⌂ WiX managed SDK WixToolset.Dtf.WindowsInstaller.Package namespace InstallPackage Class

InstallPackage Class

Provides access to powerful build, maintenance, and analysis operations on an installation package (.MSI or .MSM).

Methods

Method	Description
ApplyPatch(patchPackage, transform)	Applies a patch package to the database, resulting in an installation package that has the patch built-in.
Consolidate(mediaCabinet)	Consolidates a package by combining and re-compressing all files into a single internal or external cabinet.
ExtractFiles()	Extracts all files to the «see P:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.WorkingDirectory» . The files are extracted to the relative directory matching their «see P:WixToolset.Dtf.WindowsInstaller.Package.InstallPath.SourcePath» .
FindFiles(longFileName)	Gets the list of file keys that have the specified long file name.
FindFiles(pattern)	Gets the list of file keys whose long file names match a specified regular-expression search

Method	Description
	pattern.
LogMessage(format, args)	Sends a message to the «see E:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.Message» event-handler.
UpdateDirectories()	Rebuilds the cached directory structure information accessed by the «see P:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.Directories» and «see P:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.Files» properties. This should be done after modifying the File, Component, or Directory tables, or else the cached information may no longer be accurate.
UpdateFiles()	Updates the install package with new files from the «see P:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.WorkingDirectory» . The files must be in the relative directory matching their «see P:WixToolset.Dtf.WindowsInstaller.Package.InstallPath.SourcePath» . This method re-compresses and packages the files if necessary, and also updates the following data: File.FileSize, File.Version, File.Language, MsiFileHash.HashPart*

Properties

Property	Description
CompressionLevel	Gets or sets the compression level used by «see

Property	Description
	M:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.UpdateFiles» and «see M:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.Consolidate(System.String)» .
Directories	Gets a mapping from Directory keys to source/target paths.
Files	Gets a mapping from File keys to source/target paths.
Property	Accessor for getting and setting properties of the InstallPackage database.
SourceDirectory	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.
WorkingDirectory	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.

Events

Event	Description
Message	Handle this event to receive status messages when operations are performed on the install package.

`WixToolset.Dtf.WindowsInstaller.Package.dll` version `4.0.2+644ed0118bae1aa885bb6cc906dc6c0b904de4d9`

ApplyPatch(patchPackage, transform) Method

Applies a patch package to the database, resulting in an installation package that has the patch built-in.

Declaration

```
public void ApplyPatch(
  PatchPackage patchPackage,
  string transform
)
```

Parameters

Parameter	Type	Description
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 «see M:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.Consolidate(System.String)» to apply the file patches immediately and also discard any outdated files from the package.

Consolidate(mediaCabinet) Method

Consolidates a package by combining and re-compressing all files into a single internal or external cabinet.

Declaration

```
public void Consolidate(
  string mediaCabinet
)
```

Parameters

Parameter	Type	Description
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 uncompressed 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 «see M:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.ApplyPatch(WixToolset.Dtf.WindowsInstaller.Package.PatchPackage,Syste m.String)») then this method will apply the patches before compressing the updated files. This method edits the database summary information and the File, Media and Patch tables as necessary to maintain a valid installation package. The cabinet compression level used during re-cabbing can be configured with the «see P:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.CompressionLevel» property.

ExtractFiles() Method

Extracts all files to the «see P:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.WorkingDirectory» . The files are extracted to the relative directory matching their «see P:WixToolset.Dtf.WindowsInstaller.Package.InstallPath.SourcePath» .

Declaration

```
public void ExtractFiles()
```

Remarks

If any files have the uncompressed attribute, they will be copied from the «see P:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.SourceDirectory» .

FindFiles(longFileName) Method

Gets the list of file keys that have the specified long file name.

Declaration

```
public System.String[] FindFiles(
  string longFileName
)
```

Parameters

Parameter	Type	Description
longFileName	string	File name to search for (case-insensitive)

Return value

`System.String[]` Array of file keys, or a 0-length array if none are found

FindFiles(pattern) Method

Gets the list of file keys whose long file names match a specified regular-expression search pattern.

Declaration

```
public System.String[] FindFiles(
  System.Text.RegularExpressions.Regex pattern
)
```

Parameters

Parameter	Type	Description
pattern	System.Text.RegularExpressions.Regex	Regular expression search pattern

Return value

`System.String[]` Array of file keys, or a 0-length array if none are found

LogMessage(format, args) Method

Sends a message to the «see E:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.Message» event-handler.

Declaration

```
protected void LogMessage(
  string format,
  System.Object[] args
)
```

Parameters

Parameter	Type	Description
format	string	Message string, containing 0 or more format items
args	System.Object[]	Items to be formatted

UpdateDirectories() Method

Rebuilds the cached directory structure information accessed by the «see P:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.Directories» and «see P:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.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

```
public void UpdateDirectories()
```

UpdateFiles() Method

Updates the install package with new files from the «see P:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.WorkingDirectory» . The files must be in the relative directory matching their «see P:WixToolset.Dtf.WindowsInstaller.Package.InstallPath.SourcePath» . This method re-compresses and packages the files if necessary, and also updates the following data: File.FileSize, File.Version, File.Language, MsiFileHash.HashPart*

Declaration

```
public void UpdateFiles()
```

Remarks

The cabinet compression level used during re-cabbing can be configured with the «see P:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.CompressionLevel» property.

CompressionLevel Property

Gets or sets the compression level used by «see M:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.UpdateFiles» and «see M:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.Consolidate(System.String)» .

Declaration

```
public WixToolset.Dtf.Compression.CompressionLevel CompressionLevel { get; set; }
```

Remarks

If the Directory table is modified, this mapping will be outdated until you close and reopen the install package.

Directories Property

Gets a mapping from Directory keys to source/target paths.

Declaration

```
public InstallPathMap Directories { get; set; }
```

Remarks

If the Directory table is modified, this mapping will be outdated until you call «see M:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.UpdateDirectories» .

Files Property

Gets a mapping from File keys to source/target paths.

Declaration

```
public InstallPathMap Files { get; set; }
```

Remarks

If the File, Component, or Directory tables are modified, this mapping may be outdated until you call «see M:WixToolset.Dtf.WindowsInstaller.Package.InstallPackage.UpdateDirectories» .

Property Property

Accessor for getting and setting properties of the InstallPackage database.

Declaration

```
public InstallPackageProperties Property { get; set; }
```

SourceDirectory Property

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

```
public string SourceDirectory { get; set; }
```

WorkingDirectory Property

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

```
public string WorkingDirectory { get; set; }
```

Message Event

Handle this event to receive status messages when operations are performed on the install package.

Declaration

```
public InstallPackageMessageHandler Message
```

Value

`InstallPackageMessageHandler`