PEERNET Document Conversion Service 3.0 has been successfully installed as a Windows service on your computer. The mini-tutorials below are designed to get you converting files as soon as possible.

Converting Files and Folders

- Convert a File on the Command Line
- Batch Convert a Folder of Files
- Append or Combine Files Together
- Convert a File from my Program
- What Conversion Profiles Are Included?

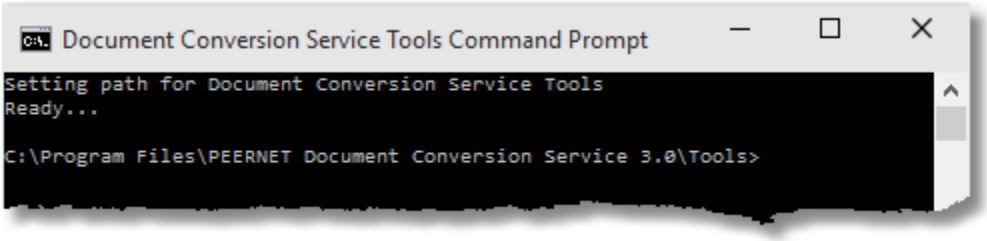
Managing the Service

- Starting and Stopping the Service
- Configuring the Service for Parallel Conversion
- Controlling Which Converters Are Running
- Common Questions and Troubleshooting

CONVERT A FILE ON THE COMMAND LINE

1. Open the Tools Command Prompt.

   From the Start menu, go to Programs - PEERNET Document Conversion Service 3.0 – Tools - Document Conversion Service Tools Command Prompt.



2. To start converting files, copy and paste in one of the sample command lines below into the Tools command prompt and simply replace the underlined text with your own filenames or folders.

   Convert a PDF file to a TIFF:
   DCSConvertFile /P="TIFF 300dpi OptimizedColor" /NE "C:\Test\File.pdf"

   Convert a Word document to a vector PDF named ConvertedFile.pdf:
   DCSConvertFile /P="Adobe PDF Multipage" /NE /N="ConvertedFile" "C:\Test\File.doc"

   Convert a Word document to a raster PDF named ConvertedFile.pdf:
   DCSConvertFile /P="PDF 300dpi OptimizedColor" /NE /N="ConvertedFile" "C:\Test\File.doc"

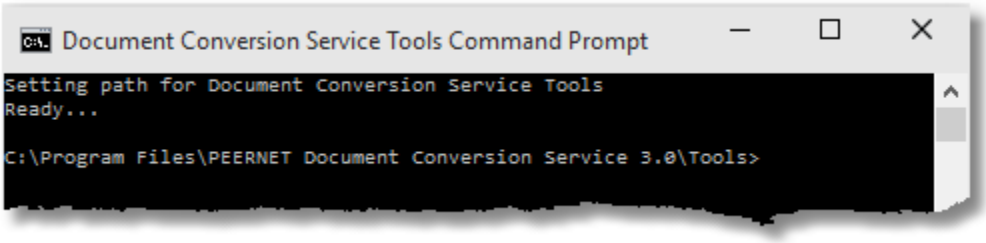
   Convert a PDF file to a TIFF and save in a different location:
   DCSConvertFile /P="TIFF 200dpi OptimizedColor" /NE /S="C:\Test\TIFFS" "C:\Test\File.pdf"

   Convert a PDF file to a TIFF, wait up to 5 minutes for the conversion service to be running:
   DCSConvertFile /P="TIFF 300dpi OptimizedColor" /NE /W=300 "C:\Test\File.pdf"

BATCH CONVERT A FOLDER OF FILES

1. Open the Tools Command Prompt.

   From the Start menu, go to Programs - PEERNET Document Conversion Service 3.0 – Tools - Document Conversion Service Tools Command Prompt.



2. Use one of our sample command lines below to begin converting. Simply copy and paste the desired command into the Tools command prompt and replace the underlined text with your own folders.

   Convert a folder of PDF files to TIFF and save in a different location:
   DCSConvertFolder /F="*.pdf" /P="TIFF 300dpi OptimizedColor" /NE /S="C:\Test\TIFF" "C:\Test\Input"

   Convert all Word and Excel documents in the folder and all its subfolders to vector PDF files, in the same location:
   DCSConvertFolder /R /F="*.doc|*.xls" /P="Adobe PDF Multipage" /S="C:\Test\Input" "C:\Test\Input"

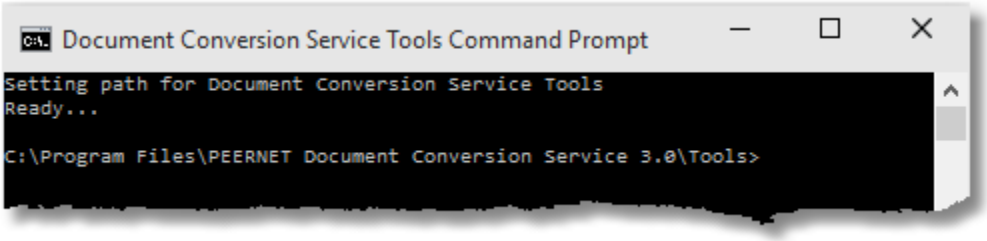
   Convert all files in the folder except text files to TIFF images and save in a *.converted* folder:
   DCSConvertFolder /X="*.txt" /P="TIFF 200dpi OptimizedColor" "C:\Test\Input"

   Convert a folder of PDF files, wait up to 5 minutes for the conversion service to be running:
   DCSConvertFile /F="*.pdf" /P="TIFF 300dpi OptimizedColor" /NE /W=300 /S="C:\Test\TIFF" "C:\Test\Input"

APPEND OR COMBINE FILES TOGETHER

1. Open the Tools Command Prompt.

   From the Start menu, go to Programs - PEERNET Document Conversion Service 3.0 - Tools - Document Conversion Service Tools Command Prompt.



2. Files are combined by passing the file names on the command line or by passing the name of a text file that contains the list of files to combine. The input files are listed one per line in the input text file and files are combined in the order in which they are given in the list.

```
C:\Input\WordFiles\File1.doc
C:\Input\WordFiles\File2.docx
C:\Input\PDF\File3.pdf
\\server\share\Input\scans\scan1.tif
```

   Use one of our sample command lines below to begin combining files. Simply copy and paste the desired command into the Tools command prompt and replace the underlined text with your own folders and files.

   Combine two PDF files to a multipage TIFF:
   DCSCombineFiles /S="C:\Test\TIFF" /N="NewFileName" /P="TIFF 300dpi OptimizedColor" "C:\Test\File1.pdf" "C:\Test\File2.pdf"

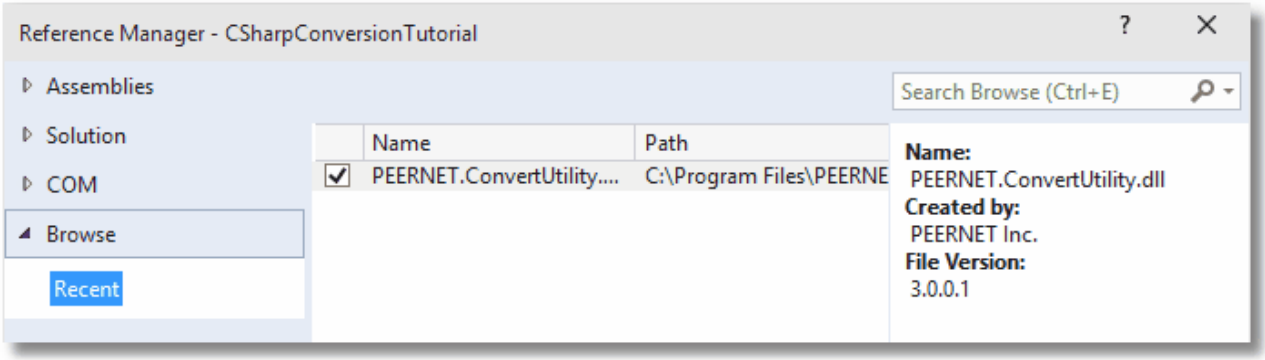
   Combine all files in the input text file to multipage PDF:
   DCSCombineFiles /S="C:\Test\TIFF" /N="NewFileName" /P="PDF 300dpi OptimizedColor" /I="C:\Test\InputFiles.txt"

   Combine all files into a multipage PDF, wait up to 5 minutes for the conversion service to be running:
   DCSCombineFiles /S="C:\Test\TIFF" /N="NewFileName" /P="TIFF 300dpi OptimizedColor" /W=300 /I="C:\Test\InputFiles.txt" "C:\Test\EndFooter.doc"

CONVERT A FILE IN MY PROGRAM

With the provided .NET library, PEERNET.ConvertUtility.dll, you can add file or folder conversion to your application with a single line of code. See the included C# and Visual Basic .NET samples for complete coding examples.

1. Add a reference to the PEERNET.ConvertUtility.dll into your project. It is located in the \Samples\Redist folder in your product installation folder.



2. Add the `using` statement to the top of your file.

```
using PEERNET.ConvertUtility;
```

3. Add the following code to your program to convert a file; replace the underlined sections of code with your filenames and output folders.

   This sample also show waiting a maximum of 5 minutes (300 seconds) for Document Conversion Service to be running and started. If this setting is not specified the call will return immediately with an error if Document Conversion Service is not running. If Document Conversion Service is already running the command executes immediately.

```
PNConversionItem resultItem = null;
Dictionary<String, String> customSettings = new Dictionary<String, String>();

 // wait 5 min for conversion service to be started
customSettings["SecondsToWaitForRunningConversionService"] = "300";

// This is the single call needed to convert a file
resultItem = PNConverter.ConvertFile(@"C:\Test\File.pdf",
                                     @"C:\Test\Output",
                                     @"ConvertedFromPDF",
                                     true, // overwrite existing
                                     false, // remove file ext
                                     false, // create log
                                     "TIFF 200dpi OptimizedColor",
                                     String.Empty,
                                     String.Empty,
                                     customSettings, // User settings
                                     String.Empty, // not using DCOM
                                     String.Empty, // use default working folder
                                     String.Empty);
```

4. To convert a folder, use the `PNConverter.ConvertFolder` method instead.

When converting a folder of files the number of documents that can be passed in parallel (*at the same time*) to Document Conversion Service to be converted is automatically determined based on the number of CPU's and cores on your system multiplied by 1.5. We recommend that you allow this value to be determined automatically, but if needed, you can customize how many documents you are processing in parallel. Please note that this value is different from the setting of the same name described in the section Configuring the Service for Parallel Conversion.

```
PNConversionItem resultItem = null;
Dictionary<String, String> customSettings = new Dictionary<String, String>();

// Add the number of threads
customSettings["NumberOfDocumentsInParallel"] = "3";

// wait 5 min for conversion service to be started
customSettings["SecondsToWaitForRunningConversionService"] = "300";

// This is the single call needed to convert all pdf files in the folder
resultItem = PNConverter.ConvertFolder(@"C:\Test\InputFiles",
                                       true, // include subfolders
                                       "*.pdf", // filter
                                       String.Empty, // exclude filter
                                       @"C:\Test\Output", // output folder
                                       true, // overwrite existing
                                       false, // remove file ext
                                       false, // create log
                                       "TIFF 200dpi OptimizedColor", // settings
                                       String.Empty, // extensison profile
                                       String.Empty, // MIME profile
                                       customSettings, // User settings
                                       String.Empty, // not using remote conversion (DCOM)
                                       String.Empty, // use default working folder
                                       string.Empty);
```

5. To convert a list of files, the `PNConverter.ConvertFileList` method allows you to pass a list of files to be converted. Like the folder conversion, the files are converted in parallel based on your system configuration. Unlike when you are converting a folder, each file in the list can have its own output directory and its own collection of custom settings that are used in addition to the profile settings.

```
IList<PNConversionItem> results = null;

Dictionary<String, String> customSettings = new Dictionary<String, String>();
// Custom conversion settings, the number of threads, wait for ready
customSettings["NumberOfDocumentsInParallel"] = "3";
// wait 5 min for conversion service to be started
customSettings["SecondsToWaitForRunningConversionService"] = "300";


// Create file list to convert
IList<PNConvertFileInfo> convertFileInfoList = new List<PNConvertFileInfo>();
// This file uses only the conversion settings from the profile
convertFileInfoList.Add(new PNConvertFileInfo(@"C:\Test\InputPDF\File1.pdf",
                                              String.Empty, null));
// This file has its own output directory
convertFileInfoList.Add(new PNConvertFileInfo(@"C:\Test\InputPDF\File2.pdf",
                                              @"C:\Test\Output\ConvertedPDF",
                                              null));
// This file also changes the conversion settings to 300 dpi and
// causes the Word converter to print markup.
IList<PNSetting> filesettings = new List<PNSetting>();
filesettings.Add(new PNSetting("Devmode settings;Resolution", "300"));
filesettings.Add(new PNSetting("Microsoft.Word.Document.PrintOut.Item",
                               "DocumentAndMarkup"));
convertFileInfoList.Add(new PNConvertFileInfo(@"C:\Test\InputWord\File1.doc",
                                              @"C:\Test\Output\ConvertedDocs",
                                              filesettings));

results = PNConverter.ConvertFileList(convertFileInfoList,
                                      @"C:\Test\FileListOut", // output folder
                                      String.Empty, // output name
                                      false, // overwrite existing
                                      true, // remove file ext
                                      false, // create log
                                      "TIFF 200dpi Monochrome", // settings
                                      String.Empty, // extension profile
                                      String.Empty, // MIME profile
                                      customSettings, // user settings
                                      String.Empty, // not using remote conversion (DCOM)
                                      String.Empty, // use default working folder
                                      String.Empty );
```

6. If you want to combine files, use the `PNConverter.CombineFiles` method. The files are appended together into a single file or a collection of serialized pages depending on the output format of the profile used, and are combined in the order in which they are given in the list.

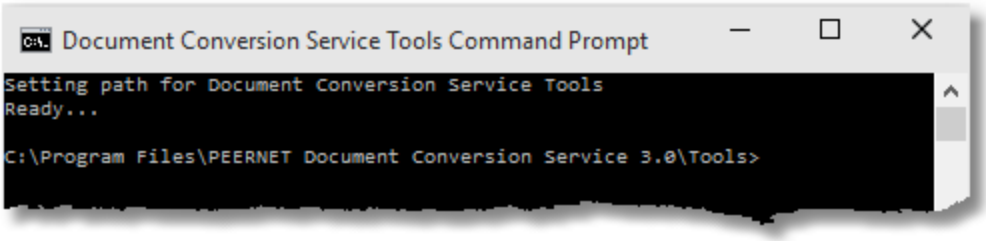
```
PNCombineItem resultItem = null;
IList<String> fileList = new List<String>();

filelist.Add(@"C:\Test\PDF\InputFile1.pdf");
filelist.Add(@"C:\Test\DOC\InputFile2.doc");
filelist.Add(@"C:\Test\XLS\InputFile3.xls");

resultItem =
    PNConverter.CombineFiles(fileList, // files collection
                             @"C:\Test\Output", // output folder
                             "CombinedFile", // name of combined file
                             false, // overwrite
                             false, // create results log
                             "TIFF 200dpi OptimizedColor", // profile
                             String.Empty, // File-ext
                             String.Empty, // MIME
                             null, // user settings
                             String.Empty, // not using remote conversion (DCOM)
                             String.Empty, // use default working folder
                             String.Empty  // Log path
                             );
```

WHAT CONVERSION PROFILES ARE AVAILABLE?

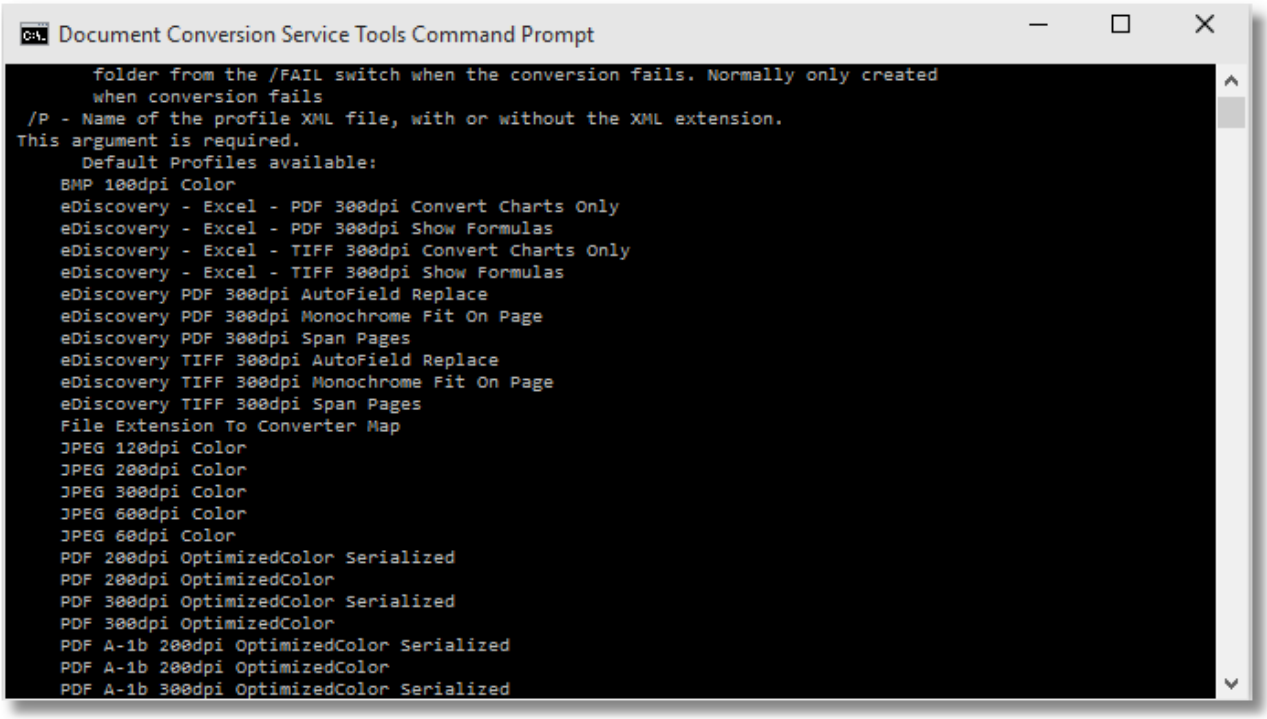
The command line tools and the PEERNET.ConvertUtility.dll use the concept of profiles to control the type of file being created. Document Conversion Service includes several read-only sample profiles for common types of output files for your use. These sample profiles can be used as a base to edit and create your own custom profiles.

1. To see a list of all profiles available, open the Tools Command Prompt.

   From the Start menu, go to Programs - PEERNET Document Conversion Service 3.0 – Tools - Document Conversion Service Tools Command Prompt.



2. Type DCSConvertFile /?, DCSConvertFolder /?, or DCSCombineFiles /?, at the prompt. This will display the command line help for the utilities as well as all profiles installed on your computer.

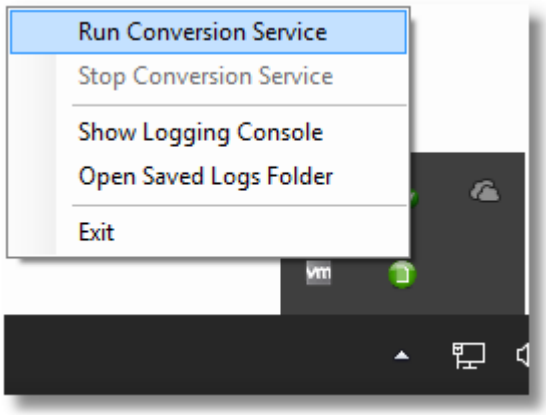


3. To open the folder containing all of the provided profiles go to Start - Programs - PEERNET Document Conversion Service 3.0 - Open Conversion Profiles Folder.

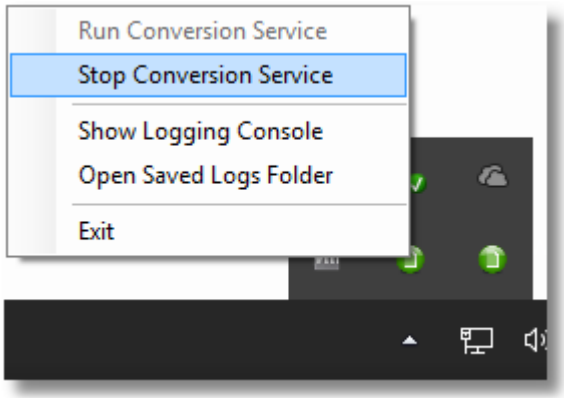
STARTING AND STOPPING THE SERVICE

The Document Conversion Service is installed as an automatic service that will start at system logon. The system tray icon, , gives you quick access to starting and stopping the service.

1. From the system tray icon menu select Run Conversion Service to start the service. If this menu is disabled the service is already running.



2. To stop the service, select Stop Conversion Service. If the service is not running this menu item will be disabled.



CONFIGURING THE SERVICE FOR PARALLEL CONVERSION

The keywords NumberOfDocumentsInParallel and NumberOfPrinters control how many documents can be processed in parallel.

These keywords are set to "*auto*" by default, which means that Document Conversion Service will automatically determine an appropriate value for these numbers based on the number of CPU's and cores on your computer. We recommend you leave this set to "auto" to get the best experience from Document Conversion Service. The maximum number of documents that can be processed in parallel is also restricted by the type of license you have.

The formula used to determine the maximum number of parallel documents is as follows:

```
(number of cores X number of CPU's) × 1.5 = max documents in parallel
```

For example, a single CPU system with 4 cores would be able to process 6 documents in parallel at a time:

```
(4 cores × 1 CPU) × 1.5 = 6 documents in parallel
```

1. To change how many documents Document Conversion Service processes in parallel, go to Start - All Programs - PEERNET Document Conversion Service 3.0 - Edit DCS Configuration File.
2. In the <Settings> section, look for the keyword NumberOfDocumentsInParallel and modify its value to the desired number. If you enter in a number higher than what is allowed by your license model, it will be reduced to the maximum allowed by your license.
3. The keyword above, NumberOfPrinters, controls the size of the Document Conversion Service printer pool. For optimal performance the size of the printer pool should match the NumberOfDocumentsInParallel setting.
4. Important! After you make changes to the configuration file the Document Conversion Service will need to be stopped and started again to pick up the changes.

```
<!-- General configuration options-->
<General>
  <Settings>
   <!-- Maximum number of printers and threads is determined by your license model. -->
   <add Name="NumberOfDocumentsInParallel" Value="3"/>
   <add Name="NumberOfPrinters" Value="3"/>
   ...
  </Settings>
</General>
```

CONTROLLING WHICH CONVERTERS ARE RUNNING

Document Conversion Service will try and load all included converters. You can customize this to only load the converters you need to use.

Most of the included converters use the document's native application to do the conversion. If the required application is not installed, that converter will not be loaded. You will need your own licenses of these applications where applicable.

Each converter can process multiple documents at a time by using a pool of applications. The size of this pool can be adjusted to match how many documents of that type you need to process. By default this pool size is set to "*auto*", which uses your computer's capabilities to determine how many instances to create in the application pool.

This application pool is dynamic and self-regulating. A pool always starts with a single instance of an application and increases the number of applications in the pool, up to the maximum specified, based on demand. When an application in the pool has been idle for a specific amount of time it is removed from the pool.

Each application in the pool will also be recycled (stopped and a new instance started) after processing a preset number of documents, or if the application begins to consume too much memory. Both of these options can be configured as well.

1. To enable or disable a converter, or to change the size of a converter's application pool, go to Start - All Programs - PEERNET Document Conversion Service 3.0 - Edit DCS Configuration File.
2. In the `<AppFactories>` section look for the `<AppFactory>` section for the desired converter.
3. Set the Enabled value to *false* to disable the converter, *true* to always load it, or *auto* to have Document Conversion Service automatically detect if the converter can be loaded.
4. Change MaxInstances to the desired number of application instances in the pool or leave it as *auto* to have the maximum number automatically determined based on your computer's capabilities.
5. The RecycleThreshold setting determines how many documents each application instance will process before it is closed and a new instance started. By default application recycling is set to zero (0), meaning the application never recycles.
6. The virtual size at which an application will recycle is set globally for all converters. To override the global setting of 1.4GB, add RecycleVirtualSizeThreshold to the `<Settings>` section for that converter's `<AppFactory>`. The size is specified in 1024KB blocks, and should be kept below the 2GB virtual size when the application is 32-bit.
7. The application resource limit is controlled through the RecycleGDIandUserHandleCountThreshold value. This value is the maximum number of combined user and GDI handles each application can have before it is recycled.
8. The global setting, AppTeardownIdleTimeout, specifies how long an idle application remains in the application pool before it is removed. The default is one hour.
9. Important! After you make changes to the configuration file the Document Conversion Service will need to be stopped and started again to pick up the changes.

```
<AppFactories>
  <Factories>

    <AppFactory Name="Microsoft Word"
                Type="PEERNET.PNDocConv.Applications.PNWordApplicationFactory"
                Assembly="PNWordApplicationFactory">
      <Settings>
        <add Name="Enabled" Value="auto"/>
        <add Name="MaxInstances" Value="3"/>
        <add Name="RecycleThreshold" Value="0"/>
      </Settings>
    </AppFactory>
    ...
  </Factories>
  <Settings>
    <!-- Global factory settings -->
    <add Name="MaxInstances" Value="auto"/>
    <add Name="RecycleThreshold" Value="0"/>
    <add Name="ReadyThreshold" Value="5000" />
    <add Name="AppInitializationThreshold" Value ="30000" />
    <add Name="RecycleVirtualSizeThreshold" Value="1400000"/>
    <add Name="RecycleGDIandUserHandleCountThreshold" Value="8000"/>
    <add Name="AppTeardownIdleTimeout" Value="3600000"/>
  </Settings>
</AppFactories>
```

COMMON QUESTIONS AND TROUBLESHOOTING

The Document Conversion Service fails to start when I select Run Conversion Service from the system tray.

Document Conversion Service needs to be run under an account with Administrative permissions. In the Services control panel applet, check the account that PEERNET Document Conversion Service Monitor 1.0 is set to run under. This must be a user account and cannot be Local System.

Another common reason for Document Conversion Service to fail to start is an error with the configuration file. If the XML in the configuration file is formatted incorrectly, the service will fail to start. When this happens an error message is added the the Windows Application event log that pinpoints the line that has the format error. Use the Event Viewer from the control panel to find and correct the error.

When I try to convert a file using the Convert a File... menu item I get the error message "*The Document Conversion Service is not running.*".

The service is not running. From the system tray icon menu ( ) select Run Conversion Service to start the service. The service can take several minutes to completely initialize depending on how many converters are configured to run.

The Document Conversion Service is running but my file always fails to convert with the error message "*Converter plug-ins are not available. Please check your Document Conversion Service configuration to ensure one of the Plug-Ins are enabled.*".

The converter for this document type is either disabled in the application configuration file, or it requires a native application that is not installed on the computer. The section Controlling Which Converters Are Running describes how to enable and disable the converters, and the Appendix lists the converters and any required applications. If the converter is enabled and its application, if needed, is installed, go to Start - All Programs - PEERNET Document Conversion Service 3.0 - Open DCS Log Viewer to check the start of the logs for any error messages.

Supported Document Type and Converter Name	Third-Party Application
Converter Name: Adobe Acrobat Reader • Adobe PDF Documents (*.pdf)	Adobe Reader X, XI, DC
Converter Name: Autodesk Design Review • Design Review Drawings (*.dwf)	Autodesk Design Review 2012-2013
Converter Name: Autodesk Design Review • AutoCAD Drawings (*.dwg)	Autodesk Design Review 2012-2013 with DWG TrueView 2012-2013 also installed.
Converter Name: Microsoft Excel • Excel Workbooks (*.xlsx, *.xlsm, *.xls) • Excel Templates (*.xltx, *.xltm, *.xlt) • Excel Binary Workbook (*.xlsb)	Microsoft Office 2003 SP3 *(with Microsoft Office Compatibility Pack)* Microsoft Office 2007 Microsoft Office 2010 *(32-bit and 64-bit)* Microsoft Office 2013 *(32-bit and 64-bit)* Microsoft Office 2016 *(32-bit and 64-bit)*
Converter Name: Ghostscript • Postscript Files (*.ps) • Encapsulated Postscript Files (.eps) • Adobe PDF Documents (*.pdf)	Ghostscript 9.05 or later *(32-bit only)* *There are known handle leak issues with earlier 9.0X versions of Ghostscript.*
Converter Name: PEERNET Image Converter • JPEG images (*.jpg) • TIFF images (*.tif) • Windows Bitmap images (*.bmp) • ZSoft PCX images (*.pcx) • ZSoft DCX images (*.dcx) • CServe Portable Network Graphics images (*.png) • Graphics Interchange Format image files (*.gif) • Icon Format (*.ico) • Windows Media Photo images (*.wdp, *.hdp, *.jxr)	Built-in, no additional applications required.
Converter Name: Internet Explorer • HTML Files (*.htm, *.html) • Secure HTML (*.shtm, *.shtml) • Web Archive (*.mht)	Internet Explorer 8.0 - 11.0
Converter Name: Microsoft Outlook • Outlook Message Files (*.msg) • Outlook Templates (*.oft) • vCard Files (*.vcf) • vCalendar Appointment Files (*.vcs) • iCalendar Appointment Files (*.ics)	Microsoft Office 2003 *(*.oft and *.msg only)* Microsoft Office 2007 Microsoft Office 2010 *(32-bit and 64-bit)* Microsoft Office 2013 *(32-bit and 64-bit)* Microsoft Office 2016 *(32-bit and 64-bit)*

Supported Document Type and Converter Name	Third-Party Application
Converter Name: Outside-In AX Supports over 500 common file formats; see the documentation that came with your Outside In Technology product.	Oracle Outside In Viewer Technology (ActiveX)
Converter Name: Microsoft PowerPoint • PowerPoint Presentations (*.pptx, *.pptm, *.ppt) • PowerPoint Shows (*.ppsx, *.ppsm, *.pps) • PowerPoint Templates (*potx, *.potm, *.pot)	Microsoft Office 2003 SP3 *(with Microsoft Office Compatibility Pack)* Microsoft Office 2007 Microsoft Office 2010 *(32-bit and 64-bit)* Microsoft Office 2013 *(32-bit and 64-bit)* Microsoft Office 2016 *(32-bit and 64-bit)*
Converter Name: Microsoft Publisher • Publisher Files (*.pub)	Microsoft Office 2003 SP3 *(with Microsoft Office Compatibility Pack)* Microsoft Office 2007 Microsoft Office 2010 *(32-bit and 64-bit)* Microsoft Office 2013 *(32-bit and 64-bit)* Microsoft Office 2016 *(32-bit and 64-bit)*
Converter Name: Microsoft Visio • Visio Drawings (*.vsd)	Microsoft Visio 2003 Microsoft Visio 2007 Microsoft Visio 2010 *(32-bit and 64-bit)* Microsoft Visio 2013 *(32-bit and 64-bit)* Microsoft Visio 2016 *(32-bit and 64-bit)*
Converter Name: Microsoft Word • Word Documents (*.docx, *.docm, *.doc) • Word Templates (*.dotx, *.dotm, *.dot) • Rich Text Documents (*.rtf) • Plain Text Files (*.txt) • Plain Text Log Files (*.log)	Microsoft Office 2003 SP3 *(with Microsoft Office Compatibility Pack)* Microsoft Office 2007 Microsoft Office 2010 *(32-bit and 64-bit)* Microsoft Office 2013 *(32-bit and 64-bit)* Microsoft Office 2016 *(32-bit and 64-bit)*
Converter Name: Microsoft XPS • XPS Documents (*.xps) • Open XPS Documents (*.oxps)	Uses Windows built-in XPS document support, no additional applications required.
Converter Name: PEERNET Passthrough • Any file type	Built-in, passes the file through the system without converting.