

comparepdfcmd

Mark Summerfield

Copyright © Qtrac Ltd. 2018-19. All Rights Reserved.

Contents

Introduction	2
Examples	3
Installing	5
System Requirements	5
Obtaining and Installing	5
How to Register a Free Trial License Key	6
How to Buy and Register a Full License Key	6
How to Buy and Register a Site License Key	8
How to View the License	8
How to Upgrade	8
How to Get Support	9
How to Uninstall	9
Tips	9
Configuration	10
Exit Status Code	10
Setting Your Own Defaults	11
Option Names and Values	11
Alternative Options	12
Appearance Mode Options	12
Accuracy Option	12
Square Size Option	13
Comparison Mode Option	13
Configuration File Option	14
Highlight Transparency Option	15
Exclude Options	15
Exclude Corners Options	16
Exclude Margins Options	16
Exclude Rectangles Options	17
Pages Options	17

Report Options	18
Report Decimals Option	18
Report Filename Option	18
Report Format Option	19
Report Output Option	20
Report Path Option	21
Report Path in Title Option	21
Report Scale Option	21
Report Show Option	21
Report Summary Option	22
Text Mode Options	22
Change Bar Option	22
Column Tolerance Option	23
Ignore Hyphens Option	23
Line Tolerance Option	23
Normalize Hyphens Option	24
Normalize Ligatures Option	24
Text Comparison Algorithm Option	24
Verbose Option	25
The PDF Files or Folders to Compare	25
Troubleshooting	25
Invalid Key Code Error	25
False Positives	25

Introduction

The *comparepdfcmd* application is a command line program for comparing two PDF files or two folders of PDF files that runs in a console (i.e., in a Command Prompt window opened by running `cmd.exe`).^{*} The program can report the differences in several ways—for example, by showing the differences visually in PDF or PNG format, or by outputting a CSV or JSON file identifying words that have been inserted, deleted, or replaced.

The application has many use cases; here are some examples:

Authors and Publishers

If you produce PDFs for publication, a PDF comparison tool can show you the exact differences between, say, the first printing, and your proposed second printing. This way you can easily spot unintended changes to the pagination. Also, you can save time by

^{*} If you need an easy-to-use graphical PDF comparison application, try *ComparePDF*: www.qtrac.eu/comparepdf.html.

checking only those pages that have changed, rather than having to review the whole publication all over again.

Journalists, Police, and Other Investigators

It is common to follow a paper trail of emails, memos, and so on, as part of an investigation. This trail will often include PDF documents. Using a PDF comparison tool will allow you to observe the evolution of a document—perhaps revealing when critical changes were made.

Legal Professionals, Legislators, and Freelancers

When a contract, legislation, or other legal document is being drawn up, how can you be sure that the changes made are correct, and are only those agreed? By using a PDF comparison tool you can see exactly what has changed between the latest draft and any previous draft.

Managers

As reports and other documents are developed they go through many changes. If your organization's process uses PDFs, using a PDF comparison tool will show you how each new version differs from its predecessor. This saves you having to reread everything, and makes it much easier to see what changes you want to comment on or change.

Software Testers and Quality Assurance Engineers

Many business processes produce PDFs as an output. Using a PDF comparison tool makes it easy to verify that the PDFs are being produced correctly, or to spot exactly where any mistakes have occurred.

Comparing PDF files is not necessarily straightforward: for example, do you consider two PDFs the same if they have the same words in the same order—but laid out differently (e.g., due to slightly different margins or word-wrapping)? The *comparepdfcmd* program provides three comparison modes: word-by-word, character-by-character, and appearance. The first two (text modes) are concerned only with the characters or words and ignore line wrapping and margins; the appearance mode compares the appearance of each pair of pages and so detects differences in layout, differences in graphics, and differences in colors and fonts. The program uses sensible defaults for ease of use, and also supports many options to support more sophisticated requirements.

Examples

These examples (and those throughout this manual) make three assumptions:

1. That you have registered a free trial licence key (which hasn't expired), or a full license key, or a site license key; (see [How to Register a Free Trial License Key](#), [How to Buy and Register a Full License Key](#), or [How to Buy and Register a Site License Key](#));
2. That you replace the paths used in the examples, e.g., C:\Users\mark for home, E:\pdfs for a folder of PDF files, and the E:\diff folder for outputting difference reports, with your own equivalents.

3. That you have added *comparepdfcmd*'s folder to your %PATH% (see [Add comparepdfcmd to the PATH](#)), or your current directory is *comparepdfcmd*'s folder, or where you see “comparepdfcmd” in the examples, you actually precede this with its full path, e.g., C:\Users\mark\comparepdfcmd\comparepdfcmd.

Here's how to return a value (%errorlevel%) to the operating system (e.g., for use in a .bat file):

```
C:\Users\mark>comparepdfcmd E:\pdfs\a1.pdf E:\pdfs\a2.pdf
```

To get some human-readable feedback use the verbose option (either `-v` or `--verbose`), e.g.:

```
C:\Users\mark>comparepdfcmd -v E:\pdfs\b1.pdf E:\pdfs\b2.pdf
Different
```

This will return a value to the operating system as described above, and will also output either “Same” or “Different” depending on whether the two PDFs have the same words or not.*

If you want to see what the differences are, tell *comparepdfcmd* to output a difference report using the `-r` or `--report` option. For example:

```
C:\Users\mark>comparepdfcmd -v -s -r E:\diff\report.pdf E:\pdfs\c1.pdf E:\pdfs\c2.pdf
Wrote "E:\diff\report.pdf"
```

The report.pdf file will contain the different pages side-by-side and with the differences highlighted.[Ⓞ] In addition, due to the `-s` or `--show` option, the report will be shown in your PDF reader.

Incidentally, if c1.pdf and c2.pdf were the same, no report would be produced and the program would display “Same” rather than “Wrote “E:\diff\report.pdf””. And, of course, no report would be shown in your PDF reader.

If you want to compare two *folders* of PDFs which have the same names, you can give the folder names instead of the names of two PDFs, e.g.:

```
C:\Users\mark>comparepdfcmd -f pdf -R E:\diff E:\pdfs\old E:\pdfs\new
"Result", "PDF#1", "PDF#2", "Report or Error"
"Missing#1", "E:\pdfs\new\only_in_new.pdf",
"Missing#2", "E:\pdfs\old\only_in_old.pdf", ,
"Same", "E:\pdfs\old\one.pdf", "E:\pdfs\new\one.pdf",
"Different", "E:\pdfs\old\two.pdf", "E:\pdfs\new\two.pdf", "E:\diff\two.pdf"
"Different", "E:\pdfs\old\three.pdf", "E:\pdfs\new\three.pdf", "E:\diff\three.pdf"
```

The `-f` or `--format` option specifies the format of the difference report files; here it is PDF format (see the [Report Format Option](#)). The `-R` or `--reportpath` option specifies where the

* Note that colors and font effects (e.g., bold) shown in the examples, have no significance—they are used purely to aid understanding.

[Ⓞ] PDF reports produced by *comparepdfcmd* are much higher quality than those produced by *DiffPDF* or *diffpdfc* since they contain scalable copies of the original pages, not fixed sized “photographs” of the pages.

reports should go (see the [Report Path Option](#)). These options are followed by the two folders to search for PDF files to compare. The program shows the results in CSV format on the console, although you can tell it to save the summary into a file instead using the `-S` or `--summary` option (see the [Report Summary Option](#)).

Many more command line options are supported, including ones to exclude margins from comparisons (e.g., to ignore timestamps in a header or footer). Even more options can be set in `.ini` configuration files. All this is explained in the [Configuration](#) section.

Installing

System Requirements

The *comparepdfcmd* program runs on 64-bit Windows.* For desktops and laptops, *comparepdfcmd* will run on Windows 7, Windows 8, and Windows 10; and for servers, on Windows Server 2008 or later. The machine must have an x86-compatible processor (i.e., most desktop and laptop computers).

Since PDF comparisons are computationally expensive, it is best to use a machine with a fast multicore processor, ideally with a processor speed of at least 2GHz that has at least two cores. *comparepdfcmd* will take advantage of all available cores, so in general will run twice as fast on a quad core machine than on a dual core machine with the same clock speed—and faster still on machines with more and faster cores.

The *comparepdfcmd* program makes use of the system's default PDF reader, e.g., to show this manual, to show the license, and to show PDF difference reports when told to. So, naturally, a PDF reader must be installed.◊

Obtaining and Installing

The program is provided in a single `.zip` file, e.g., `comparepdfcmd-1.1.5.zip`. This file can be downloaded from www.qtrac.eu/comparepdfcmd.html.

Once the `.zip` has been downloaded, unzip it in any convenient folder. This is all that is required for installation—*comparepdfcmd* does not touch the Windows registry, and does not change the `%PATH%`. However, having *comparepdfcmd* in your `%PATH%` can be very convenient, so we recommend that you permanently add it. (See the [Add *comparepdfcmd* to the PATH](#) sidebar.) Once the software has been installed, a valid license key must be registered for it to work. (This shouldn't be necessary if upgrading from one minor or patch version to another, or when using a site license key—see [How to Upgrade](#).)

Note that the software is the same regardless of the license key. A license key restricts the period of use (20 days for a free trial, one year for an annual site license), or the computer the software can be used on (for free trials and full licenses), but does *not* restrict what the software can do: the full functionality is available regardless of the license key type.

* If you require a PDF comparison program that runs on 32-bit windows, try *diffpdfc*: www.qtrac.eu/diffpdfc.html.

◊ We prefer *SumatraPDF* www.sumatrapdfreader.org but Acroread, Acrobat, Evince, or any other should work fine.

Add *comparepdfcmd* to the PATH

Click the Start button, then click Control Panel. In the Search box, enter “path”. This should give you a link to “Edit environment variables for your account”. Click this link. This should pop up the Environment Variables dialog.

Click the Path variable in the top list of user variables, then click the Edit button. Click the Variable value’s line editor, then press End to move the cursor to the very end. Now enter a semi-colon “;” followed by the drive and path to the folder you have unzipped *comparepdfcmd* into. For example, ;"C:\Users\mark\comparepdfcmd". Then click OK, then close the Control Panel window.

If you upgrade to a later version of *comparepdfcmd*, providing you unzip to the same folder as before, the %PATH%, license key file, and any configuration file, will all continue to be valid.

How to Register a Free Trial License Key

If you or someone on your behalf has bought a site license, or if you wish to buy a site license, skip ahead to [How to Buy and Register a Site License Key](#).

First, make sure that you have downloaded and installed *comparepdfcmd* (see [Obtaining and Installing](#)).

Make sure that the person logged in is the person who is to use *comparepdfcmd*.

Open a console (i.e., a Command Prompt window opened by running cmd.exe). If *comparepdfcmd*’s folder isn’t in your %PATH%, change directory (cd) to *comparepdfcmd*’s folder.

Run the program from the command line with the try option:

```
C:\Users\mark\>comparepdfcmd try
```

This will try to acquire a free trial license key over the internet and then register it. If this succeeds you can use *comparepdfcmd* for up to 20 days for free.

If this fails (e.g., if your security settings don’t allow *comparepdfcmd* to access the internet), a link for you to copy and paste into your web browser will be output on the console.

Free trial license keys are valid for 20 days. Only one free trial is allowed per computer per major version. To use *comparepdfcmd* after the trial you will need to buy either a full or site license.

How to Buy and Register a Full License Key

If you or someone on your behalf has bought a site license, or if you wish to buy a site license, skip ahead to [How to Buy and Register a Site License Key](#).

The first step is to make the purchase. For this visit www.qtrac.eu/buy?app=comparepdfcmd in your web browser. You can visit the link directly from this manual, or you can run *comparepdfcmd* with the buy option:

```
C:\Users\mark\>comparepdfcmd buy
```

Your web browser should be opened (or a new tab added if the browser is already running), at the link shown—and then immediately redirected to the MyCommerce purchase page. However, if *comparepdfcmd* isn't able to open the link, copy the link to the clipboard, then paste it into your browser's location line editor, then press Enter to visit the page.

Once you visit the link, it will immediately redirect you to the secure MyCommerce purchase page which shows the price tiers. Tiered price discounts are applied if you buy at least 10 license keys (for the same product) in a single order.

Once the purchase is complete the confirmation page will show a “get key” link with your Order ID. This link and Order ID will also be emailed to you.

The link or Order ID can be used to acquire as many license keys as licenses that were purchased. Below are the steps to take on any given computer.

First, make sure that you have downloaded and installed *comparepdfcmd* (see [Obtaining and Installing](#)).

Make sure that the person logged in is the person who is to use *comparepdfcmd*.

Open a console (i.e., a Command Prompt window opened by running `cmd.exe`). If *comparepdfcmd*'s folder isn't in your `%PATH%`, change directory (`cd`) to *comparepdfcmd*'s folder.

Run the following command, replacing *order-id* with your actual Order ID (which will look *similar* to this: 1B25F83-E9ABC.11):

```
C:\Users\mark\>comparepdfcmd getkey order-id
```

This will open your web browser at the “get key” page with the Order ID and Computer ID correctly filled in.*

With the Computer ID filled in, click the Get License Key button. This will change the page to show the full license key, and a command line to execute to register the key. The command line will look *similar* to this:

```
comparepdfcmd.exe register c8fdc993e4479c9f446e3b5b71e13c8cdc7630675518
```

Copy the command line to the clipboard, then paste it into the console window, then press Enter to run it. This will create a file called `comparepdfcmd.key` which will contain the full license key. *comparepdfcmd* will attempt to save this file in *comparepdfcmd*'s own folder, or failing that in the user's home folder (`%USERPROFILE%`).

Note that every free trial and full license key is tied to a particular computer and will not work on any other computer. If you want a license key that isn't tied to any specific computer, see [How to Buy and Register a Site License Key](#).

*If your security settings prevent *comparepdfcmd* from visiting the page, run `comparepdfcmd cid` to get the Computer ID, then visit the “get key” page in your browser, and copy and paste in the Computer ID.

How to Buy and Register a Site License Key

To buy an annual site license that can be used on up to 1000 computers and is not tied to any particular computer, visit www.qtrac.eu/sitelicense.html.

Once a site license purchase has been completed the confirmation page will show the site license key; and the key will also be emailed to the purchaser.

To register a site license key, the key must be saved in a plain text file. For example, copy the license key from the confirmation page or from the email, then run Notepad and paste in the key. Then click File→Save As. Save the key into *comparepdfcmd*'s folder in a file called `qtrac.key`; be sure to set the Save as type to All Files (*.*) and the Encoding to UTF-8.

The site license key and the paragraph of instructions above can be distributed within the organisation to everyone who is to use *comparepdfcmd*. Alternatively, a system administrator could create the `qtrac.key` file saved in *comparepdfcmd*'s folder, and then create a new .zip of the folder and distribute that so that end-users would find that *comparepdfcmd* would automatically work. (Although they would still probably need to [Add *comparepdfcmd* to the PATH.](#))

Note that everyone to whom the key or `qtrac.key` file is distributed must be made aware of the license terms and in particular that the key (or any file containing it such as `qtrac.key`) must *never* be given or sold to any third-party, and must never be made publically available in any way (e.g., in print, on social media, or the Internet).

How to View the License

The program comes with two licenses, one that governs free trial and full license keys, and another that governs site licenses. If you have registered a license key you can view the license with this command:

```
C:\Users\mark\>comparepdfcmd license
```

This will show the license in your PDF reader, or if it can't, it will output the license to the console. Alternatively, you can obtain the license as a PDF from *comparepdfcmd*'s web page www.qtrac.eu/comparepdfcmd.html. Or, you can read the license PDF from *comparepdfcmd*'s folder since both are included in the .zip file. The PDF for free trial and full licenses is `comparepdfcmd-license.pdf`, and for site licenses is `qtrac-site-license.pdf`.

How to Upgrade

To upgrade *comparepdfcmd*, download the latest version from www.qtrac.eu/comparepdfcmd.html. Then unzip the .zip file (e.g., `comparepdfcmd-1.1.5.zip`) on top of the existing installation. This will replace the application itself, but will leave any key file (`comparepdfcmd.key` or `qtrac.key`) and any configuration file (`comparepdfcmd.ini`) intact.

If you prefer to unzip to a different folder, after unzipping, copy any of the files, `comparepdfcmd.key`, `qtrac.key`, and `comparepdfcmd.ini` that exist in the original folder, into the new folder. You may also need to change your `%PATH%` to use the new folder.

A full license key entitles you to use any minor or patch release of the major version you purchased. For example, if you bought a full license for version 1, you can use *any* 1.x.y version (1.1.0, 1.2.3, etc.) that is or becomes available.

A site license key entitles you to use *any* version of *comparepdfcmd*, as well as *ComparePDF*, *DiffPDF* 5.8.3 or later, and *diffpdfc* 5.8.3 or later, within the license’s one year lifetime.

How to Get Support

It is worth reading this manual once through to become familiar with what *comparepdfcmd* can do. If this manual doesn’t answer your question, then we’ll happily try to answer.

If you believe you have found a bug, first check which version you are using:

```
C:\Users\mark\>comparepdfcmd --version
```

Then visit *comparepdfcmd*’s web site, www.qtrac.eu/comparepdfcmd.html, to see if there is a newer version. If a new version is available, upgrade to it (see [How to Upgrade](#)), and then see if you still have the bug. If the problem persists, contact us and we will try to help.

Note that changes between versions are listed at www.qtrac.eu/comparepdfcmd-changes.html.

Email support is provided on a “best efforts” basis with no formal time limit, so long as Qtrac Ltd. is trading. The support email address is support@qtrac.eu — be sure to tell us what operating system you are using (e.g., “Windows 10”) and especially which version of *comparepdfcmd* you are using. If you have a full or site license, please also give us the purchase’s Order ID.

How to Uninstall

To uninstall *comparepdfcmd*:

1. Delete *comparepdfcmd*’s folder;
2. Delete *comparepdfcmd.ini* and *comparepdfcmd.key* from your home folder (if they exist);
3. Remove *comparepdfcmd* from your %PATH% (if you added it).

No other steps are needed since *comparepdfcmd* does not touch the Windows registry.

Tips

By default, *comparepdfcmd* will compare PDFs using words mode. This will produce the best results in a wide range of situations. If you want to compare text written in a CJK—Chinese, Japanese, Korean—language, it is best to use chars (characters) mode. And if you want to compare colors, fonts, diagrams, images, or text layout, you should use appearance mode. See the [Comparison Mode Option](#).

If your PDFs have different numbers of pages or inserted or deleted pages, you can use page ranges to specify those pages from each PDF that correspond to each other. See the [Pages Options](#).

The default text comparison algorithm should produce good results for single column PDFs, but for multicolumn PDFs it is usually best to use the multicolumn comparison algorithm: see the [Text Comparison Algorithm Option](#).

It is also possible to tweak the comparison algorithm itself by setting the [Line Tolerance Option](#), and for the multicolumn algorithm, both the [Line Tolerance Option](#) and the [Column Tolerance Option](#). In particular, reducing the [Line Tolerance Option](#) from its default of 10 to 9 or 8 may help improve results for PDFs whose inter-line spacing is narrower than usual.

Configuration

When *comparepdfcmd* runs it must be told which PDF files or folders to compare. And in most cases you'll want to specify extra options, for example, to say what kind of comparison you want done, and how to report differences. You may also want to change some of the defaults it uses. This can be done by creating a `comparepdfcmd.ini` configuration file and putting it in one of the locations that *comparepdfcmd* searches for configurations; or by creating a configuration file with any name you like and using the `-C` or `--config` option to tell *comparepdfcmd* where to find it (see the [Configuration File Option](#)). Here's how the configuration process works:

1. *comparepdfcmd* starts with every option set to a sensible default wherever possible;
2. Options are read from the file `COMPAREPDFCMD\comparepdfcmd.ini` if it exists (where `COMPAREPDFCMD` is the folder where `comparepdfcmd.exe` is located);
3. Options are read from the file `%USERPROFILE%\comparepdfcmd.ini` if it exists (i.e., in your home folder);
4. Options are read from the file `comparepdfcmd.ini` (i.e., in the current folder) if it exists;
5. If the `-C` or `--config` command line option is given, options are read from the file it specifies;
6. Any options given on the command line (apart from `-C` or `--config`) are read.

Once the options have been read and a final configuration is reached, *comparepdfcmd* then performs the comparison in accordance with the configuration.

Exit Status Code

During the comparison, and when the comparison is complete, *comparepdfcmd* may output messages depending on the options that have been set. And at the end the program returns a value to the operating system (accessible as `%errorlevel%` in `.bat` files). The value will be 0 if there were no errors and no differences were found, 1 if there were no errors and differences were detected, or 2 if an error occurred.

Setting Your Own Defaults

If you want to change any of *comparepdfcmd*'s defaults, create a plain text file called `comparepdfcmd.ini` and save it in the same folder as *comparepdfcmd*'s executable or in your home folder (%USERPROFILE%). For any default you want to change, put your preferred default in this file. For example, if you want *comparepdfcmd* to default to doing appearance mode comparisons, add these lines:

```
[general]
mode=appearance
```

Examples of how to change other defaults are given wherever applicable in the explanations of *comparepdfcmd*'s options that follow. You can also copy parts of the `example.ini` file that you'll find in *comparepdfcmd*'s folder and paste them into your own `comparepdfcmd.ini` file, editing them to suit your needs.

Option Names and Values

Command line options are always specified Unix-style, e.g., `-v` or `--verbose`, *not* Windows-style (`/v`). Command line option names are case-sensitive.

In configuration files, blank lines are ignored, and so is any text that begins with a semi-colon. This makes it possible to add comments if you want. (For an example, see the `example.ini` file in *comparepdfcmd*'s folder.) Configuration file section names and key names are *not* case-sensitive.

Most option values are simple pieces of text or numbers, e.g., “words” or “30”. Numbers that represent lengths or coordinates are assumed to be points ($\frac{1}{72}$ ” or about 0.353mm). The top-left of the page is point (0, 0), with positive *y* coordinates moving down, and positive *x* coordinates moving right.

Some option values are used to turn a feature on or off. For these, a value of “1”, “true”, “t”, “yes”, “y”, or “on”, will turn the feature on; and any other value will turn it off. (And no value at all will leave it at its default.)

Page numbers are always counted from 1, regardless of the page numbers used in the PDF itself (e.g., i, ii, iii, ..., 1, 2, 3, ...).

The most important option is the [Comparison Mode Option](#). This defaults to words mode, so doesn't even need to be set if that's the kind of comparison you want (and providing you haven't changed this default in a configuration file). If you want to *see* the differences, then you'll need to use the [Report Options](#). If the pages don't exactly correspond (e.g., due to inserted or deleted pages), or you only want to compare some specific pages, then you'll need to use the [Pages Options](#). And, of course, you must specify [The PDF Files or Folders to Compare](#).

Alternative Options

It is possible to tell *comparepdfcmd* to provide some information or perform a special operation, without doing any comparisons at all. This involves using one of the alternative command line options. The most commonly used is `-h` or `--help`, e.g.,

```
E:\pdfs\>comparepdfcmd -h
```

This outputs to the console a summary of the standard and alternative command line options. Another useful alternative option, is `-m` or `--manual` or `manual`:

```
E:\pdfs\>comparepdfcmd -m
```

This shows *comparepdfcmd*'s manual in your PDF reader.

Especially when requesting support, the `-V` (*capital-V*) or `--version` option is useful:

```
E:\pdfs\>comparepdfcmd -V
```

This shows *comparepdfcmd*'s version.

The other alternative options, each used individually, are `buy` or `--buy`, `cid` or `--cid`, `getkey`, `license` or `--license`, `register` or `--register`, and `try` or `--try`. These are all covered earlier in the [Installing](#) section and its subsections.

Appearance Mode Options

These options only apply when appearance mode comparisons are made. (See also the [Comparison Mode Option](#).)

Accuracy Option

This tells *comparepdfcmd* how accurate appearance mode comparisons should be; it is ignored for text mode comparisons.

The default is 100 (i.e., 100%) which means that each pair of pages that are compared are only considered the same if they match pixel for pixel. The minimum accuracy is 80 (80%), and the maximum, 100.

Appearance mode comparisons are made by dividing each pair of pages into little squares and comparing each pair of corresponding squares. In some situations it is helpful to accept small differences (often ones that humans wouldn't even notice). This can be achieved by lowering the accuracy. For example, reducing the accuracy to 98% means that every pair of squares that are compared can be up to 2% different and still be considered "the same".

This option can be specified on the command line using the syntax `-A number` or `--accuracy=number`. Here are two equivalent examples.

```
E:\pdfs\>comparepdfcmd -a -A90 -r diff.pdf file1.pdf file2.pdf
```

```
E:\pdfs\>comparepdfcmd --appearance --accuracy=90 --report diff.pdf file1.pdf file2.pdf
```

Both specify appearance mode, and an accuracy of 90%, and tell *comparepdfcmd* that if there are differences it should create the difference report file, `diff.pdf`, containing each pair of different pages with their differences highlighted. (See the [Comparison Mode Option](#) and the [Report Filename Option](#).)

To set the default accuracy to, say, 95% for appearance mode comparisons, put these lines in a configuration file:

```
[appearance]
accuracy=95
```

Whatever the default is, it can always be overridden on the command line using the `-A` or `--accuracy` option.

See also the [Square Size Option](#).

Square Size Option

This is an advanced option that should not normally need to be changed.

The square size to use for appearance mode comparisons.

When a pair of pages is compared in appearance mode, i.e., pixel for pixel, the pages are divided into squares, and each corresponding square is compared. The default square size is 10 points, i.e., squares of size 10×10 (100 pixels) are compared.

To change the default, put these lines in a configuration file, replacing 10 with your preferred default (minimum 4 points, maximum, 144 points):

```
[appearance]
square-size=10
```

See also the [Accuracy Option](#).

Comparison Mode Option

The comparison mode tells *comparepdfcmd* how to compare the PDF files. It can be specified on the command line or in a configuration file. The default is words mode unless you change this using a configuration file.

To specify appearance mode on the command line, you can use the syntax `-a` or `--appearance`. This will override the default. Here are two equivalent examples:

```
E:\pdfs\>comparepdfcmd -a file1.pdf file2.pdf
```

```
E:\pdfs\>comparepdfcmd --appearance file1.pdf file2.pdf
```

To make appearance mode the default, put these lines in a configuration file:

```
[general]
mode=appearance
```

To specify chars (characters) mode on the command line, you can use the syntax `-c` or `--chars`. This will override the default. Here are two equivalent examples:

```
E:\pdfs\>comparepdfcmd -c file1.pdf file2.pdf
E:\pdfs\>comparepdfcmd --chars file1.pdf file2.pdf
```

To make chars mode the default, put these lines in a configuration file:

```
[general]
mode=chars
```

The default is words mode, so if this is what you want to use there is normally no need to specify it at all. However, if a different mode has been set as the default in a `comparepdfcmd.ini` file, you can override it on the command line, e.g., using the syntax `-w` or `--words` to specify words mode:

```
E:\pdfs\>comparepdfcmd -w file1.pdf file2.pdf
```

Appearance comparisons (appearance mode) works by in effect taking a “photograph” of each pair of pages being compared, and then comparing these images pixel by pixel. By default, even a single pixel difference is considered significant. However, it is possible to use the [Accuracy Option](#) to make comparisons tolerant of small differences. Another option that affects appearance mode comparisons is the [Square Size Option](#).

Word-by-word comparison (words mode) is best for comparing documents’ texts—this mode is also the fastest.* However, in some situations (perhaps when comparing documents written in CJK—Chinese, Japanese, Korean—languages), character-by-character comparisons (chars mode), may produce better results, but at the cost of being slower. Both words and chars modes (the “text” modes) only consider text: they ignore images, fonts, and layout (apart from the order of the characters or words). For complete comparisons—of text (including fonts and layout), and of images—use appearance mode. Text mode comparisons are affected by the various [Text Mode Options](#).

In addition, *all* comparisons are affected by the [Exclude Options](#) and the [Pages Options](#).

Configuration File Option

This option accepts the name of a configuration file given on the command line. The options in this file override any defaults, but themselves may be overridden on the command line.

This option may only be specified on the command line using the syntax `-C configfile` or `--config=configfile`, where `configfile` is a filename and may include a path. Here are two equivalent examples:

* In tests *comparepdfcmd*’s words mode comparisons were 2× – 4× faster than *DiffPDF* or *diffpdfc*. Furthermore, *comparepdfcmd* makes better use of processor cores, and uses less memory.

```
E:\pdfs\>comparepdfcmd -C E:\myconfig.ini file1.pdf file2.pdf
```

```
E:\pdfs\>comparepdfcmd --config E:\myconfig.ini file1.pdf file2.pdf
```

This is useful for overriding defaults that cannot be set on the command line, or for using a specific set of options for particular comparison tasks without changing the overall defaults.

To permanently set your own default options, see [Setting Your Own Defaults](#).

For an example configuration file, open `example.ini` in `comparepdfcmd`'s folder using a plain text editor.

Highlight Transparency Option

1.1.4

Differences are indicated by highlighting them as if you'd drawn over them with a highlighter pen. This makes the differences stand out and at the same time allows the highlighted text or graphics to show through.

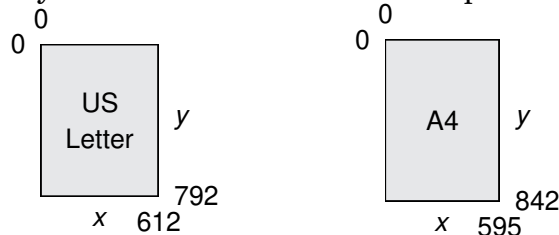
The default is 80 (i.e., 80%). The minimum is 10 (10%) and the maximum is 90 (90%).

You can change the default if you want. For example, by reducing the transparency to, say, 70%, you can make the highlighting stronger, while still allowing the differences to be visible. This can be done by putting these lines in a configuration file:

```
[general]
highlight-transparency=70
```

Exclude Options

It is sometimes convenient to exclude a region of each page from comparisons. For example, you might want to exclude a header or footer that has a timestamp that is different for every PDF you compare, but which you don't want to affect the comparison.



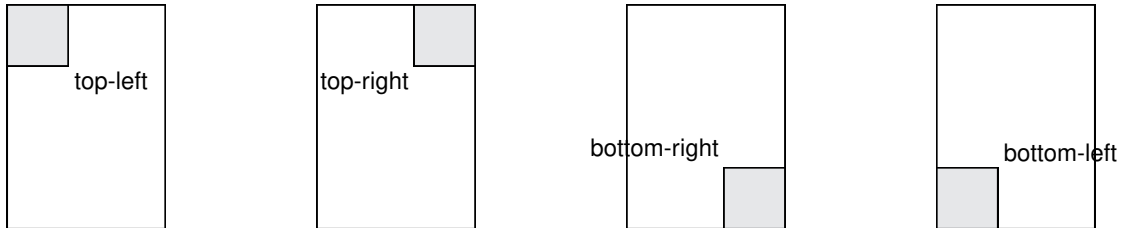
The program can exclude corners, margins, and arbitrary rectangles.

Any text wholly within an excluded corner, margin, or rectangle, is ignored for text mode comparisons, and any part of the page within an excluded corner, margin, or rectangle, is ignored for appearance mode comparisons.

Coordinate (0,0) is the top-left corner; coordinate (612,792) is the bottom-right corner of a US-Letter size page, and coordinate (595,842) is the bottom-right of an A4 page, as illustrated above.

Exclude Corners Options

Excluded corners, if set at all, are set individually, and are specified as an x, y coordinate pair relative to their nearest corner.



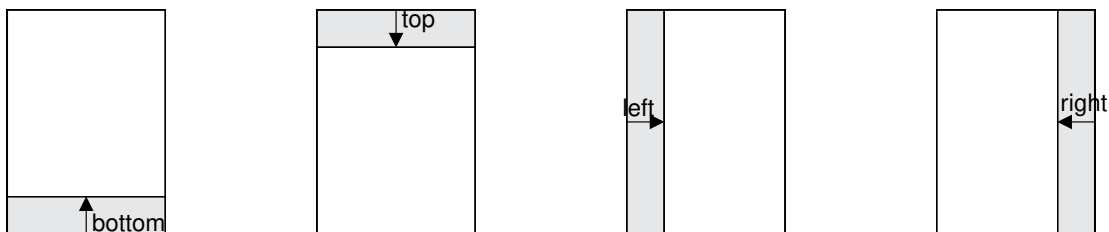
To exclude corners, add a corner exclusion section to a configuration file. For example, to exclude a one inch (25.4 mm) corner from the top-right of the PDFs being compared, add these lines:

```
[corners]
top-right=72,72
```

Corner coordinates may range from 0 to 300. You can specify up to four corners in a corners section, bottom-left, bottom-right, top-left, and top-right. The coordinates don't have to be the same—so you can exclude corner rectangles, not just squares.

Exclude Margins Options

Excluded margins, if set at all, are set individually, and are specified as a length from their nearest edge.



Margins can be excluded using the command line with the syntax `--margin-which=number`, where *which* is left, right, top, or bottom. For example, `--margin-bottom=54`. You can use none, one, two, three, or all four of these options on the command line.

It is also possible to exclude margins using a configuration file. To do this, add a margin exclusion section to a configuration file. For example, to exclude a $\frac{3}{4}$ inch (19 mm) margin from the bottom of the PDFs being compared, add these lines:

```
[margins]
bottom=54
```

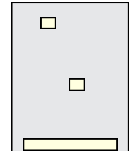
Margin coordinates may range from 0 to 300. You can specify up to four margins in a margins section, bottom, left, right, and top.

Exclude Rectangles Options

Excluded rectangles, if set at all, are set individually, and are specified as coordinates $x1, y1, x2, y2$, where $(x1, y1)$ is a rectangle's top-left corner, and $(x2, y2)$ is its bottom right corner.

To exclude rectangles, add a rectangles exclusion section to a configuration file. For example, to exclude the three rectangles illustrated, add these lines:

```
[rectangles]
rectangle1=155,79,217,158
rectangle2=300,400,350,450
rectangle3=61,533,558,594
```



Each rectangle must be individually numbered (from `rectangle1` to `rectangle20`). The numbers need not be consecutive and there can be gaps (e.g., `rectangle9`, then `rectangle15`); nor does their order within the `rectangles` section matter.

Although rectangles can be used to exclude corners and margins, as well as arbitrary rectangles, it is usually easier to specify corners and margins using the [Exclude Corners Options](#) and the [Exclude Margins Options](#).

Pages Options

By default, *comparepdfcmd* compares all the pages in the first PDF with their corresponding pages in the second PDF.

For example, if you compare two PDF files, by default *comparepdfcmd* will compare both their page 1's with each other, both their pages 2's, both their page 3's, and so on.

In the case of PDF files that have different numbers of pages, or where you only want to compare particular pages, you can specify exactly which pages *comparepdfcmd* should compare using the command line `--pages1` and `--pages2` options. If neither option is given, *comparepdfcmd* will compare all the pages. If `--pages1` is given, but not `--pages2`, then the page set for `--pages1` will also be used for `--pages2`. And if both are specified, then *comparepdfcmd* will compare as specified.

Pages can be specified using the syntaxes `--pages1=pages` and `--pages2=pages` where *pages* is a set of comma-separated page numbers, or page ranges of the form *startpage-endpage*, or a mixture of both. Here are some examples:

```
E:\pdfs\>comparepdfcmd -r diff.pdf file1.pdf file2.pdf
E:\pdfs\>comparepdfcmd -r diff.pdf --pages1=1,3,5,7,11-18 file1.pdf file2.pdf
E:\pdfs\>comparepdfcmd -r diff.pdf --pages1=3-19 --pages2=5-21 file1.pdf file2.pdf
```

In the first example, all the pages are compared. In the second example pages 1, 3, 5, 7, 11, 12, 13, 14, 15, 16, 17, and 18 are compared. And in the third example `file1.pdf`'s pages 3–19 are compared with `file2.pdf`'s pages 5–21, i.e., 3 vs. 5, 4 vs. 6, 5 vs. 7, ..., 18 vs. 20, 19 vs. 21. In each case a difference report is written to the file `diff.pdf`; see the [Report Filename Option](#).

Report Options

To get a difference report when comparing two PDFs, use the `-r` or `--report` command line option (see [Report Filename Option](#)). To have *comparepdfcmd* show this report in your PDF reader, use the `-s` or `--show` option (see [Report Show Option](#)).

To get difference reports when comparing two *folders* of PDFs, use the `-R` or `--reportpath` option (see [Report Path Option](#)) and the `-f` or `--format` option (see [Report Format Option](#)). The other options described in this section are used to configure some aspects of the reports.

Report Decimals Option

This option is used to set how many digits to show after the decimal point in CSV and JSON reports. The default is 2.

To change the default, put these lines in a configuration file, replacing 2 with your preferred default (minimum 0, maximum, 4):

```
[report]
decimals=2
```

See also the [Report Filename Option](#) and the [Report Format Option](#).

Report Filename Option

This option is used when comparing two PDFs to specify the filename of a difference report.

(If you are comparing two *folders* of PDFs, don't use this option, use the [Report Path Option](#) and the [Report Format Option](#) instead.)

There is no default for this option, so *comparepdfcmd* will not output any reports by default.

Four output formats are supported:

- CSV (comma-separated variable; `.csv`)—suitable for importing into a spreadsheet and for regression testing;
- JSON (JavaScript Object Notation; `.json` or `.json`)—suitable for regression testing;
- PDF (portable document format; `.pdf`)—containing each pair of differing pages with differences highlighted;
- PNG (portable network graphics; `.png`)—an image for each pair of pages that differ, with differences highlighted.

1.1.5

If you output a PDF report, you can tell *comparepdfcmd* to show it in your PDF reader using the [Report Show Option](#).

This option can only be specified on the command line using the syntax `-r filename` or `--report=filename` where *filename* is the filename for the difference report. The *filename* may include a path, and *must* end with `.csv`, `.json`, `.json`, `.pdf`, or `.png`, depending on which report format you want. If you want reports in two or more formats, simply repeat this option once for each format.

Here is an example:

```
E:\pdfs\>comparepdfcmd -v -r E:\diff\diff.pdf E:\pdfs\f1.pdf E:\pdfs\f2.pdf
Wrote "E:\diff\diff.pdf"
```

Here, two PDF files were compared and differences found which are shown in the `diff.pdf` file. If you give a report filename that includes a path, then what you have specified will be used. However, if you give a report filename without a path, the path specified by the [Report Path Option](#) will be used. The [Report Path Option](#) has no default, so if no path is given in the report filename and the [Report Path Option](#) isn't set, the report will be written to the current folder.*

In the following example we assume that the [Report Path Option](#) has been set in a configuration file to, say, `E:\diff`.

```
E:\pdfs\>comparepdfcmd -v -r x.csv -r y.json -r z.pdf E:\pdfs\g1.pdf E:\pdfs\g2.pdf
Wrote "E:\diff\x.csv"
Wrote "E:\diff\y.json"
Wrote "E:\diff\z.pdf"
```

Because the [Report Path Option](#) has been set, *and* no path is specified by the `-r` or `--report` options, the [Report Path Option](#) has been used.

Here's one final example:

```
E:\pdfs\>comparepdfcmd -v -r E:\diff\pagesimage-.png E:\pdfs\h1.pdf E:\pdfs\h2.pdf
Wrote "E:\diff\pagesimage-09-09.png"
Wrote "E:\diff\pagesimage-11-11.png"
Wrote "E:\diff\pagesimage-12-12.png"
```

In this example, the two PDFs differ on pages 9, 11, and 12, and each pair of pages has been saved into its own PNG file using the report filename as the *pattern* which `comparepdfcmd` fills in as `name-page1-page2.png`. Naturally, the path could have been omitted if you wanted to save the images to the current folder, or to a folder specified by the [Report Path Option](#).

Report Format Option

This option is used when comparing two *folders* of PDFs to specify the format to use for difference reports.

(If you are comparing just two PDFs, don't use this option, use the [Report Filename Option](#) instead.)

There is no default for this option, so `comparepdfcmd` will not output any reports by default when comparing two folders of PDFs.

The same four output formats as the [Report Filename Option](#) are supported. (JSON support was added in version 1.1.5.)

1.1.5

* Note that if the current folder is `C:\`, Windows will normally *silently ignore* requests to write to it.

This option can be set on the command line using the syntax `-f format` or `--format=format` where *format* is csv, json, pdf, or png. For example:

```
E:\pdfs>comparepdfcmd -f pdf -R E:\diff E:\pdfs\old E:\pdfs\new
"Result","PDF#1","PDF#2","Report or Error"
"Missing#1",,"E:\pdfs\new\only_in_new.pdf",
"Missing#2","E:\pdfs\old\only_in_old.pdf",,
"Same","E:\pdfs\old\one.pdf","E:\pdfs\new\one.pdf",
"Different","E:\pdfs\old\two.pdf","E:\pdfs\new\two.pdf","E:\diff\two.pdf"
"Different","E:\pdfs\old\three.pdf","E:\pdfs\new\three.pdf","E:\diff\three.pdf"
```

Here the report format has been set to PDF using `-f pdf` and a path for the reports has been specified using `-R E:\diff` (i.e., using the [Report Path Option](#)).

In general we recommend always using this option and the [Report Path Option](#) together, unless they are both set in configuration files.

To change the default of no format (i.e., no reports output when comparing folders), put these lines in a configuration file:

```
[report]
format=pdf
```

Naturally, you could write `format=png` or `format=csv` if you preferred either of them to PDF format for reports of PDFs that differ when comparing folders.

Report Output Option

This is an advanced option that should not normally need to be changed.

By default PDF and PNG reports show each pair of differing pages side-by-side with differences highlighted. However, if you only want to see the pages in the first PDF (or first folder's PDFs when comparing folders of PDFs), you can tell *comparepdfcmd* to only output these pages with differences highlighted. Similarly, you can tell *comparepdfcmd* to only output the pages of the second PDF (or second folder's PDFs).

To change the default of showing pairs of pages side-by-side, put these lines in a configuration file:

```
[report]
output=one
```

This will only output the first PDF's different pages (or the first folder's PDFs). If you prefer the second, then use `output=two` instead. To get the default, just delete this line or use `output=both`.

Report Path Option

Use this option to specify the folder to use for reports when comparing two PDFs using the [Report Filename Option](#) (in which case the report filenames do not need to have paths since they'll be given the path specified by this option). And use this option when comparing two *folders* of PDFs—in conjunction with the [Report Format Option](#).

This option can be set on the command line using the syntax `-R path` or `--reportpath=path`. For an example, see the [Report Format Option](#), page 20.

To change the default of no default report path, put lines like these in a configuration file:

```
[report]
path=E:\diff
```

Naturally, you would use the path relevant to your system rather than the `E:\diff` shown here.

Report Path in Title Option

1.1.7

Use this option to specify whether to include the full path to the compared PDFs in report titles.

To change the default of including the full path of compared PDFs in report titles, put lines like these in a configuration file:

```
[report]
path-in-title=false
```

Report Scale Option

1.1.6

Use this option to scale PNG reports.

By default PNG reports are output at 100% scale (i.e., at 72 DPI). However, it can sometimes be useful to use higher resolutions, especially for very small PDFs.

To change the default of 100 (100%), put lines like these in a configuration file:

```
[report]
scale=150
```

The minimum is 100 (the default), and the maximum is 800 (800%). To find the best scale, try 150, then 200, and so on.

Higher resolution output takes more time (although this may not be noticeable for small PDFs).

Report Show Option

Use this option to show *comparepdfcmd's* PDF report if two PDFs are different.

This option can only be set on the command line using the syntax `-s` or `--show`. It only applies when comparing two PDF files (*not* two folders), *and* when the [Report Filename Option](#) is used to specify a PDF report.

For example:

```
E:\pdfs>comparepdfcmd -v -s -r E:\diff\report.pdf E:\pdfs\k1.pdf E:\pdfs\k2.pdf
Wrote "E:\diff\report.pdf"
```

If the two PDFs are the same, the output will be the text “Same” on the console and no further action will be taken.

However, if the two PDFs are different (as in the example), the text “Wrote “E:\diff\report.pdf”” will be output, *and*, due to the `-s` or `--show` option, the report will be shown in your PDF reader application.

Report Summary Option

When comparing two *folders* of PDFs, *comparepdfcmd* will output a summary of the results to the console. For an example, see the [Report Format Option](#), page 20. When the summary is output to the console, the [Verbose Option](#) is ignored.

The summary can be sent to a file instead of the console (in which case the [Verbose Option](#) is respected), using the syntax `-S filename.csv` or `--summary=filename.csv`. For example:

```
E:\pdfs>comparepdfcmd -v -S E:\summary.csv -f pdf -R E:\diff E:\pdfs\old E:\pdfs\new
Same
Wrote "E:\diff\two.pdf"
Wrote "E:\diff\three.pdf"
Wrote "E:\summary.csv"
```

Text Mode Options

These options only apply when text mode comparisons are made, i.e., words and chars modes. (See also the [Comparison Mode Option](#).)

Change Bar Option

This tells *comparepdfcmd* whether to show a red vertical change bar in pages’ left margins wherever there are changes highlighted in the text. This option only applies to text mode comparisons and is ignored for appearance mode comparisons.

The default is true, i.e., to show change bars.

To change the default, i.e., to switch them off, put these lines in a configuration file:

```
[text]
bar=false
```


Column Tolerance Option

This is an advanced option that should not normally need to be changed.

This option only applies to text mode comparisons that use the multicolumn algorithm. (See the [Comparison Mode Option](#) and the [Text Comparison Algorithm Option](#).)

The default is 108. The allowed range is 10–300. If the default isn't satisfactory, try these values: 72, then 36, then 144, then 176.

To change the default, put these lines in a configuration file (with 108 changed to the default you want):

```
[text]
column-tolerance=108
```

Ignore Hyphens Option

By default, hyphens are treated like regular characters. If this option is used, hyphens are ignored. When hyphens are ignored it means that, for example, *one–two* is treated as *onetwo*. This works even if *two* is on the next line.

The default is not to ignore hyphens, but this can be overridden for any particular comparison on the command line using the syntax `--ignore-hyphens`. Or, you can make ignoring hyphens the default by adding these lines to a configuration file:

```
[text]
ignore-hyphens=true
```

If you use this option, we recommend that you also normalize hyphens (which is the default behavior). See the [Normalize Hyphens Option](#).

Line Tolerance Option

This is an advanced option that should not normally need to be changed.

To do sensible text comparisons *comparepdfcmd* needs to work out where the lines of text are on each page. This can be tricky when the lines are very close together. This option can be used to fine-tune how *comparepdfcmd* works out the lines.

The default is 10, but for PDFs that have lines very close together a setting of 9 or even 8 might produce better results. Changing the default can be done by adding lines like these to a configuration file:

```
[text]
line-tolerance=9
```

The minimum is 1 and the maximum is 27, but only 8, 9, and 10 are recommended.

Normalize Hyphens Option

This is an advanced option that should not normally need to be changed.

By default *comparepdfcmd* normalizes hyphens and a few other characters. This means that all kinds of hyphens (i.e., all hyphens in the Unicode “punctuation dash” character category, such as plain hyphens, minus signs, non-breaking hyphens, and en-dashes and em-dashes) are treated as if they were the same. In addition to normalizing hyphens, non-standard double quotes (Unicode code points U+0093 and U+0094 that are used in some PDF files) are considered to be identical to U+201C (“) and U+201D (”) for the purposes of comparison.*

If you don’t want hyphen normalization, you can turn it off on the command line using the syntax `--no-normalize-hyphens`. Or, you can make not normalizing hyphens the default by adding these lines to a configuration file:

```
[text]
normalize-hyphens=false
```

Normalize Ligatures Option

By default, a ligature is considered to be different from its spelled out form, for example, **fi** ≠ **fi**, and **fl** ≠ **fl**. If you want ligatures and their spelled out forms to be considered the same during comparisons, you can tell *comparepdfcmd* to normalize ligatures.

Ligature normalization can be turned on, on the command line, using the syntax `--normalize-ligatures`. Or, you can make normalizing ligatures the default by adding these lines to a configuration file:

```
[text]
normalize-ligatures=true
```

Text Comparison Algorithm Option

Three text mode comparison algorithms are provided. The default is the standard algorithm which gives good results in almost every case. For multicolumn PDFs using the multicolumn algorithm may produce better results. And in rare cases the special algorithm may be best.

You can change the default by adding lines like these to a configuration file:

```
[text]
algorithm=multicolumn
```

Naturally, if you prefer to use the special algorithm, you can set `algorithm=special`. And to restore the default either delete these lines or set `algorithm=standard`.

All three algorithms can be tuned using the [Line Tolerance Option](#). And the multicolumn algorithm can be further tuned using the [Column Tolerance Option](#).

* Note that *comparepdfcmd* always normalizes whitespace, that is, any kind of whitespace (space, non-breaking space, etc.), is treated as a space (i.e., as a word separator).

Verbose Option

By default *comparepdfcmd* does not output anything to the console when comparing two PDF files. And when comparing two *folders* of PDFs *comparepdfcmd* outputs a summary to the console unless the [Report Summary Option](#) is used. (Of course, if errors occur, these are always reported, no matter what this option is set to.)

If you want *comparepdfcmd* to tell you the outcome of comparisons and what reports it has written, you can tell it to be verbose. This option can only be specified on the command line, using the syntax `-v` or `--verbose`.

The PDF Files or Folders to Compare

After any command line options have been specified, two PDF filenames (with paths if needed) or two folders *must* be given. If two PDF filenames are specified, these two PDFs will be compared. If two folders are specified, then *comparepdfcmd* will look inside each folder. For every PDF file it finds in both folders (and their subfolders, and so on), *comparepdfcmd* will compare those PDF files. For every PDF file that is in one folder but not the other, *comparepdfcmd* will report the nonexistent PDF as “missing”. And any non-PDF files will be ignored.

Troubleshooting

Invalid Key Code Error

When a full license key is acquired from the “get key” page at www.qtrac.eu/getkey, the license key that is issued is tied to the Computer ID. The Computer ID depends on the computer’s Motherboard and CPU, as reported by Windows.

If—within two years of purchase—the computer it was acquired for is scrapped or replaced or has its Motherboard or CPU changed (or how these are reported is changed by a Windows upgrade), normally the license key can be transferred to another computer. Contact support@qtrac.eu with the original Order ID if a transfer is required. Only one transfer is allowed per license key.

If a license key is acquired as one class of user (e.g., Administrator), and then used by another class of user (e.g., User), even though they are on the *same* machine, the acquired license key may not be visible to *comparepdfcmd* and so result in an “Invalid key code” error. If this is the case, copy the `comparepdfcmd.key` file to the user’s home folder. (This problem should not arise if the `comparepdfcmd.key` file or the `qtrac.key` file is in the same folder as *comparepdfcmd* itself.)

False Positives

In some circumstances, *comparepdfcmd* will report differences that are not visible to the human eye. In some cases these are genuine differences (for example, two different kinds of hyphens that look the same), and in some cases these are false positives.

By default *comparepdfcmd* normalizes hyphens (i.e., treats all the different hyphen kinds as the same), but does not normalize ligatures (although it will if you tell it to). To control these, see the [Normalize Hyphens Option](#) and the [Normalize Ligatures Option](#).

False positives are usually caused by very narrow inter-line spacing, or by text which mixes Latin text (e.g., English) with non-Latin text (e.g., Japanese), and in some cases by tables. In many cases these false positives can be eliminated—or minimized—by reducing the [Line Tolerance Option](#). For example, try reducing the line tolerance from the default of 10 to 9 or 8, or try the special algorithm with these line tolerances (see the [Text Comparison Algorithm Option](#)).
