.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public IList<int> ExecuteIntegerQuery(
    string sqlFormat,
    params Object[] args
)
```

```
Public Function ExecuteIntegerQuery ( _
    sqlFormat As String, _
    ParamArray args As Object() _
) As IList(Of Integer)
```

```
public:
IList<int>^ ExecuteIntegerQuery(
    String^ sqlFormat,
    ... array<Object^>^ args
)
```

### Parameters

#### **sqlFormat (String)**

SQL query string, which may contain format items

#### **args ( Object [] )**

Zero or more objects to format

### Return Value

All results combined into an array

## [-] Remarks

The `sqlFormat` parameter is formatted using `Format(String, Object[])`.

Multiple rows columns will be collapsed into a single one-dimensional list.

Win32 MSI APIs: [MsiDatabaseOpenView](#), [MsiViewExecute](#),  
[MsiViewFetch](#)

## [-] Exceptions

Exception	Condition
<a href="#">BadQuerySyntaxException</a>	the SQL syntax is invalid
<a href="#">InstallerException</a>	the View could not be executed
<a href="#">InvalidHandleException</a>	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ExecuteQuery Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ►

**ExecuteQuery()**

C# ▾

▀ Members

Icon	Member	Description
≡	<a href="#">ExecuteQuery(String, Object[])</a>	Executes the specified SQL SELECT query and returns all results.
≡	<a href="#">ExecuteQuery(String, Record)</a>	Executes the specified SQL SELECT query and returns all results.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ExecuteQuery Method (sql, record)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ►  
**ExecuteQuery(String, Record)**

C# ▾

Executes the specified SQL SELECT query and returns all results.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public IList ExecuteQuery(  
    string sql,  
    Record record  
)
```

```
Public Function ExecuteQuery ( _  
    sql As String, _  
    record As Record _  
) As IList
```

```
public:  
IList^ ExecuteQuery(  
    String^ sql,  
    Record^ record  
)
```

### Parameters

#### sql (String)

SQL SELECT query string

#### record (Record)

Optional Record object containing the values that replace the parameter tokens (?) in the SQL query.

### Return Value

All results combined into an array

## [-] Remarks

Multiple rows columns will be collapsed into a single one-dimensional list

Win32 MSI APIs: [MsiDatabaseOpenView](#), [MsiViewExecute](#), [MsiViewFetch](#)

## [-] Exceptions

Exception	Condition
<a href="#">BadQuerySyntaxException</a>	the SQL syntax is invalid
<a href="#">InstallerException</a>	the View could not be executed
<a href="#">InvalidHandleException</a>	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ExecuteQuery Method (sqlFormat, args)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ►  
**ExecuteQuery(String, Object[])**

C# ▾

Executes the specified SQL SELECT query and returns all results.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public IList ExecuteQuery(  
    string sqlFormat,  
    params Object[] args  
)
```

```
Public Function ExecuteQuery ( _  
    sqlFormat As String, _  
    ParamArray args As Object() _  
) As IList
```

```
public:  
IList^ ExecuteQuery(  
    String^ sqlFormat,  
    ... array<Object^>^ args  
)
```

### Parameters

#### sqlFormat (String)

SQL query string, which may contain format items

#### args ( Object [] )

Zero or more objects to format

### Return Value

All results combined into an array

## **- Remarks**

The *sqlFormat* parameter is formatted using **Format(String, Object[])**.

Multiple rows columns will be collapsed into a single one-dimensional list.

Win32 MSI APIs: [MsiDatabaseOpenView](#), [MsiViewExecute](#),  
[MsiViewFetch](#)

## **- Exceptions**

Exception	Condition
<a href="#">BadQuerySyntaxException</a>	the SQL syntax is invalid
<a href="#">InstallerException</a>	the View could not be executed
<a href="#">InvalidHandleException</a>	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ExecuteScalar Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ►

### ExecuteScalar()

C# ▾

#### Members

Icon	Member	Description
≡	<a href="#">ExecuteScalar(String, Object[])</a>	Executes the specified SQL SELECT query and returns a single result.
≡	<a href="#">ExecuteScalar(String, Record)</a>	Executes the specified SQL SELECT query and returns a single result.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ExecuteScalar Method (sql, record)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ►  
**ExecuteScalar(String, Record)**

C# ▾

Executes the specified SQL SELECT query and returns a single result.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public Object ExecuteScalar(  
    string sql,  
    Record record  
)
```

```
Public Function ExecuteScalar ( _  
    sql As String, _  
    record As Record _  
) As Object
```

```
public:  
Object^ ExecuteScalar(  
    String^ sql,  
    Record^ record  
)
```

### Parameters

#### sql (String)

SQL SELECT query string

#### record (Record)

Optional Record object containing the values that replace the parameter tokens (?) in the SQL query.

### Return Value

First field of the first result

## **Remarks**

Win32 MSI APIs: [MsiDatabaseOpenView](#), [MsiViewExecute](#), [MsiViewFetch](#)

## **Exceptions**

Exception	Condition
<a href="#">BadQuerySyntaxException</a>	the SQL syntax is invalid
<a href="#">InstallerException</a>	the View could not be executed or the query returned 0 results
<a href="#">InvalidHandleException</a>	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# ExecuteScalar Method (sqlFormat, args)

Deployment Tools Foundation Namespaces ►  
 Microsoft.Deployment.WindowsInstaller ► Database ►  
**ExecuteScalar(String, Object[])**

C# ▾

Executes the specified SQL SELECT query and returns a single result.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public Object ExecuteScalar(
    string sqlFormat,
    params Object[] args
)
```

```
Public Function ExecuteScalar ( _
    sqlFormat As String, _
    ParamArray args As Object() _
) As Object
```

```
public:
Object^ ExecuteScalar(
    String^ sqlFormat,
    ... array<Object^>^ args
)
```

## Parameters

### **sqlFormat (String)**

SQL query string, which may contain format items

### **args ( Object [] )**

Zero or more objects to format

## Return Value

First field of the first result

## **Remarks**

The *sqlFormat* parameter is formatted using **Format(String, Object[])**.

Win32 MSI APIs: [MsiDatabaseOpenView](#), [MsiViewExecute](#),  
[MsiViewFetch](#)

## **Exceptions**

Exception	Condition
<a href="#">BadQuerySyntaxException</a>	the SQL syntax is invalid
<a href="#">InstallerException</a>	the View could not be executed or the query returned 0 results
<a href="#">InvalidHandleException</a>	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ExecuteStringQuery Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ►

**ExecuteStringQuery()**

C# ▾

### ▀ Members

Icon	Member	Description
≡	<a href="#">ExecuteStringQuery(String, Object[])</a>	Executes the specified SQL SELECT query and returns all results as strings.
≡	<a href="#">ExecuteStringQuery(String, Record)</a>	Executes the specified SQL SELECT query and returns all results as strings.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ExecuteStringQuery Method (sql, record)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ►  
**ExecuteStringQuery(String, Record)**

Executes the specified SQL SELECT query and returns all results as strings.

C#

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public IList<string> ExecuteStringQuery(  
    string sql,  
    Record record  
)
```

```
Public Function ExecuteStringQuery ( _  
    sql As String, _  
    record As Record _  
) As IList(of String)
```

```
public:  
IList<String^>^ ExecuteStringQuery(  
    String^ sql,  
    Record^ record  
)
```

### Parameters

#### sql (String)

SQL SELECT query string

#### record (Record)

Optional Record object containing the values that replace the parameter tokens (?) in the SQL query.

## **▀ Return Value**

All results combined into an array

## **▀ Remarks**

Multiple rows columns will be collapsed into a single on-dimensional list

Win32 MSI APIs: [MsiDatabaseOpenView](#), [MsiViewExecute](#),  
[MsiViewFetch](#)

## **▀ Exceptions**

Exception	Condition
<a href="#">BadQuerySyntaxException</a>	the SQL syntax is invalid
<a href="#">InstallerException</a>	the View could not be executed
<a href="#">InvalidHandleException</a>	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ExecuteStringQuery Method (sqlFormat, args)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ►  
**ExecuteStringQuery(String, Object[])**

C# ▾

Executes the specified SQL SELECT query and returns all results as strings.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public IList<string> ExecuteStringQuery(  
    string sqlFormat,  
    params Object[] args  
)
```

```
Public Function ExecuteStringQuery ( _  
    sqlFormat As String, _  
    ParamArray args As Object() _  
) As IList(Of String)
```

```
public:  
IList<String^>^ ExecuteStringQuery(  
    String^ sqlFormat,  
    ... array<Object^>^ args  
)
```

### Parameters

#### sqlFormat (String)

SQL query string, which may contain format items

#### args ( Object [] )

Zero or more objects to format

### Return Value

All results combined into an array

## [-] Remarks

The `sqlFormat` parameter is formatted using **Format(String, Object[])**.

Multiple rows columns will be collapsed into a single on-dimensional list

Win32 MSI APIs: [MsiDatabaseOpenView](#), [MsiViewExecute](#),  
[MsiViewFetch](#)

## [-] Exceptions

Exception	Condition
<a href="#">BadQuerySyntaxException</a>	the SQL syntax is invalid
<a href="#">InstallerException</a>	the View could not be executed
<a href="#">InvalidHandleException</a>	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Export Method (table, exportFilePath)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ► Export(String, String)

C# ▾

Copies the structure and data from a specified table to a text archive file

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Export(  
    string table,  
    string exportFilePath  
)
```

```
Public Sub Export ( _  
    table As String, _  
    exportFilePath As String _  
)
```

```
public:  
void Export(  
    String^ table,  
    String^ exportFilePath  
)
```

### Parameters

#### table (String)

Name of the table to be exported

#### exportFilePath (String)

Path to the file to be created

### Remarks

Win32 MSI API: [MsiDatabaseExport](#)

## **Exceptions**

<b>Exception</b>	<b>Condition</b>
<a href="#">FileNotFoundException</a>	the file path is invalid
<a href="#">InvalidHandleException</a>	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation ExportAll Method (directoryPath)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ►

### ExportAll(String)

C#

Exports all database tables, streams, and summary information to archive files.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void ExportAll(  
    string directoryPath  
)
```

```
Public Sub ExportAll ( _  
    directoryPath As String _  
)
```

```
public:  
void ExportAll(  
    String^ directoryPath  
)
```

#### Parameters

##### directoryPath (String)

Path to the directory where archive files will be created

#### Remarks

The directory will be created if it does not already exist.

Win32 MSI API: [MsiDatabaseExport](#)

#### Exceptions

Exception

Condition

FileNotFoundException	the directory path is invalid
InvalidHandleException	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **FilePath** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ► FilePath

C# ▾

Gets the file path the Database was originally opened from, or null if not known.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string FilePath { get; }
```

```
Public ReadOnly Property FilePath As String
```

```
public:  
property String^ FilePath {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## FromHandle Method (handle, ownsHandle)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ►  
**FromHandle(IntPtr, Boolean)**

C# ▾

Creates a new Database object from an integer database handle.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static Database FromHandle(  
    IntPtr handle,  
    bool ownsHandle  
)
```

```
Public Shared Function FromHandle ( _  
    handle As IntPtr, _  
    ownsHandle As Boolean _  
) As Database
```

```
public:  
static Database^ FromHandle(  
    IntPtr handle,  
    bool ownsHandle  
)
```

### Parameters

#### **handle (IntPtr)**

Integer database handle

#### **ownsHandle (Boolean)**

true to close the handle when this object is disposed

### Remarks

This method is only provided for interop purposes. A Database object should normally be obtained from [Database](#) or a public Database constructor.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# GenerateTransform Method (referenceDatabase, transformFile)

Deployment Tools Foundation Namespaces ►  
 Microsoft.Deployment.WindowsInstaller ► Database ►  
**GenerateTransform(Database, String)**

C# ▾

Creates a transform that, when applied to the object database, results in the reference database.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool GenerateTransform(
    Database referenceDatabase,
    string transformFile
)
```

```
Public Function GenerateTransform ( _
    referenceDatabase As Database, _
    transformFile As String _
) As Boolean
```

```
public:
bool GenerateTransform(
    Database^ referenceDatabase,
    String^ transformFile
)
```

## Parameters

### **referenceDatabase (Database)**

Database that does not include the changes

### **transformFile (String)**

Name of the generated transform file, or null to only check whether or not the two database are identical

## ■ Return Value

true if a transform is generated, or false if a transform is not generated because there are no differences between the two databases.

## ■ Remarks

A transform can add non-primary key columns to the end of a table. A transform cannot be created that adds primary key columns to a table. A transform cannot be created that changes the order, names, or definitions of columns.

If the transform is to be applied during an installation you must use the [CreateTransformSummaryInfo\(Database, String, TransformErrors, TransformValidations\)](#) method to populate the summary information stream.

Win32 MSI API: [MsiDatabaseGenerateTransform](#)

## ■ Exceptions

Exception	Condition
<a href="#">InstallerException</a>	the transform could not be generated
<a href="#">InvalidHandleException</a>	a Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Import Method (importFilePath)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ► Import(String)

C#

Imports a database table from a text archive file, dropping any existing table.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Import(  
    string importFilePath  
)
```

```
Public Sub Import (  
    importFilePath As String  
)
```

```
public:  
void Import(  
    String^ importFilePath  
)
```

### Parameters

#### importFilePath (String)

Path to the file to be imported. The table name is specified within the file.

### Remarks

Win32 MSI API: [MsiDatabaseImport](#)

### Exceptions

Exception	Condition
FileNotFoundException	the file path is invalid

[InvalidHandleException](#)

the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation ImportAll Method (directoryPath)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ►

### ImportAll(String)

C#

Imports all database tables, streams, and summary information from archive files.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void ImportAll(  
    string directoryPath  
)
```

```
Public Sub ImportAll ( _  
    directoryPath As String _  
)
```

```
public:  
void ImportAll(  
    String^ directoryPath  
)
```

#### Parameters

##### directoryPath (String)

Path to the directory from which archive files will be imported

#### Remarks

Win32 MSI API: [MsiDatabaseImport](#)

#### Exceptions

Exception	Condition
FileNotFoundException	the directory path is invalid

[InvalidHandleException](#)

the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# IsColumnPersistent Method (table, column)

Deployment Tools Foundation Namespaces ►  
 Microsoft.Deployment.WindowsInstaller ► Database ►  
**IsColumnPersistent(String, String)**

C# ▾

Checks whether a table contains a persistent column with a given name

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool IsColumnPersistent(
    string table,
    string column
)
```

```
Public Function IsColumnPersistent ( _
    table As String, _
    column As String _
) As Boolean
```

```
public:
bool IsColumnPersistent(
    String^ table,
    String^ column
)
```

## Parameters

### **table (String)**

The table to the checked

### **column (String)**

The name of the column to be checked

## Return Value

true if the column exists in the table; false if the column is temporary or

does not exist.

## [-] Remarks

To check whether a column exists regardless of persistence, use [Contains\(String\)](#).

## [-] Exceptions

Exception	Condition
<a href="#">InstallerException</a>	the View could not be executed
<a href="#">InvalidHandleException</a>	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation IsReadOnly Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ► IsReadOnly

C#

Gets a boolean value indicating whether this database was opened in read-only mode.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public bool IsReadOnly { get; }
```

```
Public ReadOnly Property IsReadOnly As Boolean
```

```
public:  
property bool IsReadOnly {  
    bool get ();  
}
```

## Remarks

Win32 MSI API: [MsiGetDatabaseState](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **IsTablePersistent Method (table)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ►

### **IsTablePersistent(String)**

C# ▾

Checks whether a table exists and is persistent in the database.

#### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool IsTablePersistent(  
    string table  
)
```

```
Public Function IsTablePersistent ( _  
    table As String _  
) As Boolean
```

```
public:  
bool IsTablePersistent(  
    String^ table  
)
```

#### [-] Parameters

##### **table (String)**

The table to be checked

#### [-] Return Value

true if the table exists and is persistent in the database; false otherwise

#### [-] Remarks

To check whether a table exists regardless of persistence, use [Contains\(String\)](#).

Win32 MSI API: [MsiDatabasesIsTablePersistent](#)

## Exceptions

Exception	Condition
ArgumentException	the table is unknown
InvalidHandleException	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## IsTransformValid Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ►

**IsTransformValid()**

C# ▾

### ▀ Members

Icon	Member	Description
≡	<a href="#">IsTransformValid(String)</a>	Checks whether a transform is valid for this Database, according to its validation data and flags.
≡	<a href="#">IsTransformValid(SummaryInfo)</a>	Checks whether a transform is valid for this Database, according to its SummaryInfo data.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## IsTransformValid Method (transformSummaryInfo)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ►  
**IsTransformValid(SummaryInfo)**

C# ▾

Checks whether a transform is valid for this Database, according to its SummaryInfo data.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool IsTransformValid(  
    SummaryInfo transformSummaryInfo  
)
```

```
Public Function IsTransformValid ( _  
    transformSummaryInfo As SummaryInfo _  
) As Boolean
```

```
public:  
bool IsTransformValid(  
    SummaryInfo^ transformSummaryInfo  
)
```

### - Parameters

#### **transformSummaryInfo (SummaryInfo)**

SummaryInfo data of a transform file

### - Return Value

true if the transform can be validly applied to this Database; false otherwise

### - Exceptions

Exception

Condition

InstallerException	error processing summary info
InvalidHandleException	the Database or SummaryInfo handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## IsTransformValid Method (transformFile)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ►  
**IsTransformValid(String)**

C# ▾

Checks whether a transform is valid for this Database, according to its validation data and flags.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool IsTransformValid(  
    string transformFile  
)
```

```
Public Function IsTransformValid ( _  
    transformFile As String _  
) As Boolean
```

```
public:  
bool IsTransformValid(  
    String^ transformFile  
)
```

### - Parameters

#### **transformFile (String)**

Path to the transform file

### - Return Value

true if the transform can be validly applied to this Database; false otherwise

### - Exceptions

Exception

Condition

InstallerException	the transform could not be applied
InvalidHandleException	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Merge Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ► Merge()

C# ▾

### Members

Icon	Member	Description
	<a href="#">Merge(Database, String)</a>	Merges another database with this database.
	<a href="#">Merge(Database)</a>	Merges another database with this database.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Merge Method (otherDatabase)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ►

### Merge(Database)

C# ▾

Merges another database with this database.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Merge(  
    Database otherDatabase  
)
```

```
Public Sub Merge ( _  
    otherDatabase As Database _  
)
```

```
public:  
void Merge(  
    Database^ otherDatabase  
)
```

#### Parameters

##### *otherDatabase* (Database)

The database to be merged into this database

#### Remarks

MsiDatabaseMerge does not copy over embedded cabinet files or embedded transforms from the reference database into the target database. Embedded data streams that are listed in the Binary table or Icon table are copied from the reference database to the target database. Storage embedded in the reference database are not copied to the target database.

The Merge method merges the data of two databases. These database must have the same codepage. The merge fails if any tables or rows in the databases conflict. A conflict exists if the data in any row in the first database differs from the data in the corresponding row of the second database. Corresponding rows are in the same table of both databases and have the same primary key in both databases. The tables of non-conflicting databases must have the same number of primary keys, same number of columns, same column types, same column names, and the same data in rows with identical primary keys. Temporary columns however don't matter in the column count and corresponding tables can have a different number of temporary columns without creating conflict as long as the persistent columns match.

If the number, type, or name of columns in corresponding tables are different, the schema of the two databases are incompatible and the installer will stop processing tables and the merge fails. The installer checks that the two databases have the same schema before checking for row merge conflicts. If the schemas are incompatible, the databases have be modified.

Win32 MSI API: [MsiDatabaseMerge](#)

## - Exceptions

Exception	Condition
<a href="#">MergeException</a>	merge failed due to a schema difference or data conflict
<a href="#">InvalidHandleException</a>	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Merge Method (otherDatabase, errorTable)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ►  
**Merge(Database, String)**

C# ▾

Merges another database with this database.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Merge(  
    Database otherDatabase,  
    string errorTable  
)
```

```
Public Sub Merge ( _  
    otherDatabase As Database, _  
    errorTable As String _  
)
```

```
public:  
void Merge(  
    Database^ otherDatabase,  
    String^ errorTable  
)
```

### Parameters

#### **otherDatabase (Database)**

The database to be merged into this database

#### **errorTable (String)**

Optional name of table to contain the names of the tables containing merge conflicts, the number of conflicting rows within the table, and a reference to the table with the merge conflict.

## **- Remarks**

Merge does not copy over embedded cabinet files or embedded transforms from the reference database into the target database. Embedded data streams that are listed in the Binary table or Icon table are copied from the reference database to the target database. Storage embedded in the reference database are not copied to the target database.

The Merge method merges the data of two databases. These database must have the same codepage. The merge fails if any tables or rows in the databases conflict. A conflict exists if the data in any row in the first database differs from the data in the corresponding row of the second database. Corresponding rows are in the same table of both databases and have the same primary key in both databases. The tables of non-conflicting databases must have the same number of primary keys, same number of columns, same column types, same column names, and the same data in rows with identical primary keys. Temporary columns however don't matter in the column count and corresponding tables can have a different number of temporary columns without creating conflict as long as the persistent columns match.

If the number, type, or name of columns in corresponding tables are different, the schema of the two databases are incompatible and the installer will stop processing tables and the merge fails. The installer checks that the two databases have the same schema before checking for row merge conflicts. If the schemas are incompatible, the databases have to be modified.

If the data in particular rows differ, this is a row merge conflict, the merge fails and creates a new table with the specified name. The first column of this table is the name of the table having the conflict. The second column gives the number of rows in the table having the conflict.

Win32 MSI API: [MsiDatabaseMerge](#)

## **- Exceptions**

Exception	Condition

MergeException	merge failed due to a schema difference or data conflict
InvalidHandleException	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **OpenMode Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ► OpenMode

C# ▾

Gets the open mode for the database.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public DatabaseOpenMode OpenMode { get; }
```

```
Public Readonly Property OpenMode As DatabaseOpenMod
```

```
public:  
property DatabaseOpenMode OpenMode {  
    DatabaseOpenMode get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## OpenView Method (sqlFormat, args)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ►  
**OpenView(String, Object[])**

Gets a View object representing the query specified by a SQL string.

C#

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public View OpenView(  
    string sqlFormat,  
    params Object[] args  
)
```

```
Public Function OpenView ( _  
    sqlFormat As String, _  
    ParamArray args As Object() _  
) As View
```

```
public:  
View^ OpenView(  
    String^ sqlFormat,  
    ... array<Object^>^ args  
)
```

### Parameters

#### sqlFormat (String)

SQL query string, which may contain format items

#### args ( Object [] )

Zero or more objects to format

### Return Value

A View object representing the query specified by a SQL string

## ■ Remarks

The *sqlFormat* parameter is formatted using **Format(String, Object[])**.

The View object should be [Close\(\)](#)d after use. It is best that the handle is closed manually as soon as it is no longer needed, as leaving lots of unused handles open can degrade performance.

Win32 MSI API: [MsiDatabaseOpenView](#)

## ■ Exceptions

Exception	Condition
<a href="#">BadQuerySyntaxException</a>	the SQL syntax is invalid
<a href="#">InvalidHandleException</a>	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **SummaryInfo Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ► SummaryInfo

C# ▾

Gets the **SummaryInfo** object for this database that can be used to examine and modify properties to the summary information stream.

## [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public SummaryInfo SummaryInfo { get; }
```

```
Public ReadOnly Property SummaryInfo As SummaryInfo
```

```
public:  
property SummaryInfo^ SummaryInfo {  
    SummaryInfo^ get ();  
}
```

## [-] Remarks

The object returned from this property does not need to be explicitly persisted or closed. Any modifications will be automatically saved when the database is committed.

Win32 MSI API: [MsiGetSummaryInformation](#)

## [-] Exceptions

Exception	Condition
<a href="#">InvalidHandleException</a>	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Tables Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ► Tables

C# ▾

Gets the collection of tables in the Database.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public TableCollection Tables { get; }
```

```
Public ReadOnly Property Tables As TableCollection
```

```
public:  
property TableCollection^ Tables {  
    TableCollection^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **ToString** Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Database ► **ToString()**

C# ▾

Returns the file path of this database, or the handle value if a file path was not specified.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override string ToString()
```

```
Public Overrides Function ToString As String
```

```
public:  
virtual String^ ToString() override
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ViewTransform Method (transformFile)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Database ►  
ViewTransform(String)

C# ▾

Apply a transform to the database, recording the changes in the  
"\_TransformView" table.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void ViewTransform(  
    string transformFile  
)
```

```
Public Sub ViewTransform ( _  
    transformFile As String _  
)
```

```
public:  
void ViewTransform(  
    String^ transformFile  
)
```

### Parameters

#### transformFile (String)

Path to the transform file

### Remarks

Win32 MSI API: [MsiDatabaseApplyTransform](#)

### Exceptions

Exception

Condition

InstallerException	the transform could not be applied
InvalidHandleException	the Database handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## DatabaseOpenMode Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► DatabaseOpenMode

C# ▾

Specifies the open mode for a [Database](#).

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public enum DatabaseOpenMode
```

```
Public Enumeration DatabaseOpenMode
```

```
public enum class DatabaseOpenMode
```

### - Members

Member	Description
<b>ReadOnly</b>	Open a database read-only, no persistent changes.
<b>Transact</b>	Open a database read/write in transaction mode.
<b>Direct</b>	Open a database direct read/write without transaction.
<b>Create</b>	Create a new database, transact mode read/write.
<b>CreateDirect</b>	Create a new database, direct mode

read/write.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation DialogAttributes Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► DialogAttributes

C#

Defines flags for the Attributes column of the Dialog table.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
[FlagsAttribute]
public enum DialogAttributes
```

```
<FlagsAttribute> _
Public Enumeration DialogAttributes
```

```
[FlagsAttribute]
public enum class DialogAttributes
```

## Members

Member	Description
<b>Visible</b>	If this bit is set, the dialog is originally created as visible, otherwise it is hidden.
<b>Modal</b>	If this bit is set, the dialog box is modal, other dialogs of the same application cannot be put on top of it, and the dialog keeps the control while it is running.
<b>Minimize</b>	If this bit is set, the dialog box can be minimized. This bit is ignored for modal dialog boxes, which cannot be minimized

<b>SysModal</b>	If this style bit is set, the dialog box will stop all other applications and no other applications can take the focus.
<b>KeepModeless</b>	If this bit is set, the other dialogs stay alive when this dialog box is created.
<b>TrackDiskSpace</b>	If this bit is set, the dialog box periodically calls the installer. If the property changes it notifies the controls on the dialog.
<b>UseCustomPalette</b>	If this bit is set, the pictures on the dialog box are created with the custom palette (one per dialog received from the first control created).
<b>RightToLeftReadingOrder</b>	If this style bit is set the text in the dialog box is displayed in right-to-left-reading order.
<b>RightAligned</b>	If this style bit is set, the text is aligned on the right side of the dialog box.
<b>LeftScroll</b>	If this style bit is set, the scroll bar is located on the left side of the dialog box.
<b>Bidirectional</b>	This is a combination of the RightToLeftReadingOrder, RightAligned, and the LeftScroll dialog style bits.

<b>Error</b>	If this bit is set, the dialog box is an error dialog.
--------------	--

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

C# ▾

Defines a callback function that the installer calls for progress notifications and error messages.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public delegate MessageResult ExternalUIHandler(
    InstallMessage messageType,
    string message,
    MessageBoxButtons buttons,
    MessageBoxIcon icon,
    MessageBoxDefaultButton defaultButton
)
```

```
Public Delegate Function ExternalUIHandler ( _
    messageType As InstallMessage, _
    message As String, _
    buttons As MessageBoxButtons, _
    icon As MessageBoxIcon, _
    defaultButton As MessageBoxDefaultButton _
) As MessageResult
```

```
public delegate MessageResult ExternalUIHandler(
    InstallMessage messageType,
    String^ message,
    MessageBoxButtons buttons,
    MessageBoxIcon icon,
    MessageBoxDefaultButton defaultButton
)
```

## Parameters

**messageType** ([InstallMessage](#))

***message*** ([String](#))

***buttons*** ([MessageBoxButtons](#))

***icon*** ([MessageBoxIcon](#))

***defaultButton*** ([MessageBoxDefaultButton](#))

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# ExternalUIRecordHandler

## Delegate

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ExternalUIRecordHandler

C#

[MSI 3.1] Defines a callback function that the installer calls for record-based progress notification and error messages.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public delegate MessageResult ExternalUIRecordHandler
    InstallMessage messageType,
    Record messageRecord,
    MessageBoxButtons buttons,
    MessageBoxIcon icon,
    MessageBoxDefaultButton defaultButton
)
```

```
Public Delegate Function ExternalUIRecordHandler ( _
    messageType As InstallMessage, _
    messageRecord As Record, _
    buttons As MessageBoxButtons, _
    icon As MessageBoxIcon, _
    defaultButton As MessageBoxDefaultButton _
) As MessageResult
```

```
public delegate MessageResult ExternalUIRecordHandler
    InstallMessage messageType,
    Record^ messageRecord,
    MessageBoxButtons buttons,
    MessageBoxIcon icon,
    MessageBoxDefaultButton defaultButton
)
```

### Parameters

*messageType* ([InstallMessage](#))

*messageRecord* ([Record](#))

*buttons* ([MessageBoxButtons](#))

*icon* ([MessageBoxIcon](#))

*defaultButton* ([MessageBoxDefaultButton](#))

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

C# ▾

Available values for the Attributes column of the Feature table.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
[FlagsAttribute]
public enum FeatureAttributes
```

```
<FlagsAttribute> _
Public Enumeration FeatureAttributes
```

```
[FlagsAttribute]
public enum class FeatureAttributes
```

## Members

Member	Description
<b>None</b>	Favor local - Components of this feature that are not marked for installation from source are installed locally.
<b>FavorSource</b>	Components of this feature not marked for local installation are installed to run from the source CD-ROM or server.
<b>FollowParent</b>	Set this attribute and the state of the feature is the same as the state of the feature's parent. You cannot use this option if the feature is located at the root of a feature tree.

<b>FavorAdvertise</b>	Set this attribute and the feature state is Advertise.
<b>DisallowAdvertise</b>	Set this attribute to prevent the feature from being advertised.
<b>UIDisallowAbsent</b>	Set this attribute and the user interface does not display an option to change the feature state to Absent. Setting this attribute forces the feature to the installation state, whether or not the feature is visible in the UI.
<b>NoUnsupportedAdvertise</b>	Set this attribute and advertising is disabled for the feature if the operating system shell does not support Windows Installer descriptors.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **FeatureInfo Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FeatureInfo

C#

Provides access to information about a feature within the context of an installation session.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public class FeatureInfo
```

```
Public Class FeatureInfo
```

```
public ref class FeatureInfo
```

### Members

All Members	Methods	Properties	
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static	<input checked="" type="checkbox"/> Declared <input checked="" type="checkbox"/> Inherited

Icon	Member	Description
	<a href="#">Attributes</a>	Gets or sets the attributes of the feature.
	<a href="#">CurrentState</a>	Gets the current install state of the feature.
	<a href="#">Description</a>	Gets the description of the feature.

	<b>Equals(Object)</b>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
💡	<b>Finalize()</b>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
💡	<b>GetCost(Boolean, Boolean, InstallState)</b>	Calculates the disk space required by the feature and its selected children and parent features.
💡	<b>GetHashCode()</b>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.  (Inherited from <a href="#">Object</a> .)
💡	<b>GetType()</b>	Gets the <a href="#">Type</a> of the current instance  (Inherited from <a href="#">Object</a> .)
💡	<b>MemberwiseClone()</b>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
📝	<b>Name</b>	Gets the name of the feature (primary key in the Feature table).
📝	<b>RequestState</b>	Gets or sets the action state of the feature.

	<b>Title</b>	Gets the title of the feature.
	<b>ToString()</b>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<b>ValidStates</b>	Gets a list of valid installation states for the feature.

## [-] Inheritance Hierarchy

[Object](#)

└ **FeatureInfo**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Attributes Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FeatureInfo ► Attributes

C# ▾

Gets or sets the attributes of the feature.

## [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public FeatureAttributes Attributes { get; set; }
```

```
Public Property Attributes As FeatureAttributes
```

```
public:  
property FeatureAttributes Attributes {  
    FeatureAttributes get ();  
    void set (FeatureAttributes value);  
}
```

## [-] Remarks

Win32 MSI APIs: [MsiGetFeatureInfo](#), [MsiSetFeatureAttributes](#)

Since the `lpAttributes` parameter of [MsiGetFeatureInfo](#) does not contain an equivalent flag for [UIDDisallowAbsent](#), this flag will not be retrieved.

Since the `dwAttributes` parameter of [MsiSetFeatureAttributes](#) does not contain an equivalent flag for [UIDDisallowAbsent](#), the presence of this flag will be ignored.

## [-] Exceptions

Exception	Condition
<a href="#">InvalidOperationException</a>	the Session handle is invalid
<a href="#">ArgumentException</a>	an unknown feature was requested

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **CurrentState** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FeatureInfo ► **CurrentState**

C# ▾

Gets the current install state of the feature.

## [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public InstallState CurrentState { get; }
```

```
Public ReadOnly Property CurrentState As InstallStat
```

```
public:  
property InstallState CurrentState {  
    InstallState get ();  
}
```

## [-] Remarks

Win32 MSI API: [MsiGetFeatureState](#)

## [-] Exceptions

Exception	Condition
<a href="#">InvalidOperationException</a>	the Session handle is invalid
<a href="#">ArgumentException</a>	an unknown feature was requested

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Description Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FeatureInfo ► **Description**

C# ▾

Gets the description of the feature.

## [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string Description { get; }
```

```
Public Readonly Property Description As String
```

```
public:  
property String^ Description {  
    String^ get ();  
}
```

## [-] Remarks

Win32 MSI API: [MsiGetFeatureInfo](#)

## [-] Exceptions

Exception	Condition
<a href="#">InvalidHandleException</a>	the Session handle is invalid
<a href="#">ArgumentException</a>	an unknown feature was requested

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetCost Method (includeParents, includeChildren, installState)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FeatureInfo ►

**GetCost(Boolean, Boolean, InstallState)**

C# ▾

Calculates the disk space required by the feature and its selected children and parent features.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public long GetCost(  
    bool includeParents,  
    bool includeChildren,  
    InstallState installState  
)
```

```
Public Function GetCost ( _  
    includeParents As Boolean, _  
    includeChildren As Boolean, _  
    installState As InstallState _  
) As Long
```

```
public:  
long long GetCost(  
    bool includeParents,  
    bool includeChildren,  
    InstallState installState  
)
```

[-] **Parameters**

**includeParents (Boolean)**

If true, the parent features are included in the cost.

**includeChildren (Boolean)**

If true, the child features are included in the cost.

### ***installState* ([InstallState](#))**

Specifies the installation state.

#### **▀ Return Value**

The disk space requirement in bytes.

#### **▀ Remarks**

Win32 MSI API: [MsiGetFeatureCost](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Name Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FeatureInfo ► Name

C# ▾

Gets the name of the feature (primary key in the Feature table).

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string Name { get; }
```

```
Public Readonly Property Name As String
```

```
public:  
property String^ Name {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation RequestState Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FeatureInfo ►

## RequestState

C# ▾

Gets or sets the action state of the feature.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public InstallState RequestState { get; set; }
```

```
Public Property RequestState As InstallState
```

```
public:  
property InstallState RequestState {  
    InstallState get ();  
    void set (InstallState value);  
}
```

### Remarks

When changing the feature action, the action state of all the Components linked to the changed Feature records are also updated appropriately, based on the new feature Select state. All Features can be configured at once by specifying the keyword ALL instead of a specific feature name.

Win32 MSI APIs: [MsiGetFeatureState](#), [MsiSetFeatureState](#)

### Exceptions

Exception	Condition
<a href="#">InvalidHandleException</a>	the Session handle is invalid
<a href="#">ArgumentException</a>	an unknown feature was requested

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Title Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FeatureInfo ► **Title**

C# ▾

Gets the title of the feature.

## [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string Title { get; }
```

```
Public Readonly Property Title As String
```

```
public:  
property String^ Title {  
    String^ get ();  
}
```

## [-] Remarks

Win32 MSI API: [MsiGetFeatureInfo](#)

## [-] Exceptions

Exception	Condition
<a href="#">InvalidHandleException</a>	the Session handle is invalid
<a href="#">ArgumentException</a>	an unknown feature was requested

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **ValidStates** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FeatureInfo ► **ValidStates**

C# ▾

Gets a list of valid installation states for the feature.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public ICollection<InstallState> ValidStates { get;
```

```
Public ReadOnly Property ValidStates As ICollection(
```

```
public:  
property ICollection<InstallState>^ ValidStates {  
    ICollection<InstallState>^ get ();  
}
```

## Remarks

Win32 MSI API: [MsiGetFeatureValidStates](#)

## Exceptions

Exception	Condition
<a href="#">InvalidHandleException</a>	the Session handle is invalid
<a href="#">ArgumentException</a>	an unknown feature was requested

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation **FeatureInfoCollection Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FeatureInfoCollection

C#

Accessor for information about features within the context of an installation session.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public sealed class FeatureInfoCollection : ICollect
    I Enumerable<FeatureInfo>, I Enumerable
```

```
Public NotInheritable Class FeatureInfoCollection _
    Implements ICollection(Of FeatureInfo), IEnu
    I Enumerable
```

```
public ref class FeatureInfoCollection sealed : ICol
    I Enumerable<FeatureInfo^>, I Enumerable
```

## Members

All Members

Methods

Properties

Public

Protected

Instance

Static

Declare  
 Inherit

Icon	Member	Description
	<a href="#">Contains(String)</a>	Checks if the collection contains a feature.
	<a href="#">CopyTo(FeatureInfo[], Int32)</a>	Copies the features into an array.
	<a href="#">Count</a>	Gets the number of features defined

		for the product.
≡	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">GetEnumerator()</a>	Enumerates the features in the collection.
≡	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance  (Inherited from <a href="#">Object</a> .)
📋	<a href="#">Item[String]</a>	Gets information about a feature within the context of an installation session.
💡	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
≡		

	<b>ToString()</b>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
--	-------------------	--

## [-] Inheritance Hierarchy

[Object](#)

└ [FeatureInfoCollection](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation Contains Method (feature)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FeatureInfoCollection ►

## Contains(String)

C# ▾

Checks if the collection contains a feature.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public bool Contains(  
    string feature  
)
```

```
Public Function Contains ( _  
    feature As String _  
) As Boolean
```

```
public:  
bool Contains(  
    String^ feature  
)
```

### Parameters

#### ***feature* (String)**

name of the feature

### Return Value

true if the feature is in the collection, else false

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CopyTo Method (array, arrayIndex)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► FeatureInfoCollection ►  
**CopyTo(FeatureInfo[], Int32)**

C# ▾

Copies the features into an array.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void CopyTo(  
    FeatureInfo[] array,  
    int arrayIndex  
)
```

```
Public Sub CopyTo ( _  
    array As FeatureInfo(), _  
    arrayIndex As Integer _  
)
```

```
public:  
virtual void CopyTo(  
    array<FeatureInfo^>^ array,  
    int arrayIndex  
) sealed
```

### Parameters

#### array ( FeatureInfo [] )

array that receives the features

#### arrayIndex (Int32)

offset into the array

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Count Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FeatureInfoCollection ►

**Count**

C# ▾

Gets the number of features defined for the product.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public int Count { get; }
```

```
Public ReadOnly Property Count As Integer
```

```
public:  
virtual property int Count {  
    int get () sealed;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetEnumerator Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FeatureInfoCollection ►

**GetEnumerator()**

C# ▾

Enumerates the features in the collection.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public IEnum<FeatureInfo> GetEnumerator()
```

```
Public Function GetEnumerator As IEnum<of Feat
```

```
public:  
virtual IEnum<FeatureInfo^>^ GetEnumerator() s
```

[-] **Return Value**

an enumerator over all features in the collection

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation Item Property (feature)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FeatureInfoCollection ►

## Item[String]

C# ▾

Gets information about a feature within the context of an installation session.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public FeatureInfo this[  
    string feature  
] { get; }
```

```
Public ReadOnly Default Property Item ( _  
    feature As String _  
) As FeatureInfo
```

```
public:  
property FeatureInfo^ default[String^ feature] {  
    FeatureInfo^ get (String^ feature);  
}
```

### Parameters

#### feature (String)

name of the feature

### Return Value

feature object

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation FeatureInstallation Class

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FeatureInstallation

C#

Represents an instance of a feature of an installed product.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public class FeatureInstallation : InstallationPart
```

```
Public Class FeatureInstallation _  
    Inherits InstallationPart
```

```
public ref class FeatureInstallation : public Instal
```

## Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon

Icon	Member	Description
	<a href="#">FeatureInstallation(String, String)</a>	Creates a new FeatureInstallation instance for a feature of a product.
	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> (Inherited from <a href="#">Object</a> .)
	<a href="#">FeatureName</a>	Gets the name of the feature.

	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithm and data structures like a hash table  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Parent</a>	Gets the parent of the feature, or null if the feature has no parent (it is a root feature).
	<a href="#">Product</a>	Gets the product that this item is a part of.  (Inherited from <a href="#">InstallationPart</a> .)
	<a href="#">State</a>	Gets the installed state of the feature.  (Overrides <a href="#">InstallationPart.State</a> .)
	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)



## Usage

Gets the usage metrics for the feature.

## ■ Inheritance Hierarchy

Object

└ InstallationPart

  └ FeatureInstallation

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## FeatureInstallation Constructor (*featureName*, *productCode*)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► FeatureInstallation ►  
**FeatureInstallation(String, String)**

Creates a new FeatureInstallation instance for a feature of a product.

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public FeatureInstallation(  
    string featureName,  
    string productCode  
)
```

```
Public Sub New ( _  
    featureName As String, _  
    productCode As String _  
)
```

```
public:  
FeatureInstallation(  
    String^ featureName,  
    String^ productCode  
)
```

### Parameters

#### *featureName* ([String](#))

feature name

#### *productCode* ([String](#))

ProductCode GUID

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# FeatureName Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FeatureInstallation ►

## FeatureName

C# ▾

Gets the name of the feature.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string FeatureName { get; }
```

```
Public ReadOnly Property FeatureName As String
```

```
public:  
property String^ FeatureName {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Parent Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FeatureInstallation ►

### Parent

C# ▾

Gets the parent of the feature, or null if the feature has no parent (it is a root feature).

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public FeatureInstallation Parent { get; }
```

```
Public ReadOnly Property Parent As FeatureInstallati
```

```
public:  
property FeatureInstallation^ Parent {  
    FeatureInstallation^ get ();  
}
```

#### Remarks

Invocation of this property may be slightly costly for products with many features, because it involves an enumeration of all the features in the product.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation State Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FeatureInstallation ► State

C# ▾

Gets the installed state of the feature.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override InstallState State { get; }
```

```
Public Overrides ReadOnly Property State As InstallS
```

```
public:  
virtual property InstallState State {  
    InstallState get () override;  
}
```

## Remarks

Win32 MSI API: [MsiQueryFeatureState](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation Usage Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FeatureInstallation ► Usage

C# ▾

Gets the usage metrics for the feature.

## - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public FeatureInstallation.UsageData Usage { get; }
```

```
Public ReadOnly Property Usage As FeatureInstallatio
```

```
public:  
property FeatureInstallation.UsageData Usage {  
    FeatureInstallation.UsageData get ();  
}
```

## - Remarks

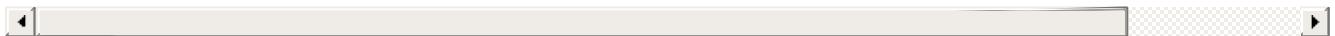
If no usage metrics are recorded, the [Usage](#) value is 0.

Win32 MSI API: [MsiGetFeatureUsage](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

# FeatureInstallation.UsageData

## Structure

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► FeatureInstallation ► FeatureInstallation.UsageData

Holds data about the usage of a feature.

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public struct UsageData
```

```
Public Structure UsageData
```

```
public value class UsageData
```

### Members

All Members

Methods

Properties

Public

Protected

Instance

Static



Declared



Inherited

Icon	Member	Description
	<a href="#">Equals(Object)</a>	Indicates whether this instance and a specified object are equal.  (Inherited from <a href="#">ValueType</a> .)
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)

	<a href="#">GetHashCode()</a>	Returns the hash code for this instance.  (Inherited from <a href="#">ValueType</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance  (Inherited from <a href="#">Object</a> .)
	<a href="#">LastUsedDate</a>	Gets the date the feature was last used.
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">ToString()</a>	Returns the fully qualified type name of this instance.  (Inherited from <a href="#">ValueType</a> .)
	<a href="#">UseCount</a>	Gets count of the number of times the feature has been used.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)  
Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## LastUsedDate Property

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ►  
FeatureInstallation.UsageData ► **LastUsedDate**

C# ▾

Gets the date the feature was last used.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public DateTime LastUsedDate { get; }
```

```
Public ReadOnly Property LastUsedDate As DateTime
```

```
public:  
property DateTime LastUsedDate {  
    DateTime get();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## UseCount Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ►

FeatureInstallation.UsageData ► **UseCount**

C# ▾

Gets count of the number of times the feature has been used.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public int UseCount { get; }
```

```
Public ReadOnly Property UseCount As Integer
```

```
public:  
property int UseCount {  
    int get();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation FileAttributes Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► FileAttributes

C#

Available values for the Attributes column of the File table.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
[FlagsAttribute]
public enum FileAttributes
```

```
<FlagsAttribute> _
Public Enumeration FileAttributes
```

```
[FlagsAttribute]
public enum class FileAttributes
```

## Members

Member	Description
<b>None</b>	No attributes.
<b>ReadOnly</b>	Read-only.
<b>Hidden</b>	Hidden.
<b>System</b>	System.
<b>Vital</b>	The file is vital for the proper operation of the component to which it belongs.

<b>Checksum</b>	The file contains a valid checksum. A checksum is required to repair a file that has become corrupted.
<b>PatchAdded</b>	This bit must only be added by a patch and if the file is being added by the patch
<b>NonCompressed</b>	The file's source type is uncompressed. If set, ignore the WordCount summary information property. If neither Noncompressed nor Compressed are set, the compression state of the file is specified by the WordCount summary information property. Do not set both Noncompressed and Compressed.
<b>Compressed</b>	The file's source type is compressed. If set, ignore the WordCount summary information property. If neither Noncompressed or Compressed are set, the compression state of the file is specified by the WordCount summary information property. Do not set both Noncompressed and Compressed.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation IEmbeddedUI Interface

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► IEmbeddedUI

C# ▾

[MSI 4.5] Interface for an embedded external user interface for an installation.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public interface IEmbeddedUI
```

```
Public Interface IEmbeddedUI
```

```
public interface class IEmbeddedUI
```

### Members

All Members

Methods

Public

Instance

Declared

Protected

Static

Inherited

Icon	Member	Description
≡	<a href="#">Initialize(Session, String, InstallUIOptions)</a>	Initializes the embedded UI.
≡	<a href="#">ProcessMessage(InstallMessage, Record, MessageBoxButtons, MessageBoxIcon, MessageBoxDefaultButton)</a>	Processes information and progress messages sent to the user interface.
≡	<a href="#">Shutdown()</a>	Shuts down the embedded UI at the end of the installation.

## Remarks

Classes which implement this interface must have a public constructor that takes no parameters.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Initialize Method (session, resourcePath, internalUILevel)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► IEmbeddedUI ►

**Initialize(Session, String, InstallUIOptions)**

C# ▾

Initializes the embedded UI.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
bool Initialize(  
    Session session,  
    string resourcePath,  
    ref InstallUIOptions internalUILevel  
)
```

```
Function Initialize ( _  
    session As Session, _  
    resourcePath As String, _  
    ByRef internalUILevel As InstallUIOptions _  
) As Boolean
```

```
bool Initialize(  
    Session^ session,  
    String^ resourcePath,  
    InstallUIOptions% internalUILevel  
)
```

### Parameters

#### **session (Session)**

Handle to the installer which can be used to get and set properties.

The handle is only valid for the duration of this method call.

#### **resourcePath (String)**

Path to the directory that contains all the files from the

MsiEmbeddedUI table.

### ***internalUILevel ( InstallUIOptions )***

On entry, contains the current UI level for the installation. After this method returns, the installer resets the UI level to the returned value of this parameter.

#### **[-] Return Value**

True if the embedded UI was successfully initialized; false if the installation should continue without the embedded UI.

#### **[-] Remarks**

Win32 MSI API: [InitializeEmbeddedUI](#)

#### **[-] Exceptions**

Exception	Condition
<a href="#">InstallCanceledException</a>	The installation was canceled by the user
<a href="#">InstallerException</a>	The embedded UI failed to initialize and causes the installation to fail.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ProcessMessage Method (messageType, messageRecord, buttons, icon, defaultButton)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► IEmbeddedUI ►

**ProcessMessage(InstallMessage, Record, MessageBoxButtons, MessageBoxIcon, MessageBoxDefaultButton)**

C#

Processes information and progress messages sent to the user interface

### Declaration Syntax

C#

Visual Basic

Visual C++

```
MessageResult ProcessMessage(  
    InstallMessage messageType,  
    Record messageRecord,  
    MessageBoxButtons buttons,  
    MessageBoxIcon icon,  
    MessageBoxDefaultButton defaultButton  
)
```

```
Function ProcessMessage ( _  
    messageType As InstallMessage, _  
    messageRecord As Record, _  
    buttons As MessageBoxButtons, _  
    icon As MessageBoxIcon, _  
    defaultButton As MessageBoxDefaultButton _  
) As MessageResult
```

```
MessageResult ProcessMessage(  
    InstallMessage messageType,  
    Record^ messageRecord,  
    MessageBoxButtons buttons,  
    MessageBoxIcon icon,  
    MessageBoxDefaultButton defaultButton  
)
```

## **[-] Parameters**

### ***messageType* ([InstallMessage](#))**

Message type.

### ***messageRecord* ([Record](#))**

Record that contains message data.

### ***buttons* ([MessageBoxButtons](#))**

Message box buttons.

### ***icon* ([MessageBoxIcon](#))**

Message box icon.

### ***defaultButton* ([MessageBoxDefaultButton](#))**

Message box default button.

## **[-] Return Value**

Result of processing the message.

## **[-] Remarks**

Win32 MSI API: [EmbeddedUIHandler](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Shutdown Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► IEmbeddedUI ►

### Shutdown()

C#

Shuts down the embedded UI at the end of the installation.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
void Shutdown()
```

```
Sub Shutdown
```

```
void Shutdown()
```

#### Remarks

If the installation was canceled during initialization, this method will not be called. If the installation was canceled or failed at any later point, this method will be called at the end.

Win32 MSI API: [ShutdownEmbeddedUI](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# InapplicablePatchHandler

## Delegate

Deployment Tools Foundation Namespaces ►  
 Microsoft.Deployment.WindowsInstaller ► InapplicablePatchHandler

C# ▾

Receives an exception from [DetermineApplicablePatches\(String, String, InapplicablePatchHandler, String, UserContexts\)](#) indicating the reason a particular patch is not applicable to a product.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public delegate void InapplicablePatchHandler(
    string patch,
    Exception exception
)
```

```
Public Delegate Sub InapplicablePatchHandler ( _
    patch As String, _
    exception As Exception _
)
```

```
public delegate void InapplicablePatchHandler(
    String^ patch,
    Exception^ exception
)
```

### Parameters

#### ***patch (String)***

MSP file path, XML file path, or XML blob that was passed to [DetermineApplicablePatches\(String, String\[\], InapplicablePatchHandler, String, UserContexts\)](#)

#### ***exception (Exception)***

exception indicating the reason the patch is not applicable

## Remarks

If *exception* is an [InstallerException](#) or subclass, then its [ErrorCode](#) and [Message](#) properties will indicate a more specific reason the patch was not applicable.

The *exception* could also be a [FileNotFoundException](#) if the patch string was a file path.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## IniFileAction Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► IniFileAction

C#

Defines values for the Action column of the IniFile and RemoveIniFile tables.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public enum IniFileAction
```

```
Public Enumeration IniFileAction
```

```
public enum class IniFileAction
```

### Members

Member	Description
AddLine	Creates or updates a .ini entry.
CreateLine	Creates a .ini entry only if the entry does not already exist.
RemoveLine	Deletes .ini entry.
AddTag	Creates a new entry or appends a new comma-separated value to an existing entry.
RemoveTag	Deletes a tag from a .ini entry.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Installation Class

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installation

C# ▾

Subclasses of this abstract class represent a unique instance of a registered product or patch installation.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public abstract class Installation
```

```
Public MustInherit Class Installation
```

```
public ref class Installation abstract
```

### Members

All Members	Methods	Properties	Events
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static	<input checked="" type="checkbox"/> Declared <input checked="" type="checkbox"/> Inherited

Icon	Member	Description
	Context	Gets the user context of this product or patch installation.
	Equals(Object)	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> . (Inherited from <a href="#">Object</a> .)
	Finalize()	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.

		(Inherited from <a href="#">Object</a> .)
	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.
		(Inherited from <a href="#">Object</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance
		(Inherited from <a href="#">Object</a> .)
	<a href="#">IsInstalled</a>	Gets a value indicating whether this product or patch is installed on the current system.
	<a href="#">Item[String]</a>	Gets a property about the product or patch installation.
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .
		(Inherited from <a href="#">Object</a> .)
	<a href="#">SourceList</a>	Gets the source list of this product or patch installation.
	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .
		(Inherited from <a href="#">Object</a> .)
	<a href="#">UserSid</a>	Gets the user security identifier (SID) under which this product or patch installation is available.

## Inheritance Hierarchy

[Object](#)

 [Installation](#)

 [PatchInstallation](#)

 [ProductInstallation](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Context Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installation ► Context

C# ▾

Gets the user context of this product or patch installation.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public UserContexts Context { get; }
```

```
Public ReadOnly Property Context As UserContexts
```

```
public:  
property UserContexts Context {  
    UserContexts get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation IsInstalled Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installation ► IsInstalled

C# ▾

Gets a value indicating whether this product or patch is installed on the current system.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public abstract bool IsInstalled { get; }
```

```
Public MustOverride ReadOnly Property IsInstalled As
```

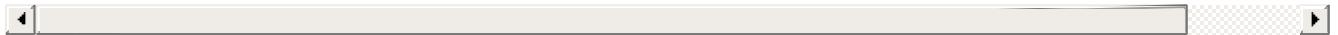
```
public:  
virtual property bool IsInstalled {  
    bool get () abstract;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation Item Property (propertyName)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installation ► Item[String]

C# ▾

Gets a property about the product or patch installation.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public abstract string this[  
    string propertyName  
] { get; }
```

```
Public MustOverride ReadOnly Default Property Item (  
    propertyName As String _  
) As String
```

```
public:  
virtual property String^ default[String^ propertyName]  
String^ get (String^ propertyName) abstract;  
}
```

## Parameters

### propertyName (String)

Name of the property being retrieved.

## Return Value

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

# SourceList Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installation ► SourceList

C# ▾

Gets the source list of this product or patch installation.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public virtual SourceList SourceList { get; }
```

```
Public Overridable ReadOnly Property SourceList As S
```

```
public:  
virtual property SourceList^ SourceList {  
    SourceList^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



## Deployment Tools Foundation **UserSid Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installation ► **UserSid**

C# ▾

Gets the user security identifier (SID) under which this product or patch installation is available.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string UserSid { get; }
```

```
Public ReadOnly Property UserSid As String
```

```
public:  
property String^ UserSid {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **InstallationPart Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallationPart

C# ▾

Subclasses of this abstract class represent an instance of a registered feature or component.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public abstract class InstallationPart
```

```
Public MustInherit Class InstallationPart
```

```
public ref class InstallationPart abstract
```

### Members

All Members	Methods	Properties	Events
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static	<input checked="" type="checkbox"/> Declared <input checked="" type="checkbox"/> Inherited

Icon	Member	Description
≡	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is

		suitable for use in hashing algorithms and data structures like a hash table.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance  (Inherited from <a href="#">Object</a> .)
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Product</a>	Gets the product that this item is a part of.
	<a href="#">State</a>	Gets the current installation state of the item.
	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)

## Inheritance Hierarchy

[Object](#)

  └ [InstallationPart](#)

- └ [ComponentInstallation](#)
- └ [FeatureInstallation](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# Product Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallationPart ► Product

C# ▾

Gets the product that this item is a part of.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public ProductInstallation Product { get; }
```

```
Public ReadOnly Property Product As ProductInstallation
```

```
public:  
property ProductInstallation^ Product {  
    ProductInstallation^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation State Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallationPart ► **State**

C# ▾

Gets the current installation state of the item.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public abstract InstallState State { get; }
```

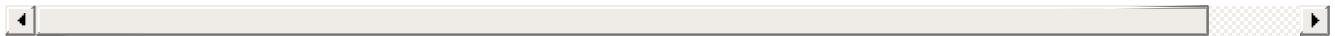
```
Public MustOverride ReadOnly Property State As Insta
```

```
public:  
virtual property InstallState State {  
    InstallState get () abstract;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation **InstallCanceledException** Class

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► **InstallCanceledException**

C# ▾

User Canceled the installation.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
[SerializableAttribute]
public class InstallCanceledException : InstallerExc
```

```
<SerializableAttribute> _
Public Class InstallCanceledException _
    Inherits InstallerException
```

```
[SerializableAttribute]
public ref class InstallCanceledException : public I
```

## Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon	Member	Description
≡	InstallCanceledException(String, Exception)	Creates a new <b>InstallCanceledException</b> object with a specified error message and a reference to the inner exception that is the cause of this exception.

 	<a href="#">InstallCanceledException(String)</a>	Creates a new <code>InstallCanceledException</code> with a specified error message.
 	<a href="#">InstallCanceledException()</a>	Creates a new <code>InstallCanceledException</code> .
 	<a href="#">InstallCanceledException(SerializationInfo, StreamingContext)</a>	Initializes a new instance of the <code>InstallCanceledException</code> class with serialized state information.
 	<a href="#">Data</a>	Gets a collection of key/value pairs that provide additional, user-defined information about the exception.  (Inherited from <a href="#">Exception</a> .)
 	<a href="#">Equals(Object)</a>	Determines whether the specified <code>Object</code> is equal to the current <code>Object</code> .  (Inherited from <code>Object</code> )
 	<a href="#">ErrorCode</a>	Gets the system error code that resulted in the exception, or 0 if not applicable.  (Inherited from <a href="#">InstallerException</a> .)
 	<a href="#">Finalize()</a>	Allows an <code>Object</code> to

		attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.
		(Inherited from <a href="#">Object</a> )
≡	<a href="#">GetBaseException()</a>	When overridden in a derived class, returns an <a href="#">Exception</a> that is the cause of one or more subsequent exceptions.
		(Inherited from <a href="#">Exception</a> .)
≡	<a href="#">GetErrorRecord()</a>	Gets extended information about the error, or null if no further information is available.
		(Inherited from <a href="#">InstallerException</a> .)
≡	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.
		(Inherited from <a href="#">Object</a> )
≡	<a href="#">GetObjectData(SerializationInfo, StreamingContext)</a>	Sets the <a href="#">SerializationInfo</a> object with information about the exception.
		(Inherited from <a href="#">Object</a> )

		<code>InstallerException.)</code>
	<a href="#">GetType()</a>	Gets the runtime type of the current instance.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">HelpLink</a>	Gets or sets a link to help file associated with this exception.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">HResult</a>	Gets or sets HRESULT coded numerical value that is assigned to a specific exception.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">InnerException</a>	Gets the <a href="#">Exception</a> instance that caused current exception.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> )
	<a href="#">Message</a>	Gets a message that describes the exception. This message may contain detailed form error data if it was available.  (Inherited from <a href="#">Object</a> )

		InstallerException.)
	Source	Gets or sets the name of the application or the object that causes the error.  (Inherited from <a href="#">Exception</a> .)
	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown.  (Inherited from <a href="#">Exception</a> .)
	TargetSite	Gets the method that throws the current exception.  (Inherited from <a href="#">Exception</a> .)
	ToString()	Creates and returns a string representation of the current exception.  (Inherited from <a href="#">Exception</a> .)

## Inheritance Hierarchy

Object

- └ Exception
- └ SystemException
- └ InstallerException
- └ **InstallCanceledException**

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

# InstallCanceledException Constructor

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► InstallCanceledException ►  
**InstallCanceledException()**

C# ▾

## ▀ Members

Icon	Member	Description
≡ ♦	<a href="#">InstallCanceledException(String, Exception)</a>	Creates a new <code>InstallCanceledException</code> object with a specified error message and a reference to the inner exception that is the cause of this exception.
≡ ♦	<a href="#">InstallCanceledException(String)</a>	Creates a new <code>InstallCanceledException</code> object with a specified error message.
≡ ♦	<a href="#">InstallCanceledException()</a>	Creates a new <code>InstallCanceledException</code> object.
≡ ♦	<a href="#">InstallCanceledException(SerializationInfo, StreamingContext)</a>	Initializes a new instance of the <code>InstallCanceledException</code> class with serialized state information.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

# InstallCanceledException Constructor

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► InstallCanceledException ►  
**InstallCanceledException()**

Creates a new InstallCanceledException.

C# ▾

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public InstallCanceledException()
```

```
Public Sub New
```

```
public:  
InstallCanceledException()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **InstallCanceledException**

## Constructor (info, context)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallCanceledException ►

**InstallCanceledException(SerializationInfo, StreamingContext)**

C#

Initializes a new instance of the `InstallCanceledException` class with serialized data.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
protected InstallCanceledException(  
    SerializationInfo info,  
    StreamingContext context  
)
```

```
Protected Sub New ( _  
    info As SerializationInfo, _  
    context As StreamingContext _  
)
```

```
protected:  
InstallCanceledException(  
    SerializationInfo^ info,  
    StreamingContext context  
)
```

### Parameters

#### **info (SerializationInfo)**

The `SerializationInfo` that holds the serialized object data about the exception being thrown.

#### **context (StreamingContext)**

The `StreamingContext` that contains contextual information about the source or destination.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **InstallCanceledException**

## Constructor (msg)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallCanceledException ►

### InstallCanceledException(String)

C#

Creates a new InstallCanceledException with a specified error message

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public InstallCanceledException(  
    string msg  
)
```

```
Public Sub New ( _  
    msg As String _  
)
```

```
public:  
InstallCanceledException(  
    String^ msg  
)
```

#### Parameters

##### *msg* (String)

The message that describes the error.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## InstallCanceledException Constructor (msg, innerException)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallCanceledException ►

InstallCanceledException(String, Exception)

C# ▾

Creates a new InstallCanceledException with a specified error message and a reference to the inner exception that is the cause of this exception.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public InstallCanceledException(  
    string msg,  
    Exception innerException  
)
```

```
Public Sub New ( _  
    msg As String, _  
    innerException As Exception _  
)
```

```
public:  
InstallCanceledException(  
    String^ msg,  
    Exception^ innerException  
)
```

### Parameters

#### *msg* (String)

The message that describes the error.

#### *innerException* (Exception)

The exception that is the cause of the current exception. If the innerException parameter is not a null reference (Nothing in Visual Basic), the current exception is raised in a catch block that handles

the inner exception.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **InstallCost** Structure

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► **InstallCost**

C# ▾

Represents a per-drive disk space cost for an installation.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public struct InstallCost
```

```
Public Structure InstallCost
```

```
public value class InstallCost
```

## Members

All Members

Methods

Properties

Public

Protected

Instance

Static



Declared



Inherited

Icon	Member	Description
	Cost	The installation cost on this drive, as a number of bytes.
	DriveName	The name of the drive this cost data applies to.
	Equals(Object)	Indicates whether this instance and a specified object are equal.  (Inherited from <a href="#">ValueType</a> .)

	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetHashCode()</a>	Returns the hash code for this instance.  (Inherited from <a href="#">ValueType</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance  (Inherited from <a href="#">Object</a> .)
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">TempCost</a>	The temporary disk space required on this drive, as a number of bytes.
	<a href="#">ToString()</a>	Returns the fully qualified type name of this instance.  (Inherited from <a href="#">ValueType</a> .)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Cost Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallCost ► Cost

C# ▾

The installation cost on this drive, as a number of bytes.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public long Cost { get; }
```

```
Public Readonly Property Cost As Long
```

```
public:  
property long long Cost {  
    long long get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## DriveName Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallCost ► DriveName

C# ▾

The name of the drive this cost data applies to.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public string DriveName { get; }
```

```
Public Readonly Property DriveName As String
```

```
public:  
property String^ DriveName {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **TempCost Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallCost ► TempCost

C# ▾

The temporary disk space required on this drive, as a number of bytes.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public long TempCost { get; }
```

```
Public Readonly Property TempCost As Long
```

```
public:  
property long long TempCost {  
    long long get ();  
}
```

### - Remarks

This temporary space requirement is space needed only for the duration of the installation, over the final footprint on disk.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Installer Class**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer

C#

Provides static methods for installing and configuring products and patches.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static class Installer
```

```
Public NotInheritable Class Installer
```

```
public ref class Installer abstract sealed
```

### Members

All Members	Methods	Properties	
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static	<input checked="" type="checkbox"/> Declared <input checked="" type="checkbox"/> Inherited

Icon	Member	Description
	AdvertiseProduct(String, Boolean, String, Int32)	Advertises a product to a computer.
	AdvertiseScript(String, Int32, Boolean)	Copies an advertise script to a local computer.
	ApplyMultiplePatches(IList<String>, String, String)	Applies one or more patches to products that are eligible for the patch. For each product, the patch package specified by the patch package argument is applied.

		receive the patch, App invokes an installation PATCH property to the patch package.
= 	ApplyPatch(String, String)	For each product listed package as eligible to patch, ApplyPatch invokes installation and sets the property to the path of package.
= 	ApplyPatch(String, String, InstallType, String)	For each product listed package as eligible to patch, ApplyPatch invokes installation and sets the property to the path of package.
= 	ConfigureFeature(String, String, InstallState)	Configures the installed product feature.
= 	ConfigureProduct(String, Int32, InstallState, String)	Installs or uninstalls a product.
= 	DetermineApplicablePatches(String, String[], InapplicablePatchHandler)	Determines which patches a specified product MS sequence.
= 	DetermineApplicablePatches(String, String[], InapplicablePatchHandler, String, UserContexts)	Determines which patches a specified product and sequence. If the produ

		this method accounts for sessions that have already been created for the product and account. It is obsolete and superceded by <a href="#">CreateSession</a> .
 <b>S</b>	<a href="#">EnableLog(InstallLogModes, String)</a>	Enables logging of the specified message type for all subsequent install sessions in the current process space.
 <b>S</b>	<a href="#">EnableLog(InstallLogModes, String, Boolean, Boolean)</a>	Enables logging of the specified message type for all subsequent install sessions in the current process space.
 <b>S</b>	<a href="#">ExtractPatchXmlData(String)</a>	Extracts information from the XML string that can be used to determine whether the patch applies to the target system. The method takes an XML string that can be passed to <a href="#">DetermineApplicablePatches(String[], String, InapplicablePatchReason, String, UserContexts)</a> to get the full patch file.
 <b>S</b>	<a href="#">GenerateAdvertiseScript(String, String, String, Int32)</a>	Generates an advertise script. This method enables the installation of a product to a script the registry and other information used to automatically publish a product.

 <a href="#">GenerateAdvertiseScript</a>	<code>GenerateAdvertiseScript(String, String, String, Int32, ProcessorArchitecture, Boolean)</code>	Generates an advertise script. This method enables the installer to write to a script the registry and other information used to advertise a product.
 <a href="#">GetErrorMessage</a>	<code>GetErrorMessage(Int32)</code>	Gets a Windows Installer error message in the system language.
 <a href="#">GetErrorMessage</a>	<code>GetErrorMessage(Int32, CultureInfo)</code>	Gets a Windows Installer error message in a specified culture.
 <a href="#">GetErrorMessage</a>	<code>GetErrorMessage(Record)</code>	Gets a formatted Windows Installer error message in the system language.
 <a href="#">GetErrorMessage</a>	<code>GetErrorMessage(Record, CultureInfo)</code>	Gets a formatted Windows Installer error message in a specified culture.
 <a href="#">GetFileHash</a>	<code>GetFileHash(String, Int32[])</code>	Gets a 128-bit hash of a file.
 <a href="#">GetFileLanguage</a>	<code>GetFileLanguage(String)</code>	Gets the language string specified using the form that the installer expects to find the database.
 <a href="#">GetFileLanguage</a>		

	<code>GetFileVersion(String)</code>	Gets the version string specified using the form installer expects to find database.
=  S	<code>GetPatchFileList(String, IList&lt;String&gt;)</code>	[MSI 4.0] Gets the list of patches that can be updated by one or more patches.
=  S	<code>GetProductInfoFromScript(String)</code>	Gets product information from the installer script file.
=  S	<code>GetShortcutTarget(String)</code>	Examines a shortcut and returns the product, feature name, component if available.
=  S	<code>InstallMissingComponent(String, String, InstallState)</code>	Installs files that are required but missing.
=  S	<code>InstallMissingFile(String, String)</code>	Installs files that are required but missing.
=  S	<code>InstallProduct(String, String)</code>	Opens an installer package and initializes an install session.
=  S	<code>NotifySidChange(String, String)</code>	[MSI 3.1] Migrates a user application configuration to a new SID.

	 <a href="#">OpenPackage(String, Boolean)</a>	Opens an installer package with functions that access the product database and returning an Session object.
	 <a href="#">OpenPackage(Database, Boolean)</a>	Opens an installer package with functions that access the product database and returning an Session object.
	 <a href="#">OpenProduct(String)</a>	Opens an installer package installed product using code.
	 <a href="#">ProcessAdvertiseScript(String, String, Boolean, Boolean)</a>	Processes an advertisement into the specified location.
	 <a href="#">ProvideAssembly(String, String, InstallMode, Boolean)</a>	Gets the full path to a \Install\component component assembly. This method adds a source and increments the usage count for the feature.
	 <a href="#">ProvideComponent(String, String, String, InstallMode)</a>	Gets the full component performing any necessary installation. This method adds for source if necessary increments the usage count for the feature.

 <a href="#">ProvideQualifiedComponent(String, String, InstallMode, String)</a>	Gets the full component qualified component that is published by a product. This method prompts for necessary and increments usage count for the feature.
 <a href="#">RebootInitiated</a>	Indicates whether a system reboot has been initiated after running an installation or configuration operation.
 <a href="#">RebootRequired</a>	Indicates whether a system reboot is required after running an installation or configuration operation.
 <a href="#">ReinstallFeature(String, String, ReinstallModes)</a>	Reinstalls a feature.
 <a href="#">ReinstallProduct(String, ReinstallModes)</a>	Reinstalls a product.
 <a href="#">RemovePatches(IList&lt;String&gt;, String, String)</a>	Removes one or more patches from a single product. To remove patches from multiple products, the RemovePatches must be called for each product.
 <a href="#">SetExternalUI(ExternalUIHandler, InstallLogModes)</a>	Enables an external user interface handler. This external UI

		called before the normal user-interface handler. UI handler has the option to suppress the internal UI by returning a non-zero value to indicate that it has handled the messages.
≡ S	<a href="#">SetExternalUI(ExternalUIRecordHandler, InstallLogModes)</a>	[MSI 3.1] Enables a registered external user-interface handler. The external UI handler is called instead of the normal internal user-interface handler. The external UI handler has the option to suppress the internal UI by returning a non-zero value to indicate that it has handled the messages.
≡ S	<a href="#">SetInternalUI(InstallUIOptions, IntPtr)</a>	Enables the installer's user-interface. Then this user-interface is used for all subsequent user-interface-generation functions in this process.
≡ S	<a href="#">SetInternalUI(InstallUIOptions)</a>	Enables the installer's user-interface. Then this user-interface is used for all subsequent user-interface-generation functions in this process. If the user interface does not change.
≡ S	<a href="#">UseFeature(String, String, InstallMode)</a>	increments the usage count of the feature.

		particular feature and returns the installation state for that feature. This method should be called before <code>Install</code> to indicate an application's dependencies and which features to use a feature.
 S	<a href="#">VerifyPackage(String)</a>	Verifies that the given file is a valid Windows installation package.
 S	<a href="#">Version</a>	Gets the current version of the Windows installer.

## Inheritance Hierarchy

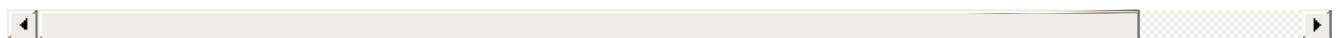
[Object](#)

 [Installer](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## AdvertiseProduct Method (**packagePath**, **perUser**, **transforms**, **locale**)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**AdvertiseProduct(String, Boolean, String, Int32)**

Advertises a product to the local computer.

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static void AdvertiseProduct(  
    string packagePath,  
    bool perUser,  
    string transforms,  
    int locale  
)
```

```
Public Shared Sub AdvertiseProduct ( _  
    packagePath As String, _  
    perUser As Boolean, _  
    transforms As String, _  
    locale As Integer _  
)
```

```
public:  
static void AdvertiseProduct(  
    String^ packagePath,  
    bool perUser,  
    String^ transforms,  
    int locale  
)
```

### Parameters

#### **packagePath** (**String**)

Path to the package of the product being advertised

***perUser* (Boolean)**

True if the product is user-assigned; false if it is machine-assigned.

***transforms* (String)**

Semi-colon delimited list of transforms to be applied. This parameter may be null.

***locale* (Int32)**

The language to use if the source supports multiple languages

**Remarks**

Win32 MSI APIs: [MsiAdvertiseProduct](#), [MsiAdvertiseProductEx](#)

**Exceptions**

Exception	Condition
<a href="#">FileNotFoundException</a>	the specified package file does not exist

**See Also**

[GenerateAdvertiseScript\(String, String, String, Int32, ProcessorArchitecture, Boolean\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## AdvertiseScript Method (*scriptFile*, *flags*, *removeItems*)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**AdvertiseScript(String, Int32, Boolean)**

C# ▾

Copies an advertise script file to the local computer.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static void AdvertiseScript(  
    string scriptFile,  
    int flags,  
    bool removeItems  
)
```

```
Public Shared Sub AdvertiseScript ( _  
    scriptFile As String, _  
    flags As Integer, _  
    removeItems As Boolean _  
)
```

```
public:  
static void AdvertiseScript(  
    String^ scriptFile,  
    int flags,  
    bool removeItems  
)
```

### Parameters

#### *scriptFile* ([String](#))

Path to a script file generated by [GenerateAdvertiseScript\(String, String, String, Int32, ProcessorArchitecture, Boolean\)](#)

#### *flags* ([Int32](#))

Flags controlling advertisement

***removeItems* (Boolean)**

True if specified items are to be removed instead of being created

**Remarks**

The process calling this function must be running under the LocalSystem account. To advertise an application for per-user installation to a targeted user, the thread that calls this function must impersonate the targeted user. If the thread calling this function is not impersonating a targeted user, the application is advertised to all users for installation with elevated privileges.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ApplyMultiplePatches Method (patchPackages, productCode, commandLine)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**ApplyMultiplePatches(IList<String>, String, String)**

C# ▾

Applies one or more patches to products that are eligible to receive the patch. For each product listed by the patch package as eligible to receive the patch, ApplyPatch invokes an installation and sets the PATCH property to the path of the patch package.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public static void ApplyMultiplePatches(  
    IList<string> patchPackages,  
    string productCode,  
    string commandLine  
)
```

```
Public Shared Sub ApplyMultiplePatches ( _  
    patchPackages As IList(Of String), _  
    productCode As String, _  
    commandLine As String _  
)
```

```
public:  
static void ApplyMultiplePatches(  
    IList<String^>^ patchPackages,  
    String^ productCode,  
    String^ commandLine  
)
```

### - Parameters

#### **patchPackages (IList<String>)**

The set of patch packages to be applied. Each item is the full path to the

an MSP file.

***productCode* ([String](#))**

Provides the ProductCode of the product being patched. If this parameter is null, the patches are applied to all products that are eligible to receive these patches.

***commandLine* ([String](#))**

optional command line property settings

**■ Remarks**

Win32 MSI API: [MsiApplyMultiplePatches](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ApplyPatch Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ► ApplyPatch()

C# ▾

### Members

Icon	Member	Description
= 	ApplyPatch(String, String)	For each product listed by the patch package as eligible to receive the patch, ApplyPatch invokes an installation and sets the PATCH property to the path of the patch package.
= 	ApplyPatch(String, String, InstallType, String)	For each product listed by the patch package as eligible to receive the patch, ApplyPatch invokes an installation and sets the PATCH property to the path of the patch package.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ApplyPatch Method (patchPackage, commandLine)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**ApplyPatch(String, String)**

C# ▾

For each product listed by the patch package as eligible to receive the patch, ApplyPatch invokes an installation and sets the PATCH property the path of the patch package.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static void ApplyPatch(  
    string patchPackage,  
    string commandLine  
)
```

```
Public Shared Sub ApplyPatch ( _  
    patchPackage As String, _  
    commandLine As String _  
)
```

```
public:  
static void ApplyPatch(  
    String^ patchPackage,  
    String^ commandLine  
)
```

### Parameters

**patchPackage (String)**  
path to the patch package

**commandLine (String)**  
optional command line property settings

## **Remarks**

The `RebootRequired` and `RebootInitiated` properties should be tested after calling this method.

Win32 MSI API: [MsiApplyPatch](#)

## **Exceptions**

Exception	Condition
<a href="#">InstallerException</a>	There was an error applying the patch

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ApplyPatch Method (patchPackage, installPackage, installType, commandLine)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ►

ApplyPatch(String, String, InstallType, String)

C#

For each product listed by the patch package as eligible to receive the patch, ApplyPatch invokes an installation and sets the PATCH property the path of the patch package.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static void ApplyPatch(  
    string patchPackage,  
    string installPackage,  
    InstallType installType,  
    string commandLine  
)
```

```
Public Shared Sub ApplyPatch ( _  
    patchPackage As String, _  
    installPackage As String, _  
    installType As InstallType, _  
    commandLine As String _  
)
```

```
public:  
static void ApplyPatch(  
    String^ patchPackage,  
    String^ installPackage,  
    InstallType installType,  
    String^ commandLine  
)
```

## [-] Parameters

### ***patchPackage (String)***

path to the patch package

### ***installPackage (String)***

path to the product to be patched, if installType is set to  
[NetworkImage](#)

### ***installType (InstallType)***

type of installation to patch

### ***commandLine (String)***

optional command line property settings

## [-] Remarks

The [RebootRequired](#) and [RebootInitiated](#) properties should be tested after calling this method.

Win32 MSI API: [MsiApplyPatch](#)

## [-] Exceptions

Exception	Condition
<a href="#">InstallerException</a>	There was an error applying the patch

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ConfigureFeature Method (**productCode**, **feature**, **installState**)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**ConfigureFeature(String, String, InstallState)**

C# ▾

Configures the installed state for a product feature.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static void ConfigureFeature(  
    string productCode,  
    string feature,  
    InstallState installState  
)
```

```
Public Shared Sub ConfigureFeature ( _  
    productCode As String, _  
    feature As String, _  
    installState As InstallState _  
)
```

```
public:  
static void ConfigureFeature(  
    String^ productCode,  
    String^ feature,  
    InstallState installState  
)
```

### Parameters

#### **productCode** (**String**)

Product code of the product to be configured.

#### **feature** (**String**)

Specifies the feature ID for the feature to be configured.

## *installState* ([InstallState](#))

Specifies the installation state for the feature.

### **Remarks**

The [RebootRequired](#) and [RebootInitiated](#) properties should be tested after calling this method.

Win32 MSI API: [MsiConfigureFeature](#)

### **Exceptions**

Exception	Condition
<a href="#">InstallerException</a>	There was an error configuring the feature.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# ConfigureProduct Method (**productCode**, **installLevel**, **installState**, **commandLine**)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**ConfigureProduct(String, Int32, InstallState, String)**

C# ▾

Installs or uninstalls a product.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public static void ConfigureProduct(  
    string productCode,  
    int installLevel,  
    InstallState installState,  
    string commandLine  
)
```

```
Public Shared Sub ConfigureProduct ( _  
    productCode As String, _  
    installLevel As Integer, _  
    installState As InstallState, _  
    commandLine As String _  
)
```

```
public:  
static void ConfigureProduct(  
    String^ productCode,  
    int installLevel,  
    InstallState installState,  
    String^ commandLine  
)
```

## Parameters

### ***productCode* ([String](#))**

Product code of the product to be configured.

### ***installLevel* ([Int32](#))**

Specifies the default installation configuration of the product. The *installLevel* parameter is ignored and all features are installed if the *installState* parameter is set to any other value than [Default](#). This parameter must be either 0 (install using authored feature levels), 65535 (install all features), or a value between 0 and 65535 to install a subset of available features.

### ***installState* ([InstallState](#))**

Specifies the installation state for the product.

### ***commandLine* ([String](#))**

Specifies the command line property settings. This should be a list in the format Property=Setting Property=Setting.

## **Remarks**

This method displays the user interface with the current settings and log mode. You can change user interface settings with the [SetInternalUI\(InstallUIOptions\)](#) and [SetExternalUI\(ExternalUIHandler, InstallLogModes\)](#) functions. You can set the log mode with the [EnableLog\(InstallLogModes, String\)](#) function.

The [RebootRequired](#) and [RebootInitiated](#) properties should be tested after calling this method.

Win32 MSI APIs: [MsiConfigureProduct](#), [MsiConfigureProductEx](#)

## **Exceptions**

Exception	Condition
<a href="#">InstallerException</a>	There was an error configuring the product.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## DetermineApplicablePatches

### Method

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**DetermineApplicablePatches()**

C# ▾

#### ▀ Members

Icon	Member	Description
	<a href="#">DetermineApplicablePatches(String, String[], InapplicablePatchHandler)</a>	Determines which patches apply to a specified product MSI and in what sequence.
	<a href="#">DetermineApplicablePatches(String, String[], InapplicablePatchHandler, String, UserContexts)</a>	Determines which patches apply to a specified product and in what sequence. If the product is installed, this method accounts for patches that have already been applied to the product and accounts for obsolete and superceded patches.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## DetermineApplicablePatches Method (productPackage, patches, errorHandler)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**DetermineApplicablePatches(String, String[], InapplicablePatchHandler)**

C# ▾

Determines which patches apply to a specified product MSI and in what sequence.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static IList<string> DetermineApplicablePatch
    string productPackage,
    string[] patches,
    InapplicablePatchHandler errorHandler
)
```

```
Public Shared Function DetermineApplicablePatches (
    productPackage As String, _
    patches As String(), _
    errorHandler As InapplicablePatchHandler _
) As IList(of String)
```

```
public:
static IList<String^>^ DetermineApplicablePatches(
    String^ productPackage,
    array<String^>^ patches,
    InapplicablePatchHandler^ errorHandler
)
```

### Parameters

#### ***productPackage (String)***

Full path to an MSI file that is the target product for the set of patches.

### ***patches* ( [String \[\]](#))**

An array of strings specifying the patches to be checked. Each item may be the path to an MSP file, the path an XML file, or just an XML blob.

### ***errorHandler* ( [InapplicablePatchHandler](#))**

Callback to be invoked for each inapplicable patch, reporting the reason the patch is not applicable. This value may be left null if that information is not desired.

## **▀ Return Value**

An array of selected patch strings from *patches*, indicating the set of applicable patches. The items are re-ordered to be in the best sequence.

## **▀ Remarks**

If an item in *patches* is a file path but does not end in .MSP or .XML, it is assumed to be an MSP file.

As this overload uses InstallContext.None, it does not consider the current state of the system.

Win32 MSI API: [MsiDetermineApplicablePatches](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **DetermineApplicablePatches**

## Method (product, patches, errorHandler, userSid, context)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**DetermineApplicablePatches(String, String[], InapplicablePatchHandler, String, UserContexts)**

C# ▾

Determines which patches apply to a specified product and in what sequence. If the product is installed, this method accounts for patches that have already been applied to the product and accounts for obsolete and superceded patches.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static IList<string> DetermineApplicablePatch
    string product,
    string[] patches,
    InapplicablePatchHandler errorHandler,
    string userSid,
    UserContexts context
)
```

```
Public Shared Function DetermineApplicablePatches (
    product As String,
    patches As String(),
    errorHandler As InapplicablePatchHandler,
    userSid As String,
    context As UserContexts
) As IList(Of String)
```

```
public:
static IList<String^>^ DetermineApplicablePatches(
    String^ product,
    array<String^>^ patches,
```

```
        InapplicablePatchHandler^ errorHandler,  
        String^ userSid,  
        UserContexts context  
    )
```

## [-] Parameters

### ***product* ([String](#))**

The product that is the target for the set of patches. This may be either a ProductCode (GUID) of a product that is currently installed or the path to a an MSI file.

### ***patches* ([String](#) [])**

An array of strings specifying the patches to be checked. Each item may be the path to an MSP file, the path an XML file, or just an XM blob.

### ***errorHandler* ([InapplicablePatchHandler](#))**

Callback to be invoked for each inapplicable patch, reporting the reason the patch is not applicable. This value may be left null if that information is not desired.

### ***userSid* ([String](#))**

Specifies a security identifier (SID) of a user. This parameter restricts the context of enumeration for this user account. This parameter cannot be the special SID strings s-1-1-0 (everyone) or s-1-5-18 (local system). If *context* is set to [None](#) or [Machine](#), then *userSid* must be null. For the current user context, *userSid* can be null and *context* can be set to [UserManaged](#) or [UserUnmanaged](#).

### ***context* ([UserContexts](#))**

Restricts the enumeration to per-user-unmanaged, per-user-managed, or per-machine context, or (if referring to an MSI) to no system context at all. This parameter can be [Machine](#), [UserManaged](#), [UserUnmanaged](#), or [None](#).

## [-] Return Value

An array of selected patch strings from *patches*, indicating the set of applicable patches. The items are re-ordered to be in the best sequence.

## [-] Remarks

If an item in *patches* is a file path but does not end in .MSP or .XML, it is assumed to be an MSP file.

Passing an InstallContext of None only analyzes the MSI file; it does not consider the current state of the system. You cannot use InstallContext.None with a ProductCode GUID.

Win32 MSI APIs: [MsiDetermineApplicablePatches](#)  
[MsiDeterminePatchSequence](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **EnableLog Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ► **EnableLog()**

C# ▾

[-] **Members**

Icon	Member	Description
= 	<a href="#">EnableLog(InstallLogModes, String)</a>	Enables logging of the selected message type for all subsequent install sessions in the current process space.
= 	<a href="#">EnableLog(InstallLogModes, String, Boolean, Boolean)</a>	Enables logging of the selected message type for all subsequent install sessions in the current process space.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## EnableLog Method (logModes, logFile)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**EnableLog(InstallLogModes, String)**

C# ▾

Enables logging of the selected message type for all subsequent install sessions in the current process space.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static void EnableLog(  
    InstallLogModes logModes,  
    string logFile  
)
```

```
Public Shared Sub EnableLog ( _  
    logModes As InstallLogModes, _  
    logFile As String _  
)
```

```
public:  
static void EnableLog(  
    InstallLogModes logModes,  
    String^ logFile  
)
```

### Parameters

#### ***logModes* (InstallLogModes)**

One or more mode flags specifying the type of messages to log

#### ***logFile* (String)**

Full path to the log file. A null path disables logging, in which case the logModes parameter is ignored.

## **Remarks**

This method takes effect on any new installation processes. Calling this method from within a custom action will not start logging for that installation.

## **Exceptions**

Exception	Condition
<a href="#">ArgumentException</a>	an invalid log mode was specified

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## EnableLog Method (logModes, logFile, append, flushEveryLine)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ►

**EnableLog(InstallLogModes, String, Boolean, Boolean)**

C# ▾

Enables logging of the selected message type for all subsequent install sessions in the current process space.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static void EnableLog(
    InstallLogModes logModes,
    string logFile,
    bool append,
    bool flushEveryLine
)
```

```
Public Shared Sub EnableLog ( _
    logModes As InstallLogModes, _
    logFile As String, _
    append As Boolean, _
    flushEveryLine As Boolean _
)
```

```
public:
static void EnableLog(
    InstallLogModes logModes,
    String^ logFile,
    bool append,
    bool flushEveryLine
)
```

### [-] Parameters

***logModes* (InstallLogModes)**

One or more mode flags specifying the type of messages to log

#### ***logFile (String)***

Full path to the log file. A null path disables logging, in which case the logModes parameter is ignored.

#### ***append (Boolean)***

If true, the log lines will be appended to any existing file content. If false, the log file will be truncated if it exists. The default is false.

#### ***flushEveryLine (Boolean)***

If true, the log will be flushed after every line. If false, the log will be flushed every 20 lines. The default is true.

### **Remarks**

This method takes effect on any new installation processes. Calling this method from within a custom action will not start logging for that installation.

Win32 MSI API: [MsiEnableLog](#)

### **Exceptions**

Exception	Condition
<a href="#">ArgumentException</a>	an invalid log mode was specified

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## ExtractPatchXmlData Method (patchPath)

Deployment Tools Foundation Namespaces ►  
 Microsoft.Deployment.WindowsInstaller ► Installer ►  
**ExtractPatchXmlData(String)**



Extracts information from a patch that can be used to determine whether the patch applies on a target system. The method returns an XML string that can be provided to [DetermineApplicablePatches\(String, String\[\], InapplicablePatchHandler, String, UserContexts\)](#) instead of the full patch file.

### Declaration Syntax

C#	Visual Basic	Visual C++
<code>public static string ExtractPatchXmlData(     string patchPath )</code>	<code>Public Shared Function ExtractPatchXmlData ( _     patchPath As String _ ) As String</code>	<code>public: static String^ ExtractPatchXmlData(     String^ patchPath )</code>

```
public static string ExtractPatchXmlData(
    string patchPath
)
```

```
Public Shared Function ExtractPatchXmlData ( _
    patchPath As String _
) As String
```

```
public:  
static String^ ExtractPatchXmlData(  
    String^ patchPath  
)
```

### Parameters

#### **patchPath (String)**

Full path to the patch being queried.

### Return Value

XML string containing patch data.

## Remarks

Win32 MSI API: [MsiExtractPatchXMLData](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GenerateAdvertiseScript Method**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**GenerateAdvertiseScript()**

C# ▾

▀ Members

Icon	Member	Description
= ♪ S	GenerateAdvertiseScript(String, String, String, Int32)	Generates an advertise script. The method enables the installer to write to a script the registry and shortcut information used to assign or publish a product.
= ♪ S	GenerateAdvertiseScript(String, String, String, Int32, ProcessorArchitecture, Boolean)	Generates an advertise script. The method enables the installer to write to a script the registry and shortcut information used to assign or publish a product.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GenerateAdvertiseScript Method  
(packagePath, scriptFilePath, transforms, locale)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ►

**GenerateAdvertiseScript(String, String, String, Int32)**

C# ▾

Generates an advertise script. The method enables the installer to write to a script the registry and shortcut information used to assign or publish a product.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public static void GenerateAdvertiseScript(  
    string packagePath,  
    string scriptFilePath,  
    string transforms,  
    int locale  
)
```

```
Public Shared Sub GenerateAdvertiseScript ( _  
    packagePath As String, _  
    scriptFilePath As String, _  
    transforms As String, _  
    locale As Integer _  
)
```

```
public:  
static void GenerateAdvertiseScript(  
    String^ packagePath,  
    String^ scriptFilePath,  
    String^ transforms,  
    int locale  
)
```

[-] **Parameters**

***packagePath* ([String](#))**

Path to the package of the product being advertised

***scriptFilePath* ([String](#))**

path to script file to be created with the advertise information

***transforms* ([String](#))**

Semi-colon delimited list of transforms to be applied. This parameter may be null.

***locale* ([Int32](#))**

The language to use if the source supports multiple languages

**[-] Remarks**

Win32 MSI APIs: [MsiAdvertiseProduct](#), [MsiAdvertiseProductEx](#)

**[-] Exceptions**

Exception	Condition
<a href="#">FileNotFoundException</a>	the specified package file does not exist

**[-] See Also**

[AdvertiseProduct\(String, Boolean, String, Int32\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# GenerateAdvertiseScript Method (**packagePath**, **scriptFilePath**, **transforms**, **locale**, **processor**, **instance**)

Deployment Tools Foundation Namespaces ►

[Microsoft.Deployment.WindowsInstaller](#) ► [Installer](#) ►

**GenerateAdvertiseScript(String, String, String, Int32,  
ProcessorArchitecture, Boolean)**

C#

Generates an advertise script. The method enables the installer to write to a script the registry and shortcut information used to assign or publish a product.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public static void GenerateAdvertiseScript(
    string packagePath,
    string scriptFilePath,
    string transforms,
    int locale,
    ProcessorArchitecture processor,
    bool instance
)
```

```
Public Shared Sub GenerateAdvertiseScript ( _
    packagePath As String, _
    scriptFilePath As String, _
    transforms As String, _
    locale As Integer, _
    processor As ProcessorArchitecture, _
    instance As Boolean _
)
```

```
public:
static void GenerateAdvertiseScript(
    String^ packagePath,
```

```
        String^ scriptFilePath,
        String^ transforms,
        int locale,
        ProcessorArchitecture processor,
        bool instance
    )
```

## [-] Parameters

### **packagePath (String)**

Path to the package of the product being advertised

### **scriptFilePath (String)**

path to script file to be created with the advertise information

### **transforms (String)**

Semi-colon delimited list of transforms to be applied. This parameter may be null.

### **locale (Int32)**

The language to use if the source supports multiple languages

### **processor (ProcessorArchitecture)**

Targeted processor architecture.

### **instance (Boolean)**

True to install multiple instances through product code changing transform. Advertises a new instance of the product. Requires that the *transforms* parameter includes the instance transform that changes the product code.

## [-] Remarks

Win32 MSI APIs: [MsiAdvertiseProduct](#), [MsiAdvertiseProductEx](#)

## [-] See Also

[AdvertiseProduct\(String, Boolean, String, Int32\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetErrorMessage Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ►

**GetErrorMessage()**

C# ▾

▀ Members

Icon	Member	Description
= ♦ S	<a href="#">GetErrorMessage(Int32)</a>	Gets a Windows Installer error message in the system default language.
= ♦ S	<a href="#">GetErrorMessage(Int32, CultureInfo)</a>	Gets a Windows Installer error message in a specified language.
= ♦ S	<a href="#">GetErrorMessage(Record)</a>	Gets a formatted Windows Installer error message in the system default language.
= ♦ S	<a href="#">GetErrorMessage(Record, CultureInfo)</a>	Gets a formatted Windows Installer error message in a specified language.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetErrorMessage Method**  
**(errorRecord)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**GetErrorMessage(Record)**

C# ▾

Gets a formatted Windows Installer error message in the system default language.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public static string GetErrorMessage(  
    Record errorRecord  
)
```

```
Public Shared Function GetErrorMessage ( _  
    errorRecord As Record _  
) As String
```

```
public:  
static String^ GetErrorMessage(  
    Record^ errorRecord  
)
```

[-] **Parameters**

**errorRecord (Record)**

Error record containing the error number in the first field, and error-specific parameters in the other fields.

[-] **Return Value**

The message string, or null if the error message is not found.

[-] **Remarks**

Error numbers greater than 2000 refer to MSI "internal" errors, and are

always returned in English.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## GetErrorMessage Method (errorRecord, culture)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**GetErrorMessage(Record, CultureInfo)**

C# ▾

Gets a formatted Windows Installer error message in a specified language.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static string GetErrorMessage(  
    Record errorRecord,  
    CultureInfo culture  
)
```

```
Public Shared Function GetErrorMessage ( _  
    errorRecord As Record, _  
    culture As CultureInfo _  
) As String
```

```
public:  
static String^ GetErrorMessage(  
    Record^ errorRecord,  
    CultureInfo^ culture  
)
```

### Parameters

#### ***errorRecord* (Record)**

Error record containing the error number in the first field, and error-specific parameters in the other fields.

#### ***culture* (CultureInfo)**

The locale for the message.

## **❑ Return Value**

The message string, or null if the error message or locale is not found.

## **❑ Remarks**

Error numbers greater than 2000 refer to MSI "internal" errors, and are always returned in English.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## GetErrorMessage Method (*errorNumber*)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**GetErrorMessage(Int32)**

C# ▾

Gets a Windows Installer error message in the system default language

### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static string GetErrorMessage(  
    int errorNumber  
)
```

```
Public Shared Function GetErrorMessage ( _  
    errorNumber As Integer _  
) As String
```

```
public:  
static String^ GetErrorMessage(  
    int errorNumber  
)
```

### [-] Parameters

#### *errorNumber* (**Int32**)

The error number.

### [-] Return Value

The message string, or null if the error message is not found.

### [-] Remarks

The returned string may have tokens such as [2] and [3] that are meant to be substituted with context-specific values.

Error numbers greater than 2000 refer to MSI "internal" errors, and are always returned in English.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## GetErrorMessage Method (errorNumber, culture)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**GetErrorMessage(Int32, CultureInfo)**

C#

Gets a Windows Installer error message in a specified language.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static string GetErrorMessage(  
    int errorNumber,  
    CultureInfo culture  
)
```

```
Public Shared Function GetErrorMessage ( _  
    errorNumber As Integer, _  
    culture As CultureInfo _  
) As String
```

```
public:  
static String^ GetErrorMessage(  
    int errorNumber,  
    CultureInfo^ culture  
)
```

### Parameters

#### ***errorNumber* (Int32)**

The error number.

#### ***culture* (CultureInfo)**

The locale for the message.

### Return Value

The message string, or null if the error message or locale is not found.

## Remarks

The returned string may have tokens such as [2] and [3] that are meant to be substituted with context-specific values.

Error numbers greater than 2000 refer to MSI "internal" errors, and are always returned in English.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetFileHash Method (path, hash)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**GetFileHash(String, Int32[])**

C# ▾

Gets a 128-bit hash of the specified file.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static void GetFileHash(  
    string path,  
    int[] hash  
)
```

```
Public Shared Sub GetFileHash ( _  
    path As String, _  
    hash As Integer() _  
)
```

```
public:  
static void GetFileHash(  
    String^ path,  
    array<int>^ hash  
)
```

### Parameters

#### **path (String)**

Path to the file

#### **hash ( Int32 [] )**

Integer array of length 4 which receives the four 32-bit parts of the hash value.

### Remarks

Win32 MSI API: [MsiGetFileHash](#)

## Exceptions

Exception	Condition
FileNotFoundException	the file does not exist or could not be read

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetFileLanguage Method (path)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**GetFileLanguage(String)**

C# ▾

Gets the language string of the path specified using the format that the installer expects to find them in in the database.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static string GetFileLanguage(  
    string path  
)
```

```
Public Shared Function GetFileLanguage ( _  
    path As String _  
) As String
```

```
public:  
static String^ GetFileLanguage(  
    String^ path  
)
```

### Parameters

#### **path (String)**

Path to the file

### Return Value

Language string in the form of a decimal language ID, or an empty string if the file does not contain a language ID

### Remarks

Win32 MSI API: [MsiGetFileVersion](#)

## Exceptions

Exception	Condition
FileNotFoundException	the file does not exist or could not be read

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetFileVersion Method (path)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**GetFileVersion(String)**

C#

Gets the version string of the path specified using the format that the installer expects to find it in in the database.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static string GetFileVersion(  
    string path  
)
```

```
Public Shared Function GetFileVersion ( _  
    path As String _  
) As String
```

```
public:  
static String^ GetFileVersion(  
    String^ path  
)
```

### Parameters

#### **path (String)**

Path to the file

### Return Value

Version string in the "#.#.#.#" format, or an empty string if the file does not contain version information

### Remarks

Win32 MSI API: [MsiGetFileVersion](#)

## Exceptions

Exception	Condition
FileNotFoundException	the file does not exist or could not be read

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetPatchFileList Method**  
**(productCode, patches)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**GetPatchFileList(String, IList<String>)**

C# ▾

[MSI 4.0] Gets the list of files that can be updated by one or more patches.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public static IList<string> GetPatchFileList(
    string productCode,
    IList<string> patches
)
```

```
Public Shared Function GetPatchFileList ( _
    productCode As String, _
    patches As IList(Of String) _
) As IList(Of String)
```

```
public:
static IList<String^>^ GetPatchFileList(
    String^ productCode,
    IList<String^>^ patches
)
```

[-] **Parameters**

***productCode* (String)**

ProductCode (GUID) of the product which is the target of the patches

***patches* (IList<String>)**

list of file paths of one or more patches to be analyzed

## **▀ Return Value**

List of absolute paths of files that can be updated when the patches are applied on this system.

## **▀ Remarks**

Win32 MSI API: [MsiGetPatchFileList](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# GetProductInfoFromScript

## Method (scriptFile)

Deployment Tools Foundation Namespaces ►  
 Microsoft.Deployment.WindowsInstaller ► Installer ►  
**GetProductInfoFromScript(String)**

C# ▾

Gets product information for an installer script file.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static ProductInstallation GetProductInfoFromScript(
    string scriptFile
)
```

```
Public Shared Function GetProductInfoFromScript ( _
    scriptFile As String _
) As ProductInstallation
```

```
public:
static ProductInstallation^ GetProductInfoFromScript(
    String^ scriptFile
)
```

### Parameters

#### *scriptFile* (String)

Path to a script file generated by [GenerateAdvertiseScript\(String, String, String, Int32, ProcessorArchitecture, Boolean\)](#)

### Return Value

ProductInstallation stub with advertise-related properties filled in.

### Remarks

Only the following properties will be filled in in the returned object:

- ProductCode
- AdvertisedLanguage
- AdvertisedVersion
- AdvertisedProductName
- AdvertisedPackageName

Other properties will be null.

Win32 MSI API: [MsiGetProductInfoFromScript](#)

## **[-] Exceptions**

Exception	Condition
<a href="#">ArgumentOutOfRangeException</a>	An invalid product property was requested

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# GetShortcutTarget Method (shortcut)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**GetShortcutTarget(String)**

C# ▾

Examines a shortcut and returns its product, feature name, and component if available.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static ShortcutTarget GetShortcutTarget(  
    string shortcut  
)
```

```
Public Shared Function GetShortcutTarget ( _  
    shortcut As String _  
) As ShortcutTarget
```

```
public:  
static ShortcutTarget GetShortcutTarget(  
    String^ shortcut  
)
```

## Parameters

### shortcut (String)

Full path to a shortcut

## Return Value

ShortcutTarget structure containing target product code, feature, and component code

## Remarks

Win32 MSI API: [MsiGetShortcutTarget](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)  
Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## InstallMissingComponent Method (product, component, installState)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ►

**InstallMissingComponent(String, String, InstallState)**

C# ▾

Installs files that are unexpectedly missing.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static void InstallMissingComponent(  
    string product,  
    string component,  
    InstallState installState  
)
```

```
Public Shared Sub InstallMissingComponent ( _  
    product As String, _  
    component As String, _  
    installState As InstallState _  
)
```

```
public:  
static void InstallMissingComponent(  
    String^ product,  
    String^ component,  
    InstallState installState  
)
```

### Parameters

#### **product (String)**

Product code for the product that owns the component to be installed

#### **component (String)**

Component to be installed

## *installState* ([InstallState](#))

Specifies the way the component should be installed.

### **Remarks**

Win32 MSI API: [MsilInstallMissingComponent](#)

### **Exceptions**

Exception	Condition
<a href="#">InstallCanceledException</a>	the user exited the installation

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## InstallMissingFile Method (product, file)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**InstallMissingFile(String, String)**

C# ▾

Installs files that are unexpectedly missing.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static void InstallMissingFile(  
    string product,  
    string file  
)
```

```
Public Shared Sub InstallMissingFile ( _  
    product As String, _  
    file As String _  
)
```

```
public:  
static void InstallMissingFile(  
    String^ product,  
    String^ file  
)
```

### Parameters

#### **product (String)**

Product code for the product that owns the file to be installed

#### **file (String)**

File to be installed

### Remarks

Win32 MSI API: [MsilInstallMissingFile](#)

## **Exceptions**

<b>Exception</b>	<b>Condition</b>
<a href="#">InstallCanceledException</a>	the user exited the installation

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## InstallProduct Method (packagePath, commandLine)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ►

**InstallProduct(String, String)**

C#

Opens an installer package and initializes an install session.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static void InstallProduct(  
    string packagePath,  
    string commandLine  
)
```

```
Public Shared Sub InstallProduct ( _  
    packagePath As String, _  
    commandLine As String _  
)
```

```
public:  
static void InstallProduct(  
    String^ packagePath,  
    String^ commandLine  
)
```

### Parameters

#### **packagePath (String)**

path to the patch package

#### **commandLine (String)**

command line property settings

### Remarks

To completely remove a product, set REMOVE=ALL in .

This method displays the user interface with the current settings and log mode. You can change user interface settings with the [SetInternalUI\(InstallUIOptions\)](#) and [SetExternalUI\(ExternalUIHandler, InstallLogModes\)](#) functions. You can set the log mode with the [EnableLog\(InstallLogModes, String\)](#) function.

The [RebootRequired](#) and [RebootInitiated](#) properties should be tested after calling this method.

Win32 MSI API: [MsiInstallProduct](#)

## **Exceptions**

Exception	Condition
<a href="#">InstallerException</a>	There was an error installing the product

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## NotifySidChange Method (oldSid, newSid)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**NotifySidChange(String, String)**

C# ▾

[MSI 3.1] Migrates a user's application configuration data to a new SID.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static void NotifySidChange(  
    string oldSid,  
    string newSid  
)
```

```
Public Shared Sub NotifySidChange ( _  
    oldSid As String, _  
    newSid As String _  
)
```

```
public:  
static void NotifySidChange(  
    String^ oldSid,  
    String^ newSid  
)
```

### Parameters

#### *oldSid* (String)

Previous user SID that data is to be migrated from

#### *newSid* (String)

New user SID that data is to be migrated to

### Remarks

## Win32 MSI API: [MsiNotifySidChange](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **OpenPackage Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ► **OpenPackage()**

C# ▾

[-] **Members**

Icon	Member	Description
=  S	<a href="#">OpenPackage(String, Boolean)</a>	Opens an installer package for use with functions that access the product database and install engine, returning an Session object.
=  S	<a href="#">OpenPackage(Database, Boolean)</a>	Opens an installer package for use with functions that access the product database and install engine, returning an Session object.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## OpenPackage Method (database, ignoreMachineState)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**OpenPackage(Database, Boolean)**

C# ▾

Opens an installer package for use with functions that access the product database and install engine, returning an Session object.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static Session OpenPackage(  
    Database database,  
    bool ignoreMachineState  
)
```

```
Public Shared Function OpenPackage ( _  
    database As Database, _  
    ignoreMachineState As Boolean _  
) As Session
```

```
public:  
static Session^ OpenPackage(  
    Database^ database,  
    bool ignoreMachineState  
)
```

### Parameters

#### **database (Database)**

Database used to create the session

#### **ignoreMachineState (Boolean)**

Specifies whether or not the create a Session object that ignores the computer state and that is incapable of changing the current computer state. A value of false yields the normal behavior. A value

of true creates a "safe" Session object that cannot change of the current machine state.

## [-] **Return Value**

A Session object allowing access to the product database and install engine

## [-] **Remarks**

Note that only one Session object can be opened by a single process. OpenPackage cannot be used in a custom action because the active installation is the only session allowed.

A "safe" Session object ignores the current computer state when opening the package and prevents changes to the current computer state.

The Session object should be [Close\(\)](#)d after use. It is best that the handle be closed manually as soon as it is no longer needed, as leaving lots of unused handles open can degrade performance.

Win32 MSI APIs: [MsiOpenPackage](#), [MsiOpenPackageEx](#)

## [-] **Exceptions**

Exception	Condition
<a href="#">InstallerException</a>	The product could not be opened
<a href="#">InstallerException</a>	The installer configuration data is corrupt

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## OpenPackage Method (*packagePath*, *ignoreMachineState*)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**OpenPackage(String, Boolean)**

C# ▾

Opens an installer package for use with functions that access the product database and install engine, returning an Session object.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static Session OpenPackage(  
    string packagePath,  
    bool ignoreMachineState  
)
```

```
Public Shared Function OpenPackage ( _  
    packagePath As String, _  
    ignoreMachineState As Boolean _  
) As Session
```

```
public:  
static Session^ OpenPackage(  
    String^ packagePath,  
    bool ignoreMachineState  
)
```

### Parameters

#### *packagePath* (**String**)

Path to the package

#### *ignoreMachineState* (**Boolean**)

Specifies whether or not the create a Session object that ignores the computer state and that is incapable of changing the current computer state. A value of false yields the normal behavior. A value

of true creates a "safe" Session object that cannot change of the current machine state.

## [-] **Return Value**

A Session object allowing access to the product database and install engine

## [-] **Remarks**

Note that only one Session object can be opened by a single process. OpenPackage cannot be used in a custom action because the active installation is the only session allowed.

A "safe" Session object ignores the current computer state when opening the package and prevents changes to the current computer state.

The Session object should be [Close\(\)](#)d after use. It is best that the handle be closed manually as soon as it is no longer needed, as leaving lots of unused handles open can degrade performance.

Win32 MSI APIs: [MsiOpenPackage](#), [MsiOpenPackageEx](#)

## [-] **Exceptions**

Exception	Condition
<a href="#">InstallerException</a>	The product could not be opened
<a href="#">InstallerException</a>	The installer configuration data is corrupt

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **OpenProduct Method  
(productCode)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**OpenProduct(String)**

Opens an installer package for an installed product using the product code.

C#

[-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public static Session OpenProduct(  
    string productCode  
)
```

```
Public Shared Function OpenProduct ( _  
    productCode As String _  
) As Session
```

```
public:  
static Session^ OpenProduct(  
    String^ productCode  
)
```

[-] Parameters

**productCode (String)**

Product code of the installed product

[-] Return Value

A Session object allowing access to the product database and install engine, or null if the specified product is not installed.

[-] Remarks

Note that only one Session object can be opened by a single process.

OpenProduct cannot be used in a custom action because the active installation is the only session allowed.

The Session object should be [Close\(\)](#)d after use. It is best that the handle be closed manually as soon as it is no longer needed, as leaving lots of unused handles open can degrade performance.

Win32 MSI API: [MsiOpenProduct](#)

## **Exceptions**

Exception	Condition
<a href="#">ArgumentException</a>	An unknown product was requested
<a href="#">InstallerException</a>	The product could not be opened
<a href="#">InstallerException</a>	The installer configuration data is corrupt

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ProcessAdvertiseScript Method (scriptFile, iconFolder, shortcuts, removeItems)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ►

**ProcessAdvertiseScript(String, String, Boolean, Boolean)**

C#

Processes an advertise script file into the specified locations.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public static void ProcessAdvertiseScript(  
    string scriptFile,  
    string iconFolder,  
    bool shortcuts,  
    bool removeItems  
)
```

```
Public Shared Sub ProcessAdvertiseScript ( _  
    scriptFile As String, _  
    iconFolder As String, _  
    shortcuts As Boolean, _  
    removeItems As Boolean _  
)
```

```
public:  
static void ProcessAdvertiseScript(  
    String^ scriptFile,  
    String^ iconFolder,  
    bool shortcuts,  
    bool removeItems  
)
```

### - Parameters

**scriptFile (String)**

Path to a script file generated by [GenerateAdvertiseScript\(String,](#)

`String, String, Int32, ProcessorArchitecture, Boolean)`

***iconFolder (String)***

An optional path to a folder in which advertised icon files and transform files are located. If this parameter is null, no icon or transform files are written.

***shortcuts (Boolean)***

True if shortcuts should be created

***removeItems (Boolean)***

True if specified items are to be removed instead of created

**[-] Remarks**

The process calling this function must be running under the LocalSystem account. To advertise an application for per-user installation to a targeted user, the thread that calls this function must impersonate the targeted user. If the thread calling this function is not impersonating a targeted user, the application is advertised to all users for installation with elevated privileges.

Win32 MSI API: [MsiProcessAdvertiseScript](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ProvideAssembly Method (assemblyName, appContext, installMode, isWin32Assembly)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ►

**ProvideAssembly(String, String, InstallMode, Boolean)**

C# ▾

Gets the full path to a Windows Installer component containing an assembly. This method prompts for a source and increments the usage count for the feature.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static string ProvideAssembly(  
    string assemblyName,  
    string appContext,  
    InstallMode installMode,  
    bool isWin32Assembly  
)
```

```
Public Shared Function ProvideAssembly ( _  
    assemblyName As String, _  
    appContext As String, _  
    installMode As InstallMode, _  
    isWin32Assembly As Boolean _  
) As String
```

```
public:  
static String^ ProvideAssembly(  
    String^ assemblyName,  
    String^ appContext,  
    InstallMode installMode,  
    bool isWin32Assembly  
)
```

## ❑ Parameters

### ***assemblyName* ([String](#))**

Assembly name

### ***appContext* ([String](#))**

Set to null for global assemblies. For private assemblies, set to the full path of the application configuration file (.cfg file) or executable file (.exe) of the application to which the assembly has been made private.

### ***installMode* ([InstallMode](#))**

Installation mode; this can also include bits from [ReinstallModes](#)

### ***isWin32Assembly* ([Boolean](#))**

True if this is a Win32 assembly, false if it is a .NET assembly

## ❑ Return Value

Path to the assembly

## ❑ Remarks

Win32 MSI API: [MsiProvideAssembly](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ProvideComponent Method (product, feature, component, installMode)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ►

**ProvideComponent(String, String, String, InstallMode)**

C# ▾

Gets the full component path, performing any necessary installation. This method prompts for source if necessary and increments the usage count for the feature.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static string ProvideComponent(  
    string product,  
    string feature,  
    string component,  
    InstallMode installMode  
)
```

```
Public Shared Function ProvideComponent ( _  
    product As String, _  
    feature As String, _  
    component As String, _  
    installMode As InstallMode _  
) As String
```

```
public:  
static String^ ProvideComponent(  
    String^ product,  
    String^ feature,  
    String^ component,  
    InstallMode installMode  
)
```

### Parameters

***product* ([String](#))**

Product code for the product that contains the feature with the necessary component

***feature* ([String](#))**

Feature ID of the feature with the necessary component

***component* ([String](#))**

Component code of the necessary component

***installMode* ([InstallMode](#))**

Installation mode; this can also include bits from [ReinstallModes](#)

**[-] Return Value**

Path to the component

**[-] Remarks**

Win32 MSI API: [MsiProvideComponent](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# ProvideQualifiedComponent Method (component, qualifier, installMode, product)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ►

**ProvideQualifiedComponent(String, String, InstallMode, String)**

C# ▾

Gets the full component path for a qualified component that is published by a product and performs any necessary installation. This method prompts for source if necessary and increments the usage count for the feature.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public static string ProvideQualifiedComponent(
    string component,
    string qualifier,
    InstallMode installMode,
    string product
)
```

```
Public Shared Function ProvideQualifiedComponent ( _
    component As String, _
    qualifier As String, _
    installMode As InstallMode, _
    product As String _
) As String
```

```
public:
static String^ ProvideQualifiedComponent(
    String^ component,
    String^ qualifier,
    InstallMode installMode,
    String^ product
)
```

## ❑ Parameters

### ***component* ([String](#))**

Specifies the component ID for the requested component. This may not be the GUID for the component itself but rather a server that provides the correct functionality, as in the ComponentId column of the PublishComponent table.

### ***qualifier* ([String](#))**

Specifies a qualifier into a list of advertising components (from PublishComponent Table).

### ***installMode* ([InstallMode](#))**

Installation mode; this can also include bits from [ReinstallModes](#)

### ***product* ([String](#))**

Optional; specifies the product to match that has published the qualified component.

## ❑ Return Value

Path to the component

## ❑ Remarks

Win32 MSI APIs: [MsiProvideQualifiedComponent](#)

[MsiProvideQualifiedComponentEx](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## RebootInitiated Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ► RebootInitiated

C#

Indicates whether a system reboot has been initiated after running an installation or configuration operation.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static bool RebootInitiated { get; }
```

```
Public Shared ReadOnly Property RebootInitiated As Boolean
```

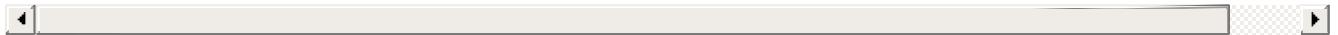
```
public:  
static property bool RebootInitiated {  
    bool get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## RebootRequired Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ►

### RebootRequired

C#

Indicates whether a system reboot is required after running an installation or configuration operation.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static bool RebootRequired { get; }
```

```
Public Shared ReadOnly Property RebootRequired As Boolean
```

```
public:  
    static property bool RebootRequired {  
        bool get ();  
    }
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## ReinstallFeature Method (product, feature, reinstallModes)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
ReinstallFeature(String, String, ReinstallModes)

C# ▾

Reinstalls a feature.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static void ReinstallFeature(  
    string product,  
    string feature,  
    ReinstallModes reinstallModes  
)
```

```
Public Shared Sub ReinstallFeature ( _  
    product As String, _  
    feature As String, _  
    reinstallModes As ReinstallModes _  
)
```

```
public:  
static void ReinstallFeature(  
    String^ product,  
    String^ feature,  
    ReinstallModes reinstallModes  
)
```

### Parameters

#### **product (String)**

Product code for the product containing the feature to be reinstalled

#### **feature (String)**

Feature to be reinstalled

## ***reinstallModes* ([ReinstallModes](#))**

Reinstall modes

### **Remarks**

Win32 MSI API: [MsiReinstallFeature](#)

### **Exceptions**

Exception	Condition
<a href="#">InstallCanceledException</a>	the user exited the installation

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ReinstallProduct Method (product, reinstallModes)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**ReinstallProduct(String, ReinstallModes)**

Reinstalls a product.

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static void ReinstallProduct(  
    string product,  
    ReinstallModes reinstallModes  
)
```

```
Public Shared Sub ReinstallProduct ( _  
    product As String, _  
    reinstallModes As ReinstallModes _  
)
```

```
public:  
static void ReinstallProduct(  
    String^ product,  
    ReinstallModes reinstallModes  
)
```

### Parameters

#### ***product* (String)**

Product code for the product to be reinstalled

#### ***reinstallModes* (ReinstallModes)**

Reinstall modes

### Remarks

Win32 MSI API: [MsiReinstallProduct](#)

## **Exceptions**

<b>Exception</b>	<b>Condition</b>
<a href="#">InstallCanceledException</a>	the user exited the installation

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## RemovePatches Method (**patches**, **productCode**, **commandLine**)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**RemovePatches(IList<String>, String, String)**

C# ▾

Removes one or more patches from a single product. To remove a patch from multiple products, RemovePatches must be called for each product.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static void RemovePatches(  
    IList<string> patches,  
    string productCode,  
    string commandLine  
)
```

```
Public Shared Sub RemovePatches ( _  
    patches As IList(Of String), _  
    productCode As String, _  
    commandLine As String _  
)
```

```
public:  
static void RemovePatches(  
    IList<String^>^ patches,  
    String^ productCode,  
    String^ commandLine  
)
```

### Parameters

#### **patches** (**IList<String>**)

List of patches to remove. Each patch can be specified by the GUI of the patch or the full path to the patch package.

### ***productCode* ([String](#))**

The ProductCode (GUID) of the product from which the patches are removed. This parameter cannot be null.

### ***commandLine* ([String](#))**

optional command line property settings

## **[-] Remarks**

The [RebootRequired](#) and [RebootInitiated](#) properties should be tested after calling this method.

Win32 MSI API: [MsiRemovePatches](#)

## **[-] Exceptions**

Exception	Condition
<a href="#">InstallerException</a>	There was an error removing the patches

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## SetExternalUI Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ► SetExternalUI()

C# ▾

### Members

Icon	Member	Description
	<code>SetExternalUI(ExternalUIHandler, InstallLogModes)</code>	Enables an external user-interface handler. This external UI handler is called before the normal internal user-interface handler. The external UI handler has the option to suppress the internal UI by returning a non-zero value to indicate that it has handled the messages.
	<code>SetExternalUI(ExternalUIRecordHandler, InstallLogModes)</code>	[MSI 3.1] Enables a record-based external user-interface handler. This external UI handler is called before the normal internal user-interface handler. The external UI handler has the option to suppress the internal UI by returning a non-zero value to indicate that it has handled the messages.

---

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)  
Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## SetExternalUI Method (uiHandler, messageFilter)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**SetExternalUI(ExternalUIHandler, InstallLogModes)**

C# ▾

Enables an external user-interface handler. This external UI handler is called before the normal internal user-interface handler. The external UI handler has the option to suppress the internal UI by returning a non-zero value to indicate that it has handled the messages.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static ExternalUIHandler SetExternalUI(  
    ExternalUIHandler uiHandler,  
    InstallLogModes messageFilter  
)
```

```
Public Shared Function SetExternalUI ( _  
    uiHandler As ExternalUIHandler, _  
    messageFilter As InstallLogModes _  
) As ExternalUIHandler
```

```
public:  
static ExternalUIHandler^ SetExternalUI(  
    ExternalUIHandler^ uiHandler,  
    InstallLogModes messageFilter  
)
```

### Parameters

#### **uiHandler (ExternalUIHandler)**

A callback delegate that handles the UI messages

#### **messageFilter (InstallLogModes)**

Specifies which messages to handle using the external message

handler. If the external handler returns a non-zero result, then that message will not be sent to the UI, instead the message will be logged if logging has been enabled.

## **■ Return Value**

The previously set external handler, or null if there was no previously set handler

## **■ Remarks**

To restore the previous UI handler, a second call is made to SetExternalUI using the ExternalUIHandler returned by the first call to SetExternalUI and specifying [None](#) as the message filter.

The external user interface handler does not have full control over the external user interface unless [SetInternalUI\(InstallUIOptions\)](#) is called with the uiLevel parameter set to [Silent](#). If SetInternalUI is not called, the internal user interface level defaults to [Basic](#). As a result, any message not handled by the external user interface handler is handled by Windows Installer. The initial "Preparing to install..." dialog always appears even if the external user interface handler handles all messages.

SetExternalUI should only be called from a bootstrapping application. You cannot call it from a custom action

Win32 MSI API: [MsiSetExternalUI](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## SetExternalUI Method (uiHandler, messageFilter)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ►

**SetExternalUI(ExternalUIRecordHandler, InstallLogModes)**

C# ▾

[MSI 3.1] Enables a record-based external user-interface handler. This external UI handler is called before the normal internal user-interface handler. The external UI handler has the option to suppress the internal UI by returning a non-zero value to indicate that it has handled the messages.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static ExternalUIRecordHandler SetExternalUI(  
    ExternalUIRecordHandler uiHandler,  
    InstallLogModes messageFilter  
)
```

```
Public Shared Function SetExternalUI ( _  
    uiHandler As ExternalUIRecordHandler, _  
    messageFilter As InstallLogModes _  
) As ExternalUIRecordHandler
```

```
public:  
static ExternalUIRecordHandler^ SetExternalUI(  
    ExternalUIRecordHandler^ uiHandler,  
    InstallLogModes messageFilter  
)
```

### Parameters

#### **uiHandler (ExternalUIRecordHandler)**

A callback delegate that handles the UI messages

#### **messageFilter (InstallLogModes)**

Specifies which messages to handle using the external message handler. If the external handler returns a non-zero result, then that message will not be sent to the UI, instead the message will be logged if logging has been enabled.

## **Return Value**

The previously set external handler, or null if there was no previously set handler

## **Remarks**

To restore the previous UI handler, a second call is made to SetExternalUI using the ExternalUIHandler returned by the first call to SetExternalUI and specifying [None](#) as the message filter.

The external user interface handler does not have full control over the external user interface unless [SetInternalUI\(InstallUIOptions\)](#) is called with the uiLevel parameter set to [Silent](#). If SetInternalUI is not called, the internal user interface level defaults to [Basic](#). As a result, any message not handled by the external user interface handler is handled by Windows Installer. The initial "Preparing to install..." dialog always appears even if the external user interface handler handles all messages.

SetExternalUI should only be called from a bootstrapping application. You cannot call it from a custom action

Win32 MSI API: [MsiSetExternalUIRecord](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## SetInternalUI Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ► SetInternalUI()

C# ▾

### Members

Icon	Member	Description
 S	<a href="#">SetInternalUI(InstallUIOptions, IntPtr)</a>	Enables the installer's internal user interface. Then this user interface is used for all subsequent calls to user-interface-generating installer functions in this process.
 S	<a href="#">SetInternalUI(InstallUIOptions)</a>	Enables the installer's internal user interface. Then this user interface is used for all subsequent calls to user-interface-generating installer functions in this process. The owner of the user interface does not change.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **SetInternalUI Method (uiOptions)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ►

### **SetInternalUI(InstallUIOptions)**

C# ▾

Enables the installer's internal user interface. Then this user interface is used for all subsequent calls to user-interface-generating installer functions in this process. The owner of the user interface does not change.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static InstallUIOptions SetInternalUI(  
    InstallUIOptions uiOptions  
)
```

```
Public Shared Function SetInternalUI ( _  
    uiOptions As InstallUIOptions _  
) As InstallUIOptions
```

```
public:  
static InstallUIOptions SetInternalUI(  
    InstallUIOptions uiOptions  
)
```

#### Parameters

##### **uiOptions (InstallUIOptions)**

Specifies the level of complexity of the user interface

#### Return Value

The previous user interface level

#### Remarks

Win32 MSI API: [MsiSetInternalUI](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## SetInternalUI Method (uiOptions, windowHandle)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**SetInternalUI(InstallUIOptions, IntPtr)**

C# ▾

Enables the installer's internal user interface. Then this user interface is used for all subsequent calls to user-interface-generating installer functions in this process.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static InstallUIOptions SetInternalUI(  
    InstallUIOptions uiOptions,  
    ref IntPtr windowHandle  
)
```

```
Public Shared Function SetInternalUI ( _  
    uiOptions As InstallUIOptions, _  
    ByRef windowHandle As IntPtr _  
) As InstallUIOptions
```

```
public:  
static InstallUIOptions SetInternalUI(  
    InstallUIOptions uiOptions,  
    IntPtr% windowHandle  
)
```

### Parameters

#### **uiOptions (InstallUIOptions)**

Specifies the level of complexity of the user interface

#### **windowHandle ( IntPtr )**

Handle to a window, which becomes the owner of any user interface created. A pointer to the previous owner of the user interface is

returned.

## ❑ Return Value

The previous user interface level

## ❑ Remarks

Win32 MSI API: [MsiSetInternalUI](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## UseFeature Method (**productCode**, **feature**, **installMode**)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**UseFeature(String, String, InstallMode)**

C# ▾

increments the usage count for a particular feature and returns the installation state for that feature. This method should be used to indicate an application's intent to use a feature.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static InstallState UseFeature(  
    string productCode,  
    string feature,  
    InstallMode installMode  
)
```

```
Public Shared Function UseFeature ( _  
    productCode As String, _  
    feature As String, _  
    installMode As InstallMode _  
) As InstallState
```

```
public:  
static InstallState UseFeature(  
    String^ productCode,  
    String^ feature,  
    InstallMode installMode  
)
```

### Parameters

#### **productCode (String)**

The product code of the product.

***feature* ([String](#))**

The feature to be used.

***installMode* ([InstallMode](#))**

Must have the value [NoDetection](#).

**▀ Return Value**

The installed state of the feature.

**▀ Remarks**

The UseFeature method should only be used on features known to be published. The application should determine the status of the feature by calling either the FeatureState method or Features method.

Win32 MSI APIs: [MsiUseFeature](#), [MsiUseFeatureEx](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation VerifyPackage Method (*packagePath*)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Installer ►  
**VerifyPackage(String)**

C# ▾

Verifies that the given file is an installation package.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static bool VerifyPackage(  
    string packagePath  
)
```

```
Public Shared Function VerifyPackage ( _  
    packagePath As String _  
) As Boolean
```

```
public:  
static bool VerifyPackage(  
    String^ packagePath  
)
```

## Parameters

### *packagePath* ([String](#))

Path to the package

## Return Value

True if the file is an installation package; false otherwise.

## Remarks

Win32 MSI API: [MsiVerifyPackage](#)

## Exceptions

Exception	Condition
FileNotFoundException	the specified package file does not exist
InstallerException	the package file could not be opened

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Version Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Installer ► Version

C# ▾

Gets the current version of the installer.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public static Version Version { get; }
```

```
Public Shared ReadOnly Property Version As Version
```

```
public:  
static property Version^ Version {  
    Version^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **InstallerException Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► **InstallerException**

C# ▾

Base class for Windows Installer exceptions.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
[SerializableAttribute]
public class InstallerException : SystemException
```

```
<SerializableAttribute> _
Public Class InstallerException _
    Inherits SystemException
```

```
[SerializableAttribute]
public ref class InstallerException : public SystemE
```

## Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon	Member	Description
	InstallerException(String, Exception)	Creates a new InstallerException object with the specified error message and inner exception.
	InstallerException(String)	Creates a new InstallerException object with the specified error message.

	 <a href="#">InstallerException()</a>	Creates a new InstallerException object.
	 <a href="#">InstallerException(SerializationInfo, StreamingContext)</a>	Initializes a new instance of the InstallerException class with data.
	<a href="#">Data</a>	Gets a collection of key/value pairs that provide additional, user-defined information about the exception.
		(Inherited from <a href="#">Exception</a> .)
	<a href="#">Equals(Object)</a>	Determines whether the specified object is equal to the current <a href="#">Object</a> .
		(Inherited from <a href="#">Object</a> .)
	<a href="#">ErrorCode</a>	Gets the system error code associated with this exception, or 0 if not applicable.
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other operations before the <a href="#">Object</a> is destroyed by garbage collection.
		(Inherited from <a href="#">Object</a> .)
	<a href="#">GetBaseException()</a>	When overridden in a derived class, returns the <a href="#">Exception</a> that is the base of one or more subsequent exceptions.
		(Inherited from <a href="#">Exception</a> .)
	<a href="#">GetErrorRecord()</a>	Gets extended information about the exception, or null if no further information is available.

	 <a href="#">GetHashCode()</a>	Serves as a hash function for a type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures that require a hash table.
		(Inherited from <a href="#">Object</a> .)
	 <a href="#">GetObjectData(SerializationInfo, StreamingContext)</a>	Sets the <a href="#">SerializationInfo</a> with information about the exception.
		(Overrides <a href="#">Exception.GetObjectData(SerializationInfo, StreamingContext)</a> .)
	 <a href="#">GetType()</a>	Gets the runtime type of the instance.
		(Inherited from <a href="#">Exception</a> .)
	<a href="#">HelpLink</a>	Gets or sets a link to the help file associated with this exception.
		(Inherited from <a href="#">Exception</a> .)
	<a href="#">HResult</a>	Gets or sets HRESULT, a numeric value that is assigned to a specific exception.
		(Inherited from <a href="#">Exception</a> .)
	<a href="#">InnerException</a>	Gets the <a href="#">Exception</a> instance that caused the current exception.
		(Inherited from <a href="#">Exception</a> .)
 	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the <a href="#">Object</a> .
		(Inherited from <a href="#">Object</a> .)
	<a href="#">Message</a>	Gets a message that describes the exception. This message may be localized.

		detailed formatted error data available.
		(Overrides <a href="#">Exception.Message</a> .)
	<a href="#">Source</a>	Gets or sets the name of the object that causes the error.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">StackTrace</a>	Gets a string representation on the call stack at the time the exception was thrown.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">TargetSite</a>	Gets the method that throws the exception.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">ToString()</a>	Creates and returns a string representation of the current exception.  (Inherited from <a href="#">Exception</a> .)

## Inheritance Hierarchy




---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

[◀]

[▶]

Deployment Tools Foundation **InstallerException Constructor**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallerException ►

**InstallerException()**

C# ▾

▀ **Members**

Icon	Member	Description
≡ ♪	<a href="#">InstallerException(String, Exception)</a>	Creates a new InstallerException with a specified error message and a reference to the inner exception that is the cause of this exception.
≡ ♪	<a href="#">InstallerException(String)</a>	Creates a new InstallerException with a specified error message.
≡ ♪	<a href="#">InstallerException()</a>	Creates a new InstallerException.
💡 ♪	<a href="#">InstallerException(SerializationInfo, StreamingContext)</a>	Initializes a new instance of the InstallerException class with serialized data.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **InstallerException Constructor**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallerException ►

**InstallerException()**

C# ▾

Creates a new InstallerException.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public InstallerException()
```

```
Public Sub New
```

```
public:  
InstallerException()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **InstallerException Constructor (info, context)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► InstallerException ►  
**InstallerException(SerializationInfo, StreamingContext)**

C# ▾

Initializes a new instance of the InstallerException class with serialized data.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
protected InstallerException(  
    SerializationInfo info,  
    StreamingContext context  
)
```

```
Protected Sub New ( _  
    info As SerializationInfo, _  
    context As StreamingContext _  
)
```

```
protected:  
InstallerException(  
    SerializationInfo^ info,  
    StreamingContext context  
)
```

[-] **Parameters**

***info* (SerializationInfo)**

The SerializationInfo that holds the serialized object data about the exception being thrown.

***context* (StreamingContext)**

The StreamingContext that contains contextual information about the source or destination.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## InstallerException Constructor (msg)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► InstallerException ►  
**InstallerException(String)**

C# ▾

Creates a new InstallerException with a specified error message.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public InstallerException(  
    string msg  
)
```

```
Public Sub New ( _  
    msg As String _  
)
```

```
public:  
InstallerException(  
    String^ msg  
)
```

### Parameters

#### *msg* (String)

The message that describes the error.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# InstallerException Constructor (msg, innerException)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallerException ►

**InstallerException(String, Exception)**

C# ▾

Creates a new InstallerException with a specified error message and a reference to the inner exception that is the cause of this exception.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public InstallerException(
    string msg,
    Exception innerException
)
```

```
Public Sub New ( _
    msg As String, _
    innerException As Exception _
)
```

```
public:
InstallerException(
    String^ msg,
    Exception^ innerException
)
```

## Parameters

### *msg* (String)

The message that describes the error.

### *innerException* (Exception)

The exception that is the cause of the current exception. If the innerException parameter is not a null reference (Nothing in Visual Basic), the current exception is raised in a catch block that handles

the inner exception.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ErrorCode Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallerException ►

### ErrorCode

C# ▾

Gets the system error code that resulted in this exception, or 0 if not applicable.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public int ErrorCode { get; }
```

```
Public ReadOnly Property ErrorCode As Integer
```

```
public:  
property int ErrorCode {  
    int get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetErrorRecord** Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallerException ►

### **GetErrorRecord()**

C# ▾

Gets extended information about the error, or null if no further information is available.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public Record GetErrorRecord()
```

```
Public Function GetErrorRecord As Record
```

```
public:  
Record^ GetErrorRecord()
```

#### Return Value

A Record object. Field 1 of the Record contains the installer message code. Other fields contain data specific to the particular error.

#### Remarks

If the record is passed to [Message\(InstallMessage, Record\)](#), it is formatted by looking up the string in the current database. If there is no installation session, the formatted error message may be obtained by a query on the Error table using the error code, followed by a call to [ToString\(\)](#). Alternatively, the standard MSI message can be retrieved by calling the [GetErrorMessage\(Record, CultureInfo\)](#) method.

The following methods and properties may report extended error data:

- [Database](#) (constructor)
- [Database.ApplyTransform\(String, TransformErrors\)](#)
- [Database.Commit\(\)](#)
- [Database.Execute\(String, Object\[\]\)](#)

- Database.ExecuteQuery(String, Object[])
- Database.ExecuteIntegerQuery(String, Object[])
- Database.ExecuteStringQuery(String, Object[])
- Database.Export(String, String)
- Database.ExportAll(String)
- Database.GenerateTransform(Database, String)
- Database.Import(String)
- Database.ImportAll(String)
- Database.Merge(Database, String)
- Database.OpenView(String, Object[])
- Database.SummaryInfo
- Database.ViewTransform(String)
- View.Assign(Record)
- View.Delete(Record)
- View.Execute(Record)
- View.Insert(Record)
- View.InsertTemporary(Record)
- View.Merge(Record)
- View.Modify(ViewModifyMode, Record)
- View.Refresh(Record)
- View.Replace(Record)
- View.Seek(Record)
- View.Update(Record)
- View.Validate(Record)
- View.ValidateFields(Record)
- View.ValidateDelete(Record)
- View.ValidateNew(Record)
- SummaryInfo (constructor)
- Record.SetStream(Int32, String)
- Session.SetInstallLevel(Int32)
- Session.GetSourcePath(String)
- Session.GetTargetPath(String)
- Session.SetTargetPath(String, String)
- ComponentInfo.CurrentState
- FeatureInfo.CurrentState
- FeatureInfo.ValidStates
- FeatureInfo.GetCost(Boolean, Boolean, InstallState)

The Record object should be [Close\(\)](#)d after use. It is best that the handl

be closed manually as soon as it is no longer needed, as leaving lots of unused handles open can degrade performance.

Win32 MSI API: [MsiGetLastErrorRecord](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetObjectData Method (info, context)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► InstallerException ►  
**GetObjectData(SerializationInfo, StreamingContext)**

C# ▾

Sets the SerializationInfo with information about the exception.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public override void GetObjectData(
    SerializationInfo info,
    StreamingContext context
)
```

```
Public Overrides Sub GetObjectData ( _
    info As SerializationInfo, _
    context As StreamingContext _ )
)
```

```
public:
virtual void GetObjectData(
    SerializationInfo^ info,
    StreamingContext context
) override
```

[-] **Parameters**

**info (SerializationInfo)**

The SerializationInfo that holds the serialized object data about the exception being thrown.

**context (StreamingContext)**

The StreamingContext that contains contextual information about the source or destination.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Message Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallerException ►

### Message

C# ▾

Gets a message that describes the exception. This message may contain detailed formatted error data if it was available.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override string Message { get; }
```

```
Public Overrides ReadOnly Property Message As String
```

```
public:  
virtual property String^ Message {  
    String^ get () override;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **InstallerHandle** Class

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► **InstallerHandle**

C#

Base class for Windows Installer handle types (Database, View, Record, SummaryInfo).

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public abstract class InstallerHandle : MarshalByRef
```

```
Public MustInherit Class InstallerHandle _  
    Inherits MarshalByRefObject _  
    Implements IDisposable
```

```
public ref class InstallerHandle abstract : public M  
    IDisposable
```

## Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon	Member	Description
💡	<a href="#">InstallerHandle(IntPtr, Boolean)</a>	Constructs a handle object from a native integer handle.
💡	<a href="#">Close()</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .

 	<a href="#">CreateObjRef(Type)</a>	Creates an object that contains all the relevant information required to generate a proxy used to communicate with a remote object.  (Inherited from <a href="#">MarshalByRefObject</a> .)
 	<a href="#">Dispose()</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .
 	<a href="#">Dispose(Boolean)</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .
 	<a href="#">Equals(Object)</a>	Tests whether this handle object is equal to another handle object. Two handle objects are equal if their types are the same and their native integer handles are the same.  (Overrides <a href="#">Object.Equals(Object)</a> .)
 	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
 	<a href="#">GetHashCode()</a>	Gets a hash value for the handle object.  (Overrides <a href="#">Object.GetHashCode()</a> .)
 	<a href="#">GetLifetimeService()</a>	Retrieves the current lifetime service object that controls the

		<p>lifetime policy for this instance.</p> <p>(Inherited from <a href="#">MarshalByRefObject</a>.)</p>
	<a href="#">GetType()</a>	<p>Gets the <a href="#">Type</a> of the current instance.</p> <p>(Inherited from <a href="#">Object</a>.)</p>
	<a href="#">Handle</a>	Gets the native integer handle.
	<a href="#">InitializeLifetimeService()</a>	<p>Obtains a lifetime service object to control the lifetime policy for this instance.</p> <p>(Inherited from <a href="#">MarshalByRefObject</a>.)</p>
	<a href="#">IsClosed</a>	<p>Checks if the handle is closed. When closed, method calls on the handle object may throw an <a href="#">InvalidOperationException</a>.</p>
	<a href="#">MemberwiseClone(Boolean)</a>	<p>Creates a shallow copy of the current <a href="#">MarshalByRefObject</a> object</p> <p>(Inherited from <a href="#">MarshalByRefObject</a>.)</p>
	<a href="#">MemberwiseClone()</a>	<p>Creates a shallow copy of the current <a href="#">Object</a>.</p> <p>(Inherited from <a href="#">Object</a>.)</p>
	<a href="#">ToString()</a>	<p>Returns a <a href="#">String</a> that represents the current <a href="#">Object</a>.</p> <p>(Inherited from <a href="#">Object</a>.)</p>

---

## ■ Remarks

These classes implement the [IDisposable](#) interface, because they hold unmanaged resources (MSI handles) that should be properly disposed when no longer needed.

## ■ Inheritance Hierarchy

Object

└ MarshalByRefObject

  └ InstallerHandle

    └ Database

    └ Transaction

    └ Record

    └ Session

    └ SummaryInfo

    └ View

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## InstallerHandle Constructor (handle, ownsHandle)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► InstallerHandle ►  
InstallerHandle(IntPtr, Boolean)

C# ▾

Constructs a handle object from a native integer handle.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected InstallerHandle(  
    IntPtr handle,  
    bool ownsHandle  
)
```

```
Protected Sub New ( _  
    handle As IntPtr, _  
    ownsHandle As Boolean _  
)
```

```
protected:  
InstallerHandle(  
    IntPtr handle,  
    bool ownsHandle  
)
```

### Parameters

#### *handle* (IntPtr)

Native integer handle.

#### *ownsHandle* (Boolean)

true to close the handle when this object is disposed or finalized

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Close Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallerHandle ► **Close()**

C# ▾

Closes the handle. After closing a handle, further method calls may throw an [InvalidOperationException](#).

### [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Close()
```

```
Public Sub Close
```

```
public:  
void Close()
```

### [-] Remarks

The finalizer of this class will NOT close the handle if it is still open, because finalization can run on a separate thread from the application, resulting in potential problems if handles are closed from that thread. It is best that the handle be closed manually as soon as it is no longer needed, as leaving lots of unused handles open can degrade performance.

This method is merely an alias for the [Dispose\(\)](#) method.

Win32 MSI API: [MsiCloseHandle](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Dispose Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallerHandle ► Dispose()

C# ▾

### Members

Icon	Member	Description
	<a href="#">Dispose()</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .
	<a href="#">Dispose(Boolean)</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Dispose Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallerHandle ► Dispose()

C#

Closes the handle. After closing a handle, further method calls may throw an [InvalidOperationException](#).

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Dispose()
```

```
Public Sub Dispose
```

```
public:  
virtual void Dispose() sealed
```

### - Remarks

The finalizer of this class will NOT close the handle if it is still open, because finalization can run on a separate thread from the application, resulting in potential problems if handles are closed from that thread. It is best that the handle be closed manually as soon as it is no longer needed, as leaving lots of unused handles open can degrade performance.

Win32 MSI API: [MsiCloseHandle](#)

### - See Also

[Close\(\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Dispose Method (disposing)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallerHandle ►

## Dispose(Boolean)

C# ▾

Closes the handle. After closing a handle, further method calls may throw an [InvalidOperationException](#).

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected virtual void Dispose(  
    bool disposing  
)
```

```
Protected Overridable Sub Dispose ( _  
    disposing As Boolean _  
)
```

```
protected:  
virtual void Dispose(  
    bool disposing  
)
```

### Parameters

#### *disposing* (Boolean)

If true, the method has been called directly or indirectly by a user's code, so managed and unmanaged resources will be disposed. If false, the method has been called by the runtime from inside the finalizer, and only unmanaged resources will be disposed.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Equals Method (obj)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallerHandle ►

### Equals(Object)

C# ▾

Tests whether this handle object is equal to another handle object. Two handle objects are equal if their types are the same and their native integer handles are the same.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override bool Equals(  
    Object obj  
)
```

```
Public Overrides Function Equals ( _  
    obj As Object _  
) As Boolean
```

```
public:  
virtual bool Equals(  
    Object^ obj  
) override
```

### Parameters

#### *obj* (Object)

The handle object to compare with the current handle object.

### Return Value

true if the specified handle object is equal to the current handle object; otherwise false

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetHashCode Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallerHandle ►

**GetHashCode()**

C# ▾

Gets a hash value for the handle object.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public override int GetHashCode()
```

```
Public Overrides Function GetHashCode As Integer
```

```
public:  
virtual int GetHashCode() override
```

[-] **Return Value**

A hash code for the handle object.

[-] **Remarks**

The hash code is derived from the native integer handle.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools FoundationHandle Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallerHandle ► Handle

C# ▾

Gets the native integer handle.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public IntPtr Handle { get; }
```

```
Public ReadOnly Property Handle As IntPtr
```

```
public:  
property IntPtr Handle {  
    IntPtr get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation IsClosed Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallerHandle ► IsClosed

C# ▾

Checks if the handle is closed. When closed, method calls on the handle object may throw an [InvalidHandleException](#).

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool IsClosed { get; }
```

```
Public ReadOnly Property IsClosed As Boolean
```

```
public:  
property bool IsClosed {  
    bool get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **InstallLogModes Enumeration**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► **InstallLogModes**

C# ▾

Log modes available for [EnableLog\(InstallLogModes, String\)](#) and [SetExternalUI\(ExternalUIHandler, InstallLogModes\)](#).

## Declaration Syntax

C#

Visual Basic

Visual C++

```
[FlagsAttribute]
public enum InstallLogModes
```

```
<FlagsAttribute> _
Public Enumeration InstallLogModes
```

```
[FlagsAttribute]
public enum class InstallLogModes
```

## Members

Member	Description
<b>None</b>	Disable logging.
<b>FatalExit</b>	Log out of memory or fatal exit information.
<b>Error</b>	Log error messages.
<b>Warning</b>	Log warning messages.

<b>User</b>	Log user requests.
<b>Info</b>	Log status messages that are not displayed.
<b>ResolveSource</b>	Log request to determine a valid source location.
<b>OutOfDiskSpace</b>	Log insufficient disk space error.
<b>ActionStart</b>	Log the start of installation actions.
<b>ActionData</b>	Log the data record for installation actions.
<b>CommonData</b>	Log parameters for user-interface initialization.
<b>PropertyDump</b>	Log the property values at termination.
<b>Verbose</b>	Sends large amounts of information to log file not generally useful to users. May be used for support.
<b>ExtraDebug</b>	Log extra debugging information.

<b>LogOnlyOnError</b>	Log only on error.
<b>Progress</b>	Log progress bar information. This message includes information on units so far and total number of units. See <a href="#">Message(InstallMessage, Record)</a> for an explanation of the message format. This message is only sent to an external user interface and is not logged.
<b>Initialize</b>	If this is not a quiet installation, then the basic UI has been initialized. If this is a full UI installation, the Full UI is not yet initialized. This message is only sent to an external user interface and is not logged.
<b>Terminate</b>	If a full UI is being used, the full UI has ended. If this is not a quiet installation, the basic UI has not yet ended. This message is only sent to an external user interface and is not logged.
<b>ShowDialog</b>	Sent prior to display of the Full UI dialog. This message is only sent to an external user interface and is not logged.
<b>FilesInUse</b>	List of files in use that need to be replaced.
<b>RMFilesInUse</b>	[MSI 4.0] List of apps that the user can request Restart Manager to shut down

and restart.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# InstallMessage Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallMessage

C# ▾

Type of message to be processed by [Message\(InstallMessage, Record\)](#), [ExternalUIHandler](#), or [ExternalUIRecordHandler](#).

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public enum InstallMessage
```

```
Public Enumeration InstallMessage
```

```
public enum class InstallMessage
```

## Members

Member	Description
<b>FatalExit</b>	Premature termination, possibly fatal OOM.
<b>Error</b>	Formatted error message.
<b>Warning</b>	Formatted warning message.
<b>User</b>	User request message.
<b>Info</b>	Informative message for log.

<b>FilesInUse</b>	List of files in use that need to be replaced.
<b>ResolveSource</b>	Request to determine a valid source location.
<b>OutOfDiskSpace</b>	Insufficient disk space message.
<b>ActionStart</b>	Start of action: action name & description
<b>ActionData</b>	Formatted data associated with individual action item.
<b>Progress</b>	Progress gauge info: units so far, total.
<b>CommonData</b>	Product info for dialog: language Id, dialog caption.
<b>Initialize</b>	Sent prior to UI initialization, no string data.
<b>Terminate</b>	Sent after UI termination, no string data.
<b>ShowDialog</b>	Sent prior to display or authored dialog or wizard.

<b>RMFilesInUse</b>	[MSI 4.0] List of apps that the user can request Restart Manager to shut down and restart.
<b>InstallStart</b>	[MSI 4.5] Sent prior to install of a product.
<b>InstallEnd</b>	[MSI 4.5] Sent after install of a product.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

C# ▾

Specifies the install mode for [ProvideComponent\(String, String, String, InstallMode\)](#) or [ProvideQualifiedComponent\(String, String, InstallMode, String\)](#).

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public enum InstallMode
```

```
Public Enumeration InstallMode
```

```
public enum class InstallMode
```

## Members

Member	Description
<b>NoSourceResolution</b>	Provide the component only if the feature's installation state is <a href="#">Local</a> .
<b>NoDetection</b>	Only check that the component is registered, without verifying that the key file of the component exists.
<b>Existing</b>	Provide the component only if the feature exists.
<b>Default</b>	Provide the component and perform any installation necessary to provide the component.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## InstallRunMode Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallRunMode

C# ▾

Specifies the run mode for [GetMode\(InstallRunMode\)](#).

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public enum InstallRunMode
```

```
Public Enumeration InstallRunMode
```

```
public enum class InstallRunMode
```

### - Members

Member	Description
<b>Admin</b>	The administrative mode is installing, or the product is installing.
<b>Advertise</b>	The advertisements are installing or the product is installing or updating.
<b>Maintenance</b>	An existing installation is being modified or there is a new installation.
<b>RollbackEnabled</b>	Rollback is enabled.
<b>LogEnabled</b>	The log file is active. It was enabled prior to the installation session.

<b>Operations</b>	Execute operations are spooling or they are in the determination phase.
<b>RebootAtEnd</b>	A reboot is necessary after a successful installation (settable).
<b>RebootNow</b>	A reboot is necessary to continue the installation (settable).
<b>Cabinet</b>	Files from cabinets and Media table files are installing.
<b>SourceShortNames</b>	The source LongFileNames is suppressed through the PID_MSISOURCE summary property.
<b>TargetShortNames</b>	The target LongFileNames is suppressed through the SHORTFILENAMES property
<b>Windows9x</b>	The operating system is Windows 95, Windows 98, or Windows ME.
<b>ZawEnabled</b>	The operating system supports demand installation.
<b>Scheduled</b>	A custom action called from install script

	execution.
<b>Rollback</b>	A custom action called from rollback execution script.
<b>Commit</b>	A custom action called from commit execution script.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## InstallState Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallState

C# ▾

Installed state of a Component or Feature.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public enum InstallState
```

```
Public Enumeration InstallState
```

```
public enum class InstallState
```

### - Members

Member	Description
<b>NotUsed</b>	The component is disabled.
<b>BadConfig</b>	The installation configuration data is corrupt.
<b>Incomplete</b>	The installation is suspended or in progress.
<b>SourceAbsent</b>	Component is set to run from source, but source is unavailable.
<b>MoreData</b>	The buffer overflow is returned.

<b>InvalidArgument</b>	An invalid parameter was passed to the function.
<b>Unknown</b>	An unrecognized product or feature name was passed to the function.
<b>Broken</b>	The component is broken.
<b>Advertised</b>	The feature is advertised.
<b>Removed</b>	The component is being removed. In action state and not settable.
<b>Absent</b>	The component is not installed, or action state is absent but clients remain.
<b>Local</b>	The component is installed on the local drive.
<b>Source</b>	The component will run from the source, CD, or network.
<b>Default</b>	The component will be installed in the default location: local or source.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## InstallType Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InstallType

C#

Specifies the type of installation for `ApplyPatch(String, String, InstallType, String)`.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public enum InstallType
```

```
Public Enumeration InstallType
```

```
public enum class InstallType
```

### Members

Member	Description
<b>Default</b>	Searches system for products to patch.
<b>NetworkImage</b>	Indicates a administrative installation.
<b>SingleInstance</b>	Indicates a particular instance.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **InstallUIOptions Enumeration**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► **InstallUIOptions**

C# ▾

Level of the installation user interface, specified with [SetInternalUI\(InstallUIOptions\)](#).

## Declaration Syntax

C#

Visual Basic

Visual C++

```
[FlagsAttribute]
public enum InstallUIOptions
```

```
<FlagsAttribute> _
Public Enumeration InstallUIOptions
```

```
[FlagsAttribute]
public enum class InstallUIOptions
```

## Members

Member	Description
NoChange	Does not change UI level.
Default	Uses Default UI level.
Silent	Silent installation.
Basic	Simple progress and error handling.
Reduced	Authored UI, wizard dialogs suppressed.

<b>Full</b>	Authored UI with wizards, progress, and errors.
<b>HideCancel</b>	When combined with the <b>Basic</b> value, the installer does not display the cancel button in the progress dialog.
<b>ProgressOnly</b>	When combined with the <b>Basic</b> value, the installer displays progress dialog boxes but does not display any modal dialog boxes or error dialog boxes.
<b>EndDialog</b>	When combined with another value, the installer displays a modal dialog box at the end of a successful installation or if there has been an error. No dialog box is displayed if the user cancels.
<b>SourceResolutionOnly</b>	Forces display of the source resolution dialog even if the UI is otherwise silent.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **InvalidOperationException Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InvalidOperationException

C# ▾

A method was called on an invalid installer handle. The handle may have been already closed.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
[SerializableAttribute]
public class InvalidOperationException : InstallerExcep
```

```
<SerializableAttribute> _
Public Class InvalidOperationException _
    Inherits InstallerException
```

```
[SerializableAttribute]
public ref class InvalidOperationException : public Ins
```

### [-] Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declare

Inherit

Icon	Member	Description
≡	InvalidOperationException(String, Exception)	Creates a new InvalidOperationException with a specified error message and a reference to the inner exception that is the cause of this exception.

	<a href="#">InvalidHandleException(String)</a>	Creates a new InvalidHandleException with a specified error message.
	<a href="#">InvalidHandleException()</a>	Creates a new InvalidHandleException
	<a href="#">InvalidHandleException(SerializationInfo, StreamingContext)</a>	Initializes a new instance of the InvalidHandleException class with serialized data.
	<a href="#">Data</a>	Gets a collection of key/value pairs that provide additional, user-defined information about the exception.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">ErrorCode</a>	Gets the system error code that resulted in this exception, or 0 if not applicable.

		(Inherited from <a href="#">InstallerException</a> .)
💡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.
		(Inherited from <a href="#">Object</a> .)
💡	<a href="#">GetBaseException()</a>	When overridden in a derived class, returns the <a href="#">Exception</a> that is the root cause of one or more subsequent exceptions.
		(Inherited from <a href="#">Exception</a> .)
💡	<a href="#">GetErrorRecord()</a>	Gets extended information about the error, or null if no further information is available.
		(Inherited from <a href="#">InstallerException</a> .)
💡	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.
		(Inherited from <a href="#">Object</a> .)
💡	<a href="#">GetObjectData(SerializationInfo,</a>	Sets the

	<code>StreamingContext)</code>	SerializationInfo with information about the exception.  (Inherited from <a href="#">InstallerException</a> .)
	<code>GetType()</code>	Gets the runtime type of the current instance.  (Inherited from <a href="#">Exception</a> .)
	<code>HelpLink</code>	Gets or sets a link to the help file associated with this exception.  (Inherited from <a href="#">Exception</a> .)
	<code>HResult</code>	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception.  (Inherited from <a href="#">Exception</a> .)
	<code>InnerException</code>	Gets the <a href="#">Exception</a> instance that caused the current exception.  (Inherited from <a href="#">Exception</a> .)
	<code>MemberwiseClone()</code>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<code>Message</code>	Gets a message that describes the exception. This

		message may contain detailed formatted error data if it was available.
		(Inherited from <a href="#">InstallerException</a> .)
	<a href="#">Source</a>	Gets or sets the name of the application or the object that causes the error.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">StackTrace</a>	Gets a string representation of the frames on the call stack at the time the current exception was thrown.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">TargetSite</a>	Gets the method that throws the current exception.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">ToString()</a>	Creates and returns a string representation of the current exception.  (Inherited from <a href="#">Exception</a> .)

## Inheritance Hierarchy

[Object](#)  
 ↘ [Exception](#)

- └ SystemException
  - └ InstallerException
  - └ InvalidHandleException
- 

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **InvalidOperationException**

## Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InvalidOperationException ►

**InvalidOperationException()**

C# ▾

### Members

Icon	Member	Description
✖	<a href="#">InvalidOperationException(String, Exception)</a>	Creates a new InvalidOperationException with a specified error message and a reference to the inner exception that is the cause of this exception.
✖	<a href="#">InvalidOperationException(String)</a>	Creates a new InvalidOperationException with a specified error message.
✖	<a href="#">InvalidOperationException()</a>	Creates a new InvalidOperationException.
💡	<a href="#">InvalidOperationException(SerializationInfo, StreamingContext)</a>	Initializes a new instance of the InvalidOperationException class with serialized data.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **InvalidOperationException**

## Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InvalidOperationException ►

**InvalidOperationException()**

C# ▾

Creates a new InvalidOperationException.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public InvalidOperationException()
```

```
Public Sub New
```

```
public:  
InvalidOperationException()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **InvalidOperationException**

## Constructor (info, context)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InvalidOperationException ►

**InvalidOperationException(SerializationInfo, StreamingContext)**

C# ▾

Initializes a new instance of the `InvalidOperationException` class with serialized data.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected InvalidOperationException(  
    SerializationInfo info,  
    StreamingContext context  
)
```

```
Protected Sub New ( _  
    info As SerializationInfo, _  
    context As StreamingContext _  
)
```

```
protected:  
InvalidOperationException(  
    SerializationInfo^ info,  
    StreamingContext context  
)
```

### Parameters

#### **info (SerializationInfo)**

The `SerializationInfo` that holds the serialized object data about the exception being thrown.

#### **context (StreamingContext)**

The `StreamingContext` that contains contextual information about the source or destination.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **InvalidOperationException**

## Constructor (msg)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InvalidOperationException ►

### InvalidOperationException(String)

C# ▾

Creates a new InvalidOperationException with a specified error message.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public InvalidOperationException(  
    string msg  
)
```

```
Public Sub New ( _  
    msg As String _  
)
```

```
public:  
InvalidOperationException(  
    String^ msg  
)
```

#### Parameters

##### *msg* (String)

The message that describes the error.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## InvalidOperationException Constructor (msg, innerException)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► InvalidOperationException ►

InvalidOperationException(String, Exception)

C#

Creates a new InvalidOperationException with a specified error message and a reference to the inner exception that is the cause of this exception.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public InvalidOperationException(  
    string msg,  
    Exception innerException  
)
```

```
Public Sub New ( _  
    msg As String, _  
    innerException As Exception _  
)
```

```
public:  
InvalidOperationException(  
    String^ msg,  
    Exception^ innerException  
)
```

### Parameters

#### msg (String)

The message that describes the error.

#### innerException (Exception)

The exception that is the cause of the current exception. If the innerException parameter is not a null reference (Nothing in Visual Basic), the current exception is raised in a catch block that handles

the inner exception.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **LocatorTypes** Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► **LocatorTypes**

C# ▾

Defines values for the Type column of the CompLocator, IniLocator, and RegLocator tables.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
[FlagsAttribute]
public enum LocatorTypes
```

```
<FlagsAttribute> _
Public Enumeration LocatorTypes
```

```
[FlagsAttribute]
public enum class LocatorTypes
```

## Members

Member	Description
<b>Directory</b>	Key path is a directory.
<b>FileName</b>	Key path is a file name.
<b>RawValue</b>	Key path is a registry value.
<b>SixtyFourBit</b>	Set this bit to have the installer search the 64-bit portion of the registry.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **MediaDisk Structure**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► **MediaDisk**

C# ▾

Represents a media disk source of a product or a patch.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public struct MediaDisk
```

```
Public Structure MediaDisk
```

```
public value class MediaDisk
```

### Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon	Member	Description
	<a href="#">MediaDisk(Int32, String, String)</a>	Creates a new media disk.
	<a href="#">DiskId</a>	Gets or sets the disk id of the media disk.
	<a href="#">DiskPrompt</a>	Gets or sets the disk prompt of the media disk.
	<a href="#">Equals(Object)</a>	Indicates whether this instance and a

		specified object are equal.  (Inherited from <a href="#">ValueType</a> .)
💡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">GetHashCode()</a>	Returns the hash code for this instance.  (Inherited from <a href="#">ValueType</a> .)
💡	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">ToString()</a>	Returns the fully qualified type name of this instance.  (Inherited from <a href="#">ValueType</a> .)
📁	<a href="#">VolumeLabel</a>	Gets or sets the volume label of the media disk.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)  
Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## MediaDisk Constructor (diskId, volumeLabel, diskPrompt)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► MediaDisk ►

**MediaDisk(Int32, String, String)**

C# ▾

Creates a new media disk.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public MediaDisk(  
    int diskId,  
    string volumeLabel,  
    string diskPrompt  
)
```

```
Public Sub New ( _  
    diskId As Integer, _  
    volumeLabel As String, _  
    diskPrompt As String _  
)
```

```
public:  
MediaDisk(  
    int diskId,  
    String^ volumeLabel,  
    String^ diskPrompt  
)
```

### Parameters

***diskId* (Int32)**

***volumeLabel* (String)**

***diskPrompt* (String)**

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)  
Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools FoundationDiskId Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► MediaDisk ► DiskId

C# ▾

Gets or sets the disk id of the media disk.

## - Declaration Syntax

C#

Visual Basic

Visual C++

```
public int DiskId { get; set; }
```

```
Public Property DiskId As Integer
```

```
public:  
property int DiskId {  
    int get ();  
    void set (int value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools FoundationDiskPrompt Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► MediaDisk ► DiskPrompt

C# ▾

Gets or sets the disk prompt of the media disk.

## - Declaration Syntax

C#

Visual Basic

Visual C++

```
public string DiskPrompt { get; set; }
```

```
Public Property DiskPrompt As String
```

```
public:  
property String^ DiskPrompt {  
    String^ get ();  
    void set (String^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **VolumeLabel** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► MediaDisk ► **VolumeLabel**

C# ▾

Gets or sets the volume label of the media disk.

## - Declaration Syntax

C#

Visual Basic

Visual C++

```
public string VolumeLabel { get; set; }
```

```
Public Property VolumeLabel As String
```

```
public:  
property String^ VolumeLabel {  
    String^ get ();  
    void set (String^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **MergeException Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► MergeException

C# ▾

A failure occurred when executing [Merge\(Database, String\)](#). The exception may contain details about the merge conflict.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
[SerializableAttribute]
public class MergeException : InstallerException
```

```
<SerializableAttribute> _
Public Class MergeException _
    Inherits InstallerException
```

```
[SerializableAttribute]
public ref class MergeException : public InstallerEx
```

### [-] Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declare

Inherit

Icon	Member	Description
	<a href="#">MergeException(String, Exception)</a>	Creates a new MergeException error message and a reference to the exception that is the cause of this exception.
	<a href="#">MergeException(String)</a>	Creates a new MergeException error message.

	 <a href="#">MergeException()</a>	Creates a new MergeException.
	 <a href="#">MergeException(SerializationInfo, StreamingContext)</a>	Initializes a new instance of the class with serialized data.
	<a href="#">ConflictCounts</a>	Gets the number of merge conflicts corresponding to the tables returned by <a href="#">ConflictTables</a> .
	<a href="#">ConflictTables</a>	Gets the list of tables containing conflicts.
	<a href="#">Data</a>	Gets a collection of key/value pairs containing additional, user-defined information about the exception.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">Equals(Object)</a>	Determines whether the specified object is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">ErrorCode</a>	Gets the system error code thrown by the exception, or 0 if not applicable.  (Inherited from <a href="#">InstallerException</a> .)
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetBaseException()</a>	When overridden in a derived class, gets the base exception.

		<p><a href="#">Exception</a> that is the root cause of subsequent exceptions.</p> <p>(Inherited from <a href="#">Exception</a>.)</p>
	<a href="#">GetErrorRecord()</a>	<p>Gets extended information about the exception if no further information is available.</p> <p>(Inherited from <a href="#">InstallerException</a>.)</p>
	<a href="#">GetHashCode()</a>	<p>Serves as a hash function for objects. The default hash function is suitable for most purposes. If you are creating a type that is intended to be used as a key in a <a href="#">Dictionary</a>, you should override this method.</p> <p>(Inherited from <a href="#">Object</a>.)</p>
	<a href="#">GetObjectData(SerializationInfo, StreamingContext)</a>	<p>Sets the <a href="#">SerializationInfo</a> with the exception.</p> <p>(Overrides <a href="#">InstallerException.GetObjectData(SerializationInfo, StreamingContext)</a>.)</p>
	<a href="#">GetType()</a>	<p>Gets the runtime type of the current instance.</p> <p>(Inherited from <a href="#">Exception</a>.)</p>
	<a href="#">HelpLink</a>	<p>Gets or sets a link to the help file for this exception.</p> <p>(Inherited from <a href="#">Exception</a>.)</p>
	<a href="#">HResult</a>	<p>Gets or sets HRESULT, a code that is assigned to a specific error.</p> <p>(Inherited from <a href="#">Exception</a>.)</p>
	<a href="#">InnerException</a>	<p>Gets the <a href="#">Exception</a> instance that is the root cause of the current exception.</p> <p>(Inherited from <a href="#">Exception</a>.)</p>
	<a href="#">MemberwiseClone()</a>	<p>Creates a shallow copy of the current instance.</p>

		(Inherited from <a href="#">Object</a> .)
	<a href="#">Message</a>	Gets a message that describes the exception.  (Overrides <a href="#">InstallerException.Message</a> .)
	<a href="#">Source</a>	Gets or sets the name of the assembly, class, method, etc. that caused the error.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">StackTrace</a>	Gets a string representation of the call stack at the time the current exception was thrown.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">TargetSite</a>	Gets the method that throws the exception.  (Inherited from <a href="#">Exception</a> .)
	<a href="#">ToString()</a>	Creates and returns a string representation of the current exception.  (Inherited from <a href="#">Exception</a> .)

## Inheritance Hierarchy

```

Object
└ Exception
  └ SystemException
    └ InstallerException
      └ MergeException

```

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## MergeException Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► MergeException ►

**MergeException()**

C# ▾

### ▀ Members

Icon	Member	Description
≡	<a href="#">MergeException(String, Exception)</a>	Creates a new MergeException with a specified error message and a reference to the inner exception that is the cause of this exception.
≡	<a href="#">MergeException(String)</a>	Creates a new MergeException with a specified error message.
≡	<a href="#">MergeException()</a>	Creates a new MergeException.
💡	<a href="#">MergeException(SerializationInfo, StreamingContext)</a>	Initializes a new instance of the MergeException class with serialized data.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## MergeException Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► MergeException ►

**MergeException()**

C# ▾

Creates a new MergeException.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public MergeException()
```

```
Public Sub New
```

```
public:  
MergeException()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## MergeException Constructor (info, context)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► MergeException ►  
MergeException(SerializationInfo, StreamingContext)

C# ▾

Initializes a new instance of the MergeException class with serialized data.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected MergeException(  
    SerializationInfo info,  
    StreamingContext context  
)
```

```
Protected Sub New ( _  
    info As SerializationInfo, _  
    context As StreamingContext _  
)
```

```
protected:  
MergeException(  
    SerializationInfo^ info,  
    StreamingContext context  
)
```

### Parameters

#### info (SerializationInfo)

The SerializationInfo that holds the serialized object data about the exception being thrown.

#### context (StreamingContext)

The StreamingContext that contains contextual information about the source or destination.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## MergeException Constructor (msg)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► MergeException ►  
**MergeException(String)**

C# ▾

Creates a new MergeException with a specified error message.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public MergeException(  
    string msg  
)
```

```
Public Sub New ( _  
    msg As String _  
)
```

```
public:  
MergeException(  
    String^ msg  
)
```

### Parameters

#### *msg* (String)

The message that describes the error.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## MergeException Constructor (msg, innerException)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► MergeException ►  
MergeException(String, Exception)

C# ▾

Creates a new MergeException with a specified error message and a reference to the inner exception that is the cause of this exception.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public MergeException(  
    string msg,  
    Exception innerException  
)
```

```
Public Sub New ( _  
    msg As String, _  
    innerException As Exception _  
)
```

```
public:  
MergeException(  
    String^ msg,  
    Exception^ innerException  
)
```

### Parameters

#### msg (String)

The message that describes the error.

#### innerException (Exception)

The exception that is the cause of the current exception. If the innerException parameter is not a null reference (Nothing in Visual Basic), the current exception is raised in a catch block that handles

the inner exception.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **ConflictCounts Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► MergeException ►

**ConflictCounts**

C#

Gets the number of merge conflicts in each table, corresponding to the tables returned by [ConflictTables](#).

Declaration Syntax

C#

Visual Basic

Visual C++

```
public IList<int> ConflictCounts { get; }
```

```
Public ReadOnly Property ConflictCounts As IList(Of
```

```
public:  
property IList<int>^ ConflictCounts {  
    IList<int>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **ConflictTables Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► MergeException ►

**ConflictTables**

C# ▾

Gets the list of tables containing merge conflicts.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public IList<string> ConflictTables { get; }
```

```
Public ReadOnly Property ConflictTables As IList(Of
```

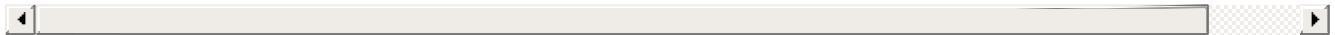
```
public:  
property IList<String^>^ ConflictTables {  
    IList<String^>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **GetObjectData Method (info, context)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► MergeException ►  
**GetObjectData(SerializationInfo, StreamingContext)**

C#

Sets the SerializationInfo with information about the exception.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public override void GetObjectData(
    SerializationInfo info,
    StreamingContext context
)
```

```
Public Overrides Sub GetObjectData ( _
    info As SerializationInfo, _
    context As StreamingContext _
)
```

```
public:
virtual void GetObjectData(
    SerializationInfo^ info,
    StreamingContext context
) override
```

[-] **Parameters**

**info (SerializationInfo)**

The SerializationInfo that holds the serialized object data about the exception being thrown.

**context (StreamingContext)**

The StreamingContext that contains contextual information about the source or destination.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Message Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► MergeException ►

### Message

C# ▾

Gets a message that describes the merge conflicts.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public override string Message { get; }
```

```
Public Overrides ReadOnly Property Message As String
```

```
public:  
virtual property String^ Message {  
    String^ get () override;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# MessageResult Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► MessageResult

C# ▾

Specifies a return status value for message handlers. These values are returned by [Message\(InstallMessage, Record\)](#), [ExternalUIHandler](#), and [ProcessMessage\(InstallMessage, Record, MessageBoxButtons, MessageBoxIcon, MessageBoxDefaultButton\)](#).

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public enum MessageResult
```

```
Public Enumeration MessageResult
```

```
public enum class MessageResult
```

## Members

Member	Description
Error	An error was found in the message handler.
None	No action was taken.
OK	IDOK
Cancel	IDCANCEL
Abort	IDABORT

<b>Retry</b>	IDRETRY
<b>Ignore</b>	IDIGNORE
<b>Yes</b>	IDYES
<b>No</b>	IDNO

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **PatchInstallation Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► PatchInstallation

C#

The Patch object represents a unique instance of a patch that has been registered or applied.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public class PatchInstallation : Installation
```

```
Public Class PatchInstallation _  
    Inherits Installation
```

```
public ref class PatchInstallation : public Installation
```

### Members

All Members	Constructors	Methods	Properties	
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static		<input checked="" type="checkbox"/> Declare <input checked="" type="checkbox"/> Inherit

Icon	Member	Description
	<a href="#">PatchInstallation(String, String)</a>	Creates a new object for accessing information about a patch installation on the current system.
	<a href="#">PatchInstallation(String, String, String, UserContexts)</a>	Creates a new object for accessing information about a patch installation on the current system.
	<a href="#">AllPatches</a>	Enumerates all patch installations on

		the system.
	<b>Context</b>	Gets the user context of this product or patch installation.  (Inherited from <a href="#">Installation</a> .)
	<b>DisplayName</b>	Get the registered display name for the patch.
	<b>Equals(Object)</b>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<b>Finalize()</b>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
	<b>GetHashCode()</b>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.  (Inherited from <a href="#">Object</a> .)
	<b>GetPatches(String, String, String, UserContexts, PatchStates)</b>	Enumerates patch installations based on certain criteria.
	<b>GetType()</b>	Gets the <a href="#">Type</a> of the current instance  (Inherited from <a href="#">Object</a> .)
	<b>InstallDate</b>	Gets the date and time when the patch is applied to the product.

	<a href="#">IsInstalled</a>	Gets a value indicating whether this patch is currently installed.  (Overrides <a href="#">Installation.IsInstalled.</a> .)
	<a href="#">IsObsoleted</a>	Gets a value indicating whether this patch is marked as obsolete.
	<a href="#">IsSuperseded</a>	Gets a value indicating whether this patch is present but has been superseded by a more recent installed patch.
	<a href="#">Item[String]</a>	Gets information about a specific patch installation.  (Overrides <a href="#">Installation.Item[String].</a> )
	<a href="#">LocalPackage</a>	Gets the cached patch file that the product uses.
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">MoreInfoUrl</a>	Gets the registered support information URL for the patch.
	<a href="#">PatchCode</a>	Gets the patch code (GUID) of the patch.

	<a href="#">ProductCode</a>	Gets the ProductCode (GUID) of the product.
	<a href="#">SourceList</a>	Gets the source list of this product or patch installation.  (Inherited from <a href="#">Installation</a> .)
	<a href="#">State</a>	Gets the installation state of this instance of the patch.
	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Transforms</a>	Gets the set of patch transforms that the last patch installation applied to the product.
	<a href="#">Uninstallable</a>	True if patch is marked as possible to uninstall from the product.
	<a href="#">UserId</a>	Gets the user security identifier (SID) under which this product or patch installation is available.  (Inherited from <a href="#">Installation</a> .)

## Inheritance Hierarchy

[Object](#)  
   └ [Installation](#)  
     └ **PatchInstallation**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)  
Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **PatchInstallation Constructor**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► PatchInstallation ►

**PatchInstallation()**

C# ▾

▀ **Members**

Icon	Member	Description
≡	<a href="#">PatchInstallation(String, String)</a>	Creates a new object for accessing information about a patch installation on the current system.
≡	<a href="#">PatchInstallation(String, String, String, UserContexts)</a>	Creates a new object for accessing information about a patch installation on the current system.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## PatchInstallation Constructor (patchCode, productCode)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► PatchInstallation ►  
PatchInstallation(String, String)

C# ▾

Creates a new object for accessing information about a patch installation on the current system.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public PatchInstallation(  
    string patchCode,  
    string productCode  
)
```

```
Public Sub New ( _  
    patchCode As String, _  
    productCode As String _  
)
```

```
public:  
PatchInstallation(  
    String^ patchCode,  
    String^ productCode  
)
```

### - Parameters

#### **patchCode (String)**

Patch code (GUID) of the patch.

#### **productCode (String)**

ProductCode (GUID) the patch has been applied to. This parameter may be null for patches that are registered only and not yet applied to any product.

## Remarks

All available user contexts will be queried.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## PatchInstallation Constructor (patchCode, productCode, userSid, context)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► PatchInstallation ►

**PatchInstallation(String, String, String, UserContexts)**

C# ▾

Creates a new object for accessing information about a patch installation on the current system.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public PatchInstallation(  
    string patchCode,  
    string productCode,  
    string userSid,  
    UserContexts context  
)
```

```
Public Sub New ( _  
    patchCode As String, _  
    productCode As String, _  
    userSid As String, _  
    context As UserContexts _  
)
```

```
public:  
PatchInstallation(  
    String^ patchCode,  
    String^ productCode,  
    String^ userSid,  
    UserContexts context  
)
```

### - Parameters

**patchCode (String)**

Registered patch code (GUID) of the patch.

***productCode* ([String](#))**

ProductCode (GUID) the patch has been applied to. This parameter may be null for patches that are registered only and not yet applied to any product.

***userSid* ([String](#))**

The specific user, when working in a user context. This parameter may be null to indicate the current user. The parameter must be null when working in a machine context.

***context* ([UserContexts](#))**

The user context. The calling process must have administrative privileges to get information for a product installed for a user other than the current user.

## **Remarks**

If the *productCode* is null, the Patch object may only be used to read and update the patch's SourceList information.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## AllPatches Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► PatchInstallation ►

### AllPatches

C# ▾

Enumerates all patch installations on the system.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static IEnumerable<PatchInstallation> AllPatches {
```

```
Public Shared ReadOnly Property AllPatches As IEnumerable<
```

```
public:  
    static property IEnumerable<PatchInstallation^>^ AllPatches {  
        IEnumerable<PatchInstallation^>^ get();  
    }
```

#### Return Value

Enumeration of patch objects.

#### Remarks

Win32 MSI API: [MsiEnumPatches](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## DisplayName Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► PatchInstallation ►

### DisplayName

C# ▾

Get the registered display name for the patch.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string DisplayName { get; }
```

```
Public ReadOnly Property DisplayName As String
```

```
public:  
property String^ DisplayName {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## GetPatches Method (patchCode, targetProductCode, userSid, context, states)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► PatchInstallation ►

**GetPatches(String, String, String, UserContexts, PatchStates)**

C#

Enumerates patch installations based on certain criteria.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static IEnumerable<PatchInstallation> GetPatches(
    string patchCode,
    string targetProductCode,
    string userSid,
    UserContexts context,
    PatchStates states
)
```

```
Public Shared Function GetPatches ( _
    patchCode As String, _
    targetProductCode As String, _
    userSid As String, _
    context As UserContexts, _
    states As PatchStates _
) As IEnumerable(Of PatchInstallation)
```

```
public:
static IEnumerable<PatchInstallation^>^ GetPatches(
    String^ patchCode,
    String^ targetProductCode,
    String^ userSid,
    UserContexts context,
    PatchStates states
)
```

## Parameters

### **patchCode (String)**

PatchCode (GUID) of the patch to be enumerated. Only instances of patches within the scope of the context specified by the *userSid* and *context* parameters will be enumerated. This parameter may be set to null to enumerate all patches in the specified context.

### **targetProductCode (String)**

ProductCode (GUID) product whose patches are to be enumerated. If non-null, patch enumeration is restricted to instances of this product within the specified context. If null, the patches for all products under the specified context are enumerated.

### **userSid (String)**

Specifies a security identifier (SID) that restricts the context of enumeration. A SID value other than s-1-1-0 is considered a user SID and restricts enumeration to the current user or any user in the system. The special SID string s-1-1-0 (Everyone) specifies enumeration across all users in the system. This parameter can be set to null to restrict the enumeration scope to the current user. When *installContext* is set to the machine context only, *userSid* must be null.

### **context (UserContexts)**

Specifies the user context.

### **states (PatchStates)**

The [PatchStates](#) of patches to return.

## Remarks

Win32 MSI APIs: [MsiEnumPatchesEx](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



## Deployment Tools Foundation **InstallDate** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► PatchInstallation ►

### InstallDate

C# ▾

Gets the date and time when the patch is applied to the product.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public DateTime InstallDate { get; }
```

```
Public ReadOnly Property InstallDate As DateTime
```

```
public:  
property DateTime InstallDate {  
    DateTime get();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation IsInstalled Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► PatchInstallation ►

## IsInstalled

C# ▾

Gets a value indicating whether this patch is currently installed.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override bool IsInstalled { get; }
```

```
Public Overrides ReadOnly Property IsInstalled As Bo
```

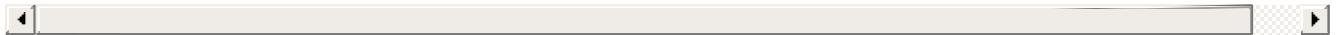
```
public:  
virtual property bool IsInstalled {  
    bool get () override;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation IsObsolete Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► PatchInstallation ►

## IsObsolete

Gets a value indicating whether this patch is marked as obsolete.

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public bool IsObsolete { get; }
```

```
Public Readonly Property IsObsolete As Boolean
```

```
public:  
property bool IsObsolete {  
    bool get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation IsSuperseded Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► PatchInstallation ►

### IsSuperseded

C# ▾

Gets a value indicating whether this patch is present but has been superseded by a more recent installed patch.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool IsSuperseded { get; }
```

```
Public ReadOnly Property IsSuperseded As Boolean
```

```
public:  
property bool IsSuperseded {  
    bool get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Item Property (propertyName)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► PatchInstallation ►

## Item[String]

C# ▾

Gets information about a specific patch installation.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override string this[  
    string propertyName  
] { get; }
```

```
Public Overrides ReadOnly Default Property Item ( _  
    propertyName As String _  
) As String
```

```
public:  
virtual property String^ default[String^ propertyName]  
    String^ get (String^ propertyName) override;  
}
```

### Parameters

#### **propertyName (String)**

The property being retrieved; see remarks for valid properties.

### Return Value

The property value, or an empty string if the property is not set for the patch.

### Remarks

Win32 MSI APIs: [MsiGetPatchInfo](#), [MsiGetPatchInfoEx](#)

### Exceptions

Exception	Condition
ArgumentOutOfRangeException	An unknown patch or property was requested
InstallerException	The installer configuration data is corrupt

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## LocalPackage Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► PatchInstallation ►

### LocalPackage

C# ▾

Gets the cached patch file that the product uses.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string LocalPackage { get; }
```

```
Public ReadOnly Property LocalPackage As String
```

```
public:  
property String^ LocalPackage {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## MoreInfoUrl Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► PatchInstallation ►

### MoreInfoUrl

C# ▾

Gets the registered support information URL for the patch.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public Uri MoreInfoUrl { get; }
```

```
Public ReadOnly Property MoreInfoUrl As Uri
```

```
public:  
property Uri^ MoreInfoUrl {  
    Uri^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **PatchCode Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► PatchInstallation ►

## **PatchCode**

C# ▾

Gets the patch code (GUID) of the patch.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string PatchCode { get; }
```

```
Public ReadOnly Property PatchCode As String
```

```
public:  
property String^ PatchCode {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# ProductCode Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► PatchInstallation ►

## ProductCode

C# ▾

Gets the ProductCode (GUID) of the product.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string ProductCode { get; }
```

```
Public ReadOnly Property ProductCode As String
```

```
public:  
property String^ ProductCode {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation State Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► PatchInstallation ► State

C# ▾

Gets the installation state of this instance of the patch.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public PatchStates State { get; }
```

```
Public ReadOnly Property State As PatchStates
```

```
public:  
property PatchStates State {  
    PatchStates get ();  
}
```

## Exceptions

Exception	Condition
ArgumentException	An unknown patch was requested
InstallerException	The installer configuration data is corrupt

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Transforms Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► PatchInstallation ►

## Transforms

Gets the set of patch transforms that the last patch installation applied to the product.

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string Transforms { get; }
```

```
Public ReadOnly Property Transforms As String
```

```
public:  
property String^ Transforms {  
    String^ get ();  
}
```

### Remarks

This value may not be available for per-user, non-managed applications if the user is not logged on.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Uninstallable Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► PatchInstallation ►

### Uninstallable

True patch is marked as possible to uninstall from the product.

C#

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public bool Uninstallable { get; }
```

```
Public Readonly Property Uninstallable As Boolean
```

```
public:  
property bool Uninstallable {  
    bool get ();  
}
```

#### Remarks

Even if this property is true, the installer can still block the uninstallation this patch is required by another patch that cannot be uninstalled.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **PatchStates Enumeration**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► PatchStates

C# ▾

Specifies the different patch states for `GetPatches(String, String, String UserContexts, PatchStates)`.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
[FlagsAttribute]  
public enum PatchStates
```

```
<FlagsAttribute> _  
Public Enumeration PatchStates
```

```
[FlagsAttribute]  
public enum class PatchStates
```

## Members

Member	Description
<b>None</b>	Invalid value.
<b>Applied</b>	Patches applied to a product.
<b>Superseded</b>	Patches that are superseded by other patches.
<b>Obsoleted</b>	Patches that are obsolesced by other patches.

<b>Registered</b>	Patches that are registered to a product but not applied.
<b>All</b>	All valid patch states.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **ProductInstallation Class**

Deployment Tools Foundation Namespaces ►

[Microsoft.Deployment.WindowsInstaller](#) ► **ProductInstallation**

C#

Represents a unique instance of a product that is either advertised, installed or unknown.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public class ProductInstallation : Installation
```

```
Public Class ProductInstallation _  
    Inherits Installation
```

```
public ref class ProductInstallation : public Installation
```

## Members

All Members	Constructors	Methods	Properties	
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static		<input checked="" type="checkbox"/> Declared <input checked="" type="checkbox"/> Inherited
Icon	Member		Description	
	<a href="#">ProductInstallation(String)</a>		Creates a new object for accessing information about a product installation on the current system.	
	<a href="#">ProductInstallation(String, String, UserContexts)</a>		Creates a new object for accessing information about a product installation on the current system.	
	<a href="#">AdvertisedLanguage</a>		Product language.	

	<a href="#">AdvertisedPackageCode</a>	Identifier of the package that a product is installed from.
	<a href="#">AdvertisedPackageName</a>	Name of the installation package for the advertised product.
	<a href="#">AdvertisedPerMachine</a>	True if the product is advertised per-machine; false if it is per-user or not advertised.
	<a href="#">AdvertisedProductIcon</a>	Primary icon for the package.
	<a href="#">AdvertisedProductName</a>	Human readable product name.
	<a href="#">AdvertisedTransforms</a>	Transforms.
	<a href="#">AdvertisedVersion</a>	Version of the advertised product.
	<a href="#">AllProducts</a>	Enumerates all product installations on the system.
	<a href="#">CollectUserInfo()</a>	Obtains and stores the user information and product ID from an installation wizard.

	<b>Context</b>	Gets the user context of this product or patch installation.  (Inherited from <a href="#">Installation</a> .)
	<b>Equals(Object)</b>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<b>Features</b>	Gets the set of published features for the product.
	<b>Finalize()</b>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
	<b>GetComponentState(String)</b>	Gets the installed state for a product component.
	<b>GetFeatureState(String)</b>	Gets the installed state for a product feature.
	<b>GetHashCode()</b>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.  (Inherited from <a href="#">Object</a> .)

	<a href="#">GetProducts(String, String, UserContexts)</a>	Enumerates product installations based on certain criteria.
	<a href="#">GetRelatedProducts(String)</a>	Gets the set of all products with a specified upgrade code. This method lists the currently installed and advertised products that have the specified UpgradeCode property in their Property table.
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
	<a href="#">HelpLink</a>	The support link.
	<a href="#">HelpTelephone</a>	The support telephone.
	<a href="#">InstallDate</a>	Date and time the product was installed.
	<a href="#">InstallLocation</a>	The installation location.
	<a href="#">InstallSource</a>	The installation source.
	<a href="#">IsAdvertised</a>	Gets a value indicating whether this product is advertised on the current system.

	<a href="#">IsElevated</a>	Checks whether the product is installed with elevated privileges. An application is called a "managed application" if elevated (system) privileges are used to install the application.
	<a href="#">IsInstalled</a>	Gets a value indicating whether this product is installed on the current system.  (Overrides <a href="#">Installation.IsInstalled</a> .)
	<a href="#">Item[String]</a>	Gets information about an installation of a product.  (Overrides <a href="#">Installation.Item[String]</a> )
	<a href="#">LocalPackage</a>	The local cached package.
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">PrivilegedPatchingAuthorized</a>	True if the advertised product can be serviced by non-administrators without elevation.
	<a href="#">ProductCode</a>	Gets the ProductCode (GUID) of the product.
	<a href="#">ProductId</a>	The product identifier.

	<a href="#">ProductName</a>	The installed product name.
	<a href="#">ProductVersion</a>	The product version.
	<a href="#">Publisher</a>	The publisher.
	<a href="#">RegCompany</a>	The company that is registered to use the product.
	<a href="#">RegOwner</a>	The owner who is registered to use the product.
	<a href="#">SourceList</a>	Gets the source list of this product installation.  (Overrides <a href="#">Installation.SourceList</a> .)
	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">UrlInfoAbout</a>	URL about information.
	<a href="#">UrlUpdateInfo</a>	The URL update information.
	<a href="#">UserId</a>	Gets the user security identifier (SID) under which this product or patch installation is available.

(Inherited from [Installation](#).)

## **[-] Inheritance Hierarchy**

[Object](#)

└ [Installation](#)

  └ **ProductInstallation**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

# ProductInstallation Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

**ProductInstallation()**

C# ▾

## Members

Icon	Member	Description
≡	<a href="#">ProductInstallation(String)</a>	Creates a new object for accessing information about a product installation on the current system.
≡	<a href="#">ProductInstallation(String, String, UserContexts)</a>	Creates a new object for accessing information about a product installation on the current system.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ProductInstallation Constructor (*productCode*)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►  
**ProductInstallation(String)**

C# ▾

Creates a new object for accessing information about a product installation on the current system.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public ProductInstallation(  
    string productCode  
)
```

```
Public Sub New ( _  
    productCode As String _  
)
```

```
public:  
ProductInstallation(  
    String^ productCode  
)
```

### [-] Parameters

#### ***productCode* (String)**

ProductCode (GUID) of the product.

### [-] Remarks

All available user contexts will be queried.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)  
Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ProductInstallation Constructor (**productCode**, **userSid**, **context**)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►  
**ProductInstallation(String, String, UserContexts)**

Creates a new object for accessing information about a product installation on the current system.

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public ProductInstallation(  
    string productCode,  
    string userSid,  
    UserContexts context  
)
```

```
Public Sub New ( _  
    productCode As String, _  
    userSid As String, _  
    context As UserContexts _  
)
```

```
public:  
ProductInstallation(  
    String^ productCode,  
    String^ userSid,  
    UserContexts context  
)
```

### Parameters

#### **productCode** (**String**)

ProductCode (GUID) of the product.

#### **userSid** (**String**)

The specific user, when working in a user context. This parameter may be null to indicate the current user. The parameter must be null when working in a machine context.

**context ([UserContexts](#))**

The user context. The calling process must have administrative privileges to get information for a product installed for a user other than the current user.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## AdvertisedLanguage Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

### AdvertisedLanguage

C# ▾

Product language.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string AdvertisedLanguage { get; }
```

```
Public ReadOnly Property AdvertisedLanguage As String
```

```
public:  
property String^ AdvertisedLanguage {  
    String^ get();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## AdvertisedPackageCode Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

### AdvertisedPackageCode

C# ▾

Identifier of the package that a product is installed from.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string AdvertisedPackageCode { get; }
```

```
Public ReadOnly Property AdvertisedPackageCode As St
```

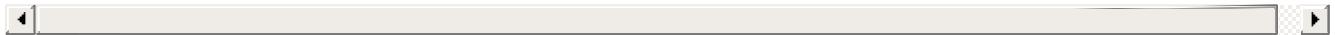
```
public:  
property String^ AdvertisedPackageCode {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# AdvertisedPackageName

## Property

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►  
**AdvertisedPackageName**

Name of the installation package for the advertised product.

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string AdvertisedPackageName { get; }
```

```
Public ReadOnly Property AdvertisedPackageName As St
```

```
public:  
property String^ AdvertisedPackageName {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## AdvertisedPerMachine Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

### AdvertisedPerMachine

C# ▾

True if the product is advertised per-machine; false if it is per-user or not advertised.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool AdvertisedPerMachine { get; }
```

```
Public ReadOnly Property AdvertisedPerMachine As Boo
```

```
public:  
property bool AdvertisedPerMachine {  
    bool get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## AdvertisedProductIcon Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

### AdvertisedProductIcon

C# ▾

Primary icon for the package.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string AdvertisedProductIcon { get; }
```

```
Public ReadOnly Property AdvertisedProductIcon As St
```

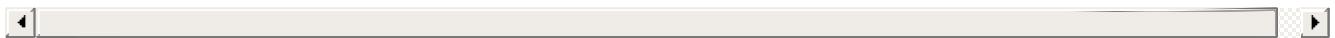
```
public:  
property String^ AdvertisedProductIcon {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## AdvertisedProductName Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

### AdvertisedProductName

C# ▾

Human readable product name.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string AdvertisedProductName { get; }
```

```
Public ReadOnly Property AdvertisedProductName As St
```

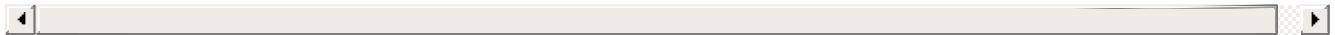
```
public:  
property String^ AdvertisedProductName {  
    String^ get();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## AdvertisedTransforms Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

### AdvertisedTransforms

C# ▾

Transforms.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string AdvertisedTransforms { get; }
```

```
Public ReadOnly Property AdvertisedTransforms As Str
```

```
public:  
property String^ AdvertisedTransforms {  
    String^ get();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## AdvertisedVersion Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

### AdvertisedVersion

C# ▾

Version of the advertised product.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public Version AdvertisedVersion { get; }
```

```
Public Readonly Property AdvertisedVersion As Version
```

```
public:  
property Version^ AdvertisedVersion {  
    Version^ get();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **AllProducts** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

## AllProducts

C# ▾

Enumerates all product installations on the system.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static IEnumerable<ProductInstallation> AllPr
```

```
Public Shared ReadOnly Property AllProducts As IEnumerable<
```

```
public:  
static property IEnumerable<ProductInstallation^>^ AllProducts  
    IEnumerable<ProductInstallation^>^ get ();  
}
```

### Return Value

An enumeration of product objects.

### Remarks

Win32 MSI API: [MsiEnumProducts](#),

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



## Deployment Tools Foundation **CollectUserInfo Method**

[Deployment Tools Foundation Namespaces](#) ►

[Microsoft.Deployment.WindowsInstaller](#) ► [ProductInstallation](#) ►

### **CollectUserInfo()**

C# ▾

Obtains and stores the user information and product ID from an installation wizard.

#### **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public void CollectUserInfo()
```

```
Public Sub CollectUserInfo
```

```
public:  
void CollectUserInfo()
```

#### **Remarks**

This method is typically called by an application during the first run of the application. The application first gets the [ProductId](#) or [RegOwner](#). If those properties are missing, the application calls [CollectUserInfo](#).

[CollectUserInfo](#) opens the product's installation package and invokes a wizard sequence that collects user information. Upon completion of the sequence, user information is registered. Since this API requires an authored user interface, the user interface level should be set to full by calling [SetInternalUI\(InstallUIOptions\)](#) as [Full](#).

The [CollectUserInfo](#) method invokes a [FirstRun](#) dialog from the product installation database.

Win32 MSI API: [MsiCollectUserInfo](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Features Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

## Features

C# ▾

Gets the set of published features for the product.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public IEnumerable<FeatureInstallation> Features { g
```

```
Public ReadOnly Property Features As IEnumerable(Of
```

```
public:  
property IEnumerable<FeatureInstallation^>^ Features  
    IEnumerable<FeatureInstallation^>^ get ();  
}
```

### Return Value

Enumeration of published features for the product.

### Remarks

Because features are not ordered, any new feature has an arbitrary index, meaning this property can return features in any order.

Win32 MSI API: [MsiEnumFeatures](#)

### Exceptions

Exception	Condition
InstallerException	The installer configuration data is corrupt

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# GetComponentState Method (component)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►  
**GetComponentState(String)**

C# ▾

Gets the installed state for a product component.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public InstallState GetComponentState(  
    string component  
)
```

```
Public Function GetComponentState ( _  
    component As String _  
) As InstallState
```

```
public:  
InstallState GetComponentState(  
    String^ component  
)
```

## Parameters

### **component (String)**

The component being queried; GUID of the component as found in the ComponentId column of the Component table.

## Return Value

Installation state of the component for the product instance: either [Local](#) or [Source](#).

## Remarks

Win32 MSI API: [MsiQueryComponentState](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)  
Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetFeatureState Method (feature)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

### **GetFeatureState(String)**

C# ▾

Gets the installed state for a product feature.

#### [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public InstallState GetFeatureState(  
    string feature  
)
```

```
Public Function GetFeatureState ( _  
    feature As String _  
) As InstallState
```

```
public:  
InstallState GetFeatureState(  
    String^ feature  
)
```

#### [-] Parameters

##### ***feature* (String)**

The feature being queried; identifier from the Feature table

#### [-] Return Value

Installation state of the feature for the product instance: either [Local](#), [Source](#), or [Advertised](#).

#### [-] Remarks

Win32 MSI APIs: [MsiQueryFeatureState](#), [MsiQueryFeatureStateEx](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## GetProducts Method (**productCode**, **userSid**, **context**)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

**GetProducts(String, String, UserContexts)**

C#

Enumerates product installations based on certain criteria.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static IEnumerable<ProductInstallation> GetPr
    string productCode,
    string userSid,
    UserContexts context
)
```

```
Public Shared Function GetProducts ( _
    productCode As String, _
    userSid As String, _
    context As UserContexts _
) As IEnumerable(Of ProductInstallation)
```

```
public:
static IEnumerable<ProductInstallation^>^ GetProduct
    String^ productCode,
    String^ userSid,
    UserContexts context
)
```

### Parameters

#### **productCode (String)**

ProductCode (GUID) of the product instances to be enumerated.

Only instances of products within the scope of the context specified by the *userSid* and *installContext* parameters will be enumerated.

This parameter may be set to null to enumerate all products in the

specified context.

#### ***userId* ([String](#))**

Specifies a security identifier (SID) that restricts the context of enumeration. A SID value other than s-1-1-0 is considered a user SID and restricts enumeration to the current user or any user in the system. The special SID string s-1-1-0 (Everyone) specifies enumeration across all users in the system. This parameter can be set to null to restrict the enumeration scope to the current user. When *context* is set to the machine context only, *userId* must be null.

#### ***context* ([UserContexts](#))**

Specifies the user context.

### **[-] Return Value**

An enumeration of product objects for enumerated product instances.

### **[-] Remarks**

Win32 MSI API: [MsiEnumProductsEx](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **GetRelatedProducts Method  
(upgradeCode)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►  
**GetRelatedProducts(String)**

C# ▾

Gets the set of all products with a specified upgrade code. This method lists the currently installed and advertised products that have the specified UpgradeCode property in their Property table.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public static IEnumerable<ProductInstallation> GetRe
    string upgradeCode
)
```

```
Public Shared Function GetRelatedProducts ( _
    upgradeCode As String _
) As IEnumerable(Of ProductInstallation)
```

```
public:
static IEnumerable<ProductInstallation^>^ GetRelated
    String^ upgradeCode
)
```

[-] **Parameters**

**upgradeCode (String)**

Upgrade code of related products

[-] **Return Value**

Enumeration of product codes

[-] **Remarks**

Win32 MSI API: [MsiEnumRelatedProducts](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)  
Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## HelpLink Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

### HelpLink

C# ▾

The support link.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string HelpLink { get; }
```

```
Public ReadOnly Property HelpLink As String
```

```
public:  
property String^ HelpLink {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

C# ▾

The support telephone.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public string HelpTelephone { get; }
```

```
Public ReadOnly Property HelpTelephone As String
```

```
public:  
property String^ HelpTelephone {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## InstallDate Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

### InstallDate

C# ▾

Date and time the product was installed.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public DateTime InstallDate { get; }
```

```
Public ReadOnly Property InstallDate As DateTime
```

```
public:  
property DateTime InstallDate {  
    DateTime get();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## InstallLocation Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

### InstallLocation

C# ▾

The installation location.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string InstallLocation { get; }
```

```
Public ReadOnly Property InstallLocation As String
```

```
public:  
property String^ InstallLocation {  
    String^ get();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# InstallSource Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

## InstallSource

C# ▾

The installation source.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string InstallSource { get; }
```

```
Public ReadOnly Property InstallSource As String
```

```
public:  
property String^ InstallSource {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## IsAdvertised Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

### IsAdvertised

C# ▾

Gets a value indicating whether this product is advertised on the current system.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool IsAdvertised { get; }
```

```
Public ReadOnly Property IsAdvertised As Boolean
```

```
public:  
property bool IsAdvertised {  
    bool get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation IsElevated Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

## IsElevated

C# ▾

Checks whether the product is installed with elevated privileges. An application is called a "managed application" if elevated (system) privileges are used to install the application.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool IsElevated { get; }
```

```
Public ReadOnly Property IsElevated As Boolean
```

```
public:  
property bool IsElevated {  
    bool get ();  
}
```

### Return Value

True if the product is elevated; false otherwise

### Remarks

Note that this property does not take into account policies such as AlwaysInstallElevated, but verifies that the local system owns the product's registry data.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **IsInstalled** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

### IsInstalled

C# ▾

Gets a value indicating whether this product is installed on the current system.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override bool IsInstalled { get; }
```

```
Public Overrides ReadOnly Property IsInstalled As Bo
```

```
public:  
virtual property bool IsInstalled {  
    bool get () override;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation Item Property (propertyName)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

## Item[String]

C# ▾

Gets information about an installation of a product.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override string this[  
    string propertyName  
] { get; }
```

```
Public Overrides ReadOnly Default Property Item ( _  
    propertyName As String _  
) As String
```

```
public:  
virtual property String^ default[String^ propertyName]  
String^ get (String^ propertyName) override;  
}
```

### Parameters

#### propertyName (String)

Name of the property being retrieved.

### Remarks

Win32 MSI APIs: [MsiGetProductInfo](#), [MsiGetProductInfoEx](#)

### Exceptions

Exception	Condition
ArgumentOutOfRangeException	An unknown product or property was requested

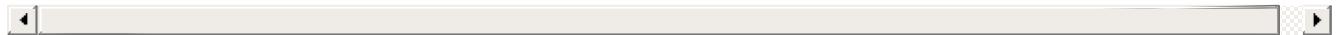
[InstallerException](#)

The installer configuration data is corrupt

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## LocalPackage Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

### LocalPackage

C# ▾

The local cached package.

#### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public string LocalPackage { get; }
```

```
Public ReadOnly Property LocalPackage As String
```

```
public:  
property String^ LocalPackage {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# PrivilegedPatchingAuthorized

## Property

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►  
**PrivilegedPatchingAuthorized**

C# ▾

True if the advertised product can be serviced by non-administrators without elevation.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool PrivilegedPatchingAuthorized { get; }
```

```
Public ReadOnly Property PrivilegedPatchingAuthorized
```

```
public:  
property bool PrivilegedPatchingAuthorized {  
    bool get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

# ProductCode Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

## ProductCode

C# ▾

Gets the ProductCode (GUID) of the product.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string ProductCode { get; }
```

```
Public ReadOnly Property ProductCode As String
```

```
public:  
property String^ ProductCode {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **ProductId Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

## ProductId

C# ▾

The product identifier.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string ProductId { get; }
```

```
Public ReadOnly Property ProductId As String
```

```
public:  
property String^ ProductId {  
    String^ get ();  
}
```

### Remarks

For more information, see [ProductID](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# ProductName Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

## ProductName

C# ▾

The installed product name.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string ProductName { get; }
```

```
Public ReadOnly Property ProductName As String
```

```
public:  
property String^ ProductName {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# ProductVersion Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

## ProductVersion

C# ▾

The product version.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public Version ProductVersion { get; }
```

```
Public ReadOnly Property ProductVersion As Version
```

```
public:  
property Version^ ProductVersion {  
    Version^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Publisher Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

## Publisher

C# ▾

The publisher.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string Publisher { get; }
```

```
Public ReadOnly Property Publisher As String
```

```
public:  
property String^ Publisher {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# RegCompany Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

## RegCompany

C# ▾

The company that is registered to use the product.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string RegCompany { get; }
```

```
Public ReadOnly Property RegCompany As String
```

```
public:  
property String^ RegCompany {  
    String^ get();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# RegOwner Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

## RegOwner

C# ▾

The owner who is registered to use the product.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string RegOwner { get; }
```

```
Public Readonly Property RegOwner As String
```

```
public:  
property String^ RegOwner {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# SourceList Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

## SourceList

C# ▾

Gets the source list of this product installation.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override SourceList SourceList { get; }
```

```
Public Overrides ReadOnly Property SourceList As Sou
```

```
public:  
virtual property SourceList^ SourceList {  
    SourceList^ get () override;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **UrlInfoAbout Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

**UrlInfoAbout**

C# ▾

URL about information.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public Uri UrlInfoAbout { get; }
```

```
Public ReadOnly Property UrlInfoAbout As Uri
```

```
public:  
property Uri^ UrlInfoAbout {  
    Uri^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# UrlUpdateInfo Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ProductInstallation ►

## UrlUpdateInfo

C# ▾

The URL update information.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public Uri UrlUpdateInfo { get; }
```

```
Public ReadOnly Property UrlUpdateInfo As Uri
```

```
public:  
property Uri^ UrlUpdateInfo {  
    Uri^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Record Class

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record

C# ▾

The Record object is a container for holding and transferring a variable number of values. Fields within the record are numerically indexed and can contain strings, integers, streams, and null values. Record fields are indexed starting with 1. Field 0 is a special format field.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public class Record : InstallerHandle
```

```
Public Class Record  
    Inherits InstallerHandle
```

```
public ref class Record : public InstallerHandle
```

### Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon	Member	Description
≡	Record(Int32)	Creates a new record object with the requested number of fields.
≡	Record(Object[])	Creates a new record object, providing values for an arbitrary number of fields.

	<a href="#">Clear()</a>	Sets all fields in a record to null.
	<a href="#">Close()</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">CreateObjRef(Type)</a>	Creates an object that contains all the relevant information required to generate a proxy used to communicate with a remote object.  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">Dispose()</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">Dispose(Boolean)</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">Equals(Object)</a>	Tests whether this handle object is equal to another handle object. Two handle objects are equal if their types are the same and their native integer handles are the same.  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">FieldCount</a>	Gets the number of fields in a record.

	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
	<a href="#">FormatString</a>	Gets or sets field 0 of the Record, which is the format string.
 	<a href="#">FromHandle(IntPtr, Boolean)</a>	Creates a new Record object from an integer record handle.
	<a href="#">GetDataSize(Int32)</a>	Gets the length of a record field. The count does not include the terminating null.
	<a href="#">GetDataSize(String)</a>	Gets the length of a record field. The count does not include the terminating null.
	<a href="#">GetHashCode()</a>	Gets a hash value for the handle object.  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">GetInteger(Int32)</a>	Gets a field value as an integer.
	<a href="#">GetInteger(String)</a>	Gets a field value as an integer.
	<a href="#">GetLifetimeService()</a>	Retrieves the current lifetime

		service object that controls the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject</a> .)
≡	<a href="#">GetNullableInteger(Int32)</a>	Gets a field value as an integer.
≡	<a href="#">GetNullableInteger(String)</a>	Gets a field value as an integer.
≡	<a href="#">GetStream(Int32, String)</a>	Reads a record stream field into a file.
≡	<a href="#">GetStream(String, String)</a>	Reads a record stream field into a file.
≡	<a href="#">GetStream(Int32)</a>	Gets a record stream field.
≡	<a href="#">GetStream(String)</a>	Gets a record stream field.
≡	<a href="#">GetString(Int32)</a>	Gets a field value as a string.
≡	<a href="#">GetString(String)</a>	Gets a field value as a string.
≡	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)

	<a href="#">Handle</a>	Gets the native integer handle.  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">InitializeLifetimeService()</a>	Obtains a lifetime service object to control the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">IsClosed</a>	Checks if the handle is closed. When closed, method calls on the handle object may throw an <a href="#">InvalidHandleException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">IsFormatStringInvalid</a>	IsFormatStringInvalid is set from several View methods that invalidate the FormatString and used to determine behavior during Record.ToString().
	<a href="#">IsNull(Int32)</a>	Reports whether a record field is null.
	<a href="#">IsNull(String)</a>	Reports whether a record field is null.
	<a href="#">Item[String]</a>	Gets or sets a record field value.
	<a href="#">Item[Int32]</a>	Gets or sets a record field value.

💡	<a href="#">MemberwiseClone(Boolean)</a>	Creates a shallow copy of the current <a href="#">MarshalByRefObject</a> object  (Inherited from <a href="#">MarshalByRefObject</a> .)
💡	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">SetInteger(Int32, Int32)</a>	Sets the value of a field to an integer.
💡	<a href="#">SetInteger(String, Int32)</a>	Sets the value of a field to an integer.
💡	<a href="#">SetNullableInteger(Int32, Nullable&lt;Int32&gt;)</a>	Sets the value of a field to a nullable integer.
💡	<a href="#">SetNullableInteger(String, Nullable&lt;Int32&gt;)</a>	Sets the value of a field to a nullable integer.
💡	<a href="#">SetStream(Int32, String)</a>	Sets a record stream field from a file. Stream data cannot be inserted into temporary fields.
💡	<a href="#">SetStream(String, String)</a>	Sets a record stream field from a file. Stream data cannot be inserted into temporary fields.
💡		

	<code>SetStream(Int32, Stream)</code>	Sets a record stream field from a Stream object. Stream data cannot be inserted into temporary fields.
≡	<code>SetStream(String, Stream)</code>	Sets a record stream field from a Stream object. Stream data cannot be inserted into temporary fields.
≡	<code>SetString(Int32, String)</code>	Sets the value of a field to a string.
≡	<code>SetString(String, String)</code>	Sets the value of a field to a string.
≡	<code>ToString()</code>	Gets a formatted string representation of the Record.  (Overrides <a href="#">Object.ToString()</a> .)
≡	<code>ToString(IFormatProvider)</code>	Gets a formatted string representation of the Record, optionally using a Session to format properties.
≡	<code>ToString(String)</code>	<b>Obsolete.</b> Gets a formatted string representation of the Record.
≡	<code>ToString(String, IFormatProvider)</code>	<b>Obsolete.</b> Gets a formatted string representation of the Record, optionally using a Session to format properties.

## **Remarks**

Most methods on the Record class have overloads that allow using either a number or a name to designate a field. However note that field names only exist when the Record is directly returned from a query on a database. For other records, attempting to access a field by name will result in an InvalidOperationException.

## **Inheritance Hierarchy**

Object

└ MarshalByRefObject

  └ InstallerHandle

    └ Record

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# Record Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► Record()

C# ▾

## Members

Icon	Member	Description
	<a href="#">Record(Int32)</a>	Creates a new record object with the requested number of fields.
	<a href="#">Record(Object[])</a>	Creates a new record object, providing values for an arbitrary number of fields.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Record Constructor (fieldCount)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► Record(Int32)

C#

Creates a new record object with the requested number of fields.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public Record(  
    int fieldCount  
)
```

```
Public Sub New ( _  
    fieldCount As Integer _  
)
```

```
public:  
Record(  
    int fieldCount  
)
```

## Parameters

### *fieldCount* (Int32)

Required number of fields, which may be 0. The maximum number of fields in a record is limited to 65535.

## Remarks

The Record object should be [Close\(\)](#)d after use. It is best that the handle be closed manually as soon as it is no longer needed, as leaving lots of unused handles open can degrade performance.

Win32 MSI API: [MsiCreateRecord](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Record Constructor (fields)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► Record(Object[])

C#

Creates a new record object, providing values for an arbitrary number of fields.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public Record(  
    params Object[] fields  
)
```

```
Public Sub New ( _  
    ParamArray fields As Object() _  
)
```

```
public:  
Record(  
    ... array<Object^>^ fields  
)
```

## Parameters

### fields ( Object [] )

The values of the record fields. The parameters should be of type Int16, Int32 or String

## Remarks

The Record object should be [Close\(\)](#)d after use. It is best that the handle be closed manually as soon as it is no longer needed, as leaving lots of unused handles open can degrade performance.

Win32 MSI API: [MsiCreateRecord](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

C# ▾

Sets all fields in a record to null.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Clear()
```

```
Public Sub Clear
```

```
public:  
void Clear()
```

## Remarks

Win32 MSI API: [MsiRecordClearData](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## FieldCount Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► FieldCount

C# ▾

Gets the number of fields in a record.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public int FieldCount { get; }
```

```
Public Readonly Property FieldCount As Integer
```

```
public:  
property int FieldCount {  
    int get ();  
}
```

### - Remarks

Win32 MSI API: [MsiRecordGetFieldCount](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **FormatString Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► **FormatString**

C# ▾

Gets or sets field 0 of the Record, which is the format string.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string FormatString { get; set; }
```

```
Public Property FormatString As String
```

```
public:  
property String^ FormatString {  
    String^ get ();  
    void set (String^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## FromHandle Method (handle, ownsHandle)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ►  
**FromHandle(IntPtr, Boolean)**

C# ▾

Creates a new Record object from an integer record handle.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static Record FromHandle(  
    IntPtr handle,  
    bool ownsHandle  
)
```

```
Public Shared Function FromHandle ( _  
    handle As IntPtr, _  
    ownsHandle As Boolean _  
) As Record
```

```
public:  
static Record^ FromHandle(  
    IntPtr handle,  
    bool ownsHandle  
)
```

### Parameters

#### **handle (IntPtr)**

Integer record handle

#### **ownsHandle (Boolean)**

true to close the handle when this object is disposed or finalized

### Remarks

This method is only provided for interop purposes. A Record object should normally be obtained by calling [Fetch\(\)](#) other methods.

The handle will be closed when this object is disposed or finalized.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetContentSize Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► **GetContentSize()**

C# ▾

**Members**

Icon	Member	Description
	<a href="#">GetContentSize(Int32)</a>	Gets the length of a record field. The count does not include the terminating null.
	<a href="#">GetContentSize(String)</a>	Gets the length of a record field. The count does not include the terminating null.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetContentSize Method (field)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ►  
**GetContentSize(Int32)**

C# ▾

Gets the length of a record field. The count does not include the terminating null.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public int GetContentSize(  
    int field  
)
```

```
Public Function GetContentSize ( _  
    field As Integer _  
) As Integer
```

```
public:  
int GetContentSize(  
    int field  
)
```

### Parameters

***field* (Int32)**

### Remarks

The returned data size is 0 if the field is null, non-existent, or an internal object pointer. The method also returns 0 if the handle is not a valid Record handle.

If the data is in integer format, the property returns 2 or 4.

If the data is in string format, the property returns the character count (n

including the NULL terminator).

If the data is in stream format, the property returns the byte count.

Win32 MSI API: [MsiRecordDataSize](#)

## Exceptions

Exception	Condition
<a href="#">ArgumentOutOfRangeException</a>	The field is less than 0 or greater than the number of fields in the Record.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetContentSize Method (fieldName)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ►  
**GetContentSize(String)**

C# ▾

Gets the length of a record field. The count does not include the terminating null.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public int GetContentSize(  
    string fieldName  
)
```

```
Public Function GetContentSize ( _  
    fieldName As String _  
) As Integer
```

```
public:  
int GetContentSize(  
    String^ fieldName  
)
```

### Parameters

#### **fieldName (String)**

Specifies the field to check.

### Remarks

The returned data size is 0 if the field is null, non-existent, or an internal object pointer. The method also returns 0 if the handle is not a valid Record handle.

If the data is in integer format, the property returns 2 or 4.

If the data is in string format, the property returns the character count (n including the NULL terminator).

If the data is in stream format, the property returns the byte count.

## - Exceptions

Exception	Condition
ArgumentOutOfRangeException	The field name does not match any of the named fields in the Record.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetInteger Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► GetInteger()

C# ▾

▀ Members

Icon	Member	Description
≡	GetInteger(Int32)	Gets a field value as an integer.
≡	GetInteger(String)	Gets a field value as an integer.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetInteger Method (field)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► **GetInteger(Int32)**

C# ▾

Gets a field value as an integer.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public int GetInteger(  
    int field  
)
```

```
Public Function GetInteger ( _  
    field As Integer _  
) As Integer
```

```
public:  
int GetInteger(  
    int field  
)
```

### Parameters

#### ***field* (Int32)**

Specifies the field to retrieve.

### Return Value

Integer value of the field, or 0 if the field is null.

### Remarks

Win32 MSI API: [MsiRecordGetInteger](#)

### Exceptions

Exception

Condition

**ArgumentOutOfRangeException**The field is less than 0 or greater than the number of fields in the Record.

## See Also

[GetNullableInteger\(Int32\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **GetInteger Method (fieldName)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ►  
**GetInteger(String)**

C# ▾

Gets a field value as an integer.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public int GetInteger(  
    string fieldName  
)
```

```
Public Function GetInteger ( _  
    fieldName As String _  
) As Integer
```

```
public:  
int GetInteger(  
    String^ fieldName  
)
```

## Parameters

### ***fieldName* (String)**

Specifies the field to retrieve.

## Return Value

Integer value of the field, or 0 if the field is null.

## Exceptions

Exception	Condition
ArgumentOutOfRangeException	The field name does not match any of the named fields in the Record.

## See Also

[GetNullableInteger\(String\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetNullableInteger Method**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ►  
**GetNullableInteger()**

C# ▾

▀ **Members**

Icon	Member	Description
≡	<a href="#">GetNullableInteger(Int32)</a>	Gets a field value as an integer.
≡	<a href="#">GetNullableInteger(String)</a>	Gets a field value as an integer.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetNullableInteger** Method (field)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ►  
**GetNullableInteger(Int32)**

C# ▾

Gets a field value as an integer.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public Nullable<int> GetNullableInteger(  
    int field  
)
```

```
Public Function GetNullableInteger ( _  
    field As Integer _  
) As Nullable(Of Integer)
```

```
public:  
Nullable<int> GetNullableInteger(  
    int field  
)
```

### Parameters

#### ***field* (Int32)**

Specifies the field to retrieve.

### Return Value

Integer value of the field, or null if the field is null.

### Remarks

Win32 MSI API: [MsiRecordGetInt](#)

### Exceptions

Exception	Condition
ArgumentOutOfRangeException	The field is less than 0 or greater than the number of fields in the Record.

## See Also

[GetInteger\(Int32\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetNullableInteger Method  
(*fieldName*)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ►  
**GetNullableInteger(String)**

C# ▾

Gets a field value as an integer.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public Nullable<int> GetNullableInteger(  
    string fieldName  
)
```

```
Public Function GetNullableInteger ( _  
    fieldName As String _  
) As Nullable(Of Integer)
```

```
public:  
Nullable<int> GetNullableInteger(  
    String^ fieldName  
)
```

[-] **Parameters**

***fieldName* (String)**

Specifies the field to retrieve.

[-] **Return Value**

Integer value of the field, or null if the field is null.

[-] **Exceptions**

Exception

Condition

ArgumentException	The field name does not match any of the
-------------------	--

named fields in the Record.

## See Also

[GetInteger\(String\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetStream Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► **GetStream()**

C# ▾

▀ **Members**

Icon	Member	Description
≡	<a href="#">GetStream(Int32, String)</a>	Reads a record stream field into a file.
≡	<a href="#">GetStream(String, String)</a>	Reads a record stream field into a file.
≡	<a href="#">GetStream(Int32)</a>	Gets a record stream field.
≡	<a href="#">GetStream(String)</a>	Gets a record stream field.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetStream Method (field)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ►  
**GetStream(Int32)**

C# ▾

Gets a record stream field.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public Stream GetStream(  
    int field  
)
```

```
Public Function GetStream ( _  
    field As Integer _  
) As Stream
```

```
public:  
Stream^ GetStream(  
    int field  
)
```

### Parameters

#### ***field* (Int32)**

Specifies the field of the Record to get.

### Return Value

A Stream that reads the field data.

### Remarks

This method is not capable of reading substorages. To extract a substorage, use [GetStream\(Int32, String\)](#).

Win32 MSI API: [MsiRecordReadStream](#)

## Exceptions

Exception	Condition
ArgumentOutOfRangeException	The field is less than 0 or greater than the number of fields in the Record.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **GetStream Method (field, filePath)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► **GetStream(Int32, String)**

C# ▾

Reads a record stream field into a file.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public void GetStream(  
    int field,  
    string filePath  
)
```

```
Public Sub GetStream ( _  
    field As Integer, _  
    filePath As String _  
)
```

```
public:  
void GetStream(  
    int field,  
    String^ filePath  
)
```

## Parameters

### ***field* (Int32)**

Specifies the field of the Record to get.

### ***filePath* (String)**

Specifies the path to the file to contain the stream.

## Remarks

This method is capable of directly extracting substorages. To do so, first select both the `Name` and `Data` column of the `\_Storages` table, then

get the stream of the `Data` field. However, substorages may only be extracted from a database that is open in read-only mode.

Win32 MSI API: [MsiRecordReadStream](#)

## **Exceptions**

Exception	Condition
ArgumentOutOfRangeException	The field is less than 0 or greater than the number of fields in the Record.
NotSupportedException	Attempt to extract a storage from a database open in read-write mode, or from a database without an associated file path.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **GetStream Method (fieldName)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ►  
**GetStream(String)**

C# ▾

Gets a record stream field.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public Stream GetStream(  
    string fieldName  
)
```

```
Public Function GetStream ( _  
    fieldName As String _  
) As Stream
```

```
public:  
Stream^ GetStream(  
    String^ fieldName  
)
```

## Parameters

### ***fieldName* (String)**

Specifies the field of the Record to get.

## Return Value

A Stream that reads the field data.

## Remarks

This method is not capable of reading substorages. To extract a substorage, use [GetStream\(String, String\)](#).

## Exceptions

Exception	Condition
ArgumentOutOfRangeException	The field name does not match any of the named fields in the Record.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetStream Method (fieldName, filePath)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ►  
**GetStream(String, String)**

C# ▾

Reads a record stream field into a file.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public void GetStream(  
    string fieldName,  
    string filePath  
)
```

```
Public Sub GetStream ( _  
    fieldName As String, _  
    filePath As String _  
)
```

```
public:  
void GetStream(  
    String^ fieldName,  
    String^ filePath  
)
```

[-] **Parameters**

**fieldName (String)**

Specifies the field of the Record to get.

**filePath (String)**

Specifies the path to the file to contain the stream.

[-] **Remarks**

This method is capable of directly extracting substorages. To do so, first select both the `Name` and `Data` column of the `\_Storages` table, then get the stream of the `Data` field. However, substorages may only be extracted from a database that is open in read-only mode.

## **Exceptions**

Exception	Condition
ArgumentOutOfRangeException	The field name does not match any of the named fields in the Record.
NotSupportedException	Attempt to extract a storage from a database open in read-write mode, or from a database without an associated file path.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetString Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► **GetString()**

C# ▾

**Members**

Icon	Member	Description
	<a href="#">GetString(Int32)</a>	Gets a field value as a string.
	<a href="#">GetString(String)</a>	Gets a field value as a string.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetString Method (field)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► **GetString(Int32)**

C# ▾

Gets a field value as a string.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string GetString(  
    int field  
)
```

```
Public Function GetString ( _  
    field As Integer _  
) As String
```

```
public:  
String^ GetString(  
    int field  
)
```

### [-] Parameters

#### ***field* (Int32)**

Specifies the field to retrieve.

### [-] Return Value

String value of the field, or an empty string if the field is null.

### [-] Remarks

Win32 MSI API: [MsiRecordGetString](#)

### [-] Exceptions

Exception	Condition
-----------	-----------

**ArgumentOutOfRangeException**The field is less than 0 or greater than the number of fields in the Record.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **GetString Method (fieldName)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ►  
**GetString(String)**

C# ▾

Gets a field value as a string.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public string GetString(  
    string fieldName  
)
```

```
Public Function GetString ( _  
    fieldName As String _  
) As String
```

```
public:  
String^ GetString(  
    String^ fieldName  
)
```

## Parameters

### ***fieldName* (String)**

Specifies the field to retrieve.

## Return Value

String value of the field, or an empty string if the field is null.

## Exceptions

Exception	Condition
ArgumentOutOfRangeException	The field name does not match any of the named fields in the Record.

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation IsFormatStringInvalid Property

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ►  
**IsFormatStringInvalid**

C# ▾

IsFormatStringInvalid is set from several View methods that invalidate the FormatString and used to determine behavior during Record.ToString().

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected bool IsFormatStringInvalid { get; set; }
```

Protected Property IsFormatStringInvalid As Boolean

```
protected:  
property bool IsFormatStringInvalid {  
    bool get ();  
    void set (bool value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **IsNull Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► **IsNull()**

C# ▾

**Members**

Icon	Member	Description
≡	<a href="#">IsNull(Int32)</a>	Reports whether a record field is null.
≡	<a href="#">IsNull(String)</a>	Reports whether a record field is null.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation IsNull Method (field)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► **IsNull(Int32)**

C# ▾

Reports whether a record field is null.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool IsNull(  
    int field  
)
```

```
Public Function IsNull ( _  
    field As Integer _  
) As Boolean
```

```
public:  
bool IsNull(  
    int field  
)
```

### Parameters

#### *field* (Int32)

Specifies the field to check.

### Return Value

True if the field is null, false otherwise.

### Remarks

Win32 MSI API: [MsiRecordIsNull](#)

### Exceptions

Exception	Condition
-----------	-----------

**ArgumentOutOfRangeException** The field is less than 0 or greater than the number of fields in the Record.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation IsNull Method (fieldName)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► IsNull(String)

C# ▾

Reports whether a record field is null.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool IsNull(  
    string fieldName  
)
```

```
Public Function IsNull ( _  
    fieldName As String _  
) As Boolean
```

```
public:  
bool IsNull(  
    String^ fieldName  
)
```

### Parameters

#### fieldName (String)

Specifies the field to check.

### Return Value

True if the field is null, false otherwise.

### Exceptions

Exception

Condition

ArgumentOutOfRangeException

The field name does not match any of the named fields in the Record.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Item Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► Item

C# ▾

**Members**

Icon	Member	Description
	<a href="#">Item[String]</a>	Gets or sets a record field value.
	<a href="#">Item[Int32]</a>	Gets or sets a record field value.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Item Property (field)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► Item[Int32]

C# ▾

Gets or sets a record field value.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public Object this[  
    int field  
] { get; set; }
```

```
Public Default Property Item ( _  
    field As Integer _  
) As Object
```

```
public:  
property Object^ default[int field] {  
    Object^ get (int field);  
    void set (int field, Object^ value);  
}
```

## Parameters

### ***field* (Int32)**

Specifies the field of the Record to get or set.

## Remarks

Record fields are indexed starting with 1. Field 0 is a special format field.

When getting a field, the type of the object returned depends on the type of the Record field. The object will be one of: Int16, Int32, String, Stream or null. If the Record was returned from a View, the type will match that the field from the View query. Otherwise, the type will match the type of the last value set for the field.

When setting a field, the type of the object provided will be converted to match the View query that returned the Record, or if Record was not returned from a View then the type of the object provided will determine the type of the Record field. The object should be one of: Int16, Int32, String, Stream, or null.

The type-specific getters and setters are slightly more efficient than this property, since they don't have to do the extra work to infer the value's type every time.

Win32 MSI APIs: [MsiRecordGetInteger](#), [MsiRecordGetString](#),  
[MsiRecordSetInteger](#), [MsiRecordSetString](#)

## [-] Exceptions

Exception	Condition
<a href="#">ArgumentOutOfRangeException</a>	The field is less than 0 or greater than the number of fields in the Record.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Item Property (fieldName)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► Item[String]

C# ▾

Gets or sets a record field value.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public Object this[  
    string fieldName  
] { get; set; }
```

```
Public Default Property Item ( _  
    fieldName As String _  
) As Object
```

```
public:  
property Object^ default[String^ fieldName] {  
    Object^ get (String^ fieldName);  
    void set (String^ fieldName, Object^ value);  
}
```

## Parameters

### fieldName (String)

Specifies the name of the field of the Record to get or set.

## Remarks

When getting a field, the type of the object returned depends on the type of the Record field. The object will be one of: Int16, Int32, String, Stream or null.

When setting a field, the type of the object provided will be converted to match the View query that returned the record, or if Record was not returned from a view then the type of the object provided will determine

the type of the Record field. The object should be one of: Int16, Int32, String, Stream, or null.

## Exceptions

Exception	Condition
ArgumentOutOfRangeException	The name does not match any known field of the Record.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **SetInteger Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► **SetInteger()**

C# ▾

▀ **Members**

Icon	Member	Description
	<a href="#">SetInteger(Int32, Int32)</a>	Sets the value of a field to an integer.
	<a href="#">SetInteger(String, Int32)</a>	Sets the value of a field to an integer.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **SetInteger Method (field, value)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► **SetInteger(Int32, Int32)**

C# ▾

Sets the value of a field to an integer.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public void SetInteger(  
    int field,  
    int value  
)
```

```
Public Sub SetInteger ( _  
    field As Integer, _  
    value As Integer _  
)
```

```
public:  
void SetInteger(  
    int field,  
    int value  
)
```

## Parameters

### ***field* (Int32)**

Specifies the field to set.

### ***value* (Int32)**

new value of the field

## Remarks

Win32 MSI API: [MsiRecordSetInteger](#)

## **Exceptions**

Exception	Condition
ArgumentOutOfRangeException	The field is less than 0 or greater than the number of fields in the Record.

## **See Also**

[SetNullableInteger\(Int32, Nullable<Int32>\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## SetInteger Method (fieldName, value)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ►  
**SetInteger(String, Int32)**

C# ▾

Sets the value of a field to an integer.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void SetInteger(  
    string fieldName,  
    int value  
)
```

```
Public Sub SetInteger (  
    fieldName As String, _  
    value As Integer _  
)
```

```
public:  
void SetInteger(  
    String^ fieldName,  
    int value  
)
```

### Parameters

#### **fieldName (String)**

Specifies the field to set.

#### **value (Int32)**

new value of the field

### Exceptions

Exception

Condition

[ArgumentOutOfRangeException](#)  
The field name does not match any of the named fields in the Record.

**See Also**

[SetNullableInteger\(String, Nullable<Int32>\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **SetNullableInteger Method**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ►  
**SetNullableInteger()**

C# ▾

▀ Members

Icon	Member	Description
≡	<a href="#">SetNullableInteger(Int32, Nullable&lt;Int32&gt;)</a>	Sets the value of a field to a nullable integer.
≡	<a href="#">SetNullableInteger(String, Nullable&lt;Int32&gt;)</a>	Sets the value of a field to a nullable integer.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## SetNullableInteger Method (field, value)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ►  
**SetNullableInteger(Int32, Nullable<Int32>)**

Sets the value of a field to a nullable integer.

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void SetNullableInteger(  
    int field,  
    Nullable<int> value  
)
```

```
Public Sub SetNullableInteger ( _  
    field As Integer, _  
    value As Nullable(Of Integer) _  
)
```

```
public:  
void SetNullableInteger(  
    int field,  
    Nullable<int> value  
)
```

### Parameters

#### ***field* (Int32)**

Specifies the field to set.

#### ***value* (Nullable<Int32>)**

new value of the field

### Remarks

Win32 MSI API: [MsiRecordSetInteger](#)

**Exceptions**

Exception	Condition
<a href="#">ArgumentOutOfRangeException</a>	The field is less than 0 or greater than the number of fields in the Record.

**See Also**

[SetInteger\(Int32, Int32\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## SetNullableInteger Method (*fieldName*, *value*)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ►  
**SetNullableInteger(String, Nullable<Int32>)**

C# ▾

Sets the value of a field to a nullable integer.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void SetNullableInteger(  
    string fieldName,  
    Nullable<int> value  
)
```

```
Public Sub SetNullableInteger ( _  
    fieldName As String, _  
    value As Nullable(Of Integer) _  
)
```

```
public:  
void SetNullableInteger(  
    String^ fieldName,  
    Nullable<int> value  
)
```

### Parameters

#### ***fieldName* (String)**

Specifies the field to set.

#### ***value* (Nullable<Int32>)**

new value of the field

### Exceptions

Exception

Condition

[ArgumentOutOfRangeException](#)  
The field name does not match any of the  
named fields in the Record.

**See Also**

[SetInteger\(String, Int32\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **SetStream Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► SetStream()

C# ▾

■ Members

Icon	Member	Description
≡ ♦	<a href="#">SetStream(Int32, String)</a>	Sets a record stream field from a file. Stream data cannot be inserted into temporary fields.
≡ ♦	<a href="#">SetStream(String, String)</a>	Sets a record stream field from a file. Stream data cannot be inserted into temporary fields.
≡ ♦	<a href="#">SetStream(Int32, Stream)</a>	Sets a record stream field from a Stream object. Stream data cannot be inserted into temporary fields.
≡ ♦	<a href="#">SetStream(String, Stream)</a>	Sets a record stream field from a Stream object. Stream data cannot be inserted into temporary fields.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **SetStream Method (field, stream)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► **SetStream(Int32, Stream)**

C# ▾

Sets a record stream field from a Stream object. Stream data cannot be inserted into temporary fields.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void SetStream(  
    int field,  
    Stream stream  
)
```

```
Public Sub SetStream ( _  
    field As Integer, _  
    stream As Stream _  
)
```

```
public:  
void SetStream(  
    int field,  
    Stream^ stream  
)
```

### Parameters

#### ***field* (Int32)**

Specifies the field of the Record to set.

#### ***stream* (Stream)**

Specifies the stream data.

### Remarks

The stream persists if the Record is inserted into the Database and the

Database is committed.

Win32 MSI API: [MsiRecordsetStream](#)

## **Exceptions**

Exception	Condition
<a href="#">ArgumentOutOfRangeException</a>	The field is less than 0 or greater than the number of fields in the Record.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **SetStream Method (field, filePath)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► **SetStream(Int32, String)**

C# ▾

Sets a record stream field from a file. Stream data cannot be inserted in temporary fields.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void SetStream(  
    int field,  
    string filePath  
)
```

```
Public Sub SetStream ( _  
    field As Integer, _  
    filePath As String _  
)
```

```
public:  
void SetStream(  
    int field,  
    String^ filePath  
)
```

### Parameters

#### ***field* (Int32)**

Specifies the field of the Record to set.

#### ***filePath* (String)**

Specifies the path to the file containing the stream.

### Remarks

The contents of the specified file are read into a stream object. The

stream persists if the Record is inserted into the Database and the Database is committed.

To reset the stream to its beginning you must pass in null for filePath. Do not pass an empty string, "", to reset the stream.

Setting a stream with this method is more efficient than setting a field to FileStream object.

Win32 MSI API: [MsiRecordsetStream](#)

## **Exceptions**

Exception	Condition
<a href="#">ArgumentOutOfRangeException</a>	The index field is less than 0 or greater than the number of fields in the Record.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## SetStream Method (fieldName, stream)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ►  
**SetStream(String, Stream)**

C# ▾

Sets a record stream field from a Stream object. Stream data cannot be inserted into temporary fields.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void SetStream(  
    string fieldName,  
    Stream stream  
)
```

```
Public Sub SetStream ( _  
    fieldName As String, _  
    stream As Stream _  
)
```

```
public:  
void SetStream(  
    String^ fieldName,  
    Stream^ stream  
)
```

### Parameters

#### **fieldName (String)**

Specifies the field name of the Record to set.

#### **stream (Stream)**

Specifies the stream data.

### Remarks

The stream persists if the Record is inserted into the Database and the Database is committed.

## [-] Exceptions

Exception	Condition
ArgumentOutOfRangeException	The field name does not match any of the named fields in the Record.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## SetStream Method (fieldName, filePath)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ►  
**SetStream(String, String)**

C# ▾

Sets a record stream field from a file. Stream data cannot be inserted in temporary fields.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void SetStream(  
    string fieldName,  
    string filePath  
)
```

```
Public Sub SetStream ( _  
    fieldName As String, _  
    filePath As String _  
)
```

```
public:  
void SetStream(  
    String^ fieldName,  
    String^ filePath  
)
```

### Parameters

#### **fieldName (String)**

Specifies the field name of the Record to set.

#### **filePath (String)**

Specifies the path to the file containing the stream.

### Remarks

The contents of the specified file are read into a stream object. The stream persists if the Record is inserted into the Database and the Database is committed. To reset the stream to its beginning you must pass in null for filePath. Do not pass an empty string, "", to reset the stream.

Setting a stream with this method is more efficient than setting a field to FileStream object.

## - Exceptions

Exception	Condition
ArgumentOutOfRangeException	The field name does not match any of the named fields in the Record.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **SetString Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► SetString()

C# ▾

▀ **Members**

Icon	Member	Description
	<a href="#">SetString(Int32, String)</a>	Sets the value of a field to a string.
	<a href="#">SetString(String, String)</a>	Sets the value of a field to a string.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **SetString Method (field, value)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► **SetString(Int32, String)**

C# ▾

Sets the value of a field to a string.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public void SetString(  
    int field,  
    string value  
)
```

```
Public Sub SetString ( _  
    field As Integer, _  
    value As String _  
)
```

```
public:  
void SetString(  
    int field,  
    String^ value  
)
```

## Parameters

### ***field* (Int32)**

Specifies the field to set.

### ***value* (String)**

new value of the field

## Remarks

Win32 MSI API: [MsiRecordSetString](#)

## Exceptions

Exception	Condition
ArgumentOutOfRangeException	The field is less than 0 or greater than the number of fields in the Record.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## SetString Method (fieldName, value)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ► SetString(String, String)

C# ▾

Sets the value of a field to a string.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void SetString(  
    string fieldName,  
    string value  
)
```

```
Public Sub SetString ( _  
    fieldName As String, _  
    value As String _  
)
```

```
public:  
void SetString(  
    String^ fieldName,  
    String^ value  
)
```

### Parameters

#### fieldName (String)

Specifies the field to set.

#### value (String)

new value of the field

### Exceptions

Exception

Condition

**ArgumentOutOfRangeException**  
The field name does not match any of the named fields in the Record.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **ToString Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► **ToString()**

C# ▾

### Members

Icon	Member	Description
=	<a href="#">ToString()</a>	Gets a formatted string representation of the Record.  (Overrides <a href="#">Object.ToString()</a> .)
=	<a href="#">ToString(IFormatProvider)</a>	Gets a formatted string representation of the Record, optionally using a Session to format properties.
=	<a href="#">ToString(String)</a>	<b>Obsolete.</b> Gets a formatted string representation of the Record.
=	<a href="#">ToString(String, IFormatProvider)</a>	<b>Obsolete.</b> Gets a formatted string representation of the Record, optionally using a Session to format properties.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **ToString** Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► **ToString()**

C# ▾

Gets a formatted string representation of the Record.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override string ToString()
```

```
Public Overrides Function ToString As String
```

```
public:  
virtual String^ ToString() override
```

### Return Value

A formatted string representation of the Record.

### Remarks

If field 0 of the Record is set to a nonempty string, it is used to format the data in the Record.

Win32 MSI API: [MsiFormatRecord](#)

### See Also

[FormatString](#)

[FormatRecord\(Record\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **ToString Method (provider)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ►  
**ToString(IFormatProvider)**

C# ▾

Gets a formatted string representation of the Record, optionally using a Session to format properties.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string ToString(  
    IFormatProvider provider  
)
```

```
Public Function ToString ( _  
    provider As IFormatProvider _  
) As String
```

```
public:  
String^ ToString(  
    IFormatProvider^ provider  
)
```

### Parameters

#### **provider (IFormatProvider)**

an optional Session instance that will be used to lookup any properties in the Record's format string

### Return Value

A formatted string representation of the Record.

### Remarks

If field 0 of the Record is set to a nonempty string, it is used to format the data in the Record.

Win32 MSI API: [MsiFormatRecord](#)

**See Also**

[FormatString](#)

[FormatRecord\(Record\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **ToString Method (format)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Record ► **ToString(String)**

C# ▾

Gets a formatted string representation of the Record.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
[ObsoleteAttribute("This method is obsolete because
public string ToString(
    string format
)
```

```
<ObsoleteAttribute("This method is obsolete because
Public Function ToString ( _
    format As String _
) As String
```

```
[ObsoleteAttribute(L"This method is obsolete because
public:
String^ ToString(
    String^ format
)
```

## Parameters

### **format (String)**

String to be used to format the data in the Record, instead of the Record's format string.

## Return Value

A formatted string representation of the Record.

## Remarks

Win32 MSI API: [MsiFormatRecord](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **ToString Method (format, provider)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Record ► **ToString(String, IFormatProvider)**

C# ▾

Gets a formatted string representation of the Record, optionally using a Session to format properties.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
[ObsoleteAttribute("This method is obsolete because
public string ToString(
    string format,
    IFormatProvider provider
)
```

```
<ObsoleteAttribute("This method is obsolete because
Public Function ToString ( _
    format As String, _
    provider As IFormatProvider _ 
) As String
```

```
[ObsoleteAttribute(L"This method is obsolete because
public:
String^ ToString(
    String^ format,
    IFormatProvider^ provider
)
```

[-] **Parameters**

***format (String)***

String to be used to format the data in the Record, instead of the Record's format string.

***provider* ([IFormatProvider](#))**

an optional Session instance that will be used to lookup any properties in the Record's format string

**▀ Return Value**

A formatted string representation of the Record.

**▀ Remarks**

Win32 MSI API: [MsiFormatRecord](#)

**▀ See Also**

[FormatString](#)

[FormatRecord\(Record\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation **RegistryRoot Enumeration**

Deployment Tools Foundation Namespaces ►

[Microsoft.Deployment.WindowsInstaller](#) ► **RegistryRoot**

C#

Defines values for the Root column of the Registry, RemoveRegistry, and RegLocator tables.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public enum RegistryRoot
```

```
Public Enumeration RegistryRoot
```

```
public enum class RegistryRoot
```

## Members

Member	Description
<b>UserOrMachine</b>	HKEY_CURRENT_USER for a per-user installation, or HKEY_LOCAL_MACHINE for a per-machine installation.
<b>ClassesRoot</b>	HKEY_CLASSES_ROOT
<b>CurrentUser</b>	HKEY_CURRENT_USER
<b>LocalMachine</b>	HKEY_LOCAL_MACHINE
<b>Users</b>	HKEY_USERS

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **ReinstallModes Enumeration**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ReinstallModes

C# ▾

Specifies the reinstall mode for `ReinstallFeature(String, String, ReinstallModes)` or `ReinstallProduct(String, ReinstallModes)`.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
[FlagsAttribute]  
public enum ReinstallModes
```

```
<FlagsAttribute> _  
Public Enumeration ReinstallModes
```

```
[FlagsAttribute]  
public enum class ReinstallModes
```

### Members

Member	Description
<b>FileMissing</b>	Reinstall only if file is missing.
<b>FileOlderVersion</b>	Reinstall if file is missing, or older version
<b>FileEqualVersion</b>	Reinstall if file is missing, or equal or older version.
<b>FileExact</b>	Reinstall if file is missing, or not exact version.

<b>FileVerify</b>	Checksum executables, reinstall if missing or corrupt.
<b>FileReplace</b>	Reinstall all files, regardless of version.
<b>MachineData</b>	Insure required machine reg entries.
<b>UserData</b>	Insure required user reg entries.
<b>Shortcut</b>	Validate shortcuts items.
<b>Package</b>	Use re-cache source install package.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **RemoveFileModes Enumeration**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► RemoveFileModes

C#

Defines values for the InstallMode column of the RemoveFile table.

[-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
[FlagsAttribute]
public enum RemoveFileModes
```

```
<FlagsAttribute> _
Public Enumeration RemoveFileModes
```

```
[FlagsAttribute]
public enum class RemoveFileModes
```

[-] Members

Member	Description
<b>None</b>	Never remove.
<b>OnInstall</b>	Remove when the associated component is being installed (install state = local or source).
<b>OnRemove</b>	Remove when the associated component is being removed (install state = absent).

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ServiceAttributes Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ServiceAttributes

C# ▾

Defines values for the ServiceType, StartType, and ErrorControl columns of the ServiceInstall table.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
[FlagsAttribute]  
public enum ServiceAttributes
```

```
<FlagsAttribute> _  
Public Enumeration ServiceAttributes
```

```
[FlagsAttribute]  
public enum class ServiceAttributes
```

### Members

Member	Description
<b>None</b>	No flags.
<b>OwnProcess</b>	A Win32 service that runs its own process.
<b>ShareProcess</b>	A Win32 service that shares a process.
<b>Interactive</b>	A Win32 service that interacts with the desktop. This value cannot be used alone and must be added to either <b>OwnProcess</b> or <b>ShareProcess</b> .

<b>AutoStart</b>	Service starts during startup of the system.
<b>DemandStart</b>	Service starts when the service control manager calls the StartService function.
<b>Disabled</b>	Specifies a service that can no longer be started.
<b>ErrorMessage</b>	Logs the error, displays a message box and continues the startup operation.
<b>ErrorCritical</b>	Logs the error if it is possible and the system is restarted with the last configuration known to be good. If the last known-good configuration is being started the startup operation fails.
<b>ErrorControlVital</b>	When combined with other error flags, specifies that the overall install should fail if the service cannot be installed into the system.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# ServiceControlEvents

## Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ServiceControlEvents

C#

Defines values for the Event column of the ServiceControl table.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
[FlagsAttribute]
public enum ServiceControlEvents
```

```
<FlagsAttribute> _
Public Enumeration ServiceControlEvents
```

```
[FlagsAttribute]
public enum class ServiceControlEvents
```

### Members

Member	Description
<b>None</b>	No control events.
<b>Start</b>	During an install, starts the service during the StartServices action.
<b>Stop</b>	During an install, stops the service during the StopServices action.
<b>Delete</b>	During an install, deletes the service during the DeleteServices action.

<b>UninstallStart</b>	During an uninstall, starts the service during the StartServices action.
<b>UninstallStop</b>	During an uninstall, stops the service during the StopServices action.
<b>UninstallDelete</b>	During an uninstall, deletes the service during the DeleteServices action.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Session Class**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Session

C# ▾

The Session object controls the installation process. It opens the install database, which contains the installation tables and data.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public sealed class Session : InstallerHandle, IFormatProvider
```

```
Public NotInheritable Class Session
    Inherits InstallerHandle
    Implements IFormatProvider
```

```
public ref class Session sealed : public InstallerHandle
    IFormatProvider
```

### Members

All Members	Methods	Properties	Events
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static	<input checked="" type="checkbox"/> Declared <input checked="" type="checkbox"/> Inherited

Icon	Member	Description
	<a href="#">Close()</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidHandleException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">Components</a>	Gets an accessor for components in the current session.

	<a href="#">CreateObjRef(Type)</a>	Creates an object that contains all the relevant information required to generate a proxy used to communicate with a remote object.  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">CustomActionData</a>	Gets custom action data for the session that was supplied by the caller.
	<a href="#">Database</a>	Gets the Database for the install session.
	<a href="#">Dispose()</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">Dispose(Boolean)</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">DoAction(String)</a>	Executes a built-in action, custom action, or user-interface wizard action.
	<a href="#">DoAction(String, CustomActionData)</a>	Executes a built-in action, custom action, or user-interface wizard action.
		

	<a href="#">DoActionSequence(String)</a>	Executes an action sequence described in the specified table.
≡	<a href="#">Equals(Object)</a>	Tests whether this handle object is equal to another handle object. Two handle objects are equal if their types are the same and their native integer handles are the same.  (Inherited from <a href="#">InstallerHandle</a> .)
≡	<a href="#">EvaluateCondition(String)</a>	Evaluates a logical expression containing symbols and values.
≡	<a href="#">EvaluateCondition(String, Boolean)</a>	Evaluates a logical expression containing symbols and values, specifying a default value to be returned in case the condition is empty.
grid	<a href="#">Features</a>	Gets an accessor for features in the current session.
💡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">Format(String)</a>	Formats a string containing installer properties.

≡	<a href="#">FormatRecord(Record)</a>	Returns a formatted string from record data.
≡	<a href="#">FormatRecord(Record, String)</a>	<b>Obsolete.</b> Returns a formatted string from record data using a specified format.
≡ S	<a href="#">FromHandle(IntPtr, Boolean)</a>	Creates a new Session object from an integer session handle.
≡	<a href="#">GetHashCode()</a>	Gets a hash value for the handle object.  (Inherited from <a href="#">InstallerHandle</a> .)
≡	<a href="#">GetLifetimeService()</a>	Retrieves the current lifetime service object that controls the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject</a> .)
≡	<a href="#">GetMode(InstallRunMode)</a>	Gets the designated mode flag for the current install session.
≡	<a href="#">GetProductProperty(String)</a>	Retrieves product properties (not session properties) from the product database.
≡	<a href="#">GetSourcePath(String)</a>	Gets the full path to the designated folder on the source media or serve

		image.
	<a href="#">GetTargetPath(String)</a>	Gets the full path to the designated folder on the installation target drive.
	<a href="#">GetTotalCost()</a>	Gets the total disk space per drive required for the installation.
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
	<a href="#">Handle</a>	Gets the native integer handle.  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">InitializeLifetimeService()</a>	Obtains a lifetime service object to control the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">IsClosed</a>	Checks if the handle is closed. When closed, method calls on the handle object may throw an <a href="#">InvalidOperationException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">Item[String]</a>	Gets or sets the string value of a named installer property, as maintained by the Session object in the in-memory Property table, or, if it is prefixed with a percent sign (%), the value of a system environment

		variable for the current process.
	<a href="#">Language</a>	Gets the numeric language ID used by the current install session.
	<a href="#">Log(String)</a>	Writes a message to the log, if logging is enabled.
	<a href="#">Log(String, Object[])</a>	Writes a formatted message to the log, if logging is enabled.
	<a href="#">MemberwiseClone(Boolean)</a>	Creates a shallow copy of the current <a href="#">MarshalByRefObject</a> object  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Message(InstallMessage, Record)</a>	Performs any enabled logging operations and defers execution to the UI handler object associated with the engine.
	<a href="#">SetInstallLevel(Int32)</a>	Sets the install level for the current installation to a specified value and recalculates the Select and Installed states for all features in the Feature table. Also sets the Action state of each component in the Component

		table based on the new level.
≡	<a href="#">SetMode(InstallRunMode, Boolean)</a>	Sets the designated mode flag for the current install session.
≡	<a href="#">SetTargetPath(String, String)</a>	Sets the full path to the designated folder on the installation target drive.
≡	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> . (Inherited from <a href="#">Object</a> .)
≡	<a href="#">VerifyDiskSpace()</a>	Checks to see if sufficient disk space is present for the current installation.

## [-] Remarks

This object is associated with a standard set of action functions, each performing particular operations on data from one or more tables. Additional custom actions may be added for particular product installations. The basic engine function is a sequencer that fetches sequential records from a designated sequence table, evaluates any specified condition expression, and executes the designated action. Actions not recognized by the engine are deferred to the UI handler object for processing, usually dialog box sequences.

Note that only one Session object can be opened by a single process.

## [-] Inheritance Hierarchy

[Object](#)

└ [MarshalByRefObject](#)

- └ InstallerHandle
  - └ Session

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **Components Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Session ► Components

C# ▾

Gets an accessor for components in the current session.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public ComponentInfoCollection Components { get; }
```

```
Public ReadOnly Property Components As ComponentInfo
```

```
public:  
property ComponentInfoCollection^ Components {  
    ComponentInfoCollection^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation **CustomActionData** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Session ►

## CustomActionData

C# ▾

Gets custom action data for the session that was supplied by the caller.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public CustomActionData CustomActionData { get; }
```

```
Public ReadOnly Property CustomActionData As CustomA
```

```
public:  
property CustomActionData^ CustomActionData {  
    CustomActionData^ get ();  
}
```

### See Also

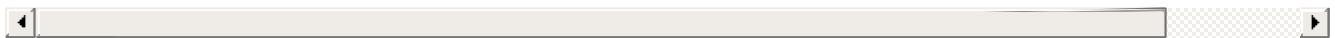
[DoAction\(String, CustomActionData\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation Database Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Session ► Database

C# ▾

Gets the Database for the install session.

## - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public Database Database { get; }
```

```
Public ReadOnly Property Database As Database
```

```
public:  
property Database^ Database {  
    Database^ get ();  
}
```

## - Remarks

Normally there is no need to close this Database object. The same object can be used throughout the lifetime of the Session, and it will be closed when the Session is closed.

Win32 MSI API: [MsiGetActiveDatabase](#)

## - Exceptions

Exception	Condition
<a href="#">InvalidOperationException</a>	the Session handle is invalid
<a href="#">InstallerException</a>	the Database cannot be accessed

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## DoAction Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Session ► DoAction()

C# ▾

### Members

Icon	Member	Description
	<a href="#">DoAction(String)</a>	Executes a built-in action, custom action, or user-interface wizard action.
	<a href="#">DoAction(String, CustomActionData)</a>	Executes a built-in action, custom action, or user-interface wizard action.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **DoAction Method (action)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Session ►  
**DoAction(String)**

Executes a built-in action, custom action, or user-interface wizard action.

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void DoAction(  
    string action  
)
```

```
Public Sub DoAction ( _  
    action As String _  
)
```

```
public:  
void DoAction(  
    String^ action  
)
```

### Parameters

#### ***action* (String)**

Name of the action to execute. Case-sensitive.

### Remarks

The DoAction method executes the action that corresponds to the name supplied. If the name is not recognized by the installer as a built-in action or as a custom action in the CustomAction table, the name is passed to the user-interface handler object, which can invoke a function or a dialog box. If a null action name is supplied, the installer uses the upper-case value of the ACTION property as the action to perform. If no property value is defined, the default action is performed, defined as "INSTALL".

Actions that update the system, such as the InstallFiles and WriteRegistryValues actions, cannot be run by calling MsiDoAction. The exception to this rule is if DoAction is called from a custom action that is scheduled in the InstallExecuteSequence table between the InstallInitialize and InstallFinalize actions. Actions that do not update the system, such as AppSearch or CostInitialize, can be called.

Win32 MSI API: [MsiDoAction](#)

## **Exceptions**

Exception	Condition
<a href="#">InvalidHandleException</a>	the Session handle is invalid
<a href="#">InstallCanceledException</a>	the user exited the installation

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## DoAction Method (action, actionData)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Session ►  
**DoAction(String, CustomActionData)**

Executes a built-in action, custom action, or user-interface wizard action.

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void DoAction(  
    string action,  
    CustomActionData actionData  
)
```

```
Public Sub DoAction (  
    action As String, _  
    actionData As CustomActionData _  
)
```

```
public:  
void DoAction(  
    String^ action,  
    CustomActionData^ actionData  
)
```

### Parameters

#### ***action* (String)**

Name of the action to execute. Case-sensitive.

#### ***actionData* (CustomActionData)**

Optional data to be passed to a deferred custom action.

### Remarks

The DoAction method executes the action that corresponds to the name supplied. If the name is not recognized by the installer as a built-in action or as a custom action in the CustomAction table, the name is passed to the user-interface handler object, which can invoke a function or a dialog box. If a null action name is supplied, the installer uses the upper-case value of the ACTION property as the action to perform. If no property value is defined, the default action is performed, defined as "INSTALL".

Actions that update the system, such as the InstallFiles and WriteRegistryValues actions, cannot be run by calling MsiDoAction. The exception to this rule is if DoAction is called from a custom action that is scheduled in the InstallExecuteSequence table between the InstallInitialize and InstallFinalize actions. Actions that do not update the system, such as AppSearch or CostInitialize, can be called.

If the called action is a deferred, rollback, or commit custom action, then the supplied *actionData* will be available via the [CustomActionData](#) property of that custom action's session.

Win32 MSI API: [MsiDoAction](#)

## Exceptions

Exception	Condition
<a href="#">InvalidHandleException</a>	the Session handle is invalid
<a href="#">InstallCanceledException</a>	the user exited the installation

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## DoActionSequence Method (sequenceTable)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Session ►  
**DoActionSequence(String)**

Executes an action sequence described in the specified table.

C#

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public void DoActionSequence(  
    string sequenceTable  
)
```

```
Public Sub DoActionSequence ( _  
    sequenceTable As String _  
)
```

```
public:  
void DoActionSequence(  
    String^ sequenceTable  
)
```

### - Parameters

#### **sequenceTable (String)**

Name of the table containing the action sequence.

### - Remarks

This method queries the specified table, ordering the actions by the numbers in the Sequence column. For each row retrieved, an action is executed, provided that any supplied condition expression does not evaluate to FALSE.

An action sequence containing any actions that update the system, such

as the InstallFiles and WriteRegistryValues actions, cannot be run by calling DoActionSequence. The exception to this rule is if DoActionSequence is called from a custom action that is scheduled in the InstallExecuteSequence table between the InstallInitialize and InstallFinalize actions. Actions that do not update the system, such as AppSearch or CostInitialize, can be called.

Win32 MSI API: [MsiSequence](#)

## **Exceptions**

Exception	Condition
<a href="#">InvalidHandleException</a>	the Session handle is invalid
<a href="#">InstallCanceledException</a>	the user exited the installation

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## EvaluateCondition Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Session ►

**EvaluateCondition()**

C# ▾

### Members

Icon	Member	Description
≡	<a href="#">EvaluateCondition(String)</a>	Evaluates a logical expression containing symbols and values.
≡	<a href="#">EvaluateCondition(String, Boolean)</a>	Evaluates a logical expression containing symbols and values, specifying a default value to be returned in case the condition is empty.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# EvaluateCondition Method (condition)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Session ►  
**EvaluateCondition(String)**

C# ▾

Evaluates a logical expression containing symbols and values.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool EvaluateCondition(  
    string condition  
)
```

```
Public Function EvaluateCondition ( _  
    condition As String _  
) As Boolean
```

```
public:  
bool EvaluateCondition(  
    String^ condition  
)
```

## Parameters

### *condition* (String)

conditional expression

## Return Value

The result of the condition evaluation

## Remarks

Win32 MSI API: [MsiEvaluateCondition](#)

## Exceptions

Exception	Condition
InvalidHandleException	the Session handle is invalid
ArgumentNullException	the condition is null or empty
SyntaxErrorException	the conditional expression is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## EvaluateCondition Method (condition, defaultValue)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Session ►  
**EvaluateCondition(String, Boolean)**

C# ▾

Evaluates a logical expression containing symbols and values, specifying a default value to be returned in case the condition is empty.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool EvaluateCondition(  
    string condition,  
    bool defaultValue  
)
```

```
Public Function EvaluateCondition ( _  
    condition As String, _  
    defaultValue As Boolean _  
) As Boolean
```

```
public:  
bool EvaluateCondition(  
    String^ condition,  
    bool defaultValue  
)
```

### [-] Parameters

#### **condition (String)**

conditional expression

#### **defaultValue (Boolean)**

value to return if the condition is empty

### [-] Return Value

The result of the condition evaluation

## **[-] Remarks**

Win32 MSI API: [MsiEvaluateCondition](#)

## **[-] Exceptions**

Exception	Condition
<a href="#">InvalidHandleException</a>	the Session handle is invalid
<a href="#">SyntaxErrorException</a>	the conditional expression is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Features Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Session ► Features

C# ▾

Gets an accessor for features in the current session.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public FeatureInfoCollection Features { get; }
```

```
Public ReadOnly Property Features As FeatureInfoColl
```

```
public:  
property FeatureInfoCollection^ Features {  
    FeatureInfoCollection^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation Format Method (format)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Session ► Format(String)

C# ▾

Formats a string containing installer properties.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string Format(  
    string format  
)
```

```
Public Function Format ( _  
    format As String _  
) As String
```

```
public:  
String^ Format(  
    String^ format  
)
```

## Parameters

### format (String)

A format string containing property tokens

## Return Value

A formatted string containing property data

## Remarks

Win32 MSI API: [MsiFormatRecord](#)

## Exceptions

Exception	Condition

[InvalidHandleException](#)

the Record handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## FormatRecord Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Session ► FormatRecord()

C#

### Members

Icon	Member	Description
	<a href="#">FormatRecord(Record)</a>	Returns a formatted string from record data.
	<a href="#">FormatRecord(Record, String)</a>	<b>Obsolete.</b> Returns a formatted string from record data using a specified format.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **FormatRecord Method (record)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Session ►  
**FormatRecord(Record)**

C# ▾

Returns a formatted string from record data.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public string FormatRecord(  
    Record record  
)
```

```
Public Function FormatRecord ( _  
    record As Record _  
) As String
```

```
public:  
String^ FormatRecord(  
    Record^ record  
)
```

[-] **Parameters**

**record (Record)**

Record object containing a template and data to be formatted. The template string must be set in field 0 followed by any referenced data parameters.

[-] **Return Value**

A formatted string containing the record data

[-] **Remarks**

Win32 MSI API: [MsiFormatRecord](#)

## Exceptions

Exception	Condition
InvalidHandleException	the Record handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## FormatRecord Method (record, format)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Session ►  
**FormatRecord(Record, String)**

C#

Returns a formatted string from record data using a specified format.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
[ObsoleteAttribute("This method is obsolete because
public string FormatRecord(
    Record record,
    string format
)
```

```
<ObsoleteAttribute("This method is obsolete because
Public Function FormatRecord ( _
    record As Record, _
    format As String _
) As String
```

```
[ObsoleteAttribute(L"This method is obsolete because
public:
String^ FormatRecord(
    Record^ record,
    String^ format
)
```

### Parameters

#### **record (Record)**

Record object containing a template and data to be formatted

#### **format (String)**

Format string to be used instead of field 0 of the Record

## **[-] Return Value**

A formatted string containing the record data

## **[-] Remarks**

Win32 MSI API: [MsiFormatRecord](#)

## **[-] Exceptions**

Exception	Condition
<a href="#">InvalidHandleException</a>	the Record handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## FromHandle Method (handle, ownsHandle)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Session ►  
**FromHandle(IntPtr, Boolean)**

C# ▾

Creates a new Session object from an integer session handle.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static Session FromHandle(  
    IntPtr handle,  
    bool ownsHandle  
)
```

```
Public Shared Function FromHandle ( _  
    handle As IntPtr, _  
    ownsHandle As Boolean _  
) As Session
```

```
public:  
static Session^ FromHandle(  
    IntPtr handle,  
    bool ownsHandle  
)
```

### Parameters

#### **handle (IntPtr)**

Integer session handle

#### **ownsHandle (Boolean)**

true to close the handle when this object is disposed or finalized

### Remarks

This method is only provided for interop purposes. A Session object should normally be obtained by calling [OpenPackage\(Database, Boolean\)](#) or [OpenProduct\(String\)](#).

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetMode Method (mode)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Session ►

### **GetMode(InstallRunMode)**

C# ▾

Gets the designated mode flag for the current install session.

#### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool GetMode(  
    InstallRunMode mode  
)
```

```
Public Function GetMode ( _  
    mode As InstallRunMode _  
) As Boolean
```

```
public:  
bool GetMode(  
    InstallRunMode mode  
)
```

#### [-] Parameters

##### **mode (InstallRunMode)**

The type of mode to be checked.

#### [-] Return Value

The value of the designated mode flag.

#### [-] Remarks

Note that only the following run modes are available to read from a deferred custom action:

- Scheduled

- Rollback
- Commit

Win32 MSI API: [MsiGetMode](#)

## **[-] Exceptions**

Exception	Condition
<a href="#">InvalidHandleException</a>	the Session handle is invalid
<a href="#">ArgumentOutOfRangeException</a>	an invalid mode flag was specified

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## GetProductProperty Method (property)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Session ►  
**GetProductProperty(String)**

C# ▾

Retrieves product properties (not session properties) from the product database.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string GetProductProperty(  
    string property  
)
```

```
Public Function GetProductProperty ( _  
    property As String _  
) As String
```

```
public:  
String^ GetProductProperty(  
    String^ property  
)
```

### [-] Parameters

*property* (String)

### [-] Return Value

Value of the property, or an empty string if the property is not set.

### [-] Remarks

Note this is not the correct method for getting ordinary session properties. For that, see the indexer on the Session class.

## Win32 MSI API: [MsiGetProperty](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetSourcePath Method**  
**(directory)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Session ►  
**GetSourcePath(String)**

C# ▾

Gets the full path to the designated folder on the source media or serve image.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public string GetSourcePath(  
    string directory  
)
```

```
Public Function GetSourcePath ( _  
    directory As String _  
) As String
```

```
public:  
String^ GetSourcePath(  
    String^ directory  
)
```

[-] **Parameters**

*directory* (**String**)

[-] **Remarks**

Win32 MSI API: [MsiGetSourcePath](#)

[-] **Exceptions**

Exception	Condition
ArgumentException	the folder was not found in the Directory

	table
InvalidOperationException	the Session handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **GetTargetPath Method (directory)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Session ►

## **GetTargetPath(String)**

C# ▾

Gets the full path to the designated folder on the installation target drive

### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string GetTargetPath(  
    string directory  
)
```

```
Public Function GetTargetPath ( _  
    directory As String _  
) As String
```

```
public:  
String^ GetTargetPath(  
    String^ directory  
)
```

### [-] Parameters

*directory* ([String](#))

### [-] Remarks

Win32 MSI API: [MsiGetTargetPath](#)

### [-] Exceptions

Exception	Condition
<a href="#">ArgumentException</a>	the folder was not found in the Directory table
<a href="#">InvalidHandleException</a>	the Session handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)  
Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetTotalCost Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Session ► **GetTotalCost()**

C# ▾

Gets the total disk space per drive required for the installation.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public IList<InstallCost> GetTotalCost()
```

```
Public Function GetTotalCost As IList(Of InstallCost)
```

```
public:  
IList<InstallCost>^ GetTotalCost()
```

### Return Value

A list of InstallCost structures, specifying the cost for each drive

### Remarks

Win32 MSI API: [MsiEnumComponentCosts](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Item Property (property)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Session ► Item[String]

C# ▾

Gets or sets the string value of a named installer property, as maintained by the Session object in the in-memory Property table, or, if it is prefixed with a percent sign (%), the value of a system environment variable for the current process.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string this[  
    string property  
] { get; set; }
```

```
Public Default Property Item ( _  
    property As String _  
) As String
```

```
public:  
property String^ default[String^ property] {  
    String^ get (String^ property);  
    void set (String^ property, String^ value);  
}
```

## Parameters

*property* (String)

## Remarks

Win32 MSI APIs: [MsiGetProperty](#), [MsiSetProperty](#)

## Exceptions

Exception

Condition

[InvalidHandleException](#)

the Session handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Language Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Session ► Language

C#

Gets the numeric language ID used by the current install session.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public int Language { get; }
```

```
Public Readonly Property Language As Integer
```

```
public:  
property int Language {  
    int get ();  
}
```

### - Remarks

Win32 MSI API: [MsiGetLanguage](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Log Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Session ► Log()

C# ▾

### Members

Icon	Member	Description
	<a href="#">Log(String)</a>	Writes a message to the log, if logging is enabled.
	<a href="#">Log(String, Object[])</a>	Writes a formatted message to the log if logging is enabled.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Log Method (msg)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Session ► Log(String)

C# ▾

Writes a message to the log, if logging is enabled.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Log(  
    string msg  
)
```

```
Public Sub Log ( _  
    msg As String _  
)
```

```
public:  
void Log(  
    String^ msg  
)
```

### Parameters

#### *msg* (String)

The line to be written to the log

### Remarks

Win32 MSI API: [MsiProcessMessage](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Log Method (format, args)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Session ► Log(String, Object[])

C# ▾

Writes a formatted message to the log, if logging is enabled.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Log(  
    string format,  
    params Object[] args  
)
```

```
Public Sub Log ( _  
    format As String, _  
    ParamArray args As Object() _  
)
```

```
public:  
void Log(  
    String^ format,  
    ... array<Object^>^ args  
)
```

### Parameters

#### **format (String)**

The line to be written to the log, containing 0 or more format specifiers

#### **args ( Object [] )**

An array containing 0 or more objects to be formatted

### Remarks

Win32 MSI API: [MsiProcessMessage](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Message Method (*messageType*, *record*)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Session ►  
**Message(InstallMessage, Record)**

C# ▾

Performs any enabled logging operations and defers execution to the U handler object associated with the engine.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public MessageResult Message(  
    InstallMessage messageType,  
    Record record  
)
```

```
Public Function Message ( _  
    messageType As InstallMessage, _  
    record As Record _  
) As MessageResult
```

```
public:  
    MessageResult Message(  
        InstallMessage messageType,  
        Record^ record  
)
```

### Parameters

#### *messageType* ([InstallMessage](#))

Type of message to be processed

#### *record* ([Record](#))

Contains message-specific fields

### Return Value

A message-dependent return value

## [-] Remarks

Logging may be selectively enabled for the various message types. See the [EnableLog\(InstallLogModes, String\)](#) method.

If record field 0 contains a formatting string, it is used to format the data in the other fields. Else if the message is an error, warning, or user message, an attempt is made to find a message template in the Error table for the current database using the error number found in field 1 of the record for message types and return values.

The *messageType* parameter may also include message-box flags from the following enumerations:

`System.Windows.Forms.MessageBoxButtons`,  
`System.Windows.Forms.MessageBoxDefaultButton`,  
`System.Windows.Forms.MessageBoxIcon`. These flags can be combined with the `InstallMessage` with a bitwise OR.

Note, this method never returns Cancel or Error values. Instead, appropriate exceptions are thrown in those cases.

Win32 MSI API: [MsiProcessMessage](#)

## [-] Exceptions

Exception	Condition
<a href="#">InvalidOperationException</a>	the Session or Record handle is invalid
<a href="#">ArgumentOutOfRangeException</a>	an invalid message kind is specified
<a href="#">InstallCanceledException</a>	the user exited the installation
<a href="#">InstallerException</a>	the message-handler failed for an unknown reason

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## SetInstallLevel Method (installLevel)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Session ►  
**SetInstallLevel(Int32)**

C#

Sets the install level for the current installation to a specified value and recalculates the Select and Installed states for all features in the Feature table. Also sets the Action state of each component in the Component table based on the new level.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void SetInstallLevel(  
    int installLevel  
)
```

```
Public Sub SetInstallLevel ( _  
    installLevel As Integer _  
)
```

```
public:  
void SetInstallLevel(  
    int installLevel  
)
```

### Parameters

*installLevel* (Int32)

New install level

### Remarks

The SetInstallLevel method sets the following:

- The installation level for the current installation to a specified value

- The Select and Installed states for all features in the Feature table
- The Action state of each component in the Component table, based on the new level

If 0 or a negative number is passed in the iInstallLevel parameter, the current installation level does not change, but all features are still updated based on the current installation level.

Win32 MSI API: [MsiSetInstallLevel](#)

## **[-] Exceptions**

Exception	Condition
<a href="#">InvalidHandleException</a>	the Session handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **SetMode Method (mode, value)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Session ►

**SetMode(InstallRunMode, Boolean)**

C# ▾

Sets the designated mode flag for the current install session.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void SetMode(  
    InstallRunMode mode,  
    bool value  
)
```

```
Public Sub SetMode ( _  
    mode As InstallRunMode, _  
    value As Boolean _  
)
```

```
public:  
void SetMode(  
    InstallRunMode mode,  
    bool value  
)
```

### Parameters

#### **mode (InstallRunMode)**

The type of mode to be set.

#### **value (Boolean)**

The desired value of the mode.

### Remarks

Win32 MSI API: [MsiSetMode](#)

## Exceptions

Exception	Condition
InvalidHandleException	the Session handle is invalid
ArgumentOutOfRangeException	an invalid mode flag was specified
InvalidOperationException	the mode cannot not be set

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## SetTargetPath Method (directory, value)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Session ►  
**SetTargetPath(String, String)**

C# ▾

Sets the full path to the designated folder on the installation target drive

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void SetTargetPath(  
    string directory,  
    string value  
)
```

```
Public Sub SetTargetPath ( _  
    directory As String, _  
    value As String _  
)
```

```
public:  
void SetTargetPath(  
    String^ directory,  
    String^ value  
)
```

### Parameters

**directory (String)**

**value (String)**

### Remarks

Setting the target path of a directory changes the path specification for the directory in the in-memory Directory table. Also, the path

specifications of all other path objects in the table that are either subordinate or equivalent to the changed path are updated to reflect the change. The properties for each affected path are also updated.

If an error occurs in this function, all updated paths and properties revert to their previous values. Therefore, it is safe to treat errors returned by this function as non-fatal.

Do not attempt to configure the target path if the components using those paths are already installed for the current user or for a different user. Check the ProductState property before setting the target path to determine if the product containing this component is installed.

Win32 MSI API: [MsiSetTargetPath](#)

## **Exceptions**

Exception	Condition
<a href="#">ArgumentException</a>	the folder was not found in the Directory table
<a href="#">InvalidHandleException</a>	the Session handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation VerifyDiskSpace Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Session ►

### VerifyDiskSpace()

C# ▾

Checks to see if sufficient disk space is present for the current installation.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool VerifyDiskSpace()
```

```
Public Function VerifyDiskSpace As Boolean
```

```
public:  
bool VerifyDiskSpace()
```

#### Return Value

True if there is sufficient disk space; false otherwise.

#### Remarks

Win32 MSI API: [MsiVerifyDiskSpace](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **ShortcutTarget** Structure

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ShortcutTarget

C# ▾

Holds information about the target of a shortcut file.

## - Declaration Syntax

C#

Visual Basic

Visual C++

```
public struct ShortcutTarget
```

```
Public Structure ShortcutTarget
```

```
public value class ShortcutTarget
```

## - Members

All Members

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon	Member	Description
	<a href="#">ComponentCode</a>	Gets the target component code of the shortcut, or null if not available.
	<a href="#">Equality(ShortcutTarget, ShortcutTarget)</a>	Tests whether two shortcut targets have the same product code, feature and/or component code.
	<a href="#">Equals(Object)</a>	Tests whether two shortcut targets have the same product code, feature and/or component code.

		(Overrides <a href="#">ValueType.Equals(Object)</a> .)
	<a href="#">Feature</a>	Gets the name of the target feature of the shortcut, or null if not available.
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetHashCode()</a>	Generates a hash code using all parts of the shortcut target.  (Overrides <a href="#">ValueType.GetHashCode()</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
	<a href="#">Inequality(ShortcutTarget, ShortcutTarget)</a>	Tests whether two shortcut targets have the same product code, feature and/or component code.
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">ProductCode</a>	Gets the target product code of the shortcut, or null if not available.
	<a href="#">ToString()</a>	Returns the fully qualified type name of this instance.

(Inherited from [ValueType](#).)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ComponentCode Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ShortcutTarget ►

### ComponentCode

C#

Gets the target component code of the shortcut, or null if not available.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string ComponentCode { get; }
```

```
Public ReadOnly Property ComponentCode As String
```

```
public:  
property String^ ComponentCode {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Equality Operator

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ShortcutTarget ►

### Equality(ShortcutTarget, ShortcutTarget)

C# ▾

Tests whether two shortcut targets have the same product code, feature and/or component code.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static bool operator ==(
    ShortcutTarget st1,
    ShortcutTarget st2
)
```

```
Public Shared Operator = ( _
    st1 As ShortcutTarget, _
    st2 As ShortcutTarget _
) As Boolean
```

```
public:
static bool operator ==(
    ShortcutTarget st1,
    ShortcutTarget st2
)
```

#### Parameters

##### st1 (ShortcutTarget)

The first shortcut target to compare.

##### st2 (ShortcutTarget)

The second shortcut target to compare.

#### Return Value

True if all parts of the targets are the same, else false.

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Equals Method (obj)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ShortcutTarget ►

### Equals(Object)

C# ▾

Tests whether two shortcut targets have the same product code, feature and/or component code.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public override bool Equals(  
    Object obj  
)
```

```
Public Overrides Function Equals ( _  
    obj As Object _  
) As Boolean
```

```
public:  
virtual bool Equals(  
    Object^ obj  
) override
```

#### Parameters

##### obj (Object)

The shortcut target to compare to the current object.

#### Return Value

True if *obj* is a shortcut target and all parts of the targets are the same, else false.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Feature Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ShortcutTarget ► Feature

C#

Gets the name of the target feature of the shortcut, or null if not available.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public string Feature { get; }
```

```
Public Readonly Property Feature As String
```

```
public:  
property String^ Feature {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetHashCode Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ShortcutTarget ►

**GetHashCode()**

C# ▾

Generates a hash code using all parts of the shortcut target.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public override int GetHashCode()
```

```
Public Overrides Function GetHashCode As Integer
```

```
public:  
virtual int GetHashCode() override
```

[-] **Return Value**

An integer suitable for hashing the shortcut target.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Inequality Operator

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ShortcutTarget ►

### Inequality(ShortcutTarget, ShortcutTarget)

C# ▾

Tests whether two shortcut targets have the same product code, feature and/or component code.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static bool operator !=(
    ShortcutTarget st1,
    ShortcutTarget st2
)
```

```
Public Shared Operator <> ( _
    st1 As ShortcutTarget, _
    st2 As ShortcutTarget _
) As Boolean
```

```
public:
static bool operator !=(
    ShortcutTarget st1,
    ShortcutTarget st2
)
```

#### Parameters

##### **st1 (ShortcutTarget)**

The first shortcut target to compare.

##### **st2 (ShortcutTarget)**

The second shortcut target to compare.

#### Return Value

True if any parts of the targets are different, else false.

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ProductCode Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ShortcutTarget ►

### ProductCode

C#

Gets the target product code of the shortcut, or null if not available.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string ProductCode { get; }
```

```
Public ReadOnly Property ProductCode As String
```

```
public:  
property String^ ProductCode {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **SourceList Class**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► SourceList

A list of sources for an installed product or patch.

C# ▾

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public class SourceList : ICollection<string>,
    IEnumerable<string>, IEnumerable
```

```
Public Class SourceList
    Implements ICollection(Of String), IEnumerable
    IEnumerable
```

```
public ref class SourceList : ICollection<String^>,
    IEnumerable<String^>, IEnumerable
```

## Members

All Members

Methods

Properties

Public

Instance

Declared

Protected

Static

Inherited

Icon	Member	Description
≡	Add(String)	Adds a network or URL source to the source list of the installed product.
≡	Clear()	Clears sources of all types: network, url, and media.
≡	ClearNetworkSources()	Removes all network sources from

		the list. URL sources are not affected.
≡	<a href="#">ClearUrlSources()</a>	Removes all URL sources from the list. Network sources are not affected.
≡	<a href="#">Contains(String)</a>	Checks if the specified source exists in the list.
≡	<a href="#">CopyTo(String[], Int32)</a>	Copies the network and URL sources from this list into an array.
	<a href="#">Count</a>	Gets the number of network and URL sources in the list.
	<a href="#">DiskPrompt</a>	Gets or sets the prompt template that is used when prompting the user for installation media.
≡	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">ForceResolution()</a>	Forces the installer to search the

		source list for a valid source the next time a source is required. For example, when the installer performs an installation or reinstallation, or when it requires the path for a component that is set to run from source.
	<a href="#">GetEnumerator()</a>	Enumerates the network and URL sources in the source list of the patch or product installation.
	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance  (Inherited from <a href="#">Object</a> .)
	<a href="#">Insert(String, Int32)</a>	Adds or reorders a network or URL source for the product or patch.
	<a href="#">IsReadOnly</a>	Gets a boolean value indicating whether the list is read-only. A SourceList is never read-only.
	<a href="#">Item[String]</a>	Gets or sets source list information properties of a product or patch installation.

	<a href="#">LastUsedSource</a>	Gets or sets the most recently used source location for the product.
	<a href="#">LastUsedType</a>	Gets the type of the last-used source.
	<a href="#">MediaList</a>	Gets the list of disks registered for the media source of the patch or product installation.
	<a href="#">MediaPackagePath</a>	Gets or sets the path relative to the root of the installation media.
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">PackageName</a>	Gets or sets the name of the Windows Installer package or patch package on the source.
	<a href="#">Remove(String)</a>	Removes a network or URL source.
	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)

## Inheritance Hierarchy

[Object](#)

## SourceList

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



## Deployment Tools Foundation **Add Method (item)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceList ► Add(String)

C# ▾

Adds a network or URL source to the source list of the installed product

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Add(  
    string item  
)
```

```
Public Sub Add ( _  
    item As String _  
)
```

```
public:  
virtual void Add(  
    String^ item  
) sealed
```

### Parameters

#### *item* (String)

Path to the source to be added. This parameter is expected to contain only the path without the filename.

### Remarks

If this method is called with a new source, the installer adds the source to the end of the source list.

If this method is called with a source already existing in the source list, it has no effect.

Win32 MSI APIs: [MsiSourceListAddSource](#), [MsiSourceListAddSourceE](#)

## See Also

[Insert\(String, Int32\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Clear Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceList ► **Clear()**

C# ▾

Clears sources of all types: network, url, and media.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public void Clear()
```

```
Public Sub Clear
```

```
public:  
virtual void Clear() sealed
```

[-] **Remarks**

Win32 MSI API: [MsiSourceListClearAll](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **ClearNetworkSources** Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceList ►

### **ClearNetworkSources()**

C#

Removes all network sources from the list. URL sources are not affected.

#### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void ClearNetworkSources()
```

```
Public Sub ClearNetworkSources
```

```
public:  
void ClearNetworkSources()
```

#### [-] Remarks

Win32 MSI API: [MsiSourceListClearAllEx](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **ClearUrlSources** Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceList ►

### **ClearUrlSources()**

C#

Removes all URL sources from the list. Network sources are not affected.

#### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void ClearUrlSources()
```

```
Public Sub ClearUrlSources
```

```
public:  
void ClearUrlSources()
```

#### [-] Remarks

Win32 MSI API: [MsiSourceListClearAllEx](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Contains Method (item)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceList ►

**Contains(String)**

C# ▾

Checks if the specified source exists in the list.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public bool Contains(  
    string item  
)
```

```
Public Function Contains ( _  
    item As String _  
) As Boolean
```

```
public:  
virtual bool Contains(  
    String^ item  
) sealed
```

[-] **Parameters**

**item (String)**

case-insensitive source to look for

[-] **Return Value**

true if the source exists in the list, false otherwise

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CopyTo Method (array, arrayIndex)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► SourceList ►  
**CopyTo(String[], Int32)**

C# ▾

Copies the network and URL sources from this list into an array.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void CopyTo(  
    string[] array,  
    int arrayIndex  
)
```

```
Public Sub CopyTo ( _  
    array As String(), _  
    arrayIndex As Integer _  
)
```

```
public:  
virtual void CopyTo(  
    array<String^>^ array,  
    int arrayIndex  
) sealed
```

### Parameters

#### array ( String [] )

destination array to be filled

#### arrayIndex (Int32)

offset into the destination array where copying begins

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Count Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceList ► Count

C# ▾

Gets the number of network and URL sources in the list.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public int Count { get; }
```

```
Public Readonly Property Count As Integer
```

```
public:  
virtual property int Count {  
    int get () sealed;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## DiskPrompt Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceList ► DiskPrompt

C# ▾

Gets or sets the prompt template that is used when prompting the user for installation media.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string DiskPrompt { get; set; }
```

```
Public Property DiskPrompt As String
```

```
public:  
property String^ DiskPrompt {  
    String^ get ();  
    void set (String^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceList ►

## ForceResolution()

C# ▾

Forces the installer to search the source list for a valid source the next time a source is required. For example, when the installer performs an installation or reinstallation, or when it requires the path for a component that is set to run from source.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void ForceResolution()
```

```
Public Sub ForceResolution
```

```
public:  
void ForceResolution()
```

### Remarks

Win32 MSI APIs: [MsiSourceListForceResolution](#),  
[MsiSourceListForceResolutionEx](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetEnumerator Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceList ►

**GetEnumerator()**

C# ▾

Enumerates the network and URL sources in the source list of the patch or product installation.

**Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public IEnumrator<string> GetEnumerator()
```

```
Public Function GetEnumerator As IEnumrator(Of Stri
```

```
public:  
virtual IEnumrator<String^>^ GetEnumerator() sealed
```

**Remarks**

Win32 MSI API: [MsiSourceListEnumSources](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Insert Method (item, index)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceList ► **Insert(String, Int32)**

C# ▾

Adds or reorders a network or URL source for the product or patch.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Insert(  
    string item,  
    int index  
)
```

```
Public Sub Insert ( _  
    item As String, _  
    index As Integer _  
)
```

```
public:  
void Insert(  
    String^ item,  
    int index  
)
```

### Parameters

#### **item (String)**

Path to the source to be added. This parameter is expected to contain only the path without the filename.

#### **index (Int32)**

Specifies the priority order in which the source will be inserted

### Remarks

If this method is called with a new source and *index* is set to 0, the

installer adds the source to the end of the source list.

If this method is called with a source already existing in the source list and *index* is set to 0, the installer retains the source's existing index.

If the method is called with an existing source in the source list and *index* is set to a non-zero value, the source is removed from its current location in the list and inserted at the position specified by *Index*, before any source that already exists at that position.

If the method is called with a new source and *Index* is set to a non-zero value, the source is inserted at the position specified by *index*, before all source that already exists at that position. The index value for all source in the list after the index specified by *Index* are updated to ensure unique index values and the pre-existing order is guaranteed to remain unchanged.

If *index* is greater than the number of sources in the list, the source is placed at the end of the list with an index value one larger than any existing source.

Win32 MSI API: [MsiSourceListAddSourceEx](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation IsReadOnly Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceList ► IsReadOnly

C# ▾

Gets a boolean value indicating whether the list is read-only. A SourceList is never read-only.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool IsReadOnly { get; }
```

```
Public ReadOnly Property IsReadOnly As Boolean
```

```
public:  
virtual property bool IsReadOnly {  
    bool get () sealed;  
}
```

## Value

read-only status of the list

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Item Property (property)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceList ► Item[String]

C#

Gets or sets source list information properties of a product or patch installation.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public string this[  
    string property  
] { get; set; }
```

```
Public Default Property Item ( _  
    property As String _  
) As String
```

```
public:  
property String^ default[String^ property] {  
    String^ get (String^ property);  
    void set (String^ property, String^ value);  
}
```

## Parameters

### *property* (String)

The source list information property name.

## Remarks

Win32 MSI API: [MsiSourceListGetInfo](#)

## Exceptions

Exception	Condition
ArgumentOutOfRangeException	An unknown product, patch, or property

was requested

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## LastUsedSource Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceList ►

### LastUsedSource

C# ▾

Gets or sets the most recently used source location for the product.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string LastUsedSource { get; set; }
```

```
Public Property LastUsedSource As String
```

```
public:  
property String^ LastUsedSource {  
    String^ get ();  
    void set (String^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **LastUsedType** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceList ►

## **LastUsedType**

C# ▾

Gets the type of the last-used source.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public string LastUsedType { get; }
```

```
Public ReadOnly Property LastUsedType As String
```

```
public:  
property String^ LastUsedType {  
    String^ get ();  
}
```

### [-] Remarks

- "n" = network location
- "u" = URL location
- "m" = media location
- (empty string) = no last used source

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## MediaList Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceList ► MediaList

C# ▾

Gets the list of disks registered for the media source of the patch or product installation.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public SourceMediaList MediaList { get; }
```

```
Public ReadOnly Property MediaList As SourceMediaList
```

```
public:  
property SourceMediaList^ MediaList {  
    SourceMediaList^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## MediaPackagePath Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceList ►

### MediaPackagePath

C# ▾

Gets or sets the path relative to the root of the installation media.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string MediaPackagePath { get; set; }
```

```
Public Property MediaPackagePath As String
```

```
public:  
property String^ MediaPackagePath {  
    String^ get ();  
    void set (String^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## PackageName Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceList ►

### PackageName

C# ▾

Gets or sets the name of the Windows Installer package or patch package on the source.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string PackageName { get; set; }
```

Public Property PackageName As String

```
public:  
property String^ PackageName {  
    String^ get ();  
    void set (String^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Remove Method (item)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceList ►

### Remove(String)

C# ▾

Removes a network or URL source.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool Remove(  
    string item  
)
```

```
Public Function Remove ( _  
    item As String _  
) As Boolean
```

```
public:  
virtual bool Remove(  
    String^ item  
) sealed
```

#### Parameters

*item* ([String](#))

#### Remarks

Win32 MSI API: [MsiSourceListClearSource](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **SourceMediaList Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceMediaList

C# ▾

A list of source media for an installed product or patch.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public class SourceMediaList : ICollection<MediaDisk  
    IEnumerable<MediaDisk>, IEnumerable
```

```
Public Class SourceMediaList _  
    Implements ICollection(Of MediaDisk), IEnumerable  
    IEnumerable
```

```
public ref class SourceMediaList : ICollection<MediaDisk>  
    IEnumerable<MediaDisk>, IEnumerable
```

### Members

All Members

Methods

Properties

Public

Instance

Declared

Protected

Static

Inherited

Icon	Member	Description
≡	Add(MediaDisk)	Adds or updates a disk of the media source for the product or patch.
≡	Clear()	Removes all source media from the list.
≡	Contains(Int32)	Checks if the specified media disk id

		exists in the list.
	<a href="#">CopyTo(MediaDisk[], Int32)</a>	Copies the source media info from this list into an array.
	<a href="#">Count</a>	Gets the number of source media in the list.
	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> . (Inherited from <a href="#">Object</a> .)
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection. (Inherited from <a href="#">Object</a> .)
	<a href="#">GetEnumerator()</a>	Enumerates the source media in the source list of the patch or product installation.
	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table. (Inherited from <a href="#">Object</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance (Inherited from <a href="#">Object</a> .)
	<a href="#">IsReadOnly</a>	Gets a boolean value indicating

		whether the list is read-only. A SourceMediaList is never read-only.
💡	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">Remove(Int32)</a>	Removes a specified disk from the set of registered disks.
💡	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)

## [-] Inheritance Hierarchy

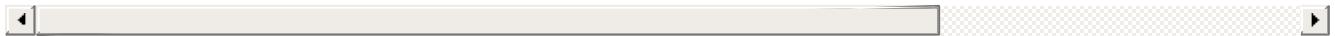
[Object](#)

└ **SourceMediaList**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



## Deployment Tools Foundation **Add Method (item)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceMediaList ►

### Add(MediaDisk)

C#

Adds or updates a disk of the media source for the product or patch.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Add(  
    MediaDisk item  
)
```

```
Public Sub Add ( _  
    item As MediaDisk _  
)
```

```
public:  
virtual void Add(  
    MediaDisk item  
) sealed
```

#### Parameters

*item* ([MediaDisk](#))

#### Remarks

To change the disk prompt only, get the existing volume label from the registry and provide it in this call along with the new disk prompt. Passin a null or empty string for *volumeLabel* registers an empty string as the volume label.

To change the volume label only, get the existing disk prompt that is registered and provide it when calling SourceListAddMediaDisk along with the new volume label. Passing null or an empty string registers an

empty string as the disk prompt.

Win32 MSI API: [MsiSourceListAddMediaDisk](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Clear Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceMediaList ► **Clear()**

C# ▾

Removes all source media from the list.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public void Clear()
```

```
Public Sub Clear
```

```
public:  
virtual void Clear() sealed
```

[-] **Remarks**

Win32 MSI API: [MsiSourceListClearAllEx](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Contains Method (diskId)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceMediaList ►

### Contains(Int32)

C# ▾

Checks if the specified media disk id exists in the list.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public bool Contains(  
    int diskId  
)
```

```
Public Function Contains ( _  
    diskId As Integer _  
) As Boolean
```

```
public:  
bool Contains(  
    int diskId  
)
```

#### Parameters

##### *diskId* (Int32)

disk id of the media to look for

#### Return Value

true if the media exists in the list, false otherwise

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CopyTo Method (array, arrayIndex)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► SourceMediaList ►  
**CopyTo(MediaDisk[], Int32)**

C# ▾

Copies the source media info from this list into an array.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void CopyTo(  
    MediaDisk[] array,  
    int arrayIndex  
)
```

```
Public Sub CopyTo ( _  
    array As MediaDisk(), _  
    arrayIndex As Integer _  
)
```

```
public:  
virtual void CopyTo(  
    array<MediaDisk>^ array,  
    int arrayIndex  
) sealed
```

### Parameters

#### array ( [MediaDisk \[\]](#) )

destination array to be filled

#### arrayIndex ([Int32](#))

offset into the destination array where copying begins

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Count Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceMediaList ► Count

C# ▾

Gets the number of source media in the list.

[-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public int Count { get; }
```

```
Public ReadOnly Property Count As Integer
```

```
public:  
virtual property int Count {  
    int get () sealed;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetEnumerator Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceMediaList ►

### **GetEnumerator()**

C#

Enumerates the source media in the source list of the patch or product installation.

#### **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public IEnum<MediaDisk> GetEnumerator()
```

```
Public Function GetEnumerator As IEnum(Of Medi
```

```
public:  
virtual IEnum<MediaDisk>^ GetEnumerator() seal
```

#### **Remarks**

Win32 MSI API: [MsiSourceListEnumMediaDisks](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



## Deployment Tools Foundation IsReadOnly Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceMediaList ►

### IsReadOnly

C# ▾

Gets a boolean value indicating whether the list is read-only. A SourceMediaList is never read-only.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool IsReadOnly { get; }
```

```
Public ReadOnly Property IsReadOnly As Boolean
```

```
public:  
virtual property bool IsReadOnly {  
    bool get () sealed;  
}
```

#### Value

read-only status of the list

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Remove Method (diskId)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SourceMediaList ►

### Remove(Int32)

C# ▾

Removes a specified disk from the set of registered disks.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool Remove(  
    int diskId  
)
```

```
Public Function Remove ( _  
    diskId As Integer _  
) As Boolean
```

```
public:  
bool Remove(  
    int diskId  
)
```

#### Parameters

##### *diskId* (Int32)

ID of the disk to remove

#### Remarks

Win32 MSI API: [MsiSourceListClearMediaDisk](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **SummaryInfo Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo

C# ▾

Provides access to summary information of a Windows Installer database.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public class SummaryInfo : InstallerHandle
```

```
Public Class SummaryInfo _  
    Inherits InstallerHandle
```

```
public ref class SummaryInfo : public InstallerHandle
```

### Members

All Members	Constructors	Methods	Properties	Events
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static		<input checked="" type="checkbox"/> Declared <input checked="" type="checkbox"/> Inherited

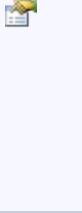
Icon	Member	Description
	<a href="#">SummaryInfo(String, Boolean)</a>	Gets a SummaryInfo object that can be used to examine, update, and add properties to the summary information stream of a package or transform.
	<a href="#">Author</a>	Gets or sets the Author summary information property.

	<a href="#">CharacterCount</a>	Gets or sets the CharacterCount summary information property.
	<a href="#">Close()</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">CodePage</a>	Gets or sets the CodePage summary information property.
	<a href="#">Comments</a>	Gets or sets the Comments summary information property.
	<a href="#">CreateObjRef(Type)</a>	Creates an object that contains all the relevant information required to generate a proxy used to communicate with a remote object.  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">CreateTime</a>	Gets or sets the CreateTime summary information property.
	<a href="#">CreatingApp</a>	Gets or sets the CreatingApp summary information property.
	<a href="#">Dispose()</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .

		(Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">Dispose(Boolean)</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidHandleException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">Equals(Object)</a>	Tests whether this handle object is equal to another handle object. Two handle objects are equal if their types are the same and their native integer handles are the same.  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetHashCode()</a>	Gets a hash value for the handle object.  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">GetLifetimeService()</a>	Retrieves the current lifetime service object that controls the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
	<a href="#">Handle</a>	Gets the native integer handle.

		(Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">InitializeLifetimeService()</a>	Obtains a lifetime service object to control the lifetime policy for this instance.
		(Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">IsClosed</a>	Checks if the handle is closed. When closed, method calls on the handle object may throw an <a href="#">InvalidOperationException</a> .
		(Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">Keywords</a>	Gets or sets the Keywords summary information property.
	<a href="#">LastPrintTime</a>	Gets or sets the LastPrintTime summary information property.
	<a href="#">LastSavedBy</a>	Gets or sets the LastSavedBy summary information property.
	<a href="#">LastSaveTime</a>	Gets or sets the LastSaveTime summary information property.
	<a href="#">MemberwiseClone(Boolean)</a>	Creates a shallow copy of the current <a href="#">MarshalByRefObject</a> object (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .

		(Inherited from <a href="#">Object</a> .)
	<a href="#">PageCount</a>	Gets or sets the PageCount summary information property.
	<a href="#">Persist()</a>	Formats and writes the previously stored properties into the standard summary information stream.
	<a href="#">RevisionNumber</a>	Gets or sets the RevisionNumber summary information property.
	<a href="#">Security</a>	Gets or sets the Security summary information property.
	<a href="#">Subject</a>	Gets or sets the Subject summary information property.
	<a href="#">Template</a>	Gets or sets the Template summary information property.
	<a href="#">Title</a>	Gets or sets the Title summary information property.
	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> . (Inherited from <a href="#">Object</a> .)

	<b>WordCount</b>	Gets or sets the WordCount summary information property.
---	------------------	--

## ❑ Inheritance Hierarchy

Object

  └ MarshalByRefObject

    └ InstallerHandle

      └ **SummaryInfo**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## SummaryInfo Constructor (**packagePath**, **enableWrite**)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo ►

**SummaryInfo(String, Boolean)**

C#

Gets a SummaryInfo object that can be used to examine, update, and add properties to the summary information stream of a package or transform.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public SummaryInfo(  
    string packagePath,  
    bool enableWrite  
)
```

```
Public Sub New ( _  
    packagePath As String, _  
    enableWrite As Boolean _  
)
```

```
public:  
SummaryInfo(  
    String^ packagePath,  
    bool enableWrite  
)
```

### Parameters

#### **packagePath (String)**

Path to the package (database) or transform

#### **enableWrite (Boolean)**

True to reserve resources for writing summary information properties.

## **Remarks**

The SummaryInfo object should be [Close\(\)](#)d after use. It is best that the handle be closed manually as soon as it is no longer needed, as leaving lots of unused handles open can degrade performance.

Win32 MSI API: [MsiGetSummaryInformation](#)

## **Exceptions**

Exception	Condition
<a href="#">FileNotFoundException</a>	the package does not exist or could not be read
<a href="#">InstallerException</a>	the package is an invalid format

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Author Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo ► Author

C# ▾

Gets or sets the Author summary information property.

## [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string Author { get; set; }
```

```
Public Property Author As String
```

```
public:  
property String^ Author {  
    String^ get ();  
    void set (String^ value);  
}
```

## [-] Remarks

The Author summary information property conveys to a file browser the manufacturer of the installation database. This value is typically set from the installer property Manufacturer.

Win32 MSI APIs: [MsiSummaryInfoGetProperty](#),  
[MsiSummaryInfo SetProperty](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **CharacterCount Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo ►

### CharacterCount

C# ▾

Gets or sets the CharacterCount summary information property.

#### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public int CharacterCount { get; set; }
```

```
Public Property CharacterCount As Integer
```

```
public:  
property int CharacterCount {  
    int get ();  
    void set (int value);  
}
```

#### [-] Remarks

The CharacterCount summary information property is only used in transforms. This part of the summary information stream is divided into two 16-bit words. The upper word contains the transform validation flags. The lower word contains the transform error condition flags.

Win32 MSI APIs: [MsiSummaryInfoGetProperty](#),  
[MsiSummaryInfo SetProperty](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **CodePage Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo ► **CodePage**

C# ▾

Gets or sets the CodePage summary information property.

## [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public short CodePage { get; set; }
```

```
Public Property CodePage As Short
```

```
public:  
property short CodePage {  
    short get ();  
    void set (short value);  
}
```

## [-] Remarks

The Codepage summary information property is the numeric value of the ANSI code page used for any strings that are stored in the summary information. Note that this is not the same code page for strings in the installation database. The Codepage summary information property is used to translate the strings in the summary information into Unicode when calling the Unicode API functions. The Codepage summary information property must be set before any string properties are set in the summary information.

Win32 MSI APIs: [MsiSummaryInfoGetProperty](#),  
[MsiSummaryInfo SetProperty](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Comments Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo ► Comments

C# ▾

Gets or sets the Comments summary information property.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public string Comments { get; set; }
```

```
Public Property Comments As String
```

```
public:  
property String^ Comments {  
    String^ get ();  
    void set (String^ value);  
}
```

## Remarks

The Comments summary information property conveys the general purpose of the installer database. By convention, the value for this summary property is set to the following:

"This installer database contains the logic and data required to install <product name>."

where <product name> is the name of the product being installed. In general the value for this summary property only changes in the product name, nothing else.

Win32 MSI APIs: [MsiSummaryInfoGetProperty](#),  
[MsiSummaryInfo SetProperty](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **CreateTime Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo ►

## CreateTime

C# ▾

Gets or sets the CreateTime summary information property.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public DateTime CreateTime { get; set; }
```

```
Public Property CreateTime As DateTime
```

```
public:  
property DateTime CreateTime {  
    DateTime get ();  
    void set (DateTime value);  
}
```

### Remarks

The CreateTime summary information property conveys when the installer database was created.

Win32 MSI APIs: [MsiSummaryInfoGetProperty](#),  
[MsiSummaryInfo SetProperty](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **CreatingApp** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo ►

## CreatingApp

C# ▾

Gets or sets the CreatingApp summary information property.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string CreatingApp { get; set; }
```

```
Public Property CreatingApp As String
```

```
public:  
property String^ CreatingApp {  
    String^ get ();  
    void set (String^ value);  
}
```

### Remarks

The CreatingApp summary information property conveys which application created the installer database. In general the value for this summary property is the name of the software used to author this database.

Win32 MSI APIs: [MsiSummaryInfoGetProperty](#),  
[MsiSummaryInfo SetProperty](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools FoundationKeywords Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo ► **Keywords**

C#

Gets or sets the Keywords summary information property.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public string Keywords { get; set; }
```

```
Public Property Keywords As String
```

```
public:  
property String^ Keywords {  
    String^ get ();  
    void set (String^ value);  
}
```

## Remarks

The Keywords summary information property is used by file browsers to hold keywords that permit the database file to be found in a keyword search. The set of keywords typically includes "Installer" as well as product-specific keywords, and may be localized.

Win32 MSI APIs: [MsiSummaryInfoGetProperty](#),  
[MsiSummaryInfo SetProperty](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **LastPrintTime Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo ►

## **LastPrintTime**

C# ▾

Gets or sets the LastPrintTime summary information property.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public DateTime LastPrintTime { get; set; }
```

```
Public Property LastPrintTime As DateTime
```

```
public:  
property DateTime LastPrintTime {  
    DateTime get ();  
    void set (DateTime value);  
}
```

### [-] Remarks

The LastPrintTime summary information property can be set to the date and time during an administrative installation to record when the administrative image was created. For non-administrative installations this property is the same as the CreateTime summary information property.

Win32 MSI APIs: [MsiSummaryInfoGetProperty](#),  
[MsiSummaryInfo SetProperty](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **LastSavedBy** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo ►

### LastSavedBy

C# ▾

Gets or sets the LastSavedBy summary information property.

#### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string LastSavedBy { get; set; }
```

```
Public Property LastSavedBy As String
```

```
public:  
property String^ LastSavedBy {  
    String^ get ();  
    void set (String^ value);  
}
```

#### [-] Remarks

The installer sets the Last Saved By summary information property to the value of the LogonUser property during an administrative installation. The installer never uses this property and a user never needs to modify it. Developers of a database editing tool may use this property to track the last person to modify the database. This property should be left set to null in a final shipping database.

In a transform, this summary property contains the platform and language ID(s) that a database should have after it has been transformed. The property specifies to what the Template should be set in the new database.

Win32 MSI APIs: [MsiSummaryInfoGetProperty](#),  
[MsiSummaryInfo SetProperty](#)

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)  
Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **LastSaveTime Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo ►

## **LastSaveTime**

C# ▾

Gets or sets the LastSaveTime summary information property.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public DateTime LastSaveTime { get; set; }
```

```
Public Property LastSaveTime As DateTime
```

```
public:  
property DateTime LastSaveTime {  
    DateTime get ();  
    void set (DateTime value);  
}
```

### [-] Remarks

The LastSaveTime summary information property conveys when the last time the installer database was modified. Each time a user changes an installation the value for this summary property is updated to the current system time/date at the time the installer database was saved. Initially the value for this summary property is set to null to indicate that no changes have yet been made.

Win32 MSI APIs: [MsiSummaryInfoGetProperty](#),  
[MsiSummaryInfo SetProperty](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **PageCount Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo ► PageCount

C# ▾

Gets or sets the PageCount summary information property.

## [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public int PageCount { get; set; }
```

```
Public Property PageCount As Integer
```

```
public:  
property int PageCount {  
    int get ();  
    void set (int value);  
}
```

## [-] Remarks

For an installation package, the PageCount summary information property contains the minimum installer version required. For Windows Installer version 1.0, this property must be set to the integer 100. For 64 bit Windows Installer Packages, this property must be set to the integer 200.

For a transform package, the PageCount summary information property contains minimum installer version required to process the transform. Set to the greater of the two PageCount summary information property values belonging to the databases used to generate the transform.

The PageCount summary information property is set to null in patch packages.

This summary property is REQUIRED.

Win32 MSI APIs: [MsiSummaryInfoGetProperty](#),  
[MsiSummaryInfoSetProperty](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Persist Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo ► Persist()

C#

Formats and writes the previously stored properties into the standard summary information stream.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Persist()
```

```
Public Sub Persist
```

```
public:  
void Persist()
```

### - Remarks

This method may only be called once after all the property values have been set. Properties may still be read after the stream is written.

Win32 MSI API: [MsiSummaryInfoPersist](#)

### - Exceptions

Exception	Condition
InstallerException	The stream cannot be successfully written.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **RevisionNumber** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo ►

## RevisionNumber

C#

Gets or sets the RevisionNumber summary information property.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string RevisionNumber { get; set; }
```

```
Public Property RevisionNumber As String
```

```
public:  
property String^ RevisionNumber {  
    String^ get ();  
    void set (String^ value);  
}
```

### Remarks

The Revision Number summary information property contains the package code for the installer package. The package code is a unique identifier of the installer package.

The Revision Number summary information property of a patch package specifies the GUID patch code for the patch. This is followed by a list of patch code GUIDs for obsolete patches that are removed when this patch is applied. The patch codes are concatenated with no delimiters separating GUIDs in the list.

The Revision Number summary information property of a transform package lists the product code GUIDs and version of the new and original products and the upgrade code GUID. The list is separated with semicolons as follows.

Original-Product-Code Original-Product-Version ; New-Product Code  
New-Product-Version; Upgrade-Code

This summary property is REQUIRED.

Win32 MSI APIs: [MsiSummaryInfoGetProperty](#),  
[MsiSummaryInfoSetProperty](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Security Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo ► Security

C# ▾

Gets or sets the Security summary information property.

## [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public int Security { get; set; }
```

```
Public Property Security As Integer
```

```
public:  
property int Security {  
    int get ();  
    void set (int value);  
}
```

## [-] Remarks

The Security summary information property conveys whether the package should be opened as read-only. The database editing tool should not modify a read-only enforced database and should issue a warning at attempts to modify a read-only recommended database. The following values of this property are applicable to Windows Installer files

- 0 - no restriction
- 2 - read only recommended
- 4 - read only enforced

This property should be set to read-only recommended (2) for an installation database and to read-only enforced (4) for a transform or patch.

Win32 MSI APIs: [MsiSummaryInfoGetProperty](#),  
[MsiSummaryInfo SetProperty](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Subject Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo ► Subject

C# ▾

Gets or sets the Subject summary information property.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public string Subject { get; set; }
```

```
Public Property Subject As String
```

```
public:  
property String^ Subject {  
    String^ get ();  
    void set (String^ value);  
}
```

## Remarks

The Subject summary information property conveys to a file browser the product that can be installed using the logic and data in this installer database. For example, the value of the summary property for Microsoft Office 97 would be "Microsoft Office 97 Professional". This value is typically set from the installer property ProductName.

Win32 MSI APIs: [MsiSummaryInfoGetProperty](#),  
[MsiSummaryInfoSetProperty](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Template Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo ► **Template**

C# ▾

Gets or sets the Template summary information property.

## [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public string Template { get; set; }
```

```
Public Property Template As String
```

```
public:  
property String^ Template {  
    String^ get ();  
    void set (String^ value);  
}
```

## [-] Remarks

The Template summary information property indicates the platform and language versions supported by the database.

The syntax of the Template Summary property information is: [platform property][,platform property][,...];[language id][,language id][,...]

For example, the following are all valid values for the Template Summary property:

- Intel;1033
- Intel64;1033
- ;1033
- ;
- Intel ;1033,2046
- Intel64;1033,2046
- Intel;0

If this is a 64-bit Windows Installer, enter Intel64 in the Template summary information property. Note that an installation package cannot have both the Intel and Intel64 properties set.

If the current platform does not match one of the platforms specified the the installer will not process the package. Not specifying a platform implies that the package is platform-independent.

Entering 0 in the language ID field of the Template summary information property, or leaving this field empty, indicates that the package is language neutral.

There are variations of this property depending on whether it is in a source installer database or a transform.

Source Installer Database - Only one language can be specified in a source installer database. Merge Modules are the only packages that may have multiple languages. For more information, see Multiple Language Merge Modules.

Transform - In a transform file, only one language may be specified. The specified platform and language determine whether a transform can be applied to a particular database. The platform property and the language property can be left blank if no transform restriction relies on them to validate the transform.

This summary property is REQUIRED.

Win32 MSI APIs: [MsiSummaryInfoGetProperty](#),  
[MsiSummaryInfo SetProperty](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Title Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo ► Title

C# ▾

Gets or sets the Title summary information property.

## [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string Title { get; set; }
```

```
Public Property Title As String
```

```
public:  
property String^ Title {  
    String^ get ();  
    void set (String^ value);  
}
```

## [-] Remarks

The Title summary information property briefly describes the type of installer package. Phrases such as "Installation Database" or "Transformer" or "Patch" may be used for this property.

Win32 MSI APIs: [MsiSummaryInfoGetProperty](#),  
[MsiSummaryInfo SetProperty](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation WordCount Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► SummaryInfo ►

## WordCount

C# ▾

Gets or sets the WordCount summary information property.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public int WordCount { get; set; }
```

```
Public Property WordCount As Integer
```

```
public:  
property int WordCount {  
    int get ();  
    void set (int value);  
}
```

### Remarks

The WordCount summary information property indicates the type of source file image. If this property is not present, it defaults to 0. Note that this property is stored in place of the standard Count property.

This property is a bit field. New bits may be added in the future. At present the following bits are available:

- Bit 0: 0 = long file names, 1 = short file names
- Bit 1: 0 = source is uncompressed, 1 = source is compressed
- Bit 2: 0 = source is original media, 1 = source is administrative installation
- [MSI 4.0] Bit 3: 0 = elevated privileges can be required to install, 1 : elevated privileges are not required to install

These are combined to give the WordCount summary information

property one of the following values indicating a type of source file image

- 0 - Original source using long file names. Matches tree in Directory table.
- 1 - Original source using short file names. Matches tree in Directory table.
- 2 - Compressed source files using long file names. Matches cabinets and files in the Media table.
- 3 - Compressed source files using short file names. Matches cabinets and files in the Media table.
- 4 - Administrative image using long file names. Matches tree in Directory table.
- 5 - Administrative image using short file names. Matches tree in Directory table.

Note that if the package is marked as compressed (bit 1 is set), the installer only installs files located at the root of the source. In this case, even files marked as uncompressed in the File table must be located at the root to be installed. To specify a source image that has both a cabinet file (compressed files) and uncompressed files that match the tree in the Directory table, mark the package as uncompressed by leaving bit 1 unset (value=0) in the WordCount summary information property and set **Compressed** (value=16384) in the Attributes column of the File table for each file in the cabinet.

For a patch package, the WordCount summary information property specifies the patch engine that was used to create the patch files. The default value is 1 and indicates that MSPATCH was used to create the patch. A value of "2" means that the patch is using smaller, optimized, files available only with Windows Installer version 1.2 or later. A patch with a WordCount of "2" fails immediately if used with a Windows Installer version earlier than 1.2. A patch with a WordCount of "3" fails immediately if used with a Windows Installer version earlier than 2.0.

This summary property is REQUIRED.

Win32 MSI APIs: [MsiSummaryInfoGetProperty](#),  
[MsiSummaryInfo SetProperty](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation TableCollection Class

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► TableCollection

C#

Contains information about all the tables in a Windows Installer database.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public class TableCollection : ICollection<TableInfo>,  
    IEnumerable<TableInfo>, IEnumerable
```

```
Public Class TableCollection  
    Implements ICollection(Of TableInfo), IEnumerable  
    IEnumerable
```

```
public ref class TableCollection : ICollection<TableInfo>,  
    IEnumerable<TableInfo^>, IEnumerable
```

## Members

All Members

Methods

Properties

Public

Instance



Declared

Protected

Static



Inherited

Icon	Member	Description
≡	Add(TableInfo)	Adds a new table to the database.
≡	Clear()	Removes all tables (and all data) from the database.
≡	Contains(String)	Checks if the database contains a table with the given name.

	<a href="#">CopyTo(TableInfo[], Int32)</a>	Copies the table information from this collection into an array.
	<a href="#">Count</a>	Gets the number of tables in the database.
	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetEnumerator()</a>	Enumerates the tables in the database.
	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance  (Inherited from <a href="#">Object</a> .)
	<a href="#">IsReadOnly</a>	Gets a boolean value indicating whether the collection is read-only. A TableCollection is read-only when the database is read-only.

	<a href="#">Item[String]</a>	Gets information about a given table.
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Remove(String)</a>	Removes a table from the database.
	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)

## Inheritance Hierarchy

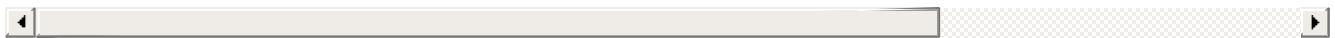
[Object](#)

[TableCollection](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



## Deployment Tools Foundation **Add Method (item)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► TableCollection ►

### Add(TableInfo)

C# ▾

Adds a new table to the database.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Add(  
    TableInfo item  
)
```

```
Public Sub Add ( _  
    item As TableInfo _  
)
```

```
public:  
virtual void Add(  
    TableInfo^ item  
) sealed
```

#### Parameters

##### *item* (TableInfo)

information about the table to be added

#### Exceptions

Exception	Condition
InvalidOperationException	a table with the same name already exists in the database

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

C# ▾

Removes all tables (and all data) from the database.

## - Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Clear()
```

```
Public Sub Clear
```

```
public:  
virtual void Clear() sealed
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Contains Method (item)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► TableCollection ►

### Contains(String)

C# ▾

Checks if the database contains a table with the given name.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool Contains(  
    string item  
)
```

```
Public Function Contains ( _  
    item As String _  
) As Boolean
```

```
public:  
bool Contains(  
    String^ item  
)
```

#### Parameters

##### *item* (String)

case-sensitive name of the table to search for

#### Return Value

True if the table exists, false otherwise.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## CopyTo Method (array, arrayIndex)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► TableCollection ►  
**CopyTo(TableInfo[], Int32)**

C# ▾

Copies the table information from this collection into an array.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void CopyTo(  
    TableInfo[] array,  
    int arrayIndex  
)
```

```
Public Sub CopyTo ( _  
    array As TableInfo(), _  
    arrayIndex As Integer _  
)
```

```
public:  
virtual void CopyTo(  
    array<TableInfo^>^ array,  
    int arrayIndex  
) sealed
```

### Parameters

#### array ( TableInfo [] )

destination array to be filled

#### arrayIndex (Int32)

offset into the destination array where copying begins

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Count Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► TableCollection ► **Count**

C# ▾

Gets the number of tables in the database.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public int Count { get; }
```

```
Public ReadOnly Property Count As Integer
```

```
public:  
virtual property int Count {  
    int get () sealed;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **GetEnumerator Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► TableCollection ►

**GetEnumerator()**

C# ▾

Enumerates the tables in the database.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public IEnum<TableInfo> GetEnumerator()
```

```
Public Function GetEnumerator As IEnum<TableInfo>
```

```
public:  
virtual IEnum<TableInfo> GetEnumerator() sea
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## IsReadOnly Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► TableCollection ►

### IsReadOnly

C# ▾

Gets a boolean value indicating whether the collection is read-only. A TableCollection is read-only when the database is read-only.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool IsReadOnly { get; }
```

```
Public ReadOnly Property IsReadOnly As Boolean
```

```
public:  
virtual property bool IsReadOnly {  
    bool get () sealed;  
}
```

#### Value

read-only status of the collection

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Item Property (table)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► TableCollection ►

## Item[String]

C# ▾

Gets information about a given table.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public TableInfo this[  
    string table  
] { get; }
```

```
Public ReadOnly Default Property Item ( _  
    table As String _  
) As TableInfo
```

```
public:  
property TableInfo^ default[String^ table] {  
    TableInfo^ get (String^ table);  
}
```

### Parameters

#### table (String)

case-sensitive name of the table

### Return Value

information about the requested table, or null if the table does not exist in the database

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Remove Method (item)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► TableCollection ►

### Remove(String)

C# ▾

Removes a table from the database.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public bool Remove(  
    string item  
)
```

```
Public Function Remove ( _  
    item As String _  
) As Boolean
```

```
public:  
bool Remove(  
    String^ item  
)
```

#### Parameters

##### *item* (String)

case-sensitive name of the table to be removed

#### Return Value

true if the table was removed, false if the table did not exist

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **TableInfo Class**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► **TableInfo**

C# ▾

Defines a table in an installation database.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public class TableInfo
```

```
Public Class TableInfo
```

```
public ref class TableInfo
```

## Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon	Member	Description
	<a href="#">TableInfo(String, ICollection&lt;ColumnInfo&gt;, IList&lt;String&gt;)</a>	Creates a table definition.
	<a href="#">Columns</a>	Gets information about the columns in this table.
	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> (Inherited from <a href="#">Object</a> .)
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free

		resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithm and data structures like a hash table  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
📋	<a href="#">Name</a>	Gets the name of the table.
📋	<a href="#">PrimaryKeys</a>	Gets the names of the columns that are primary keys of the table.
📋	<a href="#">SqlCreateString</a>	Gets an SQL CREATE string that can be used to create the table.
📋	<a href="#">SqlInsertString</a>	Gets an SQL INSERT string that can be used insert a new record into the table.

	<code>SqlSelectString</code>	Gets an SQL SELECT string that can be used to select all columns of the table.
≡	<code>ToString()</code>	Gets a string representation of the table.  (Overrides <code>Object.ToString()</code> .)

## ▀ Inheritance Hierarchy

`Object`

  └ `TableInfo`

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## TableInfo Constructor (name, columns, primaryKeys)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► TableInfo ►

**TableInfo(String, ICollection<ColumnInfo>, IList<String>)**

C# ▾

Creates a table definition.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public TableInfo(  
    string name,  
    ICollection<ColumnInfo> columns,  
    IList<string> primaryKeys  
)
```

```
Public Sub New ( _  
    name As String, _  
    columns As ICollection(Of ColumnInfo), _  
    primaryKeys As IList(Of String) _  
)
```

```
public:  
TableInfo(  
    String^ name,  
    ICollection<ColumnInfo^>^ columns,  
    IList<String^>^ primaryKeys  
)
```

### Parameters

#### **name (String)**

Name of the table.

#### **columns (ICollection<ColumnInfo>)**

Columns in the table.

***primaryKeys* ([IList<String>](#))**

The primary keys of the table.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Columns Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► TableInfo ► Columns

C# ▾

Gets information about the columns in this table.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public ColumnCollection Columns { get; }
```

```
Public ReadOnly Property Columns As ColumnCollection
```

```
public:  
property ColumnCollection^ Columns {  
    ColumnCollection^ get ();  
}
```

### - Remarks

This property queries the database every time it is called, to ensure the returned values are up-to-date. For best performance, hold onto the returned collection if using it more than once.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Name Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► TableInfo ► Name

C# ▾

Gets the name of the table.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public string Name { get; }
```

```
Public Readonly Property Name As String
```

```
public:  
property String^ Name {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **PrimaryKeys Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► TableInfo ► PrimaryKeys

C# ▾

Gets the names of the columns that are primary keys of the table.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public IList<string> PrimaryKeys { get; }
```

```
Public ReadOnly Property PrimaryKeys As IList(Of Str
```

```
public:  
property IList<String^>^ PrimaryKeys {  
    IList<String^>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation **SqlCreateString** Property

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► TableInfo ►  
**SqlCreateString**

Gets an SQL CREATE string that can be used to create the table.

C# ▾

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string SqlCreateString { get; }
```

```
Public ReadOnly Property SqlCreateString As String
```

```
public:  
property String^ SqlCreateString {  
    String^ get();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **SqlInsertString** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► TableInfo ► **SqlInsertString**

C#

Gets an SQL INSERT string that can be used insert a new record into the table.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string SqlInsertString { get; }
```

```
Public ReadOnly Property SqlInsertString As String
```

```
public:  
property String^ SqlInsertString {  
    String^ get();  
}
```

### Remarks

The values are expressed as question-mark tokens, to be supplied by the record.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **SqlSelectString** Property

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► TableInfo ►  
**SqlSelectString**

Gets an SQL SELECT string that can be used to select all columns of the table.

C#

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string SqlSelectString { get; }
```

```
Public ReadOnly Property SqlSelectString As String
```

```
public:  
property String^ SqlSelectString {  
    String^ get ();  
}
```

### Remarks

The columns are listed explicitly in the SELECT string, as opposed to using "SELECT \*".

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **ToString** Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► TableInfo ► **ToString()**

C# ▾

Gets a string representation of the table.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public override string ToString()
```

```
Public Overrides Function ToString As String
```

```
public:  
virtual String^ ToString() override
```

### Return Value

The name of the table.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **TextStyles Enumeration**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► **TextStyles**

C# ▾

Defines values for the StyleBits column of the TextStyle table.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
[FlagsAttribute]
public enum TextStyles
```

```
<FlagsAttribute> _
Public Enumeration TextStyles
```

```
[FlagsAttribute]
public enum class TextStyles
```

### Members

Member	Description
<b>Bold</b>	Bold
<b>Italic</b>	Italic
<b>Underline</b>	Underline
<b>Strike</b>	Strike out

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Transaction Class

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Transaction

C# ▾

[MSI 4.5] Handle to a multi-session install transaction.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public class Transaction : InstallerHandle
```

```
Public Class Transaction _  
    Inherits InstallerHandle
```

```
public ref class Transaction : public InstallerHandle
```

## Members

All Members	Constructors	Methods	Properties	Events
<input checked="" type="checkbox"/> Public		<input checked="" type="checkbox"/> Instance		<input checked="" type="checkbox"/> Declared
<input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Static		<input checked="" type="checkbox"/> Inherited

Icon	Member	Description
≡	<a href="#">Transaction(String, TransactionAttributes)</a>	[MSI 4.5] Begins transaction processing of a multi-package installation.
≡	<a href="#">Close()</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
≡	<a href="#">Commit()</a>	Ends the install transaction and commits all changes to the system belonging to the transaction.

	 <a href="#">CreateObjRef(Type)</a>	Creates an object that contains all the relevant information required to generate a proxy used to communicate with a remote object.  (Inherited from <a href="#">MarshalByRefObject</a> .)
	 <a href="#">Dispose()</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
	 <a href="#">Dispose(Boolean)</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
	 <a href="#">Equals(Object)</a>	Tests whether this handle object is equal to another handle object. Two handle objects are equal if their types are the same and their native integer handles are the same.  (Inherited from <a href="#">InstallerHandle</a> .)
	 <a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
 <b>S</b>	<a href="#">FromHandle(IntPtr, Boolean)</a>	Creates a new Transaction object from an integer handle.

	<a href="#">GetHashCode()</a>	Gets a hash value for the handle object.  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">GetLifetimeService()</a>	Retrieves the current lifetime service object that controls the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
	<a href="#">Handle</a>	Gets the native integer handle.  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">InitializeLifetimeService()</a>	Obtains a lifetime service object to control the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">IsClosed</a>	Checks if the handle is closed. When closed, method calls on the handle object may throw an <a href="#">InvalidOperationException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">Join(TransactionAttributes)</a>	Makes the current process the owner of the multi-package installation transaction.
	<a href="#">MemberwiseClone(Boolean)</a>	Creates a shallow copy of the current <a href="#">MarshalByRefObject</a> object

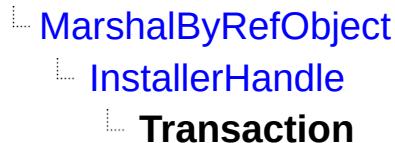
		(Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Name</a>	Gets the name of the transaction.
	<a href="#">OwnerChanged</a>	Notifies listeners when the process that owns the transaction changed.
	<a href="#">Rollback()</a>	Ends the install transaction and undoes changes to the system belonging to the transaction.
	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)

## Remarks

Win32 MSI APIs: [MsiBeginTransaction](#) [MsiJoinTransaction](#)  
[MsiEndTransaction](#)

## Inheritance Hierarchy

[Object](#)



Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Transaction Constructor (name, attributes)

Deployment Tools Foundation Namespaces ►  
 Microsoft.Deployment.WindowsInstaller ► Transaction ►  
**Transaction(String, TransactionAttributes)**

[MSI 4.5] Begins transaction processing of a multi-package installation.

C# ▾

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public Transaction(
    string name,
    TransactionAttributes attributes
)
```

```
Public Sub New ( _
    name As String, _
    attributes As TransactionAttributes _
)
```

```
public:
Transaction(
    String^ name,
    TransactionAttributes attributes
)
```

## Parameters

### ***name* (String)**

Name of the multi-package installation.

### ***attributes* (TransactionAttributes)**

Select optional behavior when beginning the transaction.

## Remarks

Win32 MSI API: [MsiBeginTransaction](#)

## **Exceptions**

<b>Exception</b>	<b>Condition</b>
<a href="#">InstallerException</a>	The transaction could not be initialized.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Commit Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Transaction ► **Commit()**

C#

Ends the install transaction and commits all changes to the system belonging to the transaction.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Commit()
```

```
Public Sub Commit
```

```
public:  
void Commit()
```

### - Remarks

Runs any Commit Custom Actions and commits to the system any changes to Win32 or common language runtime assemblies. Deletes the rollback script, and after using this option, the transaction's changes can no longer be undone with a Rollback Installation.

This method can only be called by the current owner of the transaction.

Win32 MSI API: [MsiEndTransaction](#)

### - Exceptions

Exception	Condition
<a href="#">InstallerException</a>	The transaction could not be committed.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## FromHandle Method (handle, ownsHandle)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► Transaction ►  
**FromHandle(IntPtr, Boolean)**

C# ▾

Creates a new Transaction object from an integer handle.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static Transaction FromHandle(  
    IntPtr handle,  
    bool ownsHandle  
)
```

```
Public Shared Function FromHandle ( _  
    handle As IntPtr, _  
    ownsHandle As Boolean _  
) As Transaction
```

```
public:  
static Transaction^ FromHandle(  
    IntPtr handle,  
    bool ownsHandle  
)
```

### Parameters

#### **handle (IntPtr)**

Integer transaction handle

#### **ownsHandle (Boolean)**

true to close the handle when this object is disposed

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:

Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools FoundationJoin Method (attributes)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Transaction ►

### Join(TransactionAttributes)

C# ▾

Makes the current process the owner of the multi-package installation transaction.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Join(  
    TransactionAttributes attributes  
)
```

```
Public Sub Join ( _  
    attributes As TransactionAttributes _  
)
```

```
public:  
void Join(  
    TransactionAttributes attributes  
)
```

#### Parameters

##### attributes (TransactionAttributes)

Select optional behavior when joining the transaction.

#### Remarks

Win32 MSI API: [MsiJoinTransaction](#)

#### Exceptions

Exception	Condition
<a href="#">InvalidHandleException</a>	The transaction handle is not valid.

[InstallerException](#)

The transaction could not be joined.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Name Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Transaction ► Name

C# ▾

Gets the name of the transaction.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public string Name { get; }
```

```
Public Readonly Property Name As String
```

```
public:  
property String^ Name {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **OwnerChanged Event**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Transaction ►

**OwnerChanged**

C# ▾

Notifies listeners when the process that owns the transaction changed.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public event EventHandler<EventArgs> OwnerChanged
```

```
Public Event OwnerChanged As EventHandler(Of EventArgs)
```

```
public:  
    event EventHandler<EventArgs>^ OwnerChanged {  
        void add (EventHandler<EventArgs>^ value);  
        void remove (EventHandler<EventArgs>^ value);  
    }
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Rollback Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► Transaction ► **Rollback()**

C#

Ends the install transaction and undoes changes to the system belonging to the transaction.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Rollback()
```

```
Public Sub Rollback
```

```
public:  
void Rollback()
```

### [-] Remarks

This method can only be called by the current owner of the transaction.

Win32 MSI API: [MsiEndTransaction](#)

### [-] Exceptions

Exception	Condition
<a href="#">InstallerException</a>	The transaction could not be rolled back.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# TransactionAttributes Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► TransactionAttributes

C# ▾

Attributes for Transaction methods.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
[FlagsAttribute]
public enum TransactionAttributes
```

```
<FlagsAttribute> _
Public Enumeration TransactionAttributes
```

```
[FlagsAttribute]
public enum class TransactionAttributes
```

## Members

Member	Description
<b>None</b>	No attributes.
<b>ChainEmbeddedUI</b>	Request that the Windows Installer not shutdown the embedded UI until the transaction is complete.
<b>JoinExistingEmbeddedUI</b>	Request that the Windows Installer transfer the embedded UI from the original installation.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **TransformErrors Enumeration**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► **TransformErrors**

C# ▾

Transform error conditions available for  
CreateTransformSummaryInfo(Database, String, TransformErrors,  
TransformValidations) or ApplyTransform(String, TransformErrors).

## Declaration Syntax

C#

Visual Basic

Visual C++

```
[FlagsAttribute]
public enum TransformErrors
```

```
<FlagsAttribute> _
Public Enumeration TransformErrors
```

```
[FlagsAttribute]
public enum class TransformErrors
```

## Members

Member	Description
<b>None</b>	No error conditions.
<b>AddExistingRow</b>	Adding a row that already exists.
<b>DelMissingRow</b>	Deleting a row that doesn't exist.
<b>AddExistingTable</b>	Adding a table that already exists.

<b>DelMissingTable</b>	Deleting a table that doesn't exist.
<b>UpdateMissingRow</b>	Updating a row that doesn't exist.
<b>ChangeCodePage</b>	Transform and database code pages do not match and neither code page is neutral.
<b>ViewTransform</b>	Create the temporary _TransformView table when applying the transform.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# TransformValidations

## Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► TransformValidations

C# ▾

Transform validation flags available for  
`CreateTransformSummaryInfo(Database, String, TransformErrors,  
 TransformValidations).`

### Declaration Syntax

C#

Visual Basic

Visual C++

```
[FlagsAttribute]
public enum TransformValidations
```

```
<FlagsAttribute> _
Public Enumeration TransformValidations
```

```
[FlagsAttribute]
public enum class TransformValidations
```

### Members

Member	Description
<b>None</b>	Validate no properties.
<b>Language</b>	Default language must match base database.
<b>Product</b>	Product must match base database.
<b>MajorVersion</b>	Check major version only.

<b>MinorVersion</b>	Check major and minor versions only
<b>UpdateVersion</b>	Check major, minor, and update versions.
<b>NewLessBaseVersion</b>	Installed version < base version.
<b>NewLessEqualBaseVersion</b>	Installed version <= base version.
<b>NewEqualBaseVersion</b>	Installed version = base version.
<b>NewGreaterEqualBaseVersion</b>	Installed version >= base version.
<b>NewGreaterBaseVersion</b>	Installed version > base version.
<b>UpgradeCode</b>	UpgradeCode must match base database.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)  
Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **UpgradeAttributes Enumeration**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► UpgradeAttributes

C# ▾

Defines values for the Attributes column of the Upgrade table.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
[FlagsAttribute]
public enum UpgradeAttributes
```

```
<FlagsAttribute> _
Public Enumeration UpgradeAttributes
```

```
[FlagsAttribute]
public enum class UpgradeAttributes
```

[-] **Members**

Member	Description
<b>MigrateFeatures</b>	Migrates feature states by enabling the logic in the MigrateFeatureStates action.
<b>OnlyDetect</b>	Detects products and applications but does not remove.
<b>IgnoreRemoveFailure</b>	Continues installation upon failure to remove a product or application.
<b>VersionMinInclusive</b>	Detects the range of versions including the value in VersionMin.

<b>VersionMaxInclusive</b>	Detects the range of versions including the value in VersionMax.
<b>LanguagesExclusive</b>	Detects all languages, excluding the languages listed in the Language column

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

C# ▾

Specifies the installation context for [ProductInstallations](#), [PatchInstallations](#), and [DetermineApplicablePatches\(String, String\[\], InapplicablePatchHandler, String, UserContexts\)](#)

## Declaration Syntax

C#

Visual Basic

Visual C++

```
[FlagsAttribute]
public enum UserContexts
```

```
<FlagsAttribute> _
Public Enumeration UserContexts
```

```
[FlagsAttribute]
public enum class UserContexts
```

## Members

Member	Description
<b>None</b>	Not installed.
<b>UserManaged</b>	User managed install context.
<b>UserUnmanaged</b>	User non-managed context.
<b>Machine</b>	Per-machine context.

All	All contexts, or all valid values.
AllUserManaged	All user-managed contexts.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **ValidationError** Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► **ValidationError**

C# ▾

Defines the type of error encountered by the [Validate\(Record\)](#), [ValidateNew\(Record\)](#), or [ValidateFields\(Record\)](#) methods of the [View](#) class.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public enum ValidationError
```

```
Public Enumeration ValidationError
```

```
public enum class ValidationError
```

## Members

Member	Description
<b>None</b>	No error.
<b>DuplicateKey</b>	The new record duplicates primary keys of the existing record in a table.
<b>Required</b>	There are no null values allowed, or the column is about to be deleted but is referenced by another row.
<b>BadLink</b>	The corresponding record in a foreign table was not found.

<b>Overflow</b>	The data is greater than the maximum value allowed.
<b>Underflow</b>	The data is less than the minimum value allowed.
<b>NotInSet</b>	The data is not a member of the values permitted in the set.
<b>BadVersion</b>	An invalid version string was supplied.
<b>BadCase</b>	The case was invalid. The case must be all uppercase or all lowercase.
<b>BadGuid</b>	An invalid GUID was supplied.
<b>BadWildcard</b>	An invalid wildcard file name was supplied, or the use of wildcards was invalid.
<b>BadIdentifier</b>	An invalid identifier was supplied.
<b>BadLanguage</b>	Invalid language IDs were supplied.
<b>BadFileName</b>	An invalid file name was supplied.

<b>BadPath</b>	An invalid path was supplied.
<b>BadCondition</b>	An invalid conditional statement was supplied.
<b>BadFormatted</b>	An invalid format string was supplied.
<b>BadTemplate</b>	An invalid template string was supplied.
<b>BadDefaultDir</b>	An invalid string was supplied in the DefaultDir column of the Directory table.
<b>BadRegPath</b>	An invalid registry path string was supplied.
<b>BadCustomSource</b>	An invalid string was supplied in the CustomSource column of the CustomAction table.
<b>BadProperty</b>	An invalid property string was supplied.
<b>MissingData</b>	The _Validation table is missing a reference to a column.
<b>BadCategory</b>	The category column of the _Validation

	table for the column is invalid.
<b>BadKeyTable</b>	The table in the Keytable column of the _Validation table was not found or loaded
<b>BadMaxMinValues</b>	The value in the MaxValue column of the _Validation table is less than the value in the MinValue column.
<b>BadCabinet</b>	An invalid cabinet name was supplied.
<b>BadShortcut</b>	An invalid shortcut target name was supplied.
<b>StringOverflow</b>	The string is too long for the length specified by the column definition.
<b>BadLocalizeAttrib</b>	An invalid localization attribute was supplied. (Primary keys cannot be localized.)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)  
 Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation ValidationErrorHandler Structure

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ValidationErrorHandler

C# ▾

Contains specific information about an error encountered by the [Validate\(Record\)](#), [ValidateNew\(Record\)](#), or [ValidateFields\(Record\)](#) methods of the [View](#) class.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public struct ValidationErrorHandler
```

Public Structure ValidationErrorHandler

```
public value class ValidationErrorHandler
```

## Members

All Members	Methods	Properties	Events
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static	<input checked="" type="checkbox"/> Declared <input checked="" type="checkbox"/> Inherited

Icon	Member	Description
	<a href="#">Column</a>	Gets the column containing the error, or null if the error applies to the whole row.
	<a href="#">Equals(Object)</a>	Indicates whether this instance and a specified object are equal. (Inherited from <a href="#">ValueType</a> .)
	<a href="#">Error</a>	Gets the type of validation error encountered.

💡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">GetHashCode()</a>	Returns the hash code for this instance.  (Inherited from <a href="#">ValueType</a> .)
💡	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">ToString()</a>	Returns the fully qualified type name of this instance.  (Inherited from <a href="#">ValueType</a> .)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Column Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ValidationErrorsInfo ►

**Column**

C# ▾

Gets the column containing the error, or null if the error applies to the whole row.

Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string Column { get; }
```

```
Public ReadOnly Property Column As String
```

```
public:  
property String^ Column {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Error Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ValidationCreateInfo ► Error

C# ▾

Gets the type of validation error encountered.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public ValidationError Error { get; }
```

```
Public Readonly Property Error As ValidationError
```

```
public:  
property ValidationError Error {  
    ValidationError get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation View Class

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► View

C# ▾

A View represents a result set obtained when processing a query using the [OpenView\(String, Object\[\]\)](#) method of a [Database](#). Before any data can be transferred, the query must be executed using the [Execute\(Record\)](#) method, passing to it all replaceable parameters designated within the SQL query string.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public class View : InstallerHandle, IEnumerable<Record>, IEnumberable
```

```
Public Class View
    Inherits InstallerHandle
    Implements IEnumerable(Of Record), IEnumberab
```

```
public ref class View : public InstallerHandle,
    IEnumerable<Record^>, IEnumberable
```

### Members

All Members

Methods

Properties

Public  
 Protected

Instance  
 Static

Declares  
 Inherited

Icon	Member	Description
	<a href="#">Assign(Record)</a>	Updates or inserts a Record into the View.
	<a href="#">Close()</a>	Closes the handle. After closing a

		handle, further method calls may throw an <a href="#">InvalidOperationException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">Columns</a>	Gets the set of columns that were included in the query for this View, or null if this view is not a SELECT query.
	<a href="#">CreateObjRef(Type)</a>	Creates an object that contains all the relevant information required to generate a proxy used to communicate with a remote object.  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">Database</a>	Gets the Database on which this View was opened.
	<a href="#">Delete(Record)</a>	Deletes a Record from the View.
	<a href="#">Dispose()</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">Dispose(Boolean)</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">Equals(Object)</a>	Tests whether this handle object is equal to another handle object. Two

		handle objects are equal if their types are the same and their native integer handles are the same.  (Inherited from <a href="#">InstallerHandle</a> .)
≡	<a href="#">Execute(Record)</a>	Executes a SQL View query and supplies any required parameters. The query uses the question mark token to represent parameters as described in SQL Syntax. The values of these parameters are passed in as the corresponding fields of a parameter record.
≡	<a href="#">Execute()</a>	Executes a SQL View query.
≡	<a href="#">Fetch()</a>	Fetches the next sequential record from the view, or null if there are no more records.
≡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">GetEnumerator()</a>	Enumerates over the Records retrieved by the View.
≡	<a href="#">GetHashCode()</a>	Gets a hash value for the handle object.

		(Inherited from <a href="#">InstallerHandle</a> .)
≡	<a href="#">GetLifetimeService()</a>	<p>Retrieves the current lifetime service object that controls the lifetime policy for this instance.</p> <p>(Inherited from <a href="#">MarshalByRefObject</a>.)</p>
≡	<a href="#">GetType()</a>	<p>Gets the <a href="#">Type</a> of the current instance.</p> <p>(Inherited from <a href="#">Object</a>.)</p>
⊕	<a href="#">Handle</a>	<p>Gets the native integer handle.</p> <p>(Inherited from <a href="#">InstallerHandle</a>.)</p>
≡	<a href="#">InitializeLifetimeService()</a>	<p>Obtains a lifetime service object to control the lifetime policy for this instance.</p> <p>(Inherited from <a href="#">MarshalByRefObject</a>.)</p>
≡	<a href="#">Insert(Record)</a>	Inserts a Record into the view.
≡	<a href="#">InsertTemporary(Record)</a>	Inserts a Record into the View. The inserted data is not persistent.
⊕	<a href="#">IsClosed</a>	<p>Checks if the handle is closed. When closed, method calls on the handle object may throw an <a href="#">InvalidOperationException</a>.</p> <p>(Inherited from <a href="#">InstallerHandle</a>.)</p>
💡	<a href="#">MemberwiseClone(Boolean)</a>	Creates a shallow copy of the current <a href="#">MarshalByRefObject</a> object

		(Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Merge(Record)</a>	Inserts or validates a record.
	<a href="#">Modify(ViewModifyMode, Record)</a>	Updates a fetched Record.
	<a href="#">QueryString</a>	Gets the SQL query string used to open this View.
	<a href="#">Refresh(Record)</a>	Refreshes the data in a Record.
	<a href="#">Replace(Record)</a>	Updates or deletes and inserts a Record into the View.
	<a href="#">Seek(Record)</a>	Refreshes the information in the supplied record without changing the position in the result set and without affecting subsequent fetch operations.
	<a href="#">Tables</a>	Gets the set of tables that were included in the SQL query for this View.
		

	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
=	<a href="#">Update(Record)</a>	Updates the View with new data from the Record.
=	<a href="#">Validate(Record)</a>	Validates a record, returning information about any errors.
=	<a href="#">ValidateDelete(Record)</a>	Validates a record that will be deleted later, returning information about any errors.
=	<a href="#">ValidateFields(Record)</a>	Validates fields of a fetched or new record, returning information about any errors. Can validate one or more fields of an incomplete record.
=	<a href="#">ValidateNew(Record)</a>	Validates a new record, returning information about any errors.

## [-] Inheritance Hierarchy

Object

- └ MarshalByRefObject
- └ InstallerHandle
- └ **View**

Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)



## Deployment Tools Foundation **Assign Method (record)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► View ► Assign(Record)

C# ▾

Updates or inserts a Record into the View.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Assign(  
    Record record  
)
```

```
Public Sub Assign ( _  
    record As Record _  
)
```

```
public:  
void Assign(  
    Record^ record  
)
```

### Parameters

#### *record* ([Record](#))

the Record to be assigned

### Remarks

Updates record if the primary keys match an existing row and inserts if they do not match. Fails with a read-only database. This method cannot be used with a View containing joins.

See [Modify\(ViewModifyMode, Record\)](#) for more remarks.

Win32 MSI API: [MsiViewModify](#)

## Exceptions

Exception	Condition
InstallerException	the assignment failed
InvalidHandleException	the View handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Columns Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► View ► Columns

C#

Gets the set of columns that were included in the query for this View, or null if this view is not a SELECT query.

## [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public ColumnCollection Columns { get; }
```

```
Public ReadOnly Property Columns As ColumnCollection
```

```
public:  
property ColumnCollection^ Columns {  
    ColumnCollection^ get ();  
}
```

## [-] Remarks

Win32 MSI API: [MsiViewGetColumnInfo](#)

## [-] Exceptions

Exception	Condition
InstallerException	the View is not in an active state
InvalidHandleException	the View handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Database Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► View ► Database

C# ▾

Gets the Database on which this View was opened.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public Database Database { get; }
```

```
Public Readonly Property Database As Database
```

```
public:  
property Database^ Database {  
    Database^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Delete Method (record)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► View ► Delete(Record)

C# ▾

Deletes a Record from the View.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Delete(  
    Record record  
)
```

```
Public Sub Delete ( _  
    record As Record _  
)
```

```
public:  
void Delete(  
    Record^ record  
)
```

### Parameters

#### record (Record)

the Record to be deleted

### Remarks

The Record must have been obtained by calling [Fetch\(\)](#). Fails if the row has been deleted. Works only with read-write records. This method cannot be used with a View containing joins.

See [Modify\(ViewModifyMode, Record\)](#) for more remarks.

Win32 MSI API: [MsiViewModify](#)

## Exceptions

Exception	Condition
InstallerException	the deletion failed
InvalidHandleException	the View handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Execute Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► View ► Execute()

C# ▾

## Members

Icon	Member	Description
	<a href="#">Execute(Record)</a>	Executes a SQL View query and supplies any required parameters. The query uses the question mark token to represent parameters as described in SQL Syntax. The values of these parameters are passed in as the corresponding fields of a parameter record.
	<a href="#">Execute()</a>	Executes a SQL View query.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Execute Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► View ► Execute()

C# ▾

Executes a SQL View query.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Execute()
```

```
Public Sub Execute
```

```
public:  
void Execute()
```

### Remarks

Win32 MSI API: [MsiViewExecute](#)

### Exceptions

Exception	Condition
<a href="#">InstallerException</a>	the View could not be executed
<a href="#">InvalidHandleException</a>	the View handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Execute Method (executeParams)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► View ► Execute(Record)

C# ▾

Executes a SQL View query and supplies any required parameters. The query uses the question mark token to represent parameters as described in SQL Syntax. The values of these parameters are passed in as the corresponding fields of a parameter record.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Execute(  
    Record executeParams  
)
```

```
Public Sub Execute ( _  
    executeParams As Record _  
)
```

```
public:  
void Execute(  
    Record^ executeParams  
)
```

### Parameters

#### executeParams (Record)

Optional Record that supplies the parameters. This Record contains values to replace the parameter tokens in the SQL query.

### Remarks

Win32 MSI API: [MsiViewExecute](#)

### Exceptions

Exception

Condition

InstallerException	the View could not be executed
InvalidHandleException	the View handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Fetch Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► View ► Fetch()

C#

Fetches the next sequential record from the view, or null if there are no more records.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public Record Fetch()
```

```
Public Function Fetch As Record
```

```
public:  
Record^ Fetch()
```

### - Remarks

The Record object should be [Close\(\)](#)d after use. It is best that the handle be closed manually as soon as it is no longer needed, as leaving lots of unused handles open can degrade performance.

Win32 MSI API: [MsiViewFetch](#)

### - Exceptions

Exception	Condition
<a href="#">InstallerException</a>	the View is not in an active state
<a href="#">InvalidHandleException</a>	the View handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetEnumerator Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► View ► **GetEnumerator()**

C# ▾

Enumerates over the Records retrieved by the View.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public IEnum<Record> GetEnumerator()
```

```
Public Function GetEnumerator As IEnum<Record>
```

```
public:  
virtual IEnum<Record^>^ GetEnumerator() sealed
```

### - Return Value

An enumerator of Record objects.

### - Remarks

Each Record object should be [Close\(\)](#)d after use. It is best that the handle be closed manually as soon as it is no longer needed, as leaving lots of unused handles open can degrade performance. However, note that it is not necessary to complete the enumeration just for the purpose of closing handles, because Records are fetched lazily on each step of the enumeration.

Win32 MSI API: [MsiViewFetch](#)

### - Exceptions

Exception	Condition
<a href="#">InstallerException</a>	The View was not <a href="#">Execute(Record)</a> d before attempting the enumeration.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Insert Method (record)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► View ► Insert(Record)

C# ▾

Inserts a Record into the view.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Insert(  
    Record record  
)
```

```
Public Sub Insert ( _  
    record As Record _  
)
```

```
public:  
void Insert(  
    Record^ record  
)
```

### [-] Parameters

#### *record* ([Record](#))

the Record to be inserted

### [-] Remarks

Fails if a row with the same primary keys exists. Fails with a read-only database. This method cannot be used with a View containing joins.

See [Modify\(ViewModifyMode, Record\)](#) for more remarks.

Win32 MSI API: [MsiViewModify](#)

### [-] Exceptions

Exception	Condition
InstallerException	the insertion failed
InvalidHandleException	the View handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **InsertTemporary Method (record)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► View ►  
**InsertTemporary(Record)**

C#

Inserts a Record into the View. The inserted data is not persistent.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public void InsertTemporary(  
    Record record  
)
```

```
Public Sub InsertTemporary ( _  
    record As Record _  
)
```

```
public:  
void InsertTemporary(  
    Record^ record  
)
```

[-] **Parameters**

**record (Record)**

the Record to be inserted

[-] **Remarks**

Fails if a row with the same primary key exists. Works only with read-write records. This method cannot be used with a View containing joins.

See [Modify\(ViewModifyMode, Record\)](#) for more remarks.

Win32 MSI API: [MsiViewModify](#)

## Exceptions

Exception	Condition
InstallerException	the insertion failed
InvalidHandleException	the View handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Merge Method (record)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► View ► Merge(Record)

C# ▾

Inserts or validates a record.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public bool Merge(  
                  Record record  
)
```

```
Public Function Merge ( _  
                      record As Record _  
) As Boolean
```

```
public:  
bool Merge(  
          Record^ record  
)
```

### - Parameters

#### *record* (Record)

the Record to be merged

### - Return Value

true if the record was inserted or validated, false if there is an existing record with the same primary keys that is not identical

### - Remarks

Works only with read-write records. This method cannot be used with a View containing joins.

See [Modify\(ViewModifyMode, Record\)](#) for more remarks.

Win32 MSI API: [MsiViewModify](#)

## [-] Exceptions

Exception	Condition
<a href="#">InstallerException</a>	the merge failed (for a reason other than invalid data)
<a href="#">InvalidHandleException</a>	the View handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Modify Method (mode, record)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► View ►  
**Modify(ViewModifyMode, Record)**

C# ▾

Updates a fetched Record.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Modify(  
    ViewModifyMode mode,  
    Record record  
)
```

```
Public Sub Modify ( _  
    mode As ViewModifyMode, _  
    record As Record _  
)
```

```
public:  
void Modify(  
    ViewModifyMode mode,  
    Record^ record  
)
```

### Parameters

#### **mode (ViewModifyMode)**

specifies the modify mode

#### **record (Record)**

the Record to modify

### Remarks

You can update or delete a record immediately after inserting, or seekin provided you have NOT modified the 0th field of the inserted or sought

record.

To execute any SQL statement, a View must be created. However, a View that does not create a result set, such as CREATE TABLE, or INSERT INTO, cannot be used with any of the Modify methods to update tables through the view.

You cannot fetch a record containing binary data from one database and then use that record to insert the data into another database. To move binary data from one database to another, you should export the data to a file and then import it into the new database using a query and the [SetStream\(Int32, String\)](#). This ensures that each database has its own copy of the binary data.

Note that custom actions can only add, modify, or remove temporary rows, columns, or tables from a database. Custom actions cannot modify persistent data in a database, such as data that is a part of the database stored on disk.

Win32 MSI API: [MsiViewModify](#)

## Exceptions

Exception	Condition
<a href="#">InstallerException</a>	the modification failed, or a validation was requested and the data did not pass
<a href="#">InvalidHandleException</a>	the View handle is invalid

## See Also

[Refresh\(Record\)](#)  
[Insert\(Record\)](#)  
[Update\(Record\)](#)  
[Assign\(Record\)](#)  
[Replace\(Record\)](#)  
[Delete\(Record\)](#)  
[InsertTemporary\(Record\)](#)  
[Seek\(Record\)](#)  
[Merge\(Record\)](#)

[Validate\(Record\)](#)  
[ValidateNew\(Record\)](#)  
[ValidateFields\(Record\)](#)  
[ValidateDelete\(Record\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module:  
Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# QueryString Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► View ► QueryString

C# ▾

Gets the SQL query string used to open this View.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string QueryString { get; }
```

```
Public Readonly Property QueryString As String
```

```
public:  
property String^ QueryString {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Refresh Method (record)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► View ► Refresh(Record)

C# ▾

Refreshes the data in a Record.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Refresh(  
    Record record  
)
```

```
Public Sub Refresh ( _  
    record As Record _  
)
```

```
public:  
void Refresh(  
    Record^ record  
)
```

## Parameters

### record (Record)

the Record to be refreshed

## Remarks

The Record must have been obtained by calling [Fetch\(\)](#). Fails with a deleted Record. Works only with read-write Records.

See [Modify\(ViewModifyMode, Record\)](#) for more remarks.

Win32 MSI API: [MsiViewModify](#)

## Exceptions

Exception	Condition
InstallerException	the refresh failed
InvalidHandleException	the View handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Replace Method (record)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► View ► Replace(Record)

C# ▾

Updates or deletes and inserts a Record into the View.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Replace(  
    Record record  
)
```

```
Public Sub Replace ( _  
    record As Record _  
)
```

```
public:  
void Replace(  
    Record^ record  
)
```

## Parameters

### record (Record)

the Record to be replaced

## Remarks

The Record must have been obtained by calling [Fetch\(\)](#). Updates record if the primary keys are unchanged. Deletes old row and inserts new if primary keys have changed. Fails with a read-only database. This method cannot be used with a View containing joins.

See [Modify\(ViewModifyMode, Record\)](#) for more remarks.

Win32 MSI API: [MsiViewModify](#)

## Exceptions

Exception	Condition
InstallerException	the replacement failed
InvalidHandleException	the View handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Seek Method (record)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► View ► Seek(Record)

C# ▾

Refreshes the information in the supplied record without changing the position in the result set and without affecting subsequent fetch operations.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool Seek(  
    Record record  
)
```

```
Public Function Seek ( _  
    record As Record _  
) As Boolean
```

```
public:  
bool Seek(  
    Record^ record  
)
```

### Parameters

#### record (Record)

the Record to be filled with the result of the seek

### Remarks

After seeking, the Record may then be used for subsequent Update, Delete, and Refresh operations. All primary key columns of the table must be in the query and the Record must have at least as many fields as the query. Seek cannot be used with multi-table queries. This method cannot be used with a View containing joins.

See [Modify\(ViewModifyMode, Record\)](#) for more remarks.

Win32 MSI API: [MsiViewModify](#)

## [-] Exceptions

Exception	Condition
<a href="#">InstallerException</a>	the seek failed
<a href="#">InvalidHandleException</a>	the View handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Tables Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► View ► Tables

C# ▾

Gets the set of tables that were included in the SQL query for this View.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public IList<TableInfo> Tables { get; }
```

```
Public ReadOnly Property Tables As IList(Of TableInfo)
```

```
public:  
property IList<TableInfo^>^ Tables {  
    IList<TableInfo^>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Update Method (record)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► View ► Update(Record)

C# ▾

Updates the View with new data from the Record.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Update(  
    Record record  
)
```

```
Public Sub Update ( _  
    record As Record _  
)
```

```
public:  
void Update(  
    Record^ record  
)
```

## Parameters

### *record* ([Record](#))

the new data

## Remarks

Only non-primary keys can be updated. The Record must have been obtained by calling [Fetch\(\)](#). Fails with a deleted Record. Works only with read-write Records.

See [Modify\(ViewModifyMode, Record\)](#) for more remarks.

Win32 MSI API: [MsiViewModify](#)

## Exceptions

Exception	Condition
InstallerException	the update failed
InvalidHandleException	the View handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Validate Method (record)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► View ► Validate(Record)

C# ▾

Validates a record, returning information about any errors.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public ICollection<ValidationErrorResponse> Validate(  
    Record record  
)
```

```
Public Function Validate ( _  
    record As Record _  
) As ICollection(Of ValidationErrorResponse)
```

```
public:  
ICollection<ValidationErrorResponse>^ Validate(  
    Record^ record  
)
```

## Parameters

### record (Record)

the Record to be validated

## Return Value

null if the record was validated; if there is an existing record with the same primary keys that has conflicting data then error information is returned

## Remarks

The Record must have been obtained by calling [Fetch\(\)](#). Works with read-write and read-only records. This method cannot be used with a View containing joins.

See [Modify\(ViewModifyMode, Record\)](#) for more remarks.

Win32 MSI APIs: [MsiViewModify](#), [MsiViewGetError](#)

## [-] Exceptions

Exception	Condition
<a href="#">InstallerException</a>	the validation failed (for a reason other than invalid data)
<a href="#">InvalidHandleException</a>	the View handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **ValidateDelete Method (record)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► View ►  
**ValidateDelete(Record)**

C# ▾

Validates a record that will be deleted later, returning information about any errors.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public ICollection<ValidationCreateInfo> ValidateDelete(
    Record record
)
```

```
Public Function ValidateDelete ( _
    record As Record _ 
) As ICollection(Of ValidationCreateInfo)
```

```
public:
ICollection<ValidationCreateInfo>^ ValidateDelete(
    Record^ record
)
```

### Parameters

#### **record (Record)**

the Record to be validated

### Return Value

null if the record is safe to delete; if another row refers to the primary keys of this row then error information is returned

### Remarks

Validation does not check for the existence of the primary keys of this record in properties or strings. Does not check if a column is a foreign key to

multiple tables. Works with read-write and read-only records. This method cannot be used with a View containing joins.

See [Modify\(ViewModifyMode, Record\)](#) for more remarks.

Win32 MSI APIs: [MsiViewModify](#), [MsiViewGetError](#)

## **Exceptions**

Exception	Condition
<a href="#">InstallerException</a>	the validation failed (for a reason other than invalid data)
<a href="#">InvalidHandleException</a>	the View handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **ValidateFields Method (record)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► View ►  
**ValidateFields(Record)**

C# ▾

Validates fields of a fetched or new record, returning information about any errors. Can validate one or more fields of an incomplete record.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public ICollection<ValidationErrorResponse> ValidateFields(
    Record record
)
```

```
Public Function ValidateFields ( _
    record As Record _
) As ICollection(Of ValidationErrorResponse)
```

```
public:
ICollection<ValidationErrorResponse>^ ValidateFields(
    Record^ record
)
```

### Parameters

#### **record (Record)**

the Record to be validated

### Return Value

null if the record was validated; if there is an existing record with the same primary keys that has conflicting data then error information is returned

### Remarks

Works with read-write and read-only records. This method cannot be

used with a View containing joins.

See [Modify\(ViewModifyMode, Record\)](#) for more remarks.

Win32 MSI APIs: [MsiViewModify](#), [MsiViewGetError](#)

## [-] Exceptions

Exception	Condition
<a href="#">InstallerException</a>	the validation failed (for a reason other than invalid data)
<a href="#">InvalidHandleException</a>	the View handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation ValidateNew Method (record)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller ► View ►  
**ValidateNew(Record)**

Validates a new record, returning information about any errors.

C# ▾

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public ICollection<ValidationErrorResponse> ValidateNew(  
    Record record  
)
```

```
Public Function ValidateNew ( _  
    record As Record _  
) As ICollection(Of ValidationErrorResponse)
```

```
public:  
ICollection<ValidationErrorResponse>^ ValidateNew(  
    Record^ record  
)
```

### Parameters

#### *record* (**Record**)

the Record to be validated

### Return Value

null if the record was validated; if there is an existing record with the same primary keys then error information is returned

### Remarks

Checks for duplicate keys. The Record must have been obtained by calling [Fetch\(\)](#). Works with read-write and read-only records. This method cannot be used with a View containing joins.

See [Modify\(ViewModifyMode, Record\)](#) for more remarks.

Win32 MSI APIs: [MsiViewModify](#), [MsiViewGetError](#)

## [-] Exceptions

Exception	Condition
<a href="#">InstallerException</a>	the validation failed (for a reason other than invalid data)
<a href="#">InvalidHandleException</a>	the View handle is invalid

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation ViewModifyMode Enumeration

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller ► ViewModifyMode

C# ▾

Specifies the modify mode for [Modify\(ViewModifyMode, Record\)](#).

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public enum ViewModifyMode
```

```
Public Enumeration ViewModifyMode
```

```
public enum class ViewModifyMode
```

## Members

Member	Description
<b>Seek</b>	Refreshes the information in the supplied record without changing the position in the result set and without affecting subsequent fetch operations.
<b>Refresh</b>	Refreshes the data in a Record.
<b>Insert</b>	Inserts a Record into the view.
<b>Update</b>	Updates the View with new data from the Record.
<b>Assign</b>	Updates or inserts a Record into the View

<b>Replace</b>	Updates or deletes and inserts a Record into the View.
<b>Merge</b>	Inserts or validates a record.
<b>Delete</b>	Deletes a Record from the View.
<b>InsertTemporary</b>	Inserts a Record into the View. The inserted data is not persistent.
<b>Validate</b>	Validates a record.
<b>ValidateNew</b>	Validates a new record.
<b>ValidateField</b>	Validates fields of a fetched or new record. Can validate one or more fields of an incomplete record.
<b>ValidateDelete</b>	Validates a record that will be deleted later.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)  
Assembly: Microsoft.Deployment.WindowsInstaller (Module: Microsoft.Deployment.WindowsInstaller) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools

# Foundation **Microsoft.Deployment.WindowsInstaller.Linq** Namespace

Deployment Tools Foundation Namespaces ►  
**Microsoft.Deployment.WindowsInstaller.Linq**

C# ▾

LINQ extensions for querying Windows Installer databases  
(experimental).

## Declaration Syntax

C#

Visual Basic

Visual C++

```
namespace Microsoft.Deployment.WindowsInstaller.Linq
```

```
Namespace Microsoft.Deployment.WindowsInstaller.Linq
```

```
namespace Microsoft.Deployment.WindowsInstaller.Linq
```

## Types

All Types

Classes

Icon	Type	Description
	<b>DatabaseColumnAttribute</b>	Apply to a property on a subclass of QRecord to indicate the name of the column the property is to be associated with.
	<b>DatabaseTableAttribute</b>	Apply to a subclass of QRecord to indicate the name of the table the record type is to be used with.
	<b>QDatabase</b>	Queryable MSI database - extends the base Database class with LINQ

		query functionality along with predefined entity types for common tables.
	<b>QRecord</b>	Generic record entity for queryable databases, and base for strongly-typed entity subclasses.
	<b>QTable&lt;TRecord&gt;</b>	Represents one table in a LINQ-queryable Database.
	<b>Queryable</b>	Allows any Database instance to be converted into a queryable database

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

## Deployment Tools Foundation DatabaseColumnAttribute Class

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ►

### DatabaseColumnAttribute

C# ▾

Apply to a property on a subclass of QRecord to indicate the name of the column the property is to be associated with.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public class DatabaseColumnAttribute : Attribute
```

```
Public Class DatabaseColumnAttribute _  
    Inherits Attribute
```

```
public ref class DatabaseColumnAttribute : public At
```

#### Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon	Member	Description
	<a href="#">DatabaseColumnAttribute(String)</a>	Creates a new DatabaseColumnAttribute which maps a record property to a column.
	<a href="#">Column</a>	Gets or sets the column associated with the record property.

≡	<a href="#">Equals(Object)</a>	Returns a value that indicates whether this instance is equal to a specified object.  (Inherited from <a href="#">Attribute</a> .)
💡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">GetHashCode()</a>	Returns the hash code for this instance.  (Inherited from <a href="#">Attribute</a> .)
≡	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">IsDefaultAttribute()</a>	When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class.  (Inherited from <a href="#">Attribute</a> .)
≡	<a href="#">Match(Object)</a>	When overridden in a derived class, returns a value that indicates whether this instance equals a specified object.  (Inherited from <a href="#">Attribute</a> .)
💡	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .

		(Inherited from <a href="#">Object</a> .)
≡	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> . (Inherited from <a href="#">Object</a> .)
grid	<a href="#">TypeId</a>	When implemented in a derive class, gets a unique identifier for this <a href="#">Attribute</a> . (Inherited from <a href="#">Attribute</a> .)

## ■ Remarks

If this attribute is not used on a property, the default column name will be the same as the property name.

## ■ Inheritance Hierarchy

[Object](#)

└ [Attribute](#)

└ **DatabaseColumnAttribute**

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)



# DatabaseColumnAttribute

## Constructor (column)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ►

DatabaseColumnAttribute ► DatabaseColumnAttribute(String)

C# ▾

Creates a new DatabaseColumnAttribute which maps a record property to a column.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public DatabaseColumnAttribute(  
    string column  
)
```

```
Public Sub New ( _  
    column As String _  
)
```

```
public:  
DatabaseColumnAttribute(  
    String^ column  
)
```

### Parameters

#### column (String)

name of the column associated with the property

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Column Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ►

DatabaseColumnAttribute ► **Column**

C# ▾

Gets or sets the column associated with the record property.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public string Column { get; set; }
```

```
Public Property Column As String
```

```
public:  
property String^ Column {  
    String^ get ();  
    void set (String^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

## Deployment Tools Foundation DatabaseTableAttribute Class

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ►

### DatabaseTableAttribute

C#

Apply to a subclass of QRecord to indicate the name of the table the record type is to be used with.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public class DatabaseTableAttribute : Attribute
```

```
Public Class DatabaseTableAttribute _  
    Inherits Attribute
```

```
public ref class DatabaseTableAttribute : public Att
```

#### Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon	Member	Description
	<a href="#">DatabaseTableAttribute(String)</a>	Creates a new DatabaseTableAttribute for the specified table.
	<a href="#">Equals(Object)</a>	Returns a value that indicates whether this instance is equal to specified object.  (Inherited from <a href="#">Attribute</a> .)

	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetHashCode()</a>	Returns the hash code for this instance.  (Inherited from <a href="#">Attribute</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
	<a href="#">IsDefaultAttribute()</a>	When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class.  (Inherited from <a href="#">Attribute</a> .)
	<a href="#">Match(Object)</a>	When overridden in a derived class, returns a value that indicates whether this instance equals a specified object.  (Inherited from <a href="#">Attribute</a> .)
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Table</a>	Gets or sets the table associated with the record type.
	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents

		the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<b>TypeId</b>	When implemented in a derived class, gets a unique identifier for this <a href="#">Attribute</a> .  (Inherited from <a href="#">Attribute</a> .)

## [-] Remarks

If this attribute is not used on a record type, the default table name will be derived from the record type name. (An optional underscore suffix is stripped.)

## [-] Inheritance Hierarchy

[Object](#)

└ [Attribute](#)

└ **DatabaseTableAttribute**

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## DatabaseTableAttribute Constructor (table)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► DatabaseTableAttribute

► DatabaseTableAttribute(String)

C# ▾

Creates a new DatabaseTableAttribute for the specified table.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public DatabaseTableAttribute(  
    string table  
)
```

```
Public Sub New ( _  
    table As String _  
)
```

```
public:  
DatabaseTableAttribute(  
    String^ table  
)
```

### Parameters

#### *table* (String)

name of the table associated with the record type

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Table Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► DatabaseTableAttribute

## ► Table

C# ▾

Gets or sets the table associated with the record type.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public string Table { get; set; }
```

```
Public Property Table As String
```

```
public:  
property String^ Table {  
    String^ get ();  
    void set (String^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

## Deployment Tools Foundation **QDatabase Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase

C# ▾

Queryable MSI database - extends the base Database class with LINQ query functionality along with predefined entity types for common tables

### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public class QDatabase : Database
```

```
Public Class QDatabase
    Inherits Database
```

```
public ref class QDatabase : public Database
```

### [-] Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declare

Inherit

Icon

Member

Description



[QDatabase\(String\)](#)

Opens an existing database in read-only mode.



[QDatabase\(String, String\)](#)

Opens an existing database with another database as output.



[QDatabase\(String, DatabaseOpenMode\)](#)

Opens an existing

		database or creates a new one.
💡	<a href="#">QDatabase(IntPtr, Boolean, String, DatabaseOpenMode)</a>	Creates a new database from an MSI handle.
💡	<a href="#">ApplyTransform(String)</a>	Apply a transform to the database, suppressing any error conditions specified by the transform's summary information.  (Inherited from <a href="#">Database</a> .)
💡	<a href="#">ApplyTransform(String, TransformErrors)</a>	Apply a transform to the database, specifying error conditions to suppress.  (Inherited from <a href="#">Database</a> .)
💡	<a href="#">Close()</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidHandleException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
📁	<a href="#">CodePage</a>	Gets or sets the code page of the Database.  (Inherited from

		<a href="#">Database.)</a>
	<a href="#">Commit()</a>	Finalizes the persistor form of the database. All persistent data is written to the writeable database, and no temporary columns or rows are written.  (Inherited from <a href="#">Database.</a> )
	<a href="#">Components</a>	Queryable standard table with predefined specialized record type
	<a href="#">CountRows(String)</a>	Gets the count of all rows in the table.  (Inherited from <a href="#">Database.</a> )
	<a href="#">CountRows(String, String)</a>	Gets the count of all rows in the table that satisfy a given condition.  (Inherited from <a href="#">Database.</a> )
	<a href="#">CreateFolders</a>	Queryable standard table with predefined specialized record type
	<a href="#">CreateObjRef(Type)</a>	Creates an object that contains all the relevant information required to

		generate a proxy used to communicate with a remote object.  (Inherited from <a href="#">MarshalByRefObject</a> .)
≡	<a href="#">CreateRecord(Int32)</a>	Creates a new record object with the requested number of fields.  (Inherited from <a href="#">Database</a> .)
≡	<a href="#">CreateTransformSummaryInfo(Database, String, TransformErrors, TransformValidations)</a>	Creates and populates the summary information stream of an existing transform file, and fills in the properties with the base and reference ProductCode and ProductVersion.  (Inherited from <a href="#">Database</a> .)
↙	<a href="#">CustomActions</a>	Queryable standard table with predefined specialized record type
≡	<a href="#">DeleteOnClose(String)</a>	Schedules a file or directory for deletion after the database handle is closed.  (Inherited from <a href="#">Database</a> .)

	<a href="#">Directories</a>	Queryable standard table with predefined specialized record type
	<a href="#">Dispose(Boolean)</a>	Closes the database handle. After closing a handle, further method calls may throw <a href="#">InvalidOperationException</a>  (Inherited from <a href="#">Database</a> .)
	<a href="#">Dispose()</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a>  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">DuplicateFiles</a>	Queryable standard table with predefined specialized record type
	<a href="#">Equals(Object)</a>	Tests whether this handle object is equal to another handle object. Two handle objects are equal if their types are the same and their native integer handles are the same.  (Inherited from <a href="#">InstallerHandle</a> .)

≡	<a href="#">Execute(String, Object[])</a>	Executes the query specified by a SQL string. The query may not be a SELECT statement.
≡	<a href="#">(Inherited from Database.)</a>	(Inherited from Database.)
≡	<a href="#">Execute(String, Record)</a>	Executes the query specified by a SQL string. The query may not be a SELECT statement.
≡	<a href="#">(Inherited from Database.)</a>	(Inherited from Database.)
≡	<a href="#">ExecuteIntegerQuery(String, Object[])</a>	Executes the specified SQL SELECT query and returns all results as integers.
≡	<a href="#">(Inherited from Database.)</a>	(Inherited from Database.)
≡	<a href="#">ExecuteIntegerQuery(String, Record)</a>	Executes the specified SQL SELECT query and returns all results as integers.
≡	<a href="#">(Inherited from Database.)</a>	(Inherited from Database.)
≡	<a href="#">ExecuteQuery(String, Object[])</a>	Executes the specified SQL SELECT query and returns all results.
≡	<a href="#">(Inherited from Database.)</a>	(Inherited from Database.)
≡	<a href="#">ExecuteQuery(String, Record)</a>	Executes the specified

		SQL SELECT query and returns all results.  (Inherited from <a href="#">Database</a> .)
≡	<a href="#">ExecuteScalar(String, Object[])</a>	Executes the specified SQL SELECT query and returns a single result.  (Inherited from <a href="#">Database</a> .)
≡	<a href="#">ExecuteScalar(String, Record)</a>	Executes the specified SQL SELECT query and returns a single result.  (Inherited from <a href="#">Database</a> .)
≡	<a href="#">ExecuteStringQuery(String, Object[])</a>	Executes the specified SQL SELECT query and returns all results as strings.  (Inherited from <a href="#">Database</a> .)
≡	<a href="#">ExecuteStringQuery(String, Record)</a>	Executes the specified SQL SELECT query and returns all results as strings.  (Inherited from <a href="#">Database</a> .)
≡	<a href="#">Export(String, String)</a>	Copies the structure and data from a specified table to a text archive file.

		(Inherited from Database.)
	<a href="#">ExportAll(String)</a>	Exports all database tables, streams, and summary information to archive files.  (Inherited from Database.)
	<a href="#">FeatureComponents</a>	Queryable standard table with predefined specialized record type
	<a href="#">Features</a>	Queryable standard table with predefined specialized record type
	<a href="#">FileHashes</a>	Queryable standard table with predefined specialized record type
	<a href="#">FilePath</a>	Gets the file path the Database was originally opened from, or null if not known.  (Inherited from Database.)
	<a href="#">Files</a>	Queryable standard table with predefined specialized record type

	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> )
	<a href="#">GenerateTransform(Database, String)</a>	Creates a transform that, when applied to the object database, results in the reference database.  (Inherited from <a href="#">Database</a> .)
	<a href="#">GetHashCode()</a>	Gets a hash value for the handle object.  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">GetLifetimeService()</a>	Retrieves the current lifetime service object that controls the lifetime policy for this instance  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> )
	<a href="#">Handle</a>	Gets the native integer handle.  (Inherited from

		<a href="#">InstallerHandle.)</a>
	<a href="#">Import(String)</a>	Imports a database table from a text archive file, dropping any existing table.  (Inherited from <a href="#">Database.</a> )
	<a href="#">ImportAll(String)</a>	Imports all database tables, streams, and summary information from archive files.  (Inherited from <a href="#">Database.</a> )
	<a href="#">InitializeLifetimeService()</a>	Obtains a lifetime service object to control the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject.</a> )
	<a href="#">InstallExecuteSequences</a>	Queryable standard table with predefined specialized record type
	<a href="#">InstallUISequences</a>	Queryable standard table with predefined specialized record type
	<a href="#">IsClosed</a>	Checks if the handle is closed. When closed, method calls on the handle object may

		throw an <a href="#">InvalidHandleException</a>  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">IsColumnPersistent(String, String)</a>	Checks whether a table contains a persistent column with a given name.  (Inherited from <a href="#">Database</a> .)
	<a href="#">IsReadOnly</a>	Gets a boolean value indicating whether this database was opened in read-only mode.  (Inherited from <a href="#">Database</a> .)
	<a href="#">IsTablePersistent(String)</a>	Checks whether a table exists and is persistent in the database.  (Inherited from <a href="#">Database</a> .)
	<a href="#">IsTransformValid(String)</a>	Checks whether a transform is valid for this Database, according to its validation data and flags.  (Inherited from <a href="#">Database</a> .)
	<a href="#">IsTransformValid(SummaryInfo)</a>	Checks whether a transform is valid for this Database,

		according to its <a href="#">SummaryInfo</a> data.
		(Inherited from <a href="#">Database</a> .)
	<a href="#">Item[String]</a>	Gets a queryable table from the database.
	<a href="#">LaunchConditions</a>	Queryable standard table with predefined specialized record type
	<a href="#">Log</a>	Gets or sets a log where all MSI SQL queries are written.
	<a href="#">Medias</a>	Queryable standard table with predefined specialized record type
	<a href="#">MemberwiseClone(Boolean)</a>	Creates a shallow copy of the current <a href="#">MarshalByRefObject</a> object. (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> . (Inherited from <a href="#">Object</a> )
	<a href="#">Merge(Database, String)</a>	Merges another

		database with this database.
		(Inherited from <a href="#">Database</a> .)
	<a href="#">Merge(Database)</a>	Merges another database with this database.  (Inherited from <a href="#">Database</a> .)
	<a href="#">OpenMode</a>	Gets the open mode for the database.  (Inherited from <a href="#">Database</a> .)
	<a href="#">OpenView(String, Object[])</a>	Gets a View object representing the query specified by a SQL string.  (Inherited from <a href="#">Database</a> .)
	<a href="#">Properties</a>	Queryable standard table with predefined specialized record type
	<a href="#">Registries</a>	Queryable standard table with predefined specialized record type
	<a href="#">RemoveFiles</a>	Queryable standard table with predefined specialized record type

	<a href="#">SummaryInfo</a>	Gets the SummaryInfo object for this database that can be used to examine and modify properties to the summary information stream.  (Inherited from <a href="#">Database</a> .)
	<a href="#">Tables</a>	Gets the collection of tables in the Database  (Inherited from <a href="#">Database</a> .)
	<a href="#">ToString()</a>	Returns the file path of this database, or the handle value if a file path was not specified  (Inherited from <a href="#">Database</a> .)
	<a href="#">ViewTransform(String)</a>	Apply a transform to the database, recording the changes in the "_TransformView" table  (Inherited from <a href="#">Database</a> .)

## Inheritance Hierarchy

[Object](#)

-  [MarshalByRefObject](#)
-  [InstallerHandle](#)
-  [Database](#)

## └ **QDatabase**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►

**QDatabase()**

C# ▾

## Members

Icon	Member	Description
≡	<a href="#">QDatabase(String)</a>	Opens an existing database in read-only mode.
≡	<a href="#">QDatabase(String, String)</a>	Opens an existing database with another database as output.
≡	<a href="#">QDatabase(String, DatabaseOpenMode)</a>	Opens an existing database or creates a new one.
💡	<a href="#">QDatabase(IntPtr, Boolean, String, DatabaseOpenMode)</a>	Creates a new database from an MS handle.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation

## QDatabase Constructor (handle, ownsHandle, filePath, openMode)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►  
QDatabase(IntPtr, Boolean, String, DatabaseOpenMode)

C# ▾

Creates a new database from an MSI handle.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected QDatabase(  
    IntPtr handle,  
    bool ownsHandle,  
    string filePath,  
    DatabaseOpenMode openMode  
)
```

```
Protected Sub New ( _  
    handle As IntPtr, _  
    ownsHandle As Boolean, _  
    filePath As String, _  
    openMode As DatabaseOpenMode _  
)
```

```
protected:  
QDatabase(  
    IntPtr handle,  
    bool ownsHandle,  
    String^ filePath,  
    DatabaseOpenMode openMode  
)
```

### - Parameters

#### **handle (IntPtr)**

Native MSI database handle.

***ownsHandle* ([Boolean](#))**

True if the handle should be closed when the database object is disposed

***filePath* ([String](#))**

Path of the database file, if known

***openMode* ([DatabaseOpenMode](#))**

Mode the handle was originally opened in

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►

## QDatabase(String)

C# ▾

Opens an existing database in read-only mode.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public QDatabase(
    string filePath
)
```

```
Public Sub New ( _
    filePath As String _
)
```

```
public:
QDatabase(
    String^ filePath
)
```

### Parameters

#### **filePath (String)**

Path to the database file.

### Remarks

Because this constructor initiates database access, it cannot be used with a running installation.

The Database object should be [Close\(\)](#)d after use. The finalizer will close the handle if it is still open, however due to the nondeterministic nature of finalization it is best that the handle be closed manually as soon as it is no longer needed, as leaving lots of unused handles open can degrade performance.

## Exceptions

Exception	Condition
InstallerException	the database could not be created/opened

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)

# QDatabase Constructor (filePath, mode)

Deployment Tools Foundation Namespaces ►  
 Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►  
**QDatabase(String, DatabaseOpenMode)**

C# ▾

Opens an existing database or creates a new one.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public QDatabase(
    string filePath,
    DatabaseOpenMode mode
)
```

```
Public Sub New ( _
    filePath As String, _
    mode As DatabaseOpenMode _
)
```

```
public:
QDatabase(
    String^ filePath,
    DatabaseOpenMode mode
)
```

## Parameters

### **filePath (String)**

Path to the database file. If an empty string is supplied, a temporary database is created that is not persisted.

### **mode (DatabaseOpenMode)**

Open mode for the database

## Remarks

To make and save changes to a database first open the database in transaction, create or, or direct mode. After making the changes, always call the Commit method before closing the database handle. The Comm method flushes all buffers.

Always call the Commit method on a database that has been opened in direct mode before closing the database. Failure to do this may corrupt the database.

Because this constructor initiates database access, it cannot be used with a running installation.

The Database object should be [Close\(\)](#)d after use. The finalizer will close the handle if it is still open, however due to the nondeterministic nature of finalization it is best that the handle be closed manually as soon as it is no longer needed, as leaving lots of unused handles open can degrade performance.

## Exceptions

Exception	Condition
<a href="#">InstallerException</a>	the database could not be created/opened

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)

# QDatabase Constructor (filePath, outputPath)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►

**QDatabase(String, String)**

C#

Opens an existing database with another database as output.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public QDatabase(
    string filePath,
    string outputPath
)
```

```
Public Sub New ( _
    filePath As String, _
    outputPath As String _
)
```

```
public:
QDatabase(
    String^ filePath,
    String^ outputPath
)
```

## Parameters

### **filePath (String)**

Path to the database to be read.

### **outputPath (String)**

Open mode for the database

## Return Value

Database object representing the created or opened database

## **Remarks**

When a database is opened as the output of another database, the summary information stream of the output database is actually a read-only mirror of the original database and thus cannot be changed. Additionally, it is not persisted with the database. To create or modify the summary information for the output database it must be closed and re-opened.

The returned Database object should be [Close\(\)](#)d after use. The finalize will close the handle if it is still open, however due to the nondeterminist nature of finalization it is best that the handle be closed manually as soon as it is no longer needed, as leaving lots of unused handles open can degrade performance.

## **Exceptions**

Exception	Condition
<a href="#">InstallerException</a>	the database could not be created/opened

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Components Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►

**Components**

C# ▾

Queryable standard table with predefined specialized record type.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public QTable<Component_> Components { get; }
```

```
Public ReadOnly Property Components As QTable(Of Com
```

```
public:  
property QTable<Component_>^ Components {  
    QTable<Component_>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)



# Deployment Tools Foundation **CreateFolders** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►

## CreateFolders

C# ▾

Queryable standard table with predefined specialized record type.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public QTable<CreateFolder_> CreateFolders { get; }
```

```
Public ReadOnly Property CreateFolders As QTable(Of
```

```
public:  
property QTable<CreateFolder_>^ CreateFolders {  
    QTable<CreateFolder_>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)



# Deployment Tools Foundation **CustomActions** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►

## CustomActions

C# ▾

Queryable standard table with predefined specialized record type.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public QTable<CustomAction_> CustomActions { get; }
```

```
Public ReadOnly Property CustomActions As QTable(Of
```

```
public:  
property QTable<CustomAction_>^ CustomActions {  
    QTable<CustomAction_>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)



# Deployment Tools Foundation Directories Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►

## Directories

C# ▾

Queryable standard table with predefined specialized record type.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public QTable<Directory_> Directories { get; }
```

```
Public ReadOnly Property Directories As QTable(Of Di
```

```
public:  
property QTable<Directory_>^ Directories {  
    QTable<Directory_>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)



Deployment Tools Foundation

## DuplicateFiles Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►

### DuplicateFiles

C#

Queryable standard table with predefined specialized record type.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public QTable<DuplicateFile_> DuplicateFiles { get;
```

```
Public ReadOnly Property DuplicateFiles As QTable(Of
```

```
public:  
property QTable<DuplicateFile_>^ DuplicateFiles {  
    QTable<DuplicateFile_>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)



Deployment Tools Foundation

## FeatureComponents Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►

### FeatureComponents

C#

Queryable standard table with predefined specialized record type.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public QTable<FeatureComponent_> FeatureComponents {
```

```
Public ReadOnly Property FeatureComponents As QTable
```

```
public:  
property QTable<FeatureComponent_>^ FeatureComponents  
    QTable<FeatureComponent_>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)



## Deployment Tools Foundation **Features Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►

### Features

C# ▾

Queryable standard table with predefined specialized record type.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public QTable<Feature_> Features { get; }
```

```
Public ReadOnly Property Features As QTable(Of Feature)
```

```
public:  
property QTable<Feature_>^ Features {  
    QTable<Feature_>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)



# Deployment Tools Foundation **FileHashes** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►

## FileHashes

C# ▾

Queryable standard table with predefined specialized record type.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public QTable<FileHash_> FileHashes { get; }
```

```
Public ReadOnly Property FileHashes As QTable(Of Fil
```

```
public:  
property QTable<FileHash_>^ FileHashes {  
    QTable<FileHash_>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)



# Deployment Tools Foundation **Files Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ► Files

C#

Queryable standard table with predefined specialized record type.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public QTable<File_> Files { get; }
```

```
Public ReadOnly Property Files As QTable(Of File_)
```

```
public:  
property QTable<File_>^ Files {  
    QTable<File_>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation **InstallExecuteSequences**

## Property

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►  
**InstallExecuteSequences**

C# ▾

Queryable standard table with predefined specialized record type.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public QTable<InstallSequence_> InstallExecuteSequen
```

```
Public ReadOnly Property InstallExecuteSequences As
```

```
public:  
property QTable<InstallSequence_>^ InstallExecuteSe  
    QTable<InstallSequence_>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:  
Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0  
(3.5.1623.0)



## Deployment Tools Foundation **InstallUISequences** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►

### InstallUISequences

C# ▾

Queryable standard table with predefined specialized record type.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public QTable<InstallSequence_> InstallUISequences {
```

```
Public ReadOnly Property InstallUISequences As QTabl
```

```
public:  
property QTable<InstallSequence_>^ InstallUISequences  
    QTable<InstallSequence_>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)



Deployment Tools Foundation

## Item Property (table)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►

### Item[String]

C# ▾

Gets a queryable table from the database.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public QTable<QRecord> this[  
    string table  
] { get; }
```

```
Public ReadOnly Default Property Item ( _  
    table As String _  
) As QTable(Of QRecord)
```

```
public:  
property QTable<QRecord^>^ default[String^ table] {  
    QTable<QRecord^>^ get (String^ table);  
}
```

#### Parameters

##### table (String)

name of the table

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

# Deployment Tools Foundation **LaunchConditions** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►

## LaunchConditions

C# ▾

Queryable standard table with predefined specialized record type.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public QTable<LaunchCondition_> LaunchConditions { g
```

```
Public ReadOnly Property LaunchConditions As QTable(
```

```
public:  
property QTable<LaunchCondition_>^ LaunchConditions  
    QTable<LaunchCondition_>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)



# Deployment Tools Foundation Log Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ► Log

C# ▾

Gets or sets a log where all MSI SQL queries are written.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public TextWriter Log { get; set; }
```

```
Public Property Log As TextWriter
```

```
public:  
property TextWriter^ Log {  
    TextWriter^ get ();  
    void set (TextWriter^ value);  
}
```

## Remarks

The log can be useful for debugging, or simply to watch the LINQ magic in action.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Medias Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ► Medias

C# ▾

Queryable standard table with predefined specialized record type.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public QTable<Media_> Medias { get; }
```

```
Public ReadOnly Property Medias As QTable(Of Media_)
```

```
public:  
property QTable<Media_>^ Medias {  
    QTable<Media_>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation

# Properties Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►

## Properties

C# ▾

Queryable standard table with predefined specialized record type.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public QTable<Property_> Properties { get; }
```

```
Public ReadOnly Property Properties As QTable(Of Pro
```

```
public:  
property QTable<Property_>^ Properties {  
    QTable<Property_>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)



# Deployment Tools Foundation Registries Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►

## Registries

C# ▾

Queryable standard table with predefined specialized record type.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public QTable<Registry_> Registries { get; }
```

```
Public ReadOnly Property Registries As QTable(Of Registry)
```

```
public:  
property QTable<Registry_>^ Registries {  
    QTable<Registry_>^ get();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)



# Deployment Tools Foundation RemoveFiles Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QDatabase ►

## RemoveFiles

C# ▾

Queryable standard table with predefined specialized record type.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public QTable<RemoveFile_> RemoveFiles { get; }
```

```
Public ReadOnly Property RemoveFiles As QTable(Of Re
```

```
public:  
property QTable<RemoveFile_>^ RemoveFiles {  
    QTable<RemoveFile_>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)



C#

Generic record entity for queryable databases, and base for strongly-typed entity subclasses.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public class QRecord
```

```
Public Class QRecord
```

```
public ref class QRecord
```

## Members

All Members

Constructors

Methods

Properties

 Public Instance

Declared

 Protected Static

Inherited

Icon

Icon	Member	Description
≡	<a href="#">QRecord()</a>	Do not call. Use QTable.NewRecord() instead.
≡	<a href="#">Assign()</a>	Not yet implemented.
≡	<a href="#">Delete()</a>	Deletes the record from the table if it exists.
≡		

	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">FieldCount</a>	Gets the number of fields in the record.
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance  (Inherited from <a href="#">Object</a> .)
	<a href="#">I(Int32)</a>	Used by subclasses to get a field as an integer.
	<a href="#">Insert()</a>	Inserts the record in the database.
	<a href="#">Insert(Boolean)</a>	Inserts the record into the table.
	<a href="#">Item[String]</a>	Gets or sets a record field.

	<a href="#">Item[Int32]</a>	Gets or sets a record field.
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Merge()</a>	Not yet implemented.
	<a href="#">NI(Int32)</a>	Used by subclasses to get a field as a nullable integer.
	<a href="#">Refresh()</a>	Not yet implemented.
	<a href="#">ToString()</a>	Dumps all record fields to a string.  (Overrides <a href="#">Object.ToString()</a> .)
	<a href="#">Update(IList&lt;String&gt;, IList&lt;String&gt;)</a>	Update multiple fields in the record (and the database).
	<a href="#">Update(IList&lt;Int32&gt;, IList&lt;String&gt;)</a>	Update multiple fields in the record (and the database).
	<a href="#">Validate()</a>	Not yet implemented.
	<a href="#">ValidateDelete()</a>	Not yet implemented.
	<a href="#">ValidateFields()</a>	Not yet implemented.

≡	<a href="#">ValidateNew()</a>	Not yet implemented.

## [-] **Remarks**

Several predefined specialized subclasses are provided for common standard tables. Subclasses for additional standard tables or custom tables are not necessary, but they are easy to create and make the coding experience much nicer.

When creating subclasses, the following attributes may be useful:

[DatabaseTableAttribute](#), [DatabaseColumnAttribute](#)

## [-] **Inheritance Hierarchy**

[Object](#)

└ [QRecord](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)

Do not call. Use QTable.NewRecord() instead.

## Declaration Syntax

[C#](#)[Visual Basic](#)[Visual C++](#)

```
public QRecord()
```

```
Public Sub New
```

```
public:  
QRecord()
```

## Remarks

Subclasses must also provide a public parameterless constructor.

QRecord constructors are only public due to implementation reasons (to satisfy the new() constraint on the QTable generic class). They are not intended to be called by user code other than a subclass constructor. If the constructor is invoked directly, the record instance will not be properly initialized (associated with a database table) and calls to methods on the instance will throw a NullReferenceException.

## See Also

[NewRecord\(\)](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Assign Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ► **Assign()**

C# ▾

Not yet implemented.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Assign()
```

```
Public Sub Assign
```

```
public:  
void Assign()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)

C# ▾

Deletes the record from the table if it exists.

## - Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Delete()
```

```
Public Sub Delete
```

```
public:  
void Delete()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# FieldCount Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ►

## FieldCount

C# ▾

Gets the number of fields in the record.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public int FieldCount { get; }
```

```
Public ReadOnly Property FieldCount As Integer
```

```
public:  
property int FieldCount {  
    int get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation

## I Method (index)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ► I(Int32)

C# ▾

Used by subclasses to get a field as an integer.

### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected int I(  
    int index  
)
```

```
Protected Function I ( _  
    index As Integer _  
) As Integer
```

```
protected:  
int I(  
    int index  
)
```

### - Parameters

#### *index* (Int32)

zero-based column index of the field

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation **Insert Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ► **Insert()**

C# ▾

**Members**

Icon	Member	Description
	<a href="#">Insert()</a>	Inserts the record in the database.
	<a href="#">Insert(Boolean)</a>	Inserts the record into the table.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation **Insert Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ► **Insert()**

C# ▾

Inserts the record in the database.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public void Insert()
```

```
Public Sub Insert
```

```
public:  
void Insert()
```

[-] **Remarks**

The record (primary keys) may not already exist in the table.

Use [NewRecord\(\)](#) to get a new record. Primary keys and all required fields must be filled in before insertion.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

## Deployment Tools Foundation **Insert Method (temporary)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ►

### **Insert(Boolean)**

C# ▾

Inserts the record into the table.

#### [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Insert(  
    bool temporary  
)
```

```
Public Sub Insert ( _  
    temporary As Boolean _  
)
```

```
public:  
void Insert(  
    bool temporary  
)
```

#### [-] Parameters

##### **temporary (Boolean)**

true if the record is temporarily inserted, to be visible only as long as the database is open

#### [-] Remarks

The record (primary keys) may not already exist in the table.

Use [NewRecord\(\)](#) to get a new record. Primary keys and all required fields must be filled in before insertion.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Item Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ► Item

C# ▾

**Members**

Icon	Member	Description
	<a href="#">Item[String]</a>	Gets or sets a record field.
	<a href="#">Item[Int32]</a>	Gets or sets a record field.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

# Deployment Tools Foundation Item Property (index)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ► Item[Int32]

C# ▾

Gets or sets a record field.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string this[  
    int index  
] { get; set; }
```

```
Public Default Property Item ( _  
    index As Integer _  
) As String
```

```
public:  
property String^ default[int index] {  
    String^ get (int index);  
    void set (int index, String^ value);  
}
```

## Parameters

### index (Int32)

zero-based column index of the field

## Remarks

Setting a field value will automatically update the database.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

# Deployment Tools Foundation Item Property (field)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ►

## Item[String]

C# ▾

Gets or sets a record field.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string this[  
    string field  
] { get; set; }
```

```
Public Default Property Item ( _  
    field As String _  
) As String
```

```
public:  
property String^ default[String^ field] {  
    String^ get (String^ field);  
    void set (String^ field, String^ value);  
}
```

### Parameters

#### field (String)

column name of the field

### Remarks

Setting a field value will automatically update the database.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation

## Merge Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ► Merge()

C# ▾

Not yet implemented.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public bool Merge()
```

```
Public Function Merge As Boolean
```

```
public:  
bool Merge()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## NI Method (index)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ► NI(Int32)

C# ▾

Used by subclasses to get a field as a nullable integer.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
protected Nullable<int> NI(  
    int index  
)
```

```
Protected Function NI ( _  
    index As Integer _  
) As Nullable(Of Integer)
```

```
protected:  
Nullable<int> NI(  
    int index  
)
```

### Parameters

#### *index* (Int32)

zero-based column index of the field

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation

## Refresh Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ► Refresh()

C# ▾

Not yet implemented.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Refresh()
```

```
Public Sub Refresh
```

```
public:  
void Refresh()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **ToString Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ► **ToString()**

C# ▾

Dumps all record fields to a string.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public override string ToString()
```

```
Public Overrides Function ToString As String
```

```
public:  
virtual String^ ToString() override
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation

## Update Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ► **Update()**

C# ▾

### Members

Icon	Member	Description
	<a href="#">Update(IList&lt;String&gt;, IList&lt;String&gt;)</a>	Update multiple fields in the record (and the database).
	<a href="#">Update(IList&lt;Int32&gt;, IList&lt;String&gt;)</a>	Update multiple fields in the record (and the database).

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

## Deployment Tools Foundation **Update Method (indexes, values)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ►

**Update(IList<Int32>, IList<String>)**

C# ▾

Update multiple fields in the record (and the database).

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Update(  
    IList<int> indexes,  
    IList<string> values  
)
```

```
Public Sub Update ( _  
    indexes As IList(Of Integer), _  
    values As IList(Of String) _  
)
```

```
public:  
void Update(  
    IList<int>^ indexes,  
    IList<String>^ values  
)
```

### Parameters

#### **indexes (IList<Int32>)**

column indexes of fields to update

#### **values (IList<String>)**

new values for each field being updated

### Remarks

The record (primary keys) must already exist in the table.

Updating primary key fields is not yet implemented; use Delete() and

Insert() instead.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Update Method (fields, values)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ►

**Update(IList<String>, IList<String>)**

C# ▾

Update multiple fields in the record (and the database).

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Update(  
    IList<string> fields,  
    IList<string> values  
)
```

```
Public Sub Update ( _  
    fields As IList(Of String), _  
    values As IList(Of String) _  
)
```

```
public:  
void Update(  
    IList<String>^ fields,  
    IList<String>^ values  
)
```

### Parameters

#### **fields (IList<String>)**

column names of fields to update

#### **values (IList<String>)**

new values for each field being updated

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation **Validate Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ► **Validate()**

C# ▾

Not yet implemented.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public ICollection<ValidationFileInfo> Validate()
```

```
Public Function Validate As ICollection(Of Validation
```

```
public:  
ICollection<ValidationFileInfo>^ Validate()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **ValidateDelete Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ►

**ValidateDelete()**

C# ▾

Not yet implemented.

Declaration Syntax

C#

Visual Basic

Visual C++

```
public ICollection<ValidationFileInfo> ValidateDelete()
```

```
Public Function ValidateDelete As ICollection(Of Val
```

```
public:  
ICollection<ValidationFileInfo>^ ValidateDelete()
```

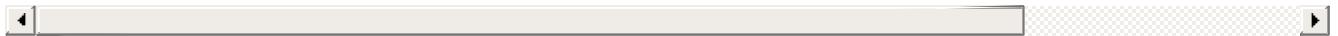
---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)



Deployment Tools Foundation **ValidateFields Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ►

**ValidateFields()**

C# ▾

Not yet implemented.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public ICollection<ValidationFileInfo> ValidateFiel
```

```
Public Function ValidateFields As ICollection(Of Val
```

```
public:  
ICollection<ValidationFileInfo>^ ValidateFields()
```

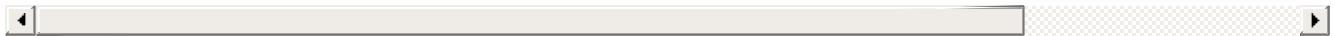
---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)



Deployment Tools Foundation **ValidateNew Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QRecord ►

**ValidateNew()**

C# ▾

Not yet implemented.

Declaration Syntax

C#

Visual Basic

Visual C++

```
public ICollection<ValidationFileInfo> ValidateNew(
```

```
Public Function ValidateNew As ICollection(Of Valida
```

```
public:  
ICollection<ValidationFileInfo>^ ValidateNew( )
```

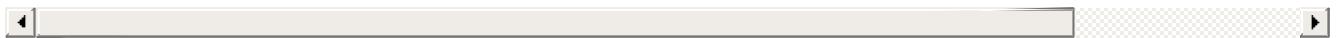
---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)



C# ▾

Represents one table in a LINQ-queryable Database.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public sealed class QTable<TRecord> : IOrderedQueryable<TRecord>, IEnumerable<TRecord>, IQueryable<TRecord>, IEnumerable
```

```
Public NotInheritable Class QTable(Of TRecord As {Ne
    Implements IOrderedQueryable(Of TRecord), IQ
    IEnumerable(Of TRecord), IOrderedQueryable,
    IQueryable
```

```
generic<typename TRecord>
where TRecord : gcnew(), QRecord
public ref class QTable sealed : IOrderedQueryable<T
    IQueryable<TRecord>, IEnumerable<TRecord>, I
    IEnumerable, IQueryable
```

## Generic Template Parameters

### TRecord

type that represents one record in the table

## Members

All Members

Constructors

Methods

Properties

Public  
 Protected

Instance  
 Static

Declared  
 Inherited

Icon

Member

Description

	<code>QTable&lt;TRecord&gt;(QDatabase)</code>	Creates a new QTable, inferring the table name from the name of the record type parameter.
	<code>QTable&lt;TRecord&gt;(QDatabase, String)</code>	Creates a new QTable with an explicit table name.
	<code>Database</code>	Gets the database this table is associated with.
	<code>Equals(Object)</code>	Determines whether the specified <code>Object</code> is equal to the current <code>Object</code> .  (Inherited from <code>Object</code> .)
	<code>Finalize()</code>	Allows an <code>Object</code> to attempt to free resources and perform other cleanup operations before the <code>Object</code> is reclaimed by garbage collection.  (Inherited from <code>Object</code> .)
	<code>GetEnumerator()</code>	Enumerates over all records in the table.
	<code>GetHashCode()</code>	Serves as a hash function for a particular type. <code>GetHashCode()</code> is suitable for use in hashing algorithms and data structures like a hash table.  (Inherited from <code>Object</code> .)
		

	<code>GetType()</code>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
	<code>MemberwiseClone()</code>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<code>NewRecord()</code>	Creates a new record that can be inserted into this table.
	<code>TableInfo</code>	Gets schema information about the table.
	<code>ToString()</code>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)

## Remarks

This class is the primary gateway to all LINQ to MSI query functionality.

The TRecord generic parameter may be the general [QRecord](#) class, or specialized subclass of [QRecord](#).

## Inheritance Hierarchy

[Object](#)

 [QTable<TRecord>](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)

[◀]

[▶]

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QTable<TRecord> ►

**QTable<TRecord>()**

C# ▾

## ▀ Members

Icon	Member	Description
≡	<a href="#">QTable&lt;TRecord&gt;(QDatabase)</a>	Creates a new QTable, inferring the table name from the name of the record type parameter.
≡	<a href="#">QTable&lt;TRecord&gt;(QDatabase, String)</a>	Creates a new QTable with an explicit table name.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

## QTable<TRecord> Constructor (db)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Linq ► QTable<TRecord> ►  
QTable<TRecord>(QDatabase)

C#

Creates a new QTable, inferring the table name from the name of the record type parameter.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public QTable(  
    QDatabase db  
)
```

```
Public Sub New ( _  
    db As QDatabase _  
)
```

```
public:  
QTable(  
    QDatabase^ db  
)
```

### Parameters

#### db (QDatabase)

database that contains the table

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)

# QTable<TRecord> Constructor (db, table)

Deployment Tools Foundation Namespaces ►  
 Microsoft.Deployment.WindowsInstaller.Linq ► QTable<TRecord> ►  
**QTable<TRecord>(QDatabase, String)**

C# ▾

Creates a new QTable with an explicit table name.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public QTable(
    QDatabase db,
    string table
)
```

```
Public Sub New ( _
    db As QDatabase, _
    table As String _
)
```

```
public:
QTable(
    QDatabase^ db,
    String^ table
)
```

## Parameters

### **db (QDatabase)**

database that contains the table

### **table (String)**

name of the table

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0  
(3.5.1623.0)

Deployment Tools Foundation

## Database Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QTable<TRecord> ►

### Database

C# ▾

Gets the database this table is associated with.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public QDatabase Database { get; }
```

```
Public ReadOnly Property Database As QDatabase
```

```
public:  
property QDatabase^ Database {  
    QDatabase^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation **GetEnumerator Method**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Linq ► QTable<TRecord> ►  
**GetEnumerator()**

C# ▾

Enumerates over all records in the table.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public IEnumarator<TRecord> GetEnumerator()
```

```
Public Function GetEnumerator As IEnumarator(Of TRec
```

```
public:  
virtual IEnumarator<TRecord>^ GetEnumerator() sealed
```

[-] **Return Value**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

## NewRecord Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QTable<TRecord> ►

### NewRecord()

C# ▾

Creates a new record that can be inserted into this table.

#### - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public TRecord NewRecord()
```

```
Public Function NewRecord As TRecord
```

```
public:  
TRecord NewRecord()
```

#### - Return Value

a record with all fields initialized to null

#### - Remarks

Primary keys and required fields must be filled in with non-null values before the record can be inserted.

The record is tied to this table in this database; it cannot be inserted into another table or database.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

# Deployment Tools Foundation **TableInfo** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► QTable<TRecord> ►

## TableInfo

C# ▾

Gets schema information about the table.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public TableInfo TableInfo { get; }
```

```
Public ReadOnly Property TableInfo As TableInfo
```

```
public:  
property TableInfo^ TableInfo {  
    TableInfo^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module:

Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

# Deployment Tools Foundation **Queryable Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► **Queryable**

C# ▾

Allows any Database instance to be converted into a queryable database.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static class Queryable
```

```
Public NotInheritable Class Queryable
```

```
public ref class Queryable abstract sealed
```

## Members

All Members

Methods

Public

Instance

Declared

Protected

Static

Inherited

Icon	Member	Description
	<a href="#">AsQueryable(Database)</a>	Converts any Database instance into a queryable database.

## Inheritance Hierarchy

Object

└ **Queryable**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0

(3.5.1623.0)

## Deployment Tools Foundation AsQueryable Method (db)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Linq ► Queryable ►

### AsQueryable(Database)

C# ▾

Converts any Database instance into a queryable database.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static QDatabase AsQueryable(  
    Database db  
)
```

```
Public Shared Function AsQueryable ( _  
    db As Database _  
) As QDatabase
```

```
public:  
static QDatabase^ AsQueryable(  
    Database^ db  
)
```

#### Parameters

*db* (Database)

#### Return Value

Queryable database instance that operates on the same MSI handle.

#### Remarks

This extension method is meant for convenient on-the-fly conversion. If the existing database instance already happens to be a QDatabase, the it is returned unchanged. Otherwise since the new database carries the same MSI handle, only one of the instances needs to be closed, not bot

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Linq (Module: Microsoft.Deployment.WindowsInstaller.Linq) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools

# Foundation Microsoft.Deployment.WindowsInstaller Namespace

Deployment Tools Foundation Namespaces ►  
**Microsoft.Deployment.WindowsInstaller.Package**

C# ▾

Extended classes for working with Windows Installer installation and patch packages.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
namespace Microsoft.Deployment.WindowsInstaller.Pack
```

```
Namespace Microsoft.Deployment.WindowsInstaller.Pack
```

```
namespace Microsoft.Deployment.WindowsInstaller.Pack
```

## Types

All Types

Classes

Delegates

Icon	Type	Description
	<a href="#">InstallPackage</a>	Provides access to powerful build, maintenance, and analysis operations on an installation package (.MSI or .MSM).
	<a href="#">InstallPackageMessageHandler</a>	Handles status messages generated when operations are performed on an <a href="#">InstallPackage</a> or <a href="#">PatchPackage</a> .

	<b>InstallPackageProperties</b>	Accessor for getting and setting properties of the <a href="#">InstallPackage</a> database.
	<b>InstallPath</b>	Represents the installation path of a file or directory from an installer product database.
	<b>InstallPathCollection</b>	Represents a collection of InstallPaths that are the child paths of the same parent directory.
	<b>InstallPathMap</b>	Represents a mapping of install paths for all directories, components, or files in an installation database.
	<b>PatchPackage</b>	Provides access to convenient properties and operations on a patch package (.MSP).
	<b>TransformInfo</b>	Contains properties of a transform package (.MST).

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

# Deployment Tools Foundation **InstallPackage Class**

Deployment Tools Foundation Namespaces ►

[Microsoft.Deployment.WindowsInstaller.Package](#) ► [InstallPackage](#)

C# ▾

Provides access to powerful build, maintenance, and analysis operations on an installation package (.MSI or .MSM).

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public class InstallPackage : Database
```

```
Public Class InstallPackage
    Inherits Database
```

```
public ref class InstallPackage : public Database
```

## Members

All Members	Constructors	Methods	Properties	Events
<input checked="" type="checkbox"/> Public		<input checked="" type="checkbox"/> Instance		<input checked="" type="checkbox"/> Declared
<input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Static		<input checked="" type="checkbox"/> Inherited

Icon	Member	Description
≡	<a href="#">InstallPackage(String, DatabaseOpenMode)</a>	Creates a new <code>InstallPackage</code> object. The file source directory and working directory are the same as the location as the package file.
≡	<a href="#">InstallPackage(String, DatabaseOpenMode, String, String)</a>	Creates a new <code>InstallPackage</code> object, specifying an alternate file source directory

		and/or working directory.
≡	<a href="#">ApplyPatch(PatchPackage, String)</a>	Applies a patch package to the database, resulting in an installation package that has the patch built in.
≡	<a href="#">ApplyTransform(String)</a>	Apply a transform to the database, suppressing any error conditions specified by the transform's summary information.  (Inherited from <a href="#">Database</a> .)
≡	<a href="#">ApplyTransform(String, TransformErrors)</a>	Apply a transform to the database, specifying error conditions to suppress.  (Inherited from <a href="#">Database</a> .)
≡	<a href="#">Close()</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">CodePage</a>	Gets or sets the code

		page of the Database.  (Inherited from <a href="#">Database</a> .)
	<a href="#">Commit()</a>	Finalizes the persistor form of the database. All persistent data is written to the writeable database, and no temporary columns or rows are written.  (Inherited from <a href="#">Database</a> .)
	<a href="#">CompressionLevel</a>	Gets or sets the compression level use by <a href="#">UpdateFiles()</a> and <a href="#">Consolidate(String)</a> .
	<a href="#">Consolidate(String)</a>	Consolidates a package by combining and recompressing all files into a single internal or external cabinet.
	<a href="#">CountRows(String)</a>	Gets the count of all rows in the table.  (Inherited from <a href="#">Database</a> .)
	<a href="#">CountRows(String, String)</a>	Gets the count of all rows in the table that satisfy a given condition.

		(Inherited from <a href="#">Database</a> .)
≡	<a href="#">CreateObjRef(Type)</a>	Creates an object that contains all the relevant information required to generate a proxy used to communicate with a remote object.  (Inherited from <a href="#">MarshalByRefObject</a> .)
≡	<a href="#">CreateRecord(Int32)</a>	Creates a new record object with the requested number of fields.  (Inherited from <a href="#">Database</a> .)
≡	<a href="#">CreateTransformSummaryInfo(Database, String, TransformErrors, TransformValidations)</a>	Creates and populates the summary information stream of an existing transform file, and fills in the properties with the base and reference ProductCode and ProductVersion.  (Inherited from <a href="#">Database</a> .)
≡	<a href="#">DeleteOnClose(String)</a>	Schedules a file or directory for deletion after the database handle is closed.  (Inherited from <a href="#">Database</a> .)

	<a href="#">Directories</a>	Gets a mapping from Directory keys to source/target paths.
	<a href="#">Dispose(Boolean)</a>	Closes the database handle. After closing a handle, further method calls may throw <a href="#">InvalidHandleException</a>  (Inherited from <a href="#">Database</a> .)
	<a href="#">Dispose()</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidHandleException</a>  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">Equals(Object)</a>	Tests whether this handle object is equal another handle object. Two handle objects are equal if their types are the same and their native integer handles are the same.  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">Execute(String, Object[])</a>	Executes the query specified by a SQL string. The query may not be a SELECT statement.

		(Inherited from <a href="#">Database</a> .)
≡	<a href="#">Execute(String, Record)</a>	Executes the query specified by a SQL string. The query may not be a SELECT statement.  (Inherited from <a href="#">Database</a> .)
≡	<a href="#">ExecuteIntegerQuery(String, Object[])</a>	Executes the specified SQL SELECT query and returns all results as integers.  (Inherited from <a href="#">Database</a> .)
≡	<a href="#">ExecuteIntegerQuery(String, Record)</a>	Executes the specified SQL SELECT query and returns all results as integers.  (Inherited from <a href="#">Database</a> .)
≡	<a href="#">ExecuteQuery(String, Object[])</a>	Executes the specified SQL SELECT query and returns all results.  (Inherited from <a href="#">Database</a> .)
≡	<a href="#">ExecuteQuery(String, Record)</a>	Executes the specified SQL SELECT query and returns all results.  (Inherited from <a href="#">Database</a> .)

	<code>ExecuteScalar(String, Object[])</code>	Executes the specified SQL SELECT query and returns a single result.  (Inherited from <a href="#">Database</a> .)
≡	<code>ExecuteScalar(String, Record)</code>	Executes the specified SQL SELECT query and returns a single result.  (Inherited from <a href="#">Database</a> .)
≡	<code>ExecuteStringQuery(String, Object[])</code>	Executes the specified SQL SELECT query and returns all results as strings.  (Inherited from <a href="#">Database</a> .)
≡	<code>ExecuteStringQuery(String, Record)</code>	Executes the specified SQL SELECT query and returns all results as strings.  (Inherited from <a href="#">Database</a> .)
≡	<code>Export(String, String)</code>	Copies the structure and data from a specified table to a text archive file.  (Inherited from <a href="#">Database</a> .)
≡	<code>ExportAll(String)</code>	Exports all database tables, streams, and

		summary information for archive files.
		(Inherited from <a href="#">Database</a> .)
	<a href="#">ExtractFiles()</a>	Extracts all files to the <a href="#">WorkingDirectory</a> . The files are extracted to the relative directory matching their <a href="#">SourcePath</a> .
	<a href="#">ExtractFiles(Collection&lt;String&gt;)</a>	Extracts a specified list of files to the <a href="#">WorkingDirectory</a> . The files are extracted to the relative directory matching their <a href="#">SourcePath</a> .
	<a href="#">FilePath</a>	Gets the file path the Database was originally opened from, or null if not known.  (Inherited from <a href="#">Database</a> .)
	<a href="#">Files</a>	Gets a mapping from File keys to source/target paths.
	<a href="#">Finalize()</a>	Allows an Object to attempt to free resources and perform

		other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.
		(Inherited from <a href="#">Object</a> )
≡	<a href="#">FindFiles(String)</a>	Gets the list of file key that have the specified long file name.
≡	<a href="#">FindFiles(Regex)</a>	Gets the list of file key whose long file names match a specified regular-expression search pattern.
≡	<a href="#">GenerateTransform(Database, String)</a>	Creates a transform that, when applied to the object database, results in the reference database.  (Inherited from <a href="#">Database</a> .)
≡	<a href="#">GetHashCode()</a>	Gets a hash value for the handle object.  (Inherited from <a href="#">InstallerHandle</a> .)
≡	<a href="#">GetLifetimeService()</a>	Retrieves the current lifetime service object that controls the lifetime policy for this instance  (Inherited from

		<a href="#">MarshalByRefObject. )</a>
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
	<a href="#">Handle</a>	Gets the native integer handle.  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">Import(String)</a>	Imports a database table from a text archive file, dropping any existing table.  (Inherited from <a href="#">Database</a> .)
	<a href="#">ImportAll(String)</a>	Imports all database tables, streams, and summary information from archive files.  (Inherited from <a href="#">Database</a> .)
	<a href="#">InitializeLifetimeService()</a>	Obtains a lifetime service object to control the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject</a> .)
	<a href="#">IsClosed</a>	Checks if the handle is closed. When closed, method calls on the handle object may throw an

		<a href="#">InvalidHandleException</a>  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">IsColumnPersistent(String, String)</a>	Checks whether a table contains a persistent column with a given name.  (Inherited from <a href="#">Database</a> .)
	<a href="#">IsReadOnly</a>	Gets a boolean value indicating whether this database was opened in read-only mode.  (Inherited from <a href="#">Database</a> .)
	<a href="#">IsTablePersistent(String)</a>	Checks whether a table exists and is persistent in the database.  (Inherited from <a href="#">Database</a> .)
	<a href="#">IsTransformValid(String)</a>	Checks whether a transform is valid for this Database, according to its validation data and flags.  (Inherited from <a href="#">Database</a> .)
	<a href="#">IsTransformValid(SummaryInfo)</a>	Checks whether a transform is valid for this Database, according to its

		SummaryInfo data.  (Inherited from <a href="#">Database</a> .)
💡	<a href="#">LogMessage(String, Object[])</a>	Sends a message to the <a href="#">Message</a> event-handler.
💡	<a href="#">MemberwiseClone(Boolean)</a>	Creates a shallow copy of the current <a href="#">MarshalByRefObject</a> object.  (Inherited from <a href="#">MarshalByRefObject</a> .)
💡	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> )
💡	<a href="#">Merge(Database, String)</a>	Merges another database with this database.  (Inherited from <a href="#">Database</a> .)
💡	<a href="#">Merge(Database)</a>	Merges another database with this database.  (Inherited from <a href="#">Database</a> .)
⚡	<a href="#">Message</a>	Handle this event to receive status messages when operations are performed on the instance package.

	<a href="#">OpenMode</a>	Gets the open mode for the database.  (Inherited from <a href="#">Database</a> .)
	<a href="#">OpenView(String, Object[])</a>	Gets a View object representing the query specified by a SQL string.  (Inherited from <a href="#">Database</a> .)
	<a href="#">Property</a>	Accessor for getting and setting properties of the InstallPackage database.
	<a href="#">SourceDirectory</a>	Gets or sets the location to obtain source files and cabinets when extracting or updating files in the working directory. This is often the location of an original copy of the package that is not meant to be modified.
	<a href="#">SummaryInfo</a>	Gets the SummaryInfo object for this database that can be used to examine and modify properties to the

		summary information stream.
		(Inherited from <a href="#">Database</a> .)
	<a href="#">Tables</a>	Gets the collection of tables in the Database
		(Inherited from <a href="#">Database</a> .)
	<a href="#">ToString()</a>	Returns the file path of this database, or the handle value if a file path was not specified
		(Inherited from <a href="#">Database</a> .)
	<a href="#">UpdateDirectories()</a>	Rebuilds the cached directory structure information accessed via the <a href="#">Directories</a> and <a href="#">Files</a> properties. This should be done after modifying the File, Component, or Directory tables, or else the cached information may no longer be accurate.
	<a href="#">UpdateFiles()</a>	Updates the install package with new files from the <a href="#">WorkingDirectory</a> . The files must be in the relative directory matching their

		<p><a href="#">SourcePath</a>. This method re-compresses and packages the files necessary, and also updates the following data: File.FileSize, File.Version, File.Language, MsiFileHash.HashPart</p>
	<a href="#">UpdateFiles(ICollection&lt;String&gt;)</a>	<p>Updates the install package with new files from the <a href="#">WorkingDirectory</a>. The files must be in the relative directory matching their <a href="#">SourcePath</a>. This method re-compresses and packages the files necessary, and also updates the following data: File.FileSize, File.Version, File.Language, MsiFileHash.HashPart</p>
	<a href="#">ViewTransform(String)</a>	<p>Apply a transform to the database, recording the changes in the "_TransformView" table.</p> <p>(Inherited from <a href="#">Database</a>.)</p>
	<a href="#">WorkingDirectory</a>	<p>Gets or sets the location where files will be</p>

extracted to/updated from. Also the location where a temporary folder is created during some operations.

## ■ Inheritance Hierarchy

Object

└ MarshalByRefObject

  └ InstallerHandle

    └ Database

      └ **InstallPackage**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# InstallPackage Constructor

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►

InstallPackage()

C# ▾

## Members

Icon	Member	Description
	<a href="#">InstallPackage(String, DatabaseOpenMode)</a>	Creates a new InstallPackage object The file source directory and working directory are the same as the location of the package file.
	<a href="#">InstallPackage(String, DatabaseOpenMode, String, String)</a>	Creates a new InstallPackage object specifying an alternate file source directory and/or working directory.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

# InstallPackage Constructor (*packagePath*, *openMode*)

Deployment Tools Foundation Namespaces ►  
 Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►  
**InstallPackage(String, DatabaseOpenMode)**



Creates a new InstallPackage object. The file source directory and working directory are the same as the location as the package file.

## Declaration Syntax

[C#](#)
[Visual Basic](#)
[Visual C++](#)

```
public InstallPackage(  
    string packagePath,  
    DatabaseOpenMode openMode  
)
```

```
Public Sub New ( _  
    packagePath As String, _  
    openMode As DatabaseOpenMode _  
)
```

```
public:  
InstallPackage(  
    String^ packagePath,  
    DatabaseOpenMode openMode  
)
```

## Parameters

### *packagePath* ([String](#))

Path to the install package to be created or opened

### *openMode* ([DatabaseOpenMode](#))

Open mode for the database

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

# InstallPackage Constructor (**packagePath**, **openMode**, **sourceDir**, **workingDir**)

Deployment Tools Foundation Namespaces ►

[Microsoft.Deployment.WindowsInstaller.Package](#) ► [InstallPackage](#) ►

**InstallPackage(String, DatabaseOpenMode, String, String)**

C# ▾

Creates a new InstallPackage object, specifying an alternate file source directory and/or working directory.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public InstallPackage(
    string packagePath,
    DatabaseOpenMode openMode,
    string sourceDir,
    string workingDir
)
```

```
Public Sub New ( _
    packagePath As String, _
    openMode As DatabaseOpenMode, _
    sourceDir As String, _
    workingDir As String _
)
```

```
public:
InstallPackage(
    String^ packagePath,
    DatabaseOpenMode openMode,
    String^ sourceDir,
    String^ workingDir
)
```

## Parameters

***packagePath* ([String](#))**

Path to the install package to be created or opened

***openMode* ([DatabaseOpenMode](#))**

Open mode for the database

***sourceDir* ([String](#))**

Location to obtain source files and cabinets when extracting or updating files in the working directory. This is often the location of a original copy of the package that is not meant to be modified. If this parameter is null, it defaults to the directory of *packagePath*.

***workingDir* ([String](#))**

Location where files will be extracted to/updated from. Also the location where a temporary folder is created during some operation  
If this parameter is null, it defaults to the directory of *packagePath*.

## **Remarks**

If the source location is different than the working directory, then no files will be modified at the source location.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ApplyPatch Method (patchPackage, transform)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►  
**ApplyPatch(PatchPackage, String)**

C#

Applies a patch package to the database, resulting in an installation package that has the patch built-in.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public void ApplyPatch(  
    PatchPackage patchPackage,  
    string transform  
)
```

```
Public Sub ApplyPatch ( _  
    patchPackage As PatchPackage, _  
    transform As String _  
)
```

```
public:  
void ApplyPatch(  
    PatchPackage^ patchPackage,  
    String^ transform  
)
```

### - Parameters

#### **patchPackage (PatchPackage)**

The patch package to be applied

#### **transform (String)**

Optional name of the specific transform to apply. This parameter is usually left null, which causes the patch to be searched for a transform that is valid to apply to this database.

## **Remarks**

If the patch contains any binary file patches, they will not immediately be applied to the target files, though they will at installation time.

After calling this method you can use [Consolidate\(String\)](#) to apply the file patches immediately and also discard any outdated files from the package.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **CompressionLevel Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►

**CompressionLevel**

C# ▾

Gets or sets the compression level used by [UpdateFiles\(\)](#) and [Consolidate\(String\)](#).

▀ **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public CompressionLevel CompressionLevel { get; set;
```

```
Public Property CompressionLevel As CompressionLevel
```

```
public:  
property CompressionLevel CompressionLevel {  
    CompressionLevel get ();  
    void set (CompressionLevel value);  
}
```

▀ **Remarks**

If the Directory table is modified, this mapping will be outdated until you close and reopen the install package.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Consolidate Method (mediaCabinet)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►  
**Consolidate(String)**

C#

Consolidates a package by combining and re-compressing all files into a single internal or external cabinet.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Consolidate(  
    string mediaCabinet  
)
```

```
Public Sub Consolidate ( _  
    mediaCabinet As String _  
)
```

```
public:  
void Consolidate(  
    String^ mediaCabinet  
)
```

### Parameters

**mediaCabinet (String)**

### Remarks

If an installation package was built from many merge modules, this method can somewhat decrease package size, complexity, and installation time.

This method will also convert a package with all or mostly uncompressse files into a package where all files are compressed.

If the package contains any not-yet-applied binary file patches (for example, a package generated by a call to [ApplyPatch\(PatchPackage, String\)](#)) then this method will apply the patches before compressing the updated files.

This method edits the database summary information and the File, Med and Patch tables as necessary to maintain a valid installation package.

The cabinet compression level used during re-cabbing can be configured with the [CompressionLevel](#) property.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Directories Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ► Directories

C# ▾

Gets a mapping from Directory keys to source/target paths.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public InstallPathMap Directories { get; }
```

```
Public ReadOnly Property Directories As InstallPathM
```

```
public:  
property InstallPathMap^ Directories {  
    InstallPathMap^ get ();  
}
```

## Remarks

If the Directory table is modified, this mapping will be outdated until you call [UpdateDirectories\(\)](#).

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# ExtractFiles Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►

**ExtractFiles()**

C# ▾

## Members

Icon	Member	Description
≡	<a href="#">ExtractFiles()</a>	Extracts all files to the <a href="#">WorkingDirectory</a> . The files are extracted to the relative directory matching their <a href="#">SourcePath</a> .
≡	<a href="#">ExtractFiles(IICollection&lt;String&gt;)</a>	Extracts a specified list of files to the <a href="#">WorkingDirectory</a> . The files are extracted to the relative directory matching their <a href="#">SourcePath</a> .

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation

## ExtractFiles Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►

### ExtractFiles()

C#

Extracts all files to the [WorkingDirectory](#). The files are extracted to the relative directory matching their [SourcePath](#).

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void ExtractFiles()
```

```
Public Sub ExtractFiles
```

```
public:  
void ExtractFiles()
```

#### Remarks

If any files have the uncompressed attribute, they will be copied from the [SourceDirectory](#).

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

## Deployment Tools Foundation ExtractFiles Method (fileKeys)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►

### ExtractFiles(ICollection<String>)

C#

Extracts a specified list of files to the [WorkingDirectory](#). The files are extracted to the relative directory matching their [SourcePath](#).

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void ExtractFiles(  
    ICollection<string> fileKeys  
)
```

```
Public Sub ExtractFiles ( _  
    fileKeys As ICollection(Of String) _  
)
```

```
public:  
void ExtractFiles(  
    ICollection<String>^ fileKeys  
)
```

#### Parameters

##### fileKeys (ICollection<String>)

List of file key strings to extract

#### Remarks

If any files have the uncompressed attribute, they will be copied from the [SourceDirectory](#).

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

# Deployment Tools Foundation **Files Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►

## Files

C# ▾

Gets a mapping from File keys to source/target paths.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public InstallPathMap Files { get; }
```

```
Public ReadOnly Property Files As InstallPathMap
```

```
public:  
property InstallPathMap^ Files {  
    InstallPathMap^ get ();  
}
```

### Remarks

If the File, Component, or Directory tables are modified, this mapping may be outdated until you call [UpdateDirectories\(\)](#).

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## FindFiles Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►

FindFiles()

C# ▾

### Members

Icon	Member	Description
≡	<a href="#">FindFiles(String)</a>	Gets the list of file keys that have the specified long file name.
≡	<a href="#">FindFiles(Regex)</a>	Gets the list of file keys whose long file names match a specified regular-expression search pattern.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

## Deployment Tools Foundation **FindFiles Method (longFileName)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►

**FindFiles(String)**

C# ▾

Gets the list of file keys that have the specified long file name.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string[] FindFiles(  
    string longFileName  
)
```

```
Public Function FindFiles ( _  
    longFileName As String _  
) As String()
```

```
public:  
array<String^>^ FindFiles(  
    String^ longFileName  
)
```

### Parameters

#### **longFileName (String)**

File name to search for (case-insensitive)

### Return Value

Array of file keys, or a 0-length array if none are found

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

## Deployment Tools Foundation **FindFiles Method (pattern)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►

**FindFiles(Regex)**

C# ▾

Gets the list of file keys whose long file names match a specified regular expression search pattern.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string[] FindFiles(  
    Regex pattern  
)
```

```
Public Function FindFiles ( _  
    pattern As Regex _  
) As String()
```

```
public:  
array<String^>^ FindFiles(  
    Regex^ pattern  
)
```

### Parameters

***pattern* (Regex)**

Regular expression search pattern

### Return Value

Array of file keys, or a 0-length array if none are found

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## LogMessage Method (format, args)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►  
**LogMessage(String, Object[])**

C# ▾

Sends a message to the [Message](#) event-handler.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
protected void LogMessage(  
    string format,  
    params Object[] args  
)
```

```
Protected Sub LogMessage ( _  
    format As String, _  
    ParamArray args As Object() _  
)
```

```
protected:  
void LogMessage(  
    String^ format,  
    ... array<Object^>^ args  
)
```

### Parameters

#### ***format* (String)**

Message string, containing 0 or more format items

#### ***args* ( Object [] )**

Items to be formatted

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0  
(3.5.1623.0)

Deployment Tools Foundation

## Message Event

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►

### Message

C#

Handle this event to receive status messages when operations are performed on the install package.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public event InstallPackageMessageHandler Message
```

```
Public Event Message As InstallPackageMessageHandler
```

```
public:  
    event InstallPackageMessageHandler^ Message {  
        void add (InstallPackageMessageHandler^ value);  
        void remove (InstallPackageMessageHandler^ value);  
    }
```

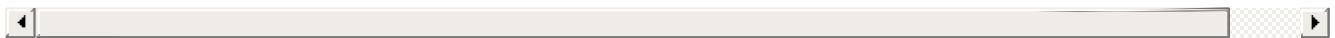
#### Examples

```
installPackage.Message += new  
InstallPackageMessageHandler(Console.WriteLine)
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

# Property Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►

## Property

C# ▾

Accessor for getting and setting properties of the InstallPackage database.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public InstallPackageProperties Property { get; }
```

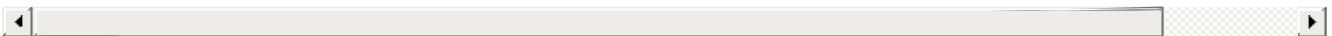
```
Public ReadOnly Property Property As InstallPackageP
```

```
public:  
property InstallPackageProperties^ Property {  
    InstallPackageProperties^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation **SourceDirectory Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►

## SourceDirectory

C# ▾

Gets or sets the location to obtain source files and cabinets when extracting or updating files in the working directory. This is often the location of an original copy of the package that is not meant to be modified.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string SourceDirectory { get; set; }
```

```
Public Property SourceDirectory As String
```

```
public:  
property String^ SourceDirectory {  
    String^ get ();  
    void set (String^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## UpdateDirectories Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►

### UpdateDirectories()

C# ▾

Rebuilds the cached directory structure information accessed by the **Directories** and **Files** properties. This should be done after modifying the File, Component, or Directory tables, or else the cached information may no longer be accurate.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void UpdateDirectories()
```

```
Public Sub UpdateDirectories
```

```
public:  
void UpdateDirectories()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation **UpdateFiles Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►

**UpdateFiles()**

C# ▾

▀ **Members**

Icon	Member	Description
≡ ♦	<a href="#">UpdateFiles()</a>	Updates the install package with new files from the <a href="#">WorkingDirectory</a> . The files must be in the relative directory matching their <a href="#">SourcePath</a> . This method re-compresses and packages the files if necessary, and also updates the following data: File.FileSize, File.Version, File.Language, MsiFileHash.HashPart*
≡ ♦	<a href="#">UpdateFiles(ICollection&lt;String&gt;)</a>	Updates the install package with new files from the <a href="#">WorkingDirectory</a> . The files must be in the relative directory matching their <a href="#">SourcePath</a> . This method re-compresses and packages the files if necessary, and also updates the following data: File.FileSize, File.Version, File.Language, MsiFileHash.HashPart?.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0  
(3.5.1623.0)

## Deployment Tools Foundation **UpdateFiles Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►

### **UpdateFiles()**

C# ▾

Updates the install package with new files from the [WorkingDirectory](#). The files must be in the relative directory matching their [SourcePath](#). The method re-compresses and packages the files if necessary, and also updates the following data: File.FileSize, File.Version, File.Language, MsiFileHash.HashPart\*

#### **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public void UpdateFiles()
```

```
Public Sub UpdateFiles
```

```
public:  
void UpdateFiles()
```

#### **Remarks**

The cabinet compression level used during re-cabbing can be configured with the [CompressionLevel](#) property.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

## Deployment Tools Foundation **UpdateFiles Method (fileKeys)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►

### UpdateFiles(ICollection<String>)

C# ▾

Updates the install package with new files from the [WorkingDirectory](#). The files must be in the relative directory matching their [SourcePath](#). The method re-compresses and packages the files if necessary, and also updates the following data: File.FileSize, File.Version, File.Language, MsiFileHash.HashPart?.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void UpdateFiles(  
    ICollection<string> fileKeys  
)
```

```
Public Sub UpdateFiles ( _  
    fileKeys As ICollection(Of String) _  
)
```

```
public:  
void UpdateFiles(  
    ICollection<String>^ fileKeys  
)
```

### Parameters

#### *fileKeys* ([ICollection<String>](#))

List of file key strings to update

### Remarks

This method does not change the media structure of the package, so it may require extracting and re-compressing a large cabinet just to update one file.

The cabinet compression level used during re-cabbing can be configured with the [CompressionLevel](#) property.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## WorkingDirectory Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPackage ►

### WorkingDirectory

C# ▾

Gets or sets the location where files will be extracted to/updated from.  
Also the location where a temporary folder is created during some operations.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string WorkingDirectory { get; set; }
```

```
Public Property WorkingDirectory As String
```

```
public:  
property String^ WorkingDirectory {  
    String^ get ();  
    void set (String^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

# InstallPackageMessageHandler Delegate

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ►  
**InstallPackageMessageHandler**

C# ▾

Handles status messages generated when operations are performed on an [InstallPackage](#) or [PatchPackage](#).

## - Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public delegate void InstallPackageMessageHandler(  
    string format,  
    params Object[] args  
)
```

```
Public Delegate Sub InstallPackageMessageHandler ( _  
    format As String, _  
    ParamArray args As Object() _  
)
```

```
public delegate void InstallPackageMessageHandler(  
    String^ format,  
    ... array<Object^>^ args  
)
```

## - Parameters

***format* (String)**

***args* ( Object [] )**

## - Examples

```
installPackage.Message += new  
InstallPackageMessageHandler(Console.WriteLine
```

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **InstallPackageProperties** Class

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ►  
**InstallPackageProperties**

C# ▾

Accessor for getting and setting properties of the [InstallPackage](#) database.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public class InstallPackageProperties
```

```
Public Class InstallPackageProperties
```

```
public ref class InstallPackageProperties
```

## Members

All Members	Methods	Properties	
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static	<input checked="" type="checkbox"/> Declared <input checked="" type="checkbox"/> Inherited

Icon	Member	Description
≡	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
💡	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
≡	<a href="#">GetHashCode()</a>	Serves as a hash function for a

		particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.
		(Inherited from <a href="#">Object</a> .)
≡	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance (Inherited from <a href="#">Object</a> .)
☰	<a href="#">Item[String]</a>	Gets or sets a property in the database. When getting a property that does not exist in the database, an empty string is returned. To remove a property from the database, set it to an empty string.
💡	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> . (Inherited from <a href="#">Object</a> .)
≡	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> . (Inherited from <a href="#">Object</a> .)

## [-] Inheritance Hierarchy

[Object](#)

└ **InstallPackageProperties**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Item Property (name)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ►  
InstallPackageProperties ► Item[String]

C# ▾

Gets or sets a property in the database. When getting a property that does not exist in the database, an empty string is returned. To remove a property from the database, set it to an empty string.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string this[  
    string name  
] { get; set; }
```

```
Public Default Property Item ( _  
    name As String _  
) As String
```

```
public:  
property String^ default[String^ name] {  
    String^ get (String^ name);  
    void set (String^ name, String^ value);  
}
```

## Parameters

*name* (String)

## Remarks

This has the same results as direct SQL queries on the Property table; it's only meant to be a more convenient way of access.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0  
(3.5.1623.0)

# Deployment Tools Foundation **InstallPath Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► **InstallPath**

C#

Represents the installation path of a file or directory from an installer product database.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public class InstallPath
```

```
Public Class InstallPath
```

```
public ref class InstallPath
```

## Members

All Members	Constructors	Methods	Properties
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static	<input checked="" type="checkbox"/> Declared <input checked="" type="checkbox"/> Inherited
Icon	Member	Description	
	<a href="#">InstallPath(String)</a>	Creates a new InstallPath, specifying a filename.	
	<a href="#">InstallPath(String, Boolean)</a>	Creates a new InstallPath, parsing out either the short or long filename.	
	<a href="#">ChildPaths</a>	Gets the set of child paths if this InstallPath object represents a directory.	

	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance  (Inherited from <a href="#">Object</a> .)
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">ParentPath</a>	Gets the path of the parent directory.
	<a href="#">SourceName</a>	Gets or sets the source name of the InstallPath.
	<a href="#">SourcePath</a>	Gets the full source path.
	<a href="#">TargetName</a>	Gets or sets the target name of the

		install path.
	<a href="#">TargetPath</a>	Gets the full target path.
	<a href="#">ToString()</a>	Gets the full source path. (Overrides <a href="#">Object.ToString()</a> .)

## Inheritance Hierarchy

[Object](#)

└ **InstallPath**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **InstallPath Constructor**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPath ►

**InstallPath()**

C# ▾

▀ **Members**

Icon	Member	Description
≡ ♪	<a href="#">InstallPath(String)</a>	Creates a new InstallPath, specifying a filename.
≡ ♪	<a href="#">InstallPath(String, Boolean)</a>	Creates a new InstallPath, parsing out either the short or long filename.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation

## InstallPath Constructor (name)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPath ►

InstallPath(String)

C# ▾

Creates a new InstallPath, specifying a filename.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public InstallPath(  
    string name  
)
```

```
Public Sub New ( _  
    name As String _  
)
```

```
public:  
InstallPath(  
    String^ name  
)
```

### Parameters

#### ***name* (String)**

The name of the file or directory. Not a full path.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## InstallPath Constructor (name, useShortNames)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ► InstallPath ►  
**InstallPath(String, Boolean)**

C# ▾

Creates a new InstallPath, parsing out either the short or long filename.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public InstallPath(  
    string name,  
    bool useShortNames  
)
```

```
Public Sub New ( _  
    name As String, _  
    useShortNames As Boolean _  
)
```

```
public:  
InstallPath(  
    String^ name,  
    bool useShortNames  
)
```

### Parameters

#### **name (String)**

The name of the file or directory, in short|long syntax for a filename or targetshort|targetlong:sourceshort|sourcelong syntax for a directory.

#### **useShortNames (Boolean)**

true to parse the short part of the combined filename; false to parse the long part

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ChildPaths Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPath ►

### ChildPaths

C#

Gets the set of child paths if this InstallPath object represents a directory.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public InstallPathCollection ChildPaths { get; }
```

```
Public ReadOnly Property ChildPaths As InstallPathCo
```

```
public:  
property InstallPathCollection^ ChildPaths {  
    InstallPathCollection^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation

# ParentPath Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPath ►

## ParentPath

C# ▾

Gets the path of the parent directory.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public InstallPath ParentPath { get; }
```

```
Public ReadOnly Property ParentPath As InstallPath
```

```
public:  
property InstallPath^ ParentPath {  
    InstallPath^ get();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation

# SourceName Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPath ►

## SourceName

C# ▾

Gets or sets the source name of the InstallPath.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string SourceName { get; set; }
```

```
Public Property SourceName As String
```

```
public:  
property String^ SourceName {  
    String^ get ();  
    void set (String^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation

# SourcePath Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPath ►

## SourcePath

C# ▾

Gets the full source path.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string SourcePath { get; set; }
```

```
Public Property SourcePath As String
```

```
public:  
property String^ SourcePath {  
    String^ get ();  
    void set (String^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

# Deployment Tools Foundation **TargetName** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPath ►

## TargetName

C# ▾

Gets or sets the target name of the install path.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string TargetName { get; set; }
```

```
Public Property TargetName As String
```

```
public:  
property String^ TargetName {  
    String^ get ();  
    void set (String^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

# Deployment Tools Foundation **TargetPath Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPath ►

## TargetPath

C# ▾

Gets the full target path.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string TargetPath { get; set; }
```

```
Public Property TargetPath As String
```

```
public:  
property String^ TargetPath {  
    String^ get ();  
    void set (String^ value);  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

## Deployment Tools Foundation **ToString** Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPath ►

### **ToString()**

C# ▾

Gets the full source path.

#### [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public override string ToString()
```

```
Public Overrides Function ToString As String
```

```
public:  
virtual String^ ToString() override
```

#### [-] Return Value

SourcePath

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **InstallPathCollection Class**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ►  
**InstallPathCollection**

C# ▾

Represents a collection of InstallPaths that are the child paths of the same parent directory.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public class InstallPathCollection : IList<InstallPa  
    ICollection<InstallPath>, IEnumerable<Instal
```

```
Public Class InstallPathCollection _  
    Implements IList(Of InstallPath), ICollection  
    IEnumerable(Of InstallPath), IEnumerable
```

```
public ref class InstallPathCollection : IList<Insta  
    ICollection<InstallPath^>, IEnumerable<Insta
```

### Members

All Members	Methods	Properties	Events
<input checked="" type="checkbox"/> Public <input checked="" type="checkbox"/> Protected		<input checked="" type="checkbox"/> Instance <input checked="" type="checkbox"/> Static	<input checked="" type="checkbox"/> Declared <input checked="" type="checkbox"/> Inherited

Icon	Member	Description
	Add(InstallPath)	Adds a new child path to the collection.
	Clear()	Removes all items from the collection.

	<a href="#">Contains(InstallPath)</a>	Tests if the collection contains a child path.
	<a href="#">CopyTo(InstallPath[], Int32)</a>	Copies the collection into an array.
	<a href="#">Count</a>	Gets the number of items in the collection.
	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetEnumerator()</a>	Gets an enumerator over all items in the collection.
	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance  (Inherited from <a href="#">Object</a> .)
	<a href="#">IndexOf(InstallPath)</a>	Gets the index of a child path in the collection.

	 <a href="#">Insert(Int32, InstallPath)</a>	Inserts a child path into the collection.
	 <a href="#">Item[Int32]</a>	Gets or sets the element at the specified index.
	 <a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	 <a href="#">Remove(InstallPath)</a>	Removes a child path to the collection.
	 <a href="#">RemoveAt(Int32)</a>	Removes an item from the collection.
	 <a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)

## Inheritance Hierarchy

[Object](#)

└ **InstallPathCollection**

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Add Method (item)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ►  
InstallPathCollection ► Add(InstallPath)

C# ▾

Adds a new child path to the collection.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Add(  
    InstallPath item  
)
```

```
Public Sub Add ( _  
    item As InstallPath _  
)
```

```
public:  
virtual void Add(  
    InstallPath^ item  
) sealed
```

### Parameters

#### *item* (InstallPath)

The InstallPath to add.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Clear Method**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ►  
InstallPathCollection ► **Clear()**

C# ▾

Removes all items from the collection.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public void Clear()
```

```
Public Sub Clear
```

```
public:  
virtual void Clear() sealed
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation Contains Method (item)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ►  
InstallPathCollection ► Contains(InstallPath)

C# ▾

Tests if the collection contains a child path.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool Contains(  
    InstallPath item  
)
```

```
Public Function Contains ( _  
    item As InstallPath _  
) As Boolean
```

```
public:  
virtual bool Contains(  
    InstallPath^ item  
) sealed
```

## Parameters

### item (InstallPath)

The InstallPath to search for.

## Return Value

true if the item is found; false otherwise

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **CopyTo Method (array, index)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ►  
InstallPathCollection ► **CopyTo(InstallPath[], Int32)**

C# ▾

Copies the collection into an array.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void CopyTo(  
    InstallPath[] array,  
    int index  
)
```

```
Public Sub CopyTo ( _  
    array As InstallPath(), _  
    index As Integer _  
)
```

```
public:  
virtual void CopyTo(  
    array<InstallPath^>^ array,  
    int index  
) sealed
```

### Parameters

#### **array ( InstallPath [] )**

The array to copy into.

#### **index ( Int32 )**

The starting index in the destination array.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation **Count Property**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ►  
InstallPathCollection ► Count

C# ▾

Gets the number of items in the collection.

Declaration Syntax

C#

Visual Basic

Visual C++

```
public int Count { get; }
```

```
Public ReadOnly Property Count As Integer
```

```
public:  
virtual property int Count {  
    int get () sealed;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)  
Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetEnumerator Method**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ►  
InstallPathCollection ► **GetEnumerator()**

C# ▾

Gets an enumerator over all items in the collection.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public IEnum<InstallPath> GetEnumerator()
```

```
Public Function GetEnumerator As IEnum<of Inst
```

```
public:  
virtual IEnum<InstallPath^>^ GetEnumerator() s
```

### [-] Return Value

An enumerator for the collection.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)



## Deployment Tools Foundation **IndexOf Method (item)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ►  
InstallPathCollection ► **IndexOf(InstallPath)**

C# ▾

Gets the index of a child path in the collection.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public int IndexOf(  
    InstallPath item  
)
```

```
Public Function IndexOf ( _  
    item As InstallPath _  
) As Integer
```

```
public:  
virtual int IndexOf(  
    InstallPath^ item  
) sealed
```

### Parameters

#### *item* ([InstallPath](#))

The InstallPath to search for.

### Return Value

The index of the item, or -1 if not found.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Insert Method (index, item)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ►  
InstallPathCollection ► **Insert(Int32, InstallPath)**

C# ▾

Inserts a child path into the collection.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Insert(  
    int index,  
    InstallPath item  
)
```

```
Public Sub Insert ( _  
    index As Integer, _  
    item As InstallPath _  
)
```

```
public:  
virtual void Insert(  
    int index,  
    InstallPath^ item  
) sealed
```

### Parameters

#### *index* ([Int32](#))

The insertion index.

#### *item* ([InstallPath](#))

The InstallPath to insert.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

# Deployment Tools Foundation **Item Property (index)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ►  
InstallPathCollection ► **Item[Int32]**

C# ▾

Gets or sets the element at the specified index.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public InstallPath this[  
    int index  
] { get; set; }
```

```
Public Default Property Item ( _  
    index As Integer _  
) As InstallPath
```

```
public:  
virtual property InstallPath^ default[int index] {  
    InstallPath^ get (int index) sealed;  
    void set (int index, InstallPath^ value) sea  
}
```

## Parameters

***index* (Int32)**

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)



## Deployment Tools Foundation Remove Method (item)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ►  
InstallPathCollection ► Remove(InstallPath)

C# ▾

Removes a child path to the collection.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool Remove(  
    InstallPath item  
)
```

```
Public Function Remove ( _  
    item As InstallPath _  
) As Boolean
```

```
public:  
virtual bool Remove(  
    InstallPath^ item  
) sealed
```

### Parameters

#### *item* (InstallPath)

The InstallPath to remove.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation RemoveAt Method (index)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ►  
InstallPathCollection ► RemoveAt(Int32)

C# ▾

Removes an item from the collection.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void RemoveAt(  
    int index  
)
```

```
Public Sub RemoveAt ( _  
    index As Integer _  
)
```

```
public:  
virtual void RemoveAt(  
    int index  
) sealed
```

### Parameters

#### index (Int32)

The index of the item to remove.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **InstallPathMap Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► **InstallPathMap**

C# ▾

Represents a mapping of install paths for all directories, components, or files in an installation database.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public class InstallPathMap : IDictionary<String, InstallPath>,  
    ICollection<KeyValuePair<String, InstallPath>>,  
    IEnumerable
```

```
Public Class InstallPathMap  
    Implements IDictionary(Of String, InstallPath),  
    ICollection(Of KeyValuePair(Of String, InstallPath)),  
    IEnumerable
```

```
public ref class InstallPathMap : IDictionary<String^>,  
    ICollection<KeyValuePair<String^>, InstallPath>,  
    IEnumerable
```

## Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon	Member	Description
	InstallPathMap()	Creates a new empty InstallPathMap.
	Add(String, InstallPath)	Sets an install path for a

		directory, component, or file key.
	<a href="#">BuildComponentPathMap(Database, InstallPathMap)</a>	Builds a mapping from Component keys to installation paths.
	<a href="#">BuildDirectoryPathMap(Database, Boolean)</a>	Builds a mapping from Directory keys to installation paths.
	<a href="#">BuildDirectoryPathMap(Database, Boolean, String, String)</a>	Builds a mapping of Directory keys to directory paths, specifying root directories for the source and target paths.
	<a href="#">BuildFilePathMap(Database, InstallPathMap, Boolean)</a>	Builds a mapping from File keys to installation paths.
	<a href="#">Clear()</a>	Removes all entries from the dictionary.
	<a href="#">ContainsKey(String)</a>	Tests whether a directory, component, or file key exists in the map.
	<a href="#">Count</a>	Gets the number of entries in the dictionary.

 <a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
 <a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
 <a href="#">GetEnumerator()</a>	Gets an enumerator over all entries in the dictionary.
 <a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table  (Inherited from <a href="#">Object</a> .)
 <a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
 <a href="#">Item[String]</a>	Gets or sets an install path for a directory, component, or file key.
 <a href="#">Keys</a>	Gets the collection of keys

		in the InstallPathMap. Depending on the type of InstallPathMap, they are all directory, component, or file key strings.
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Remove(String)</a>	Removes an install path from the map.
	<a href="#">SourcePaths</a>	Gets a mapping from keys to source paths.
	<a href="#">TargetPaths</a>	Gets a mapping from keys to target paths.
	<a href="#">ToString()</a>	Returns a <a href="#">String</a> that represents the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">TryGetValue(String, InstallPath)</a>	Attempts to get a value from the dictionary.
	<a href="#">Values</a>	Gets the collection of InstallPath values in the InstallPathMap.

## Inheritance Hierarchy

[Object](#)

 [InstallPathMap](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)



Deployment Tools Foundation **InstallPathMap Constructor**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPathMap ►

**InstallPathMap()**

C# ▾

Creates a new empty InstallPathMap.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public InstallPathMap()
```

```
Public Sub New
```

```
public:  
InstallPathMap()
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **Add Method (key, installPath)**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPathMap ►

### Add(String, InstallPath)

C# ▾

Sets an install path for a directory, component, or file key.

#### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public void Add(  
    string key,  
    InstallPath installPath  
)
```

```
Public Sub Add ( _  
    key As String,  
    installPath As InstallPath _  
)
```

```
public:  
virtual void Add(  
    String^ key,  
    InstallPath^ installPath  
) sealed
```

#### Parameters

##### **key (String)**

Depending on the type of InstallPathMap, this is the primary key from the either the Directory, Component, or File table.

##### **installPath (InstallPath)**

The install path of the key item.

#### Remarks

Changing an install path does not modify the Database used to generat

this InstallPathMap.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## BuildComponentPathMap Method (db, directoryPathMap)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ► InstallPathMap ►  
**BuildComponentPathMap(Database, InstallPathMap)**

C#

Builds a mapping from Component keys to installation paths.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public static InstallPathMap BuildComponentPathMap(  
    Database db,  
    InstallPathMap directoryPathMap  
)
```

```
Public Shared Function BuildComponentPathMap ( _  
    db As Database, _  
    directoryPathMap As InstallPathMap _  
) As InstallPathMap
```

```
public:  
static InstallPathMap^ BuildComponentPathMap(  
    Database^ db,  
    InstallPathMap^ directoryPathMap  
)
```

### Parameters

#### db (Database)

Installation database.

#### directoryPathMap (InstallPathMap)

Directory mapping returned by [BuildDirectoryPathMap\(Database, Boolean\)](#).

### Return Value

An InstallPathMap with the described mapping.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## BuildDirectoryPathMap Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPathMap ►

**BuildDirectoryPathMap()**

C# ▾

### Members

Icon	Member	Description
= 	<a href="#">BuildDirectoryPathMap(Database, Boolean)</a>	Builds a mapping from Directory keys to installation paths.
= 	<a href="#">BuildDirectoryPathMap(Database, Boolean, String, String)</a>	Builds a mapping of Directory keys to directory paths, specifying root directories for the source and target paths.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## BuildDirectoryPathMap Method (db, useShortNames)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ► InstallPathMap ►  
**BuildDirectoryPathMap(Database, Boolean)**

C# ▾

Builds a mapping from Directory keys to installation paths.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static InstallPathMap BuildDirectoryPathMap(  
    Database db,  
    bool useShortNames  
)
```

```
Public Shared Function BuildDirectoryPathMap ( _  
    db As Database, _  
    useShortNames As Boolean _  
) As InstallPathMap
```

```
public:  
static InstallPathMap^ BuildDirectoryPathMap(  
    Database^ db,  
    bool useShortNames  
)
```

### Parameters

#### db (Database)

Installation database.

#### useShortNames (Boolean)

true to use short directory names; false to use long names

### Return Value

An InstallPathMap with the described mapping.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## BuildDirectoryPathMap Method (db, useShortNames, sourceRootDir, targetRootDir)

Deployment Tools Foundation Namespaces ►  
[Microsoft.Deployment.WindowsInstaller.Package](#) ► [InstallPathMap](#) ►  
**BuildDirectoryPathMap(Database, Boolean, String, String)**

C#

Builds a mapping of Directory keys to directory paths, specifying root directories for the source and target paths.

### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public static InstallPathMap BuildDirectoryPathMap(  
    Database db,  
    bool useShortNames,  
    string sourceRootDir,  
    string targetRootDir  
)
```

```
Public Shared Function BuildDirectoryPathMap ( _  
    db As Database, _  
    useShortNames As Boolean, _  
    sourceRootDir As String, _  
    targetRootDir As String _  
) As InstallPathMap
```

```
public:  
static InstallPathMap^ BuildDirectoryPathMap(  
    Database^ db,  
    bool useShortNames,  
    String^ sourceRootDir,  
    String^ targetRootDir  
)
```

### - Parameters

***db*** ([Database](#))

Database containing the Directory table.

***useShortNames*** ([Boolean](#))

true to use short directory names; false to use long names

***sourceRootDir*** ([String](#))

The root directory path of all source paths, or null to leave them relative.

***targetRootDir*** ([String](#))

The root directory path of all source paths, or null to leave them relative.

**[-] Return Value**

An InstallPathMap with the described mapping.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## BuildFilePathMap Method (db, componentPathMap, useShortNames)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ► InstallPathMap ►  
**BuildFilePathMap(Database, InstallPathMap, Boolean)**

C# ▾

Builds a mapping from File keys to installation paths.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public static InstallPathMap BuildFilePathMap(  
    Database db,  
    InstallPathMap componentPathMap,  
    bool useShortNames  
)
```

```
Public Shared Function BuildFilePathMap ( _  
    db As Database, _  
    componentPathMap As InstallPathMap, _  
    useShortNames As Boolean _  
) As InstallPathMap
```

```
public:  
static InstallPathMap^ BuildFilePathMap(  
    Database^ db,  
    InstallPathMap^ componentPathMap,  
    bool useShortNames  
)
```

### Parameters

**db (Database)**

Installation database.

**componentPathMap (InstallPathMap)**

Component mapping returned by

`BuildComponentPathMap(Database, InstallPathMap).`

**`useShortNames (Boolean)`**

true to use short file names; false to use long names

**Return Value**

An `InstallPathMap` with the described mapping.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

C# ▾

Removes all entries from the dictionary.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public void Clear()
```

```
Public Sub Clear
```

```
public:  
virtual void Clear() sealed
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation ContainsKey Method (key)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPathMap ►

## ContainsKey(String)

C#

Tests whether a directory, component, or file key exists in the map.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public bool ContainsKey(  
    string key  
)
```

```
Public Function ContainsKey ( _  
    key As String _  
) As Boolean
```

```
public:  
virtual bool ContainsKey(  
    String^ key  
) sealed
```

### Parameters

#### key (String)

Depending on the type of InstallPathMap, this is the primary key from the either the Directory, Component, or File table.

### Return Value

true if the key is found; false otherwise

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **Count Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPathMap ►

**Count**

C# ▾

Gets the number of entries in the dictionary.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public int Count { get; }
```

```
Public ReadOnly Property Count As Integer
```

```
public:  
virtual property int Count {  
    int get () sealed;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation **GetEnumerator Method**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ► InstallPathMap ►  
**GetEnumerator()**

C# ▾

Gets an enumerator over all entries in the dictionary.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public IEnumarator<KeyValuePair<string, InstallPath>
```

```
Public Function GetEnumerator As IEnumarator(Of KeyV
```

```
public:  
virtual IEnumarator<KeyValuePair<String^, InstallPat
```

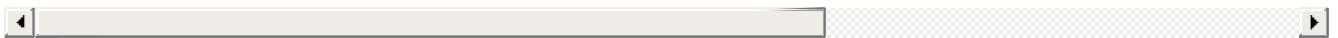
[-] **Return Value**

An enumerator for the dictionary.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation Item Property (key)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPathMap ►

## Item[String]

C#

Gets or sets an install path for a directory, component, or file key.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public InstallPath this[  
    string key  
] { get; set; }
```

```
Public Default Property Item ( _  
    key As String _  
) As InstallPath
```

```
public:  
virtual property InstallPath^ default[String^ key] {  
    InstallPath^ get (String^ key) sealed;  
    void set (String^ key, InstallPath^ value) s  
}
```

### Parameters

#### key (String)

Depending on the type of InstallPathMap, this is the primary key from the either the Directory, Component, or File table.

### Remarks

Changing an install path does not modify the Database used to generate this InstallPathMap.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0  
(3.5.1623.0)



# Deployment Tools Foundation Keys Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPathMap ► Keys

C# ▾

Gets the collection of keys in the InstallPathMap. Depending on the type of InstallPathMap, they are all directory, component, or file key strings.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public ICollection<string> Keys { get; }
```

```
Public ReadOnly Property Keys As ICollection(Of String)
```

```
public:  
virtual property ICollection<String^>^ Keys {  
    ICollection<String^>^ get () sealed;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation Remove Method (key)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPathMap ►

### Remove(String)

C# ▾

Removes an install path from the map.

#### - Declaration Syntax

C#

Visual Basic

Visual C++

```
public bool Remove(  
    string key  
)
```

```
Public Function Remove ( _  
    key As String _  
) As Boolean
```

```
public:  
virtual bool Remove(  
    String^ key  
) sealed
```

#### - Parameters

##### key (String)

Depending on the type of InstallPathMap, this is the primary key from the either the Directory, Component, or File table.

#### - Return Value

true if the item was removed, false if it did not exist

#### - Remarks

Changing an install path does not modify the Database used to generate this InstallPathMap.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **SourcePaths** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPathMap ►

## SourcePaths

C# ▾

Gets a mapping from keys to source paths.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public IDictionary<string, string> SourcePaths { get
```

```
Public ReadOnly Property SourcePaths As IDictionary(
```

```
public:  
property IDictionary<String^, String^>^ SourcePaths  
    IDictionary<String^, String^>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)



# Deployment Tools Foundation **TargetPaths** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPathMap ►

## TargetPaths

C# ▾

Gets a mapping from keys to target paths.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public IDictionary<string, string> TargetPaths { get
```

```
Public ReadOnly Property TargetPaths As IDictionary(
```

```
public:  
property IDictionary<String^, String^>^ TargetPaths  
    IDictionary<String^, String^>^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)



## Deployment Tools Foundation TryGetValue Method (key, value)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPathMap ►

### TryGetValue(String, InstallPath)

C# ▾

Attempts to get a value from the dictionary.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public bool TryGetValue(  
    string key,  
    out InstallPath value  
)
```

```
Public Function TryGetValue ( _  
    key As String, _  
    <OutAttribute> ByRef value As InstallPath _  
) As Boolean
```

```
public:  
virtual bool TryGetValue(  
    String^ key,  
    [OutAttribute] InstallPath^% value  
) sealed
```

#### Parameters

##### **key (String)**

The key to lookup.

##### **value ( InstallPath )**

Receives the value, or null if the key was not found.

#### Return Value

True if the value was found, else false.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **Values Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► InstallPathMap ►

## Values

C#

Gets the collection of InstallPath values in the InstallPathMap.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public ICollection<InstallPath> Values { get; }
```

```
Public ReadOnly Property Values As ICollection(Of In
```

```
public:  
virtual property ICollection<InstallPath^>^ Values {  
    ICollection<InstallPath^>^ get () sealed;  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)



# Deployment Tools Foundation **PatchPackage Class**

Deployment Tools Foundation Namespaces ►

[Microsoft.Deployment.WindowsInstaller.Package](#) ► **PatchPackage**

C#

Provides access to convenient properties and operations on a patch package (.MSP).

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public class PatchPackage : Database
```

```
Public Class PatchPackage
    Inherits Database
```

```
public ref class PatchPackage : public Database
```

## Members

All Members

Constructors

Methods

Properties

Events

Public

Instance

Declared

Protected

Static

Inherited

Icon

Icon	Member	Description
	<a href="#">PatchPackage(String)</a>	Creates a new patch package object; opening the patch database in read-only mode.
	<a href="#">ApplyTransform(String)</a>	Apply a transform to the database, suppressing any error conditions specified by the transform's summary information.

		(Inherited from <a href="#">Database</a> .)
	<a href="#">ApplyTransform(String, TransformErrors)</a>	Applies a transform to the database, specifying error conditions to suppress.  (Inherited from <a href="#">Database</a> .)
	<a href="#">Close()</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidOperationException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">CodePage</a>	Gets or sets the code page of the Database.  (Inherited from <a href="#">Database</a> .)
	<a href="#">Commit()</a>	Finalizes the persisting form of the database. All persistent data is written to the writeable database, and no temporary columns or rows are written.  (Inherited from <a href="#">Database</a> .)
	<a href="#">CountRows(String)</a>	Gets the count of all rows in the table.  (Inherited from

		Database.)
≡	CountRows(String, String)	Gets the count of all rows in the table that satisfy a given condition.  (Inherited from Database.)
≡	CreateObjRef(Type)	Creates an object that contains all the relevant information required to generate a proxy used to communicate with a remote object.  (Inherited from MarshalByRefObject.)
≡	CreateRecord(Int32)	Creates a new record object with the requested number of fields.  (Inherited from Database.)
≡	CreateTransformSummaryInfo(Database, String, TransformErrors, TransformValidations)	Creates and populates the summary information stream of an existing transform file, and fills in the properties with the base and reference ProductCode and ProductVersion.  (Inherited from Database.)
≡	DeleteOnClose(String)	Schedules a file or

		directory for deletion after the database handle is closed.
		(Inherited from <a href="#">Database</a> .)
💡	<a href="#">Dispose(Boolean)</a>	Closes the database handle. After closing a handle, further method calls may throw <a href="#">InvalidHandleException</a> .
		(Inherited from <a href="#">Database</a> .)
💡	<a href="#">Dispose()</a>	Closes the handle. After closing a handle, further method calls may throw an <a href="#">InvalidHandleException</a> .
		(Inherited from <a href="#">InstallerHandle</a> .)
💡	<a href="#">Equals(Object)</a>	Tests whether this handle object is equal to another handle object. Two handle objects are equal if their types are the same and their native integer handles are the same.
		(Inherited from <a href="#">InstallerHandle</a> .)
💡	<a href="#">Execute(String, Object[])</a>	Executes the query specified by a SQL string. The query may not be a SELECT

		statement.  (Inherited from <a href="#">Database</a> .)
≡◊	<a href="#">Execute(String, Record)</a>	Executes the query specified by a SQL string. The query may not be a SELECT statement.  (Inherited from <a href="#">Database</a> .)
≡◊	<a href="#">ExecuteIntegerQuery(String, Object[])</a>	Executes the specified SQL SELECT query and returns all results as integers.  (Inherited from <a href="#">Database</a> .)
≡◊	<a href="#">ExecuteIntegerQuery(String, Record)</a>	Executes the specified SQL SELECT query and returns all results as integers.  (Inherited from <a href="#">Database</a> .)
≡◊	<a href="#">ExecuteQuery(String, Object[])</a>	Executes the specified SQL SELECT query and returns all results.  (Inherited from <a href="#">Database</a> .)
≡◊	<a href="#">ExecuteQuery(String, Record)</a>	Executes the specified SQL SELECT query and returns all results.  (Inherited from

		<a href="#">Database.)</a>
≡	<a href="#">ExecuteScalar(String, Object[])</a>	Executes the specified SQL SELECT query and returns a single result.  (Inherited from <a href="#">Database.</a> )
≡	<a href="#">ExecuteScalar(String, Record)</a>	Executes the specified SQL SELECT query and returns a single result.  (Inherited from <a href="#">Database.</a> )
≡	<a href="#">ExecuteStringQuery(String, Object[])</a>	Executes the specified SQL SELECT query and returns all results as strings.  (Inherited from <a href="#">Database.</a> )
≡	<a href="#">ExecuteStringQuery(String, Record)</a>	Executes the specified SQL SELECT query and returns all results as strings.  (Inherited from <a href="#">Database.</a> )
≡	<a href="#">Export(String, String)</a>	Copies the structure and data from a specified table to a text archive file.  (Inherited from <a href="#">Database.</a> )
≡		

	<code>ExportAll(String)</code>	Exports all database tables, streams, and summary information to archive files.  (Inherited from <a href="#">Database</a> .)
	<code>ExtractTransform(String, String)</code>	Extracts a transform (.MST) from a patch package.
	<code>FilePath</code>	Gets the file path the Database was originally opened from, or null if not known.  (Inherited from <a href="#">Database</a> .)
	<code>Finalize()</code>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
	<code>GenerateTransform(Database, String)</code>	Creates a transform that, when applied to the object database, results in the reference database.  (Inherited from <a href="#">Database</a> .)
	<code>GetHashCode()</code>	Gets a hash value for

		the handle object.  (Inherited from <a href="#">InstallerHandle</a> .)
≡	<a href="#">GetLifetimeService()</a>	Retrieves the current lifetime service object that controls the lifetime policy for this instance  (Inherited from <a href="#">MarshalByRefObject</a> .)
≡	<a href="#">GetReplacedPatchCodes()</a>	Gets the list of patch codes that are replaced by this patch package
≡	<a href="#">GetTargetProductCodes()</a>	Gets the list of product codes of products targeted by this patch package.
≡	<a href="#">GetTransformInfo(String)</a>	Gets information about transforms included in the patch package.
≡	<a href="#">GetTransforms()</a>	Gets the names of the transforms included in the patch package.
≡	<a href="#">GetTransforms(Boolean)</a>	Gets the names of the transforms included in the patch package.

	<a href="#">GetTransformsInfo()</a>	Gets information about the transforms included in the patch package.
	<a href="#">GetTransformsInfo(Boolean)</a>	Gets information about the transforms included in the patch package.
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetValidTransforms(InstallPackage)</a>	Analyzes the transforms included in the patch package to find the ones that are applicable to an install package.
	<a href="#">Handle</a>	Gets the native integer handle.  (Inherited from <a href="#">InstallerHandle</a> .)
	<a href="#">Import(String)</a>	Imports a database table from a text archive file, dropping any existing table.  (Inherited from <a href="#">Database</a> .)
	<a href="#">ImportAll(String)</a>	Imports all database tables, streams, and summary information

		from archive files.
		(Inherited from <a href="#">Database</a> .)
 <a href="#">InitializeLifetimeService()</a>		Obtains a lifetime service object to control the lifetime policy for this instance.  (Inherited from <a href="#">MarshalByRefObject</a> .)
 <a href="#">IsClosed</a>		Checks if the handle is closed. When closed, method calls on the handle object may throw an <a href="#">InvalidHandleException</a> .  (Inherited from <a href="#">InstallerHandle</a> .)
 <a href="#">IsColumnPersistent(String, String)</a>		Checks whether a table contains a persistent column with a given name.  (Inherited from <a href="#">Database</a> .)
 <a href="#">IsReadOnly</a>		Gets a boolean value indicating whether this database was opened in read-only mode.  (Inherited from <a href="#">Database</a> .)
 <a href="#">IsTablePersistent(String)</a>		Checks whether a table exists and is persistent in the database.

		(Inherited from Database.)
≡	IsTransformValid(String)	<p>Checks whether a transform is valid for this Database, according to its validation data and flags.</p> <p>(Inherited from Database.)</p>
≡	IsTransformValid(SummaryInfo)	<p>Checks whether a transform is valid for this Database, according to its SummaryInfo data.</p> <p>(Inherited from Database.)</p>
💡	LogMessage(String, Object[])	Sends a message to the <a href="#">Message</a> event-handler.
💡	MemberwiseClone(Boolean)	<p>Creates a shallow copy of the current <a href="#">MarshalByRefObject</a> object.</p> <p>(Inherited from <a href="#">MarshalByRefObject</a>.)</p>
💡	MemberwiseClone()	<p>Creates a shallow copy of the current <a href="#">Object</a>.</p> <p>(Inherited from <a href="#">Object</a>)</p>
≡	Merge(Database, String)	Merges another

		database with this database.
		(Inherited from <a href="#">Database</a> .)
	<a href="#">Merge(Database)</a>	Merges another database with this database.
		(Inherited from <a href="#">Database</a> .)
	<a href="#">Message</a>	Handle this event to receive status messages when operations are performed on the patch package.
	<a href="#">OpenMode</a>	Gets the open mode for the database.
		(Inherited from <a href="#">Database</a> .)
	<a href="#">OpenView(String, Object[])</a>	Gets a View object representing the query specified by a SQL string.
		(Inherited from <a href="#">Database</a> .)
	<a href="#">PatchCode</a>	Gets the patch code (GUID) of the patch package.
	<a href="#">SummaryInfo</a>	Gets the SummaryInfo

		object for this database that can be used to examine and modify properties to the summary information stream.
		(Inherited from <a href="#">Database</a> .)
 <a href="#">Tables</a>	Gets the collection of tables in the Database	(Inherited from <a href="#">Database</a> .)
 <a href="#">ToString()</a>	Returns the file path of this database, or the handle value if a file path was not specified	(Inherited from <a href="#">Database</a> .)
 <a href="#">ViewTransform(String)</a>	Apply a transform to the database, recording the changes in the "_TransformView" table	(Inherited from <a href="#">Database</a> .)

## Inheritance Hierarchy

[Object](#)

-  [MarshalByRefObject](#)
-  [InstallerHandle](#)
-  [Database](#)
-  [PatchPackage](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

# PatchPackage Constructor (*packagePath*)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ► PatchPackage ►  
**PatchPackage(String)**

C# ▾

Creates a new patch package object; opening the patch database in read-only mode.

## [-] Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public PatchPackage(  
    string packagePath  
)
```

```
Public Sub New ( _  
    packagePath As String _  
)
```

```
public:  
PatchPackage(  
    String^ packagePath  
)
```

## [-] Parameters

### *packagePath* ([String](#))

Path to the patch package (.MSP)

## [-] Remarks

The PatchPackage object only opens the patch database in read-only mode, because transforms (sub-storages) cannot be read if the database is open in read-write mode.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## ExtractTransform Method (transform, extractFile)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ► PatchPackage ►  
**ExtractTransform(String, String)**

C# ▾

Extracts a transform (.MST) from a patch package.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public void ExtractTransform(  
    string transform,  
    string extractFile  
)
```

```
Public Sub ExtractTransform ( _  
    transform As String, _  
    extractFile As String _  
)
```

```
public:  
void ExtractTransform(  
    String^ transform,  
    String^ extractFile  
)
```

### Parameters

#### **transform (String)**

Name of the transform to extract; this may optionally be a special transform prefixed by "#"

#### **extractFile (String)**

Location where the transform will be extracted

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetReplacedPatchCodes Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► PatchPackage ►

### **GetReplacedPatchCodes()**

C#

Gets the list of patch codes that are replaced by this patch package.

#### **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public string[] GetReplacedPatchCodes()
```

```
Public Function GetReplacedPatchCodes As String()
```

```
public:  
array<String^>^ GetReplacedPatchCodes()
```

#### **Return Value**

Array of replaced patch codes (GUIDs)

#### **Remarks**

The list of replaced patch codes is stored in the RevisionNumber field of the patch summary information.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation **GetTargetProductCodes Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► PatchPackage ►

**GetTargetProductCodes()**

C# ▾

Gets the list of product codes of products targeted by this patch package.

**Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public string[] GetTargetProductCodes()
```

```
Public Function GetTargetProductCodes As String()
```

```
public:  
array<String^>^ GetTargetProductCodes()
```

**Return Value**

Array of product codes (GUIDs)

**Remarks**

The list of target product codes is stored in the Template field of the package's summary information.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation

## GetTransformInfo Method (transform)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ► PatchPackage ►  
**GetTransformInfo(String)**

C#

Gets information about a transforms included in the patch package.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public TransformInfo GetTransformInfo(  
    string transform  
)
```

```
Public Function GetTransformInfo ( _  
    transform As String _  
) As TransformInfo
```

```
public:  
TransformInfo^ GetTransformInfo(  
    String^ transform  
)
```

### Parameters

#### *transform* (String)

Name of the transform to extract; this may optionally be a special transform prefixed by "#"

### Return Value

Information about the transform

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation **GetTransforms Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► PatchPackage ►

**GetTransforms()**

C# ▾

▀ **Members**

Icon	Member	Description
≡	<a href="#">GetTransforms()</a>	Gets the names of the transforms included in the patch package.
≡	<a href="#">GetTransforms(Boolean)</a>	Gets the names of the transforms included in the patch package.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

## Deployment Tools Foundation **GetTransforms Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► PatchPackage ►

### **GetTransforms()**

C#

Gets the names of the transforms included in the patch package.

#### **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public string[] GetTransforms()
```

```
Public Function GetTransforms As String()
```

```
public:  
array<String^>^ GetTransforms()
```

#### **Return Value**

Array of transform names

#### **Remarks**

The returned list does not include the "patch special transforms" that are prefixed with "#"

The list of transform names is stored in the LastSavedBy field of the patch summary information.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

# GetTransforms Method (*includeSpecialTransforms*)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► PatchPackage ►

**GetTransforms(Boolean)**

C# ▾

Gets the names of the transforms included in the patch package.

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public string[] GetTransforms(
    bool includeSpecialTransforms
)
```

```
Public Function GetTransforms ( _
    includeSpecialTransforms As Boolean _ 
) As String()
```

```
public:
array<String^>^ GetTransforms(
    bool includeSpecialTransforms
)
```

## Parameters

### *includeSpecialTransforms* (Boolean)

Specifies whether to include the "patch special transforms" that are prefixed with "#"

## Return Value

Array of transform names

## Remarks

The list of transform names is stored in the LastSavedBy field of the patch summary information.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## GetTransformsInfo Method

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► PatchPackage ►

### GetTransformsInfo()

C# ▾

#### Members

Icon	Member	Description
≡	<a href="#">GetTransformsInfo()</a>	Gets information about the transforms included in the patch package.
≡	<a href="#">GetTransformsInfo(Boolean)</a>	Gets information about the transforms included in the patch package.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► PatchPackage ►

## GetTransformsInfo()

C#

Gets information about the transforms included in the patch package.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public TransformInfo[] GetTransformsInfo()
```

```
Public Function GetTransformsInfo As TransformInfo()
```

```
public:  
array<TransformInfo^>^ GetTransformsInfo()
```

### Return Value

Array containing information about each transform

### Remarks

The returned info does not include the "patch special transforms" that are prefixed with "#"

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

## GetTransformsInfo Method (*includeSpecialTransforms*)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► PatchPackage ►

### GetTransformsInfo(Boolean)

C#

Gets information about the transforms included in the patch package.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public TransformInfo[] GetTransformsInfo(
    bool includeSpecialTransforms
)
```

```
Public Function GetTransformsInfo ( _
    includeSpecialTransforms As Boolean _ 
) As TransformInfo()
```

```
public:
array<TransformInfo^>^ GetTransformsInfo(
    bool includeSpecialTransforms
)
```

#### Parameters

##### *includeSpecialTransforms* (Boolean)

Specifies whether to include the "patch special transforms" that are prefixed with "#"

#### Return Value

Array containing information about each transform

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation **GetValidTransforms Method  
(InstallPackage)**

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ► PatchPackage ►  
**GetValidTransforms(InstallPackage)**

C# ▾

Analyzes the transforms included in the patch package to find the ones that are applicable to an install package.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++ ▾

```
public string[] GetValidTransforms(  
    InstallPackage installPackage  
)
```

```
Public Function GetValidTransforms ( _  
    installPackage As InstallPackage _  
) As String()
```

```
public:  
array<String^>^ GetValidTransforms(  
    InstallPackage^ installPackage  
)
```

[-] **Parameters**

*installPackage* (**InstallPackage**)

The install package to validate the transforms against

[-] **Return Value**

Array of valid transform names

[-] **Remarks**

The returned list does not include the "patch special transforms" that are prefixed with "#". If a transform is valid, then its corresponding special

transform is assumed to be valid as well.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## LogMessage Method (format, args)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ► PatchPackage ►  
**LogMessage(String, Object[])**

C# ▾

Sends a message to the [Message](#) event-handler.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
protected void LogMessage(  
    string format,  
    params Object[] args  
)
```

```
Protected Sub LogMessage ( _  
    format As String, _  
    ParamArray args As Object() _  
)
```

```
protected:  
void LogMessage(  
    String^ format,  
    ... array<Object^>^ args  
)
```

### Parameters

#### ***format* (String)**

Message string, containing 0 or more format items

#### ***args* ( Object [] )**

Items to be formatted

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0  
(3.5.1623.0)

Deployment Tools Foundation

## Message Event

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► PatchPackage ►

### Message

C#

Handle this event to receive status messages when operations are performed on the patch package.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public event InstallPackageMessageHandler Message
```

```
Public Event Message As InstallPackageMessageHandler
```

```
public:  
    event InstallPackageMessageHandler^ Message {  
        void add (InstallPackageMessageHandler^ value);  
        void remove (InstallPackageMessageHandler^ value);  
    }
```

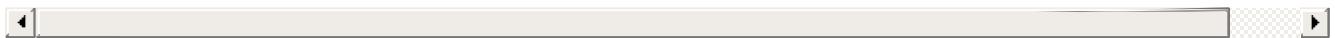
#### Examples

```
patchPackage.Message += new  
InstallPackageMessageHandler(Console.WriteLine)
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)



# Deployment Tools Foundation **PatchCode Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► PatchPackage ►

## **PatchCode**

C# ▾

Gets the patch code (GUID) of the patch package.

### [-] Declaration Syntax

C#

Visual Basic

Visual C++

```
public string PatchCode { get; }
```

```
Public ReadOnly Property PatchCode As String
```

```
public:  
property String^ PatchCode {  
    String^ get ();  
}
```

### [-] Remarks

The patch code is stored in the RevisionNumber field of the patch summary information.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

# Deployment Tools Foundation **TransformInfo Class**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► TransformInfo

C# ▾

Contains properties of a transform package (.MST).

## Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public class TransformInfo
```

```
Public Class TransformInfo
```

```
public ref class TransformInfo
```

## Members

All Members

Constructors

Methods

Properties

Public

Protected

Instance

Static

Declared

Inherited

Icon	Member	Description
	<a href="#">TransformInfo(String)</a>	Reads transform information from a transform package.
	<a href="#">TransformInfo(String, SummaryInfo)</a>	Reads transform information from the summary information of a transform package.
	<a href="#">Equals(Object)</a>	Determines whether the specified <a href="#">Object</a> is equal to the current <a href="#">Object</a> . (Inherited from <a href="#">Object</a> .)

	<a href="#">Finalize()</a>	Allows an <a href="#">Object</a> to attempt to free resources and perform other cleanup operations before the <a href="#">Object</a> is reclaimed by garbage collection.  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetHashCode()</a>	Serves as a hash function for a particular type. <a href="#">GetHashCode()</a> is suitable for use in hashing algorithms and data structures like a hash table  (Inherited from <a href="#">Object</a> .)
	<a href="#">GetType()</a>	Gets the <a href="#">Type</a> of the current instance  (Inherited from <a href="#">Object</a> .)
	<a href="#">MemberwiseClone()</a>	Creates a shallow copy of the current <a href="#">Object</a> .  (Inherited from <a href="#">Object</a> .)
	<a href="#">Name</a>	Gets the filename of the transform.
	<a href="#">TargetLanguage</a>	Gets the target language of the transform, or 0 if the transform is language-neutral.
	<a href="#">TargetPlatform</a>	Gets the target platform of the transform.
	<a href="#">TargetProductCode</a>	Gets the target product code of the transform.
	<a href="#">TargetProductVersion</a>	Gets the target product version of the transform.

		transform.
	<a href="#">ToString()</a>	Returns the name of the transform.  (Overrides <a href="#">Object.ToString()</a> .)
	<a href="#">UpgradeCode</a>	Gets the upgrade code of the transform.
	<a href="#">UpgradeProductCode</a>	Gets the upgrade product code of the transform.
	<a href="#">UpgradeProductVersion</a>	Gets the upgrade product version of the transform.
	<a href="#">Validations</a>	Gets the validation flags specified when the transform was generated.

## Inheritance Hierarchy

[Object](#)

[TransformInfo](#)

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation **TransformInfo Constructor**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► TransformInfo ►

**TransformInfo()**

C# ▾

▀ **Members**

Icon	Member	Description
≡	<a href="#">TransformInfo(String)</a>	Reads transform information from a transform package.
≡	<a href="#">TransformInfo(String, SummaryInfo)</a>	Reads transform information from the summary information of a transform package.

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

# TransformInfo Constructor (mstFile)

Deployment Tools Foundation Namespaces ►  
Microsoft.Deployment.WindowsInstaller.Package ► TransformInfo ► TransformInfo(String)

C# ▾

Reads transform information from a transform package.

## Declaration Syntax

C#

Visual Basic

Visual C++

```
public TransformInfo(  
    string mstFile  
)
```

```
Public Sub New ( _  
    mstFile As String _  
)
```

```
public:  
TransformInfo(  
    String^ mstFile  
)
```

## Parameters

### mstFile (String)

Path to a transform package (.MST file).

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## TransformInfo Constructor (name, transformSummaryInfo)

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► TransformInfo ► TransformInfo(String, SummaryInfo)

C#

Reads transform information from the summary information of a transform package.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public TransformInfo(  
    string name,  
    SummaryInfo transformSummaryInfo  
)
```

```
Public Sub New ( _  
    name As String, _  
    transformSummaryInfo As SummaryInfo _  
)
```

```
public:  
TransformInfo(  
    String^ name,  
    SummaryInfo^ transformSummaryInfo  
)
```

### Parameters

#### **name (String)**

Filename of the transform (optional).

#### **transformSummaryInfo (SummaryInfo)**

Handle to the summary information of a transform package (.MST file).

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## Name Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► TransformInfo ►

### Name

C# ▾

Gets the filename of the transform.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string Name { get; }
```

```
Public ReadOnly Property Name As String
```

```
public:  
property String^ Name {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

# Deployment Tools Foundation **TargetLanguage** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► TransformInfo ►

## TargetLanguage

C# ▾

Gets the target language of the transform, or 0 if the transform is language-neutral.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public int TargetLanguage { get; }
```

```
Public ReadOnly Property TargetLanguage As Integer
```

```
public:  
property int TargetLanguage {  
    int get();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

# Deployment Tools Foundation **TargetPlatform** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► TransformInfo ►

## TargetPlatform

C# ▾

Gets the target platform of the transform.

### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string TargetPlatform { get; }
```

```
Public ReadOnly Property TargetPlatform As String
```

```
public:  
property String^ TargetPlatform {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation **TargetProductCode Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► TransformInfo ►

**TargetProductCode**

C# ▾

Gets the target product code of the transform.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public string TargetProductCode { get; }
```

```
Public ReadOnly Property TargetProductCode As String
```

```
public:  
property String^ TargetProductCode {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation **TargetProductVersion Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► TransformInfo ►

**TargetProductVersion**

C# ▾

Gets the target product version of the transform.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public string TargetProductVersion { get; }
```

```
Public ReadOnly Property TargetProductVersion As Str
```

```
public:  
property String^ TargetProductVersion {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation **ToString Method**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► TransformInfo ►

**ToString()**

C# ▾

Returns the name of the transform.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public override string ToString()
```

```
Public Overrides Function ToString As String
```

```
public:  
virtual String^ ToString() override
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module: Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0 (3.5.1623.0)

Deployment Tools Foundation

## UpgradeCode Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► TransformInfo ►

### UpgradeCode

C# ▾

Gets the upgrade code of the transform.

#### Declaration Syntax

C#

Visual Basic

Visual C++

```
public string UpgradeCode { get; }
```

```
Public ReadOnly Property UpgradeCode As String
```

```
public:  
property String^ UpgradeCode {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation **UpgradeProductCode Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► TransformInfo ►

**UpgradeProductCode**

C# ▾

Gets the upgrade product code of the transform.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public string UpgradeProductCode { get; }
```

```
Public ReadOnly Property UpgradeProductCode As String
```

```
public:  
property String^ UpgradeProductCode {  
    String^ get();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)

Deployment Tools Foundation **UpgradeProductVersion Property**

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► TransformInfo ►

**UpgradeProductVersion**

C# ▾

Gets the upgrade product version of the transform.

[-] **Declaration Syntax**

C#

Visual Basic

Visual C++

```
public string UpgradeProductVersion { get; }
```

```
Public ReadOnly Property UpgradeProductVersion As St
```

```
public:  
property String^ UpgradeProductVersion {  
    String^ get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)



# Deployment Tools Foundation **Validations** Property

Deployment Tools Foundation Namespaces ►

Microsoft.Deployment.WindowsInstaller.Package ► TransformInfo ►

## Validations

C# ▾

Gets the validation flags specified when the transform was generated.

### Declaration Syntax

C#

Visual Basic

Visual C++ ▾

```
public TransformValidations Validations { get; }
```

```
Public ReadOnly Property Validations As TransformVal
```

```
public:  
property TransformValidations Validations {  
    TransformValidations get ();  
}
```

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)

Assembly: Microsoft.Deployment.WindowsInstaller.Package (Module:

Microsoft.Deployment.WindowsInstaller.Package) Version: 3.0.0.0

(3.5.1623.0)



## Namespaces

Namespace	Description
<a href="#">Microsoft.Deployment.Compression</a>	Framework for archive packing and unpacking.
<a href="#">Microsoft.Deployment.Compression.Cab</a>	Implements cabinet archive packing and unpacking.
<a href="#">Microsoft.Deployment.Resources</a>	Classes for reading and writing resource data in executable files.
<a href="#">Microsoft.Deployment.WindowsInstaller</a>	Complete class library for the Windows Installer APIs.
<a href="#">Microsoft.Deployment.WindowsInstaller.Linq</a>	LINQ extensions for querying Windows Installer databases (experimental).

<b>Microsoft.Deployment.WindowsInstaller.Package</b>	Extended classes for working with Windows Installer installation and patch packages.
--	--

---

Send comments on this topic to [wix-users@lists.sourceforge.net](mailto:wix-users@lists.sourceforge.net)