

EPO Merge PDF

2.4

by EPO Consulting

EPO PDF Merge is a console application which merges/collects pdf and pictures into one resulting pdf file.

E-Mail: info@epoconsulting.com

Please use this e-Mail for reporting bugs or making suggestions, how we can improve the EPO PDF merge.

Last Update: May 2010
Status: Released

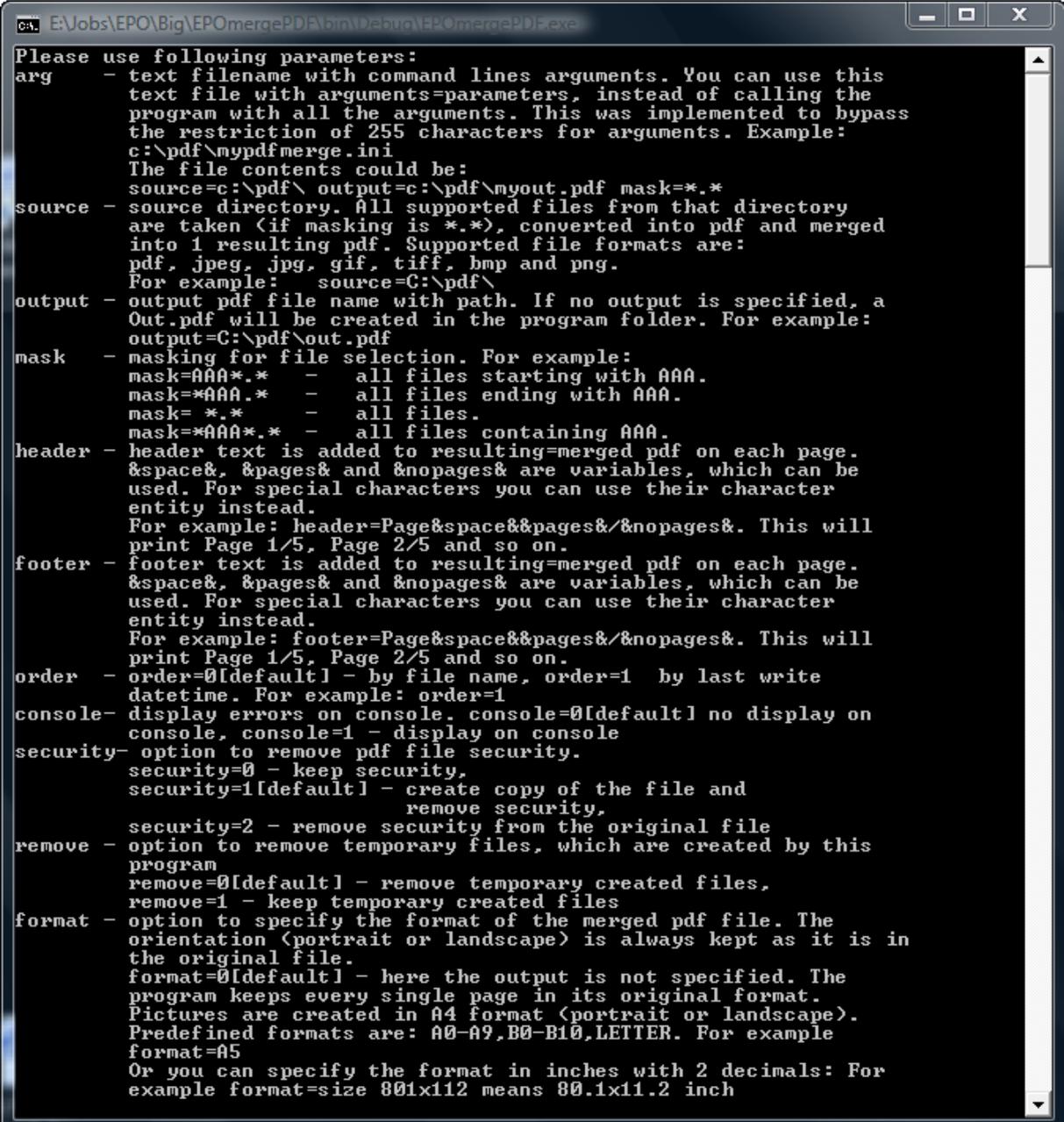


Provided by
EPO Consulting
E-Mail info@epoconsulting.com
Internet www.epoconsulting.com

*Microsoft is the trademark or registered trademark of Microsoft in USA and in several other countries.
All other trademarks or registered trademarks belonging to their respective owners. All rights reserved.*

Version changes:

- 2.0 New function - convert restricted pdf files into pictures
- 2.1 New function - remove restrictions from pdf
New function - load file with parameters as a argument (1- argument in the command line)
- 2.2 Call program with arguments change to using a names for the arguments
- 2.2 Supported different output formats for the pdf document
- 2.3 New parameters orientation, margintop, marginleft, marginright, marginbottom, see documentation about new parameters
- 2.4 New parameter error, improved pdf merging and conversions into pictures



```
C:\E:\Jobs\EPO\Big\EPOmergePDF\bin\Debug\EPOmergePDF.exe
Please use following parameters:
arg - text filename with command lines arguments. You can use this
text file with arguments=parameters, instead of calling the
program with all the arguments. This was implemented to bypass
the restriction of 255 characters for arguments. Example:
c:\pdf\mypdfmerge.ini
The file contents could be:
source=c:\pdf\ output=c:\pdf\myout.pdf mask=*. *
source - source directory. All supported files from that directory
are taken (if masking is *. *), converted into pdf and merged
into 1 resulting pdf. Supported file formats are:
pdf, jpeg, jpg, gif, tiff, bmp and png.
For example: source=C:\pdf\
output - output pdf file name with path. If no output is specified, a
Out.pdf will be created in the program folder. For example:
output=C:\pdf\out.pdf
mask - masking for file selection. For example:
mask=AAA*. * - all files starting with AAA.
mask=*AAA. * - all files ending with AAA.
mask= *. * - all files.
mask=*AAA*. * - all files containing AAA.
header - header text is added to resulting=merged pdf on each page.
&space&, &pages& and &nopages& are variables, which can be
used. For special characters you can use their character
entity instead.
For example: header=Page&space&&&pages&/&nopages&. This will
print Page 1/5, Page 2/5 and so on.
footer - footer text is added to resulting=merged pdf on each page.
&space&, &pages& and &nopages& are variables, which can be
used. For special characters you can use their character
entity instead.
For example: footer=Page&space&&&pages&/&nopages&. This will
print Page 1/5, Page 2/5 and so on.
order - order=0[default] - by file name, order=1 by last write
datetime. For example: order=1
console- display errors on console. console=0[default] no display on
console, console=1 - display on console
security- option to remove pdf file security.
security=0 - keep security.
security=1[default] - create copy of the file and
remove security.
security=2 - remove security from the original file
remove - option to remove temporary files, which are created by this
program
remove=0[default] - remove temporary created files,
remove=1 - keep temporary created files
format - option to specify the format of the merged pdf file. The
orientation (portrait or landscape) is always kept as it is in
the original file.
format=0[default] - here the output is not specified. The
program keeps every single page in its original format.
Pictures are created in A4 format (portrait or landscape).
Predefined formats are: A0-A9,B0-B10,LETTER. For example
format=A5
Or you can specify the format in inches with 2 decimals: For
example format=size 801x112 means 80.1x11.2 inch
```

```

example format=size 801x112 means 80.1x11.2 inch
orientation -option to specify page orientation of pdf pages produced from pictures.
orientation=0 [default] is generating page orientation automatically from the picture size
orientation=1 portrait for all pdf pages generated from pictures
orientation=2 landscape for all pdf pages generated from pictures
marginleft -option to specify left margin for the picture position of pdf pages produced from the picture in points.
marginright -option to specify right margin for the picture position of pdf pages produced from the picture in points.
margintop -option to specify top margin for the picture position of pdf pages produced from the picture in points.
marginbottom -option to specify bottom margin for the picture position of pdf pages produced from the picture in points.
error -what will happen, if error will be occurred
error = 0 [default], will be a new empty page created with the error message in
error = 1 , will be a new empty pdf file with one page created with the error message in
error = 2 , will be a new empty page and file created with the error message in
error = 3 , no error log

```

Installation:

There is no installation routine included, just copy files into a folder and use it with following arguments.

Program files:

- textsharp.dll open source library
- gsdll32.dll or gsdll64.dll open source library for 32bit or 64bit Windows
- EPOmergePDF.exe command line tool
- gs.dll
- gsdll32.lib or gsdll64.lib
- EPOmergePDF.pdf help

Additional files:

- argumentFileExample.txt An example for an argument file (which can be used instead of calling EPOmergePDF.exe with all arguments in the command line (only parameter arg is needed then))
- batFileExample.txt Example batch file. Rename it to [yourbatfile.bat] and start testing

EPOmergePDF.exe uses the following parameters (=arguments):

- arg** - text filename with command lines arguments. You can use this text file with arguments=parameters, instead of calling the program with all the arguments. This was implemented to bypass the restriction of 255 characters for arguments. Example:
c:\pdf\mypdfmerge.ini
The file contents could be:
source=c:\pdf\ output=c:\pdf\myout.pdf mask=*.*
- source** - source directory. All supported files from that directory are taken (if masking is *.*), converted into pdf and merged into 1 resulting pdf. Supported file formats are:
pdf, jpeg, jpg, gif, tiff, bmp and png.
For example: source=C:\pdf\

output - output pdf file name with path. If no output is specified, a Out.pdf will be created in the program folder. For example:
output=C:\pdf\out.pdf

mask - masking for file selection. For example:
mask=AAA*.* - all files starting with AAA.
mask=*AAA.* - all files ending with AAA.
mask= *.* - all files.
mask=*AAA*.* - all files containing AAA.

header - header text is added to resulting=merged pdf on each page. &space&, &pages& and &nopages& are variables, which can be used. For special characters you can use their character entity instead.
For example: header=Page&space&&pages&/&nopages&. This will print Page 1/5, Page 2/5 and so on.

footer - footer text is added to resulting=merged pdf on each page. &space&, &pages& and &nopages& are variables, which can be used. For special characters you can use their character entity instead.
For example: footer=Page&space&&pages&/&nopages&. This will print Page 1/5, Page 2/5 and so on.

order - order=0[default] - by file name, order=1 by last write datetime. For example: order=1

console - display errors on console. console=0[default] no display on console, console=1 - display on console

security - option to remove pdf file security.
security=0 - keep security,
security=1[default] - create copy of the file and remove security,
security=2 - remove security from the original file

remove - option to remove temporary files, which are created by this program
remove=0[default] - keep temporary created files,
remove=1 - remove temporary created files

format - option to specify the format of the merged pdf file. The orientation (portrait or landscape) is always kept as it is in the original file.
format=0[default] - here the output is not specified. The program keeps every single page in its original format. Pictures are created in A4 format (portrait or landscape). Predefined formats are: A0-A9,B0-B10,LETTER. For example format=A5
Or you can specify the format in inches with 2 decimals: For example format=size 801x112 means 80.1x11.2 inch

orientation - option to specify page orientation pdf's produced from the pictures
orientation = 0 (default, is generating page orientation from the picture size
orientation = 1 [portrait always]
orientation = 2[landscape always]

marginleft -option to specify left margin for the picture position in the pdf document, produced from the picture

marginright -option to specify left margin for the picture position in the pdf document, produced from the picture

margintop -option to specify left margin for the picture position in the pdf document, produced from the picture

marginbottom -option to specify left margin for the picture position in the pdf document, produced from the picture

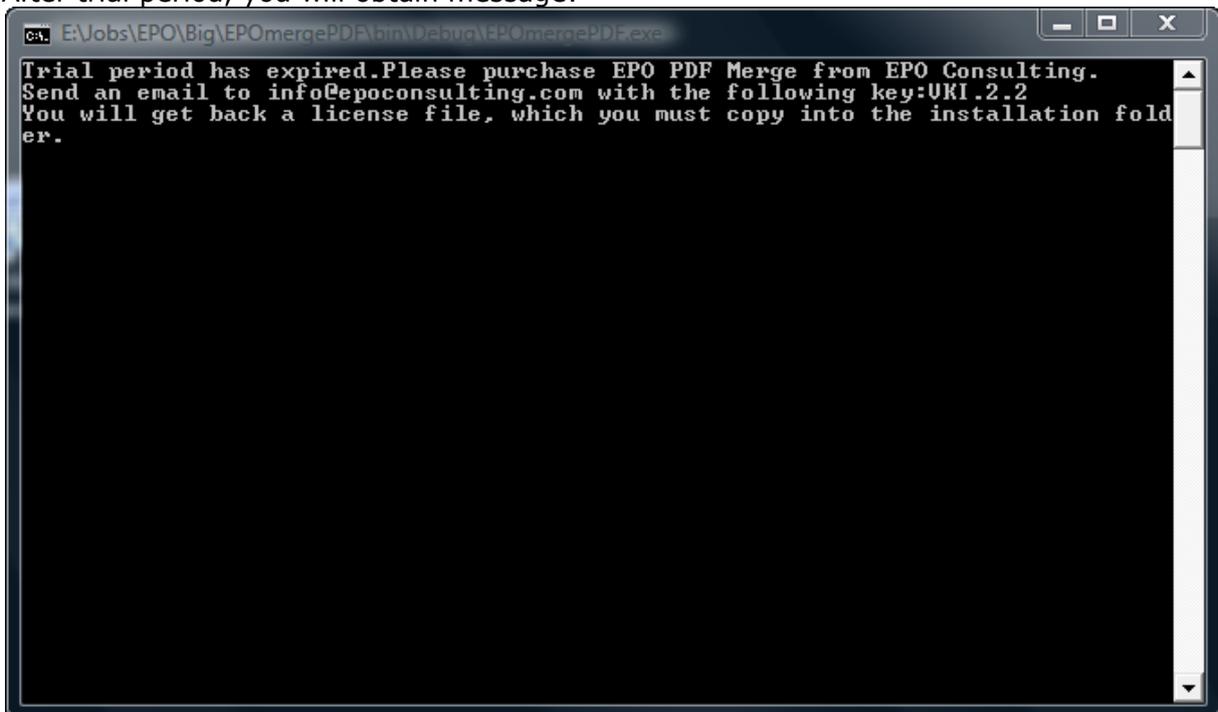
error -what will happen, if error will be occurred
error = 0 [default], will be a new empty page created with the error message in the output pdf
error = 1 , will be a new empty pdf file with one page created with the error message in (no info in the output pdf)
error = 2 , will be a new empty page and file created with the error message in the output pdf
error = 3 , no error log

Response code:

- 0 OK - Done
- 1
- 2 System.ArgumentException
- 3 System.IO.DirectoryNotFoundException
- 4 System.IO.PathTooLongException
- 5 System.IO.IOException
- 6 System.UnauthorizedAccessException
- 7 System.NotSupportedException
- 8 System.Security.SecurityException
- 9 Exception ex
- 10 Could not delete pdf file (pdf file is created and could not be accessed)
- 11 Source directory not exists!
- 12 Destination file does not contain extension!
- 13 Destination file does not contain mask!
- 14 Please use 3 or 6 arguments instead of xxx

Request for licence file:

After trial period, you will obtain message:



Please send to info@epoconsulting.com generated key (example PDF.TDV.2.2) and you will get back a EPOLicence.lic file. Then you need to copy file into program bin directory and program will be activated.

Examples:

Command line examples:

```
EPOMergePDF.exe source=c:\temp\MergePDF\ output=c:\temp\MergePDF\myresult.pdf  
mask=*.*
```

- Will merge all files from c:\temp\MergePDF\ into myresult.pdf

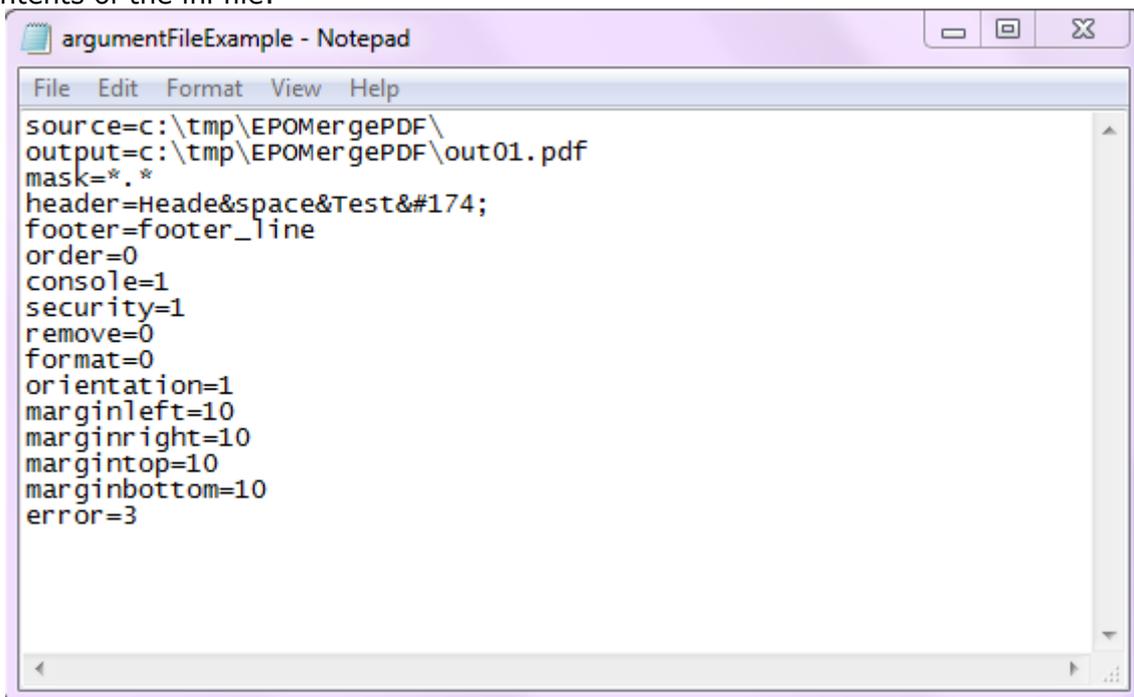
```
EPOMergePDF.exe source=c:\temp\MergePDF\ output=c:\temp\MergePDF\a.pdf  
mask=*. * header=TestHeader&space&&pages& footer=TestFooter&space&&pages&
```

- Will merge all files from c:\temp\MergePDF into a.pdf with defined header and footer

Example with using a separate argument (ini) file:

EPOMergePDF.exe arg=c:\temp\argumentFileExample.txt will take the parameters from the text (ini) file.

Contents of the ini file:



```
source=c:\tmp\EPOMergePDF\  
output=c:\tmp\EPOMergePDF\out01.pdf  
mask=*. *  
header=Heade&space&Test&#174;  
footer=footer_line  
order=0  
console=1  
security=1  
remove=0  
format=0  
orientation=1  
marginleft=10  
marginright=10  
margintop=10  
marginbottom=10  
error=3
```

Notes:

1. There are no spaces allowed in the arguments values (use parameter &space& instead of)
2. Arguments source, output, mask are mandatory (can be in arg file)
3. If there will be no path defined for the resulting pdf, then it will be created in the program directory
4. Allowed masking arguments are *xxx*, *xxx and xxx*. Masking with more stars inside is not working! Not allowed examples: *xxx*xxx*
5. Program is able to remove security settings
6. If there is a problem with file conversion, program will create a png file from the pdf and then convert the png back into pdf.

7. If you want to use special characters in the header or footer you can use UNICODE representation instead of the special characters.

Examples:

- a. Ë Ë
- b. Ö ő
- c. Š Š
- d. Ÿ ũ
- e. Ø Ǿ
- f. © ©
- g. ® ®

+++ end of document +++