HOWTO: Correctly Reference, License and Configure OfficeDecoder and Dependencies

Atalasoft's Office Decoder add-on allows for the rendering (rasterization) of Word, Excel and PowerPoint documents as images. Similar to our other optional add-on decoders, this must be licensed, and referenced and the decoder added to the RegisteredDecoders.Decoders collection. However the OfficeDecoder has additional dependencies that other Atalasoft components do not.

Licensing

In order to license the OfficeDecoder, you must have an ODD2 / ODDX2 serial or a special license that includes the OfficeDecoder. You activate these serials the same way as any other Atalasoft serial (use the activation wizard to activate for the machine in question)

Activating an ODD2 (SDK) or ODDX2 (server) license will create an Atalasoft.dotImage.Office.lic file which .. if yhou're using the SDK version on yhour local dev mahcine, it will put the file in the correct place.

If you're licensing a server deployment or your app is using local copy of IIS (instead of the built in web server in Visual studio) you'll need to copy the license to the bin directory of the application

References

If you do not include both the correct office DLL reference AND the Perceptive filters below you may see an error along these lines:

```
The type or namespace name 'Office' does not exist in the namespace
'Atalasoft.Imaging.Codec' (are you missing an assembly reference?)
```

Correcting this requires that you complete both of the following:

Atalasoft.dotImage.Office.dll

In order to use the OfficeDecoder, you must add the correct references. The Atalasoft.dotImage.Office.dll is standard.. we ship DLLs for .NET 3.5 and for 4.5.2 (and up) in both x86 (32 bit) and x64.. you need to reference the correct framework and bitness version of the Atalasoft.dotImage.Office.dll in yhour project.

Perceptive Filter DLLs

Additionally, the OfficeDecoder depends on a set of Perceptive Filter dlls that ship with the SDK but must be manually added. You can not directly reference the Perceptive filter dlls in your project references

They ship in

```
C:\Program Files (x86)\Atalasoft\DotImage 11.1\bin\PerceptiveDocumentFilters\intel-32
```

and

```
C:\Program Files (x86)\Atalasoft\DotImage 11.1\bin\PerceptiveDocumentFilters\intel-64
```

NOTE this is bitness dependent but not tied to .NET framework as our main dlls are.

You will need to add the contents of that directory:

```
ISYS11df.dll

ISYSreaders.dll

ISYSreadershd.dll

Perceptive.DocumentFilters.dll
```

to your project and then set them to copy local on build.. either that or just copy them to your application bin directory.

Decoder Registration / Usage

Finally, you are ready to use the OfficeDecoder class. You can now instantiate a new OfficeDecoder (make sure to set the Resolution to something useful like 200 or 300 as the default of 96 is too low for most needs.

Most customers will want it to "just work" so you need to add the OfficeDecoder to the RegisteredDecoders.Decoders collection. When you do this, then Atalasoft components now "just know" how to render Office documents.

You need to do this in a static constructor for your class. It should be done once and only once in the application

```
RegisteredDecoders.Decoders.Add(new OfficeDecoder() { Resolution = 200 });
```

For more on how to alter the resolution on the fly or how to safely set the decoder outside of a static constructor, please see:

HOWTO: Safely Change / Set Resolution of OfficeDecoder

Original Article:

Q10488 - HOWTO: Correctly Reference, License and Configure OfficeDecoder and Dependencies

Atalasoft Knowledge Base
https://www.atalasoft.com/kb2/KB/50016/HOWTO-Correctly-Reference-License-an...