Software License Agreement

DOC to HTML Converter

For .NET

Vesion 16

2008-2024

3200 Maysilee Street

Austin, TX 78728

512-733-2525

Software License Agreement

The Software is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. The Software is licensed, not sold. This LICENSE AGREEMENT grants you the following rights:

A. This product is licensed per developer basis only. Each developer working with this package needs to purchase a separate license.

B. The purchaser has the right to modify and link the DLL functions into their application. Such an application is free of distribution royalties with these conditions: the target application is not a stand-alone DOC to HTML or DOCX to HTML Converter; the target application uses this product for one operating system platform only; and the source code (or part) of the editor is not distributed in any form.

C. The DESKTOP LICENSE allows for the desktop application development. Your desktop application using this product can be distributed royalty-free. Each desktop license allows one developer to use this product on up to two development computers. A developer must purchase additional licenses to use the product on more than two development computers.

D. The SERVER LICENSE allows for the server application development. The server licenses must be purchased separately when using this product in a server application. Additionally, the product is licensed per developer basis. Only an UNLIMITED SERVER LICENSE allows for royalty-free distribution of your server applications using this product.

E. ENTERPRISE LICENSE: The large corporations with revenue more than $50 million and large government entities must purchase an Enterprise License. An Enterprise license is also applicable if any target customer of your product using the Software have revenue more than $500 million. Please contact us at info@subsystems.com for a quote for an Enterprise License.

F. Your license rights under this LICENSE AGREEMENT are non-exclusive. All rights not expressly granted herein are reserved by Licensor.

G. You may not sell, transfer or convey the software license to any third party without Licensor's prior express written consent.

H. The license remains valid for 12 months after the issue date. The subsequent year license renewal cost is discounted by 20 percent from the license acquisition cost. The license includes standard technical support, patches and new releases.

I. You may not disable, deactivate or remove any license enforcement mechanism used by the software.

This software is designed keeping the safety and the reliability concerns as the main considerations. Every effort has been made to make the product reliable and error free. However, Sub Systems, Inc. makes no warranties against any damage, direct or indirect, resulting from the use of the software or the manual and can not be held responsible for the same. The product is provided 'as is' without warranty of any kind, either expressed or implied, including but not limited to the implied warranties of suitability for a particular purpose. The buyer assumes the entire risk of any damage caused by this software. In no event shall Sub Systems, Inc. be liable for damage of any kind, loss of data, loss of profits, interruption of business or other financial losses arising directly or indirectly from the use of this product. Any liability of Sub Systems will be exclusively limited to refund of purchase price.

Sub Systems, Inc. offers a 30 day money back guarantee with the product. Must call for an RMA number before returning the product.

▲ ▼

Getting Started

This chapter describes the contents of the software diskettes and provides a step by step process of incorporating DOC to Html Converter into your application.

In This Chapter

▲ ▼

Files

The package contains the WHN.DLL, tesn31.dll, wrs15.dll, wrw15.dll and hsn26.dll files necessary to incorporate this product into your application.

The package also includes a set of files to construct a demo program. The demo program shows by example the process of linking the DLL to your program.

DLL Demo Files:

The following demo files are included in the c_demo.zip file.

demo.cs	Source code for the demo program

demo.exe	Executable demo program
demo.csproj	The project file to compile the demo.
AssemblyInfo.cs	Assembly information file

Visual Basic Interface and Demo Files:

Form1.vb	vb source file
dmo_vbn.vbproject	The project file for the visual basic demo program.
AssemblyInfo.vb	Assembly information file for the demo program.

▲ ▼

License Key

Your License Key and License number are e-mailed to you after your order is processed. You would set the license information using the WhsSetLicenseInfo static function. This should be preferably done before creating the Whn object to avoid pop-up nag screens.

int WhsSetLicnseInfo(String LicenseKey, String LicenseNumber, String CompanyName);

LicenseKey: Your license key is available in the product delivery email sent to you upon the purchase of the product. It consists of a string in the form of "xxxxx-yyyyy-zzzzz".

LicenseNumber: Your license number is also available in the product delivery email. The license number string starts with a "srab" or "smo" prefix.

CompanyName: Your company name as specified in your order.

Return Value: This method returns 0 when successful. A non-zero return value indicates an error condition. Here are the possible return values:

0	License application successful.
1	Invalid License Key.
2	Invalid License Number.
3	Ran out of available licenses. Please consider purchasing additional licenses.

Example:

result=Whn.WhsSetLicenseInfo("xxxxx-yyyyy-zzzzz","srabnnnnn-n","Your Company Name")

Replace the 'xxxxx-yyyyy-zzzzz' by your license key, replace "srabnnnnn-n" with your license number, and "Your Company Name" with your company name as specified in your order.

Note: *WhsSetLicenseInfo method should be called only once at the beginning of your application. Calling this method for each conversion would degrade the conversion performance.*

Sample Conversion Code

Please ensure that WHN.DLL, tesn31.dll, and hsn26.dll files are available in the project directory. The wrs15.dll and wrw15.dll file should be copied to Windows' system32 folder, or any other folder available in the 'path' setting. Set the reference for WHN.DLL in your project. The remaining dlls are referenced indirectly by WHN.DLL.

Now set namespace for the product:

```
 using SubSystems.WH;             // C# example
 Imports SubSystems.WH           ' VB Example
```

Now set the product license key and create an HRN type object:

Whn.WhsSetLicenseInfo("xxxxx-yyyyy-zzzzz","srabnnnnn-n","my company name")

```
   Whn wh = new Whn()  // C# example
   dim dh as Whn        ' VB example

   wh.InWebServer=true;  // set InWebServer to true when hosting
                         // the converter in an ASP.NET app
```

Now use one of the following calls to convert from DOC or DOCX to HTML format:

1. Convert an DOC file to an HTML file.

```
   wh.WhsConvertFile("test.doc","test.htm")
```

2. Convert an DOC byte array to an HTML string

VB Example:

```
Dim DocBytes() As Byte
Dim HtmlString As String

DocBytes = wh.WhsFileToBytes("test.doc")

If (InData.Length > 0) Then
   HtmlString = wh.WhsConvertBuffer(DocBytes)
End If
```

C# Example:

```
byte[] DocBytes=null;
string HtmlString;

if ((DocBytes=wh.WhsFileToBytes("test.doc"))!=null) {
  HtmlString=wh.WhsConvertBuffer(DocBytes));
}
```

▲ ▼

Control Methods

These methods allow you to convert between html and doc formats. Please set the namespace for the Whn class before using these methods:

```
using SubSystems.WH;             // C# example
Imports SubSystems.WH          ' VB Example
```

In This Chapter

WhsConvertBuffer

Convert DOC to HTML using text string.

String WhsConvertBuffer(DocBytes)

byte[] DocBytes; // Raw byte array containing the doc data.

Return value: This function returns a string containing the converted html documented.

A null return values indicates an error.

Examples:

```
VB Example:

  Dim DocBytes() As Byte
  Dim HtmlString As String

  DocBytes = wh.WhsFileToBytes("test.doc")

  If (InData.Length > 0) Then
     HtmlString = wh.WhsConvertBuffer(DocBytes)
  End If

  C# Example:

  byte[] DocBytes=null;
  string HtmlString;

  if ((DocBytes=wh.WhsFileToBytes("test.doc"))!=null) {
    HtmlString=wh.WhsConvertBuffer(DocBytes);
  }
```

WhsConvertFile

Convert DOC to HTML using disk files.

bool WhsConvertFile(InFile, OutFile, XlateType)

string InFile;	// Input file containing DOC document
string OutFile;	// Output files, contains the converted html document

Return value: This function returns TRUE when successful.

Examples:

```
    wh.WhsConvertFile("test.doc","test.htm")
```

▲ ▼

WhsFileToBytes

Read a doc file into a byte array.

byte[] WhsFileToBytes(DocFile)

String DocFile;	// Input doc file name.

Return value: This function returns a byte array from the given file. This method can be used to obtain the byte array to supply to the WhsConvertBuffer method.

A null return values indicates an error.

```
   Dim DocBytes() As Byte
   Dim HtmlString As String

   DocBytes = wh.WhsFileToBytes("test.doc")

   If (InData.Length > 0) Then
      HtmlString = wh.WhsConvertBuffer(DocBytes)
   End If
```

▲ ▼

WhsGetLastMessage

Get the last message.

int WhsGetLastMessage(WhsMessage, DebugMessage);

string WhsMessage;	// Returns the default user message text in English
string DebugMsg;	// Returns any debug message associated with the last message. The debug message need not be displayed to the user.

Return Value: This function returns the last message generated by the editor. This value is valid only if saving of the messages is enabled by setting the WHFLAG_RETURN_MSG_ID flag. This flag is set using the WhsSetFlags method.

▲ ▼

WhsGetPictCount

Return total number of pictures found during conversion.

int WhsGetPictCount()

Description: Thjs function is called after converting a file to get the number of pictures encountered during conversion. The picture names can be retrieved by using the WhsGetPictName function.

Return Value: When converting from DOC to html format, this method returns the number of picture files referenced in the output html file.

▲ ▼

WhsGetPictName

Return name of the picture file found during conversion.

string WhsGetPictName(idx)

int idx;	The index of the picture file. The index value must be between 0 and (Total pictures - 1). The total picture count can be retrieved using the WhsGetPictCount function.

Description: Thjs function is called after converting a file to get the individual picture file name or picture url.

Return Value: This function returns a picture file name or picture file url.

▲ ▼

WhsResetLastMessage

Reset the last editor message.

bool WhsResetLastMessage()

Description: This function can be called before calling any other function to reset the last error message.

Return Value: The function returns TRUE when successful.

See Also
WhsGetLastMessage
WhsSetFlags

▲ ▼

WhsSetFlags

Set certain flags or retrieve the values of the flags.

int WhsSetFlags(set, flags)

bool set;	// TRUE to set the given flags, FALSE to reset the given flags
int flags;	// Flags (bits) to set or reset. Currently, the following flag values are available:

WHFLAG_RETURN_MSG_ID	Do not display the error messages. Save the error code to be later retrieved using the WhsGetLastMessage function.
WHFLAG_SEGMENT_ONLY	Do not include html header and body tags. Generate just html segment.
WHFLAG_LIST_TO_TEXT	Convert document lists to text numbers in html output.
WHFLAG_NO_SPAN_TAG	Do not write the 'span' tag to html output.
WHFLAG_NO_FONT	Do not write the 'font' tag to html output.

WHFLAG_FLASH_SUPPORT	Write html tags to support Flash.
WHFLAG_XLATE_SINGLE_QUOTE	Write single quote character as &#39; to html output.

Return value: This function returns the new value of all the flags. Call this function with the 'flags' parameter set to zero to retrieve flag values without modifying it.

▲

Control Properties

The control supports the following properties:

InWebServer

This property should be set to True when this control is used in a web server. When this property is set to True, the control suppress the display of any dialog and message boxes.

ProjectFolder

Set this property to the folder containing your project, such as c:\inetpub\wwwroot\MyProject. This information helps the converter locate the images which use relative path. It is also used for creating any temporary files.

WebFolder

The folder where the embedded pictures in doc input are to be stored during the conversion.

SrcTagFolder

The folder name to add to the 'src' tag file name while saving embedded pictures in document during the converting.

PictureNamePrefix

The prefix string to use to build the picture file name during the conversion.
The default prefix string is 'HTS_'.