pgn2scid

Manual

Version 1.1

Rev. 1.1.b

Table of contents

What is pgn2scid?

pgn2scid basically is a convenient GUI wrapper (a graphical user interface for command line tools) for *pgnscid* and *scmerge*, two programs to convert PGN files to native *Scid* database files and to merge a number of *Scid* files with an existing *Scid* database. Both tools should be part of a standard *Scid vs. PC* installation.

But there's more! In detail **pgn2scid** can:

- run on multiple platforms

- keep your database up to date automatically by downloading the PGN game collections kindly provided by www.theweekinchess.com (in short TWIC).

- keep track of your downloads so that only the latest files which haven't already been downloaded are selectable.

- process PGN files from other sources, let's say from your last OTB tournament or games played on a chess server. All you have to do is to put those files in **pgn2scid**'s working directory

- automatically uncompress zipped PGN files

- automatically merge a number of PGN files into one single file

- convert any number of PGN files to the native *Scid* database format without the hassle of using command line tools

- create a zipped backup of an existing *Scid* database before executing any database merge operations

- add any number of *Scid* files to an existing database also without to use a command line tool

- every step described above is optional

- store all settings in an initialisation file so **pgn2scid** is already preconfigured next time it is started

Licensing

This software is released under the terms of the MIT License.

Copyright (c) 2017 Andreas Kreisig

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

3rd party tools

pgn2scid makes use of the tools *pgnscid* and *scmerge*, both of which had been developed by Shane Hudson and which have been released under the terms of the GNU General Public License (GPL).

Questions, comments, bugs …

If you have any questions or comments regarding **pgn2scid** please don't hesitate to send an email to andreaskreisig@gmail.com. You can also use this email address to report bugs. If you have a GitHub account you can use the GitHub issue tracker for bug reports as well. You can find the project website on https://github.com/CasualPyDev/pgn2scid.

Before you start

Although **pgn2scid** is more or less self-explanatory you should read on to understand the internal workflow of **pgn2scid**. This manual covers short descriptions of every section of **pgn2scid** as well as the prerequisites to install and run this software.

To avoid naming confusions, the command line tool *pgnscid* (as well as all other program names) is printed in italic while **pgn2scid**, the software which is the subject of this manual, is printed in italic and bold. <Buttons> you can click on are enclosed in angle brackets.

The structure of this manual follows the structure of the program. All images have been taken from the Linux version. While the visual appearance might slightly differ on other operating systems, the functionality remains the same.

1. General prerequisites for all supported platforms

pgn2scid is a cross-platform tool that runs on Unix-like operating systems as well as on MS Windows. It requires the programming language Python, version 3.4 or newer. It is recommended to download the latest stable release of Python which you can get for free on www.python.org. An exception to this is MS Windows, because there is an executable exe file available for this operating system (please read section 1.1 for more information).

To take full advantage of **pgn2scid** you also need a proper installation of the chess database *Scid vs. PC*, for the sake of simplicity hereinafter just referred to as *Scid*. This database is also available for free on http://scidvspc.sourceforge.net/. You also need some tools bundled with this database, namely *pgnscid* and *scmerge*. **pgn2scid** might also work with other data compatible chess databases. Maybe you then have to install *scmerge* and *pgnscid* manually as they are not necessarily a part of those databases.

Note

The 2.7 branch of Python, which is pre-installed on most Linux distributions, will not work. You can safely install both, the 2.7 and the 3.x version of Python on the same machine.

1.1 MS Windows only

When you use MS Windows, it's not necessary to download and install the Python programming language first as there's a pre-build executable available for this operating system. In this case you just have to download the file 'pgn2scid_win_executable.zip'. This ZIP file contains the *pgn2scid.exe* file as well as some supporting files which are necessary to run **pgn2scid**. If you have problems running this pre-build executable you can always use the *pgn2scid.pyw* file as a fallback solution. In this case you have to take care of the general prerequisites described above.

2. Installation instructions for all supported platforms

Just copy the file *pgn2scid.pyw* into a dedicated sub folder where you have full read / write access to. You shouldn't copy it into your hard disks root folder, though because **pgn2scid** itself will create some sub folders and additional files during operation which then can get a bit messy when they get mixed up with other files and folders into your computers root directory.

2.1 MS Windows only

If you want to use the pre-build executable of **pgn2scid** all you have to do is to unzip the file 'pgn2scid_win_executable.zip'. After unzipping you will find a new subfolder called 'pgn2scid' which contains the *pgn2scid.exe* file and some additional files and folders which are necessary to run **pgn2scid**. Included is also this manual and the license text.

2.2 Unix-like operating systems only

To make **pgn2scid** executable on a Unix-like operating system, you have to open a terminal, change the directory to the location where your copy of **pgn2scid** resides and run the *chmod* command. Let's assume the file *pgn2scid.pyw* is located in the directory 'pgn2scid' which, in turn, resides in your home directory.

To enter the pgn2scid directory, type

```
cd ~/pgn2scid
```

Now, to make the file pgn2scid.pyw executable, type

```
chmod +x pgn2scid.pyw
```

2.3 Mac OS X?

pgn2scid hasn't been tested on OS X systems yet. Generally it should work, but at the time of writing of this manual there are some well known and serious issues with the Tcl/Tk software package shipped with Mac OS X 10.6 and later. Tcl/Tk is necessary to run *pgn2scid* because the underlying GUI toolkit sits on top of it. If you would like to give it a try on an OS X system it's highly recommended to use a 3rd party Tcl/Tk 8.5 like ActiveTcl 8.5 (https://www.activestate.com/activetcl/downloads). You should read the Python documentation on https://www.python.org/download/mac/tcltk/ to get more information about this topic. Please keep in mind that, in contrast to a standard Tcl/Tk, ActiveTcl is not open source software so it's covered by a proprietary ActiveState license

3. The main window

pgn2scid is a single window application, all settings are directly accessible within the main window. Some options are only available when the higher-level option has been selected.
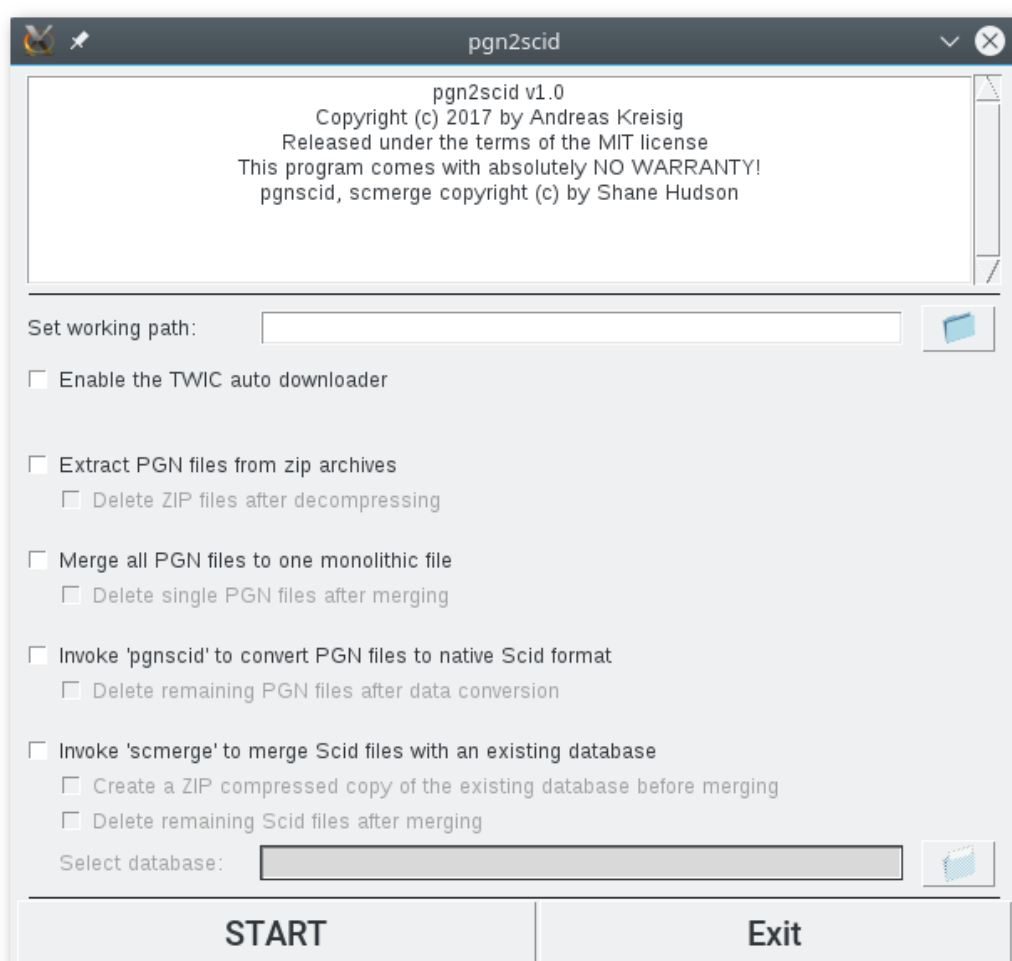
4. The status window

On top of the program there is the status window. After start up it displays some information about the author of **pgn2scid**, program version, some licensing information and some info about 3rd party tools used by **pgn2scid**.

During data processing it shows status messages and warnings or errors. In fact, it informs you about almost every step **pgn2scid** is currently performing as well as about the outcome of the step in question. A successfully finished step is indicated by the word [DONE], enclosed in square brackets.
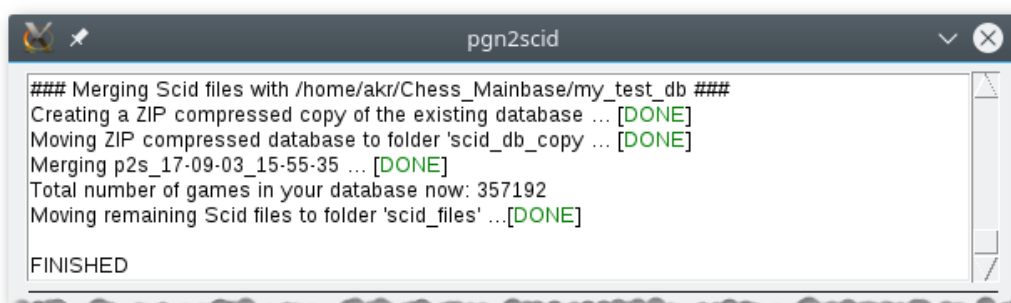
Results printed in red means that something has been [CANCELED] and / or **pgn2scid** was not able to execute a certain step. For example, if **pgn2scid** can't perform a selected task due to missing files, the selected step is [SKIPPED] and an appropriate message is displayed in the status window. In this case **pgn2scid** continues the program flow and goes to the next selected option, if available, or it stops accordingly.
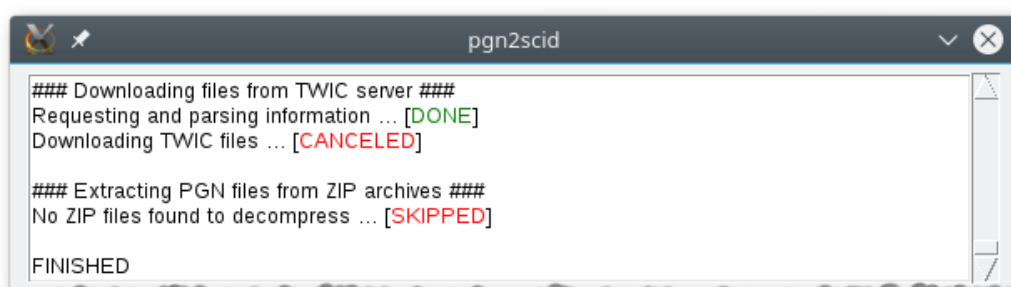
On critical errors, indicated by the word [FAILED], *pgn2scid* displays an additional error message containing some information about the occurred error. In the example below *pgn2scid* runs into a network error, caused by a missing internet connection. In this case *pgn2scid* goes to the next selected option, or, when there's no more option selected, it stops accordingly. In case of network errors, *pgn2scid* also adds an entry to a master log file named 'pgn2scid_error.log (new in version 1.1). Please read section 10 for more information about this master log file.
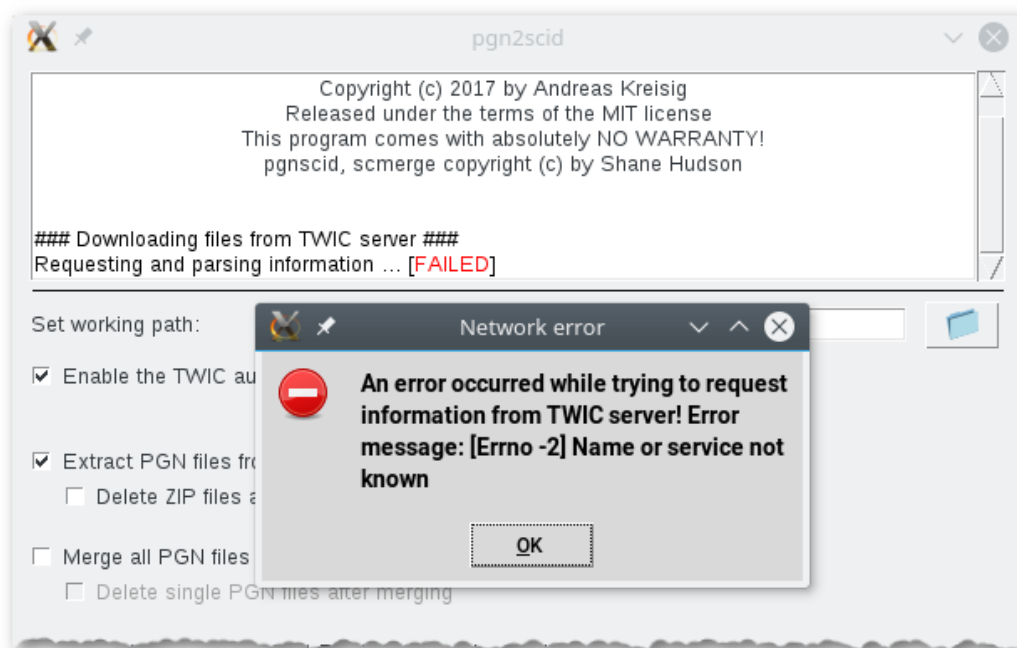
On fatal errors, indicated by the word [FAILED], an additional error message and the note STOPPED at the bottom of the status window, *pgn2scid* stops working completely. This can only happen if *pgn2scid* loses the read / write permissions on a directory or a file is suddenly inaccessible during runtime.

Note

When *pgn2scid* stops working due to a fatal error it doesn't clean up the working directory automatically. After fixing the underlying error you have to take care about what to do with the remaining files.

5. Set working path

Here you have to enter or to choose a working path. You have to put all files you'd like to get processed into this folder. When you use the TWIC auto downloader (see the section below), all downloaded files are also saved into this folder. *pgn2scid* needs full read / write access to this folder in order to work properly. If this is not the case, the program will display an error message.

6. Enable the TWIC auto downloader

When you select this option, *pgn2scid* can update your *Scid* database automatically by downloading the latest tournament games found on www.theweekinchess.com (TWIC). You need a working internet connection to use this option. If there is a proxy server between your PC and the internet it is detected automatically. Actually the TWIC auto downloader is more a semi-automatic downloader because *pgn2scid* always asks for permission before downloading any files.

6.1 The 'Select TWIC files' window

After loading and parsing the TWIC website, *pgn2scid* displays the 'Select TWIC files' window. All important information gathered from the TWIC website is displayed in this window, clearly arranged in a table. Just select the files you would like to download and click the <DOWNLOAD> button. Selecting <Cancel> cancels the download. If you need more information about a certain TWIC game collection before downloading it, you can click on the related 'link' entry in the 'Info' column. *pgn2scid* then opens your systems default web browser and loads the appropriate website.
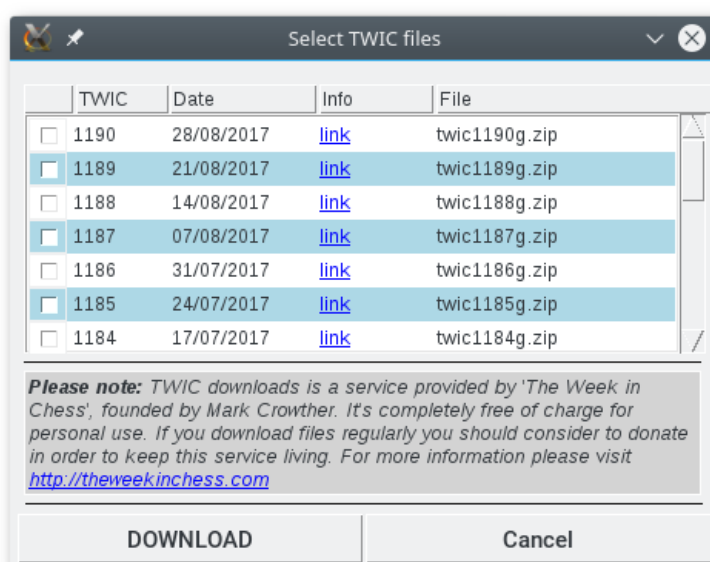

pgn2scid keeps track of your downloads. When there are no new TWIC files available after the last download, *pgn2scid* doesn't display the 'Select TWIC files' window. Instead the download step is skipped altogether and an appropriate message is displayed in the status window.

Note

Be careful which files you choose to download because *pgn2scid* remembers the highest already downloaded TWIC issue number. Every file below is not available anymore next time you use the TWIC auto downloader. For this reason it's also advisable to avoid gaps between selected TWIC files, except you are really not interested in certain issues. On the other hand you shouldn't download everything you can get to minimize server load and bandwidth and to avoid having doubled or unwanted games in your database. Please keep in mind that this service is for free for personal use so again: avoid unnecessary server load!

7. Extract PGN files from ZIP archives

When this option is selected, *pgn2scid* extracts ZIP compressed PGN files automatically. When there are no ZIP files in the working directory, *pgn2scid* skips this option, displays an appropriate message and goes on to the next selected option. When *pgn2scid* detects a filename during ZIP decompression which already exists in your working directory, a warning message pops up asking you how to proceed. You have to choose between <Skip> which skips the file in question, <Overwrite> which just overwrites the existing file or <Auto rename>. When you click on <Auto rename>, *pgn2scid* appends an ascending number in brackets to the filename, e.g. 'my_game.pgn' becomes 'my_game(1).pgn' and so on.
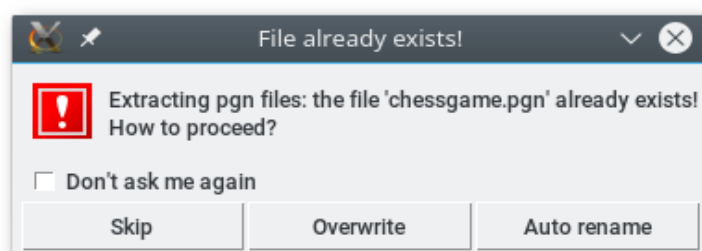
When you set a check mark on 'Don't ask me again', *pgn2scid* executes your last selected option automatically as soon as it detects an already existent file name again, without displaying a warning message. This only applies to the current session.

Note

pgn2scid only extracts files with the suffix *.pgn. When there are other type of files within a ZIP compressed file *pgn2scid* skips these files and displays an appropriate message in the status window.

7.1. Delete ZIP files after decompressing

Only available when 'Extract PGN files from ZIP archives' is selected. Use this option to delete unnecessary ZIP files after decompressing. When you don't select this option because you want to keep these ZIP files, ***pgn2scid*** creates a new directory called 'zip_files' beneath the applications root directory. It then moves all ZIP files to this folder to keep the working directory clean. This is important to avoid processing ZIP files multiple times.

8. Merge all PGN files to one monolithic file

pgn2scid can merge almost any number of PGN files to one, single file. The filename is given automatically and consists of the prefix p2s, followed be a timestamp and the suffix pgn, e.g. p2s_17-06-04_19-10-48.pgn, generally: p2s_YY-MM-DD_HH-MM-SS.pgn. The timestamp format is used to make any merged PGN file unique. Whenever you want to process more than just a few number of PGN files and you'd like to convert those files to the native *Scid* format it's recommended to select this option, because this conversion process creates three files for every single PGN file (for details see section 9).

PGN file merging is even more important when you also want to integrate your converted files into an existing *Scid* database (see section 11 for more information). Merging is quite fast. On an Intel Core i5 with 2.5 GHz and a Samsung 850 EVO SSD merging more than 315.500 lines, distributed over 5 PGN files, only took about 2-3 seconds. With a standard HDD it takes a bit longer, of course.

pgn2scid is able to recognise older merged PGN files kept in the working directory. In this case a message box pops up asking whether or not to move this file. If in doubt select <Yes> here to avoid having unwanted games in your database. ***pgn2scid*** then moves this file to the folder 'pgn_files', located in the applications root folder. <No> means that this file is going to be processed according to the subsequently set options.

Note

When there is only one PGN file inside your working directory so that there is actually nothing to merge with you don't have to unselect this option. The merge process will then act just like a file rename function using the timestamp format described above.

8.1 Delete single PGN files after merging

Only available when 'Merge all PGN files to one monolithic file' is selected. After PGN file merging, ***pgn2scid*** can delete all single PGN files as they are no longer needed. When you don't select this option, ***pgn2scid*** creates a folder called 'pgn_files' in your applications root directory and moves all PGN files to this location. As mentioned earlier this is to keep the working directory clean.

9. Invoke 'pgnscid' to convert PGN files to native Scid format

By selecting this option ***pgn2scid*** converts all PGN files in the given working directory to the native *Scid* database format. This is achieved by invoking the 3rd party command line tool *pgnscid*, which is executed in the background so you don't have to use command line instructions. During the conversion process *pgnscid* creates three database files per PGN file: the index file (.si4), the name file (.sn4) and the game file (.sg4). This is the reason why it is recommended to merge a larger number of PGN files before converting them. Otherwise, for example twenty single PGN files would end up in sixty(!) *Scid* database files.

9.1 Delete remaining PGN files after data conversion

When this option is selected, ***pgn2scid*** deletes all PGN files left in the working directory, including the optionally merged PGN file created before. These files are no longer needed for further processing so you can safely delete them.

When you don't select this option, ***pgn2scid*** moves these PGN files to the folder 'pgn_files', located in the applications root directory. If this folder doesn't exist, ***pgn2scid*** creates it.

9.2 The 'pgnscid error log' window

When *pgnscid* runs into errors during data conversion caused by a faulty PGN file, it writes an error log file into the working directory. ***pgn2scid*** detects this error log file, adds a timestamp on top of the log files content and displays the 'pgnscid error log' window. Please read these error messages carefully because you have to choose whether you'd like to ignore these errors or you'd like to suspend the faulty PGN file.
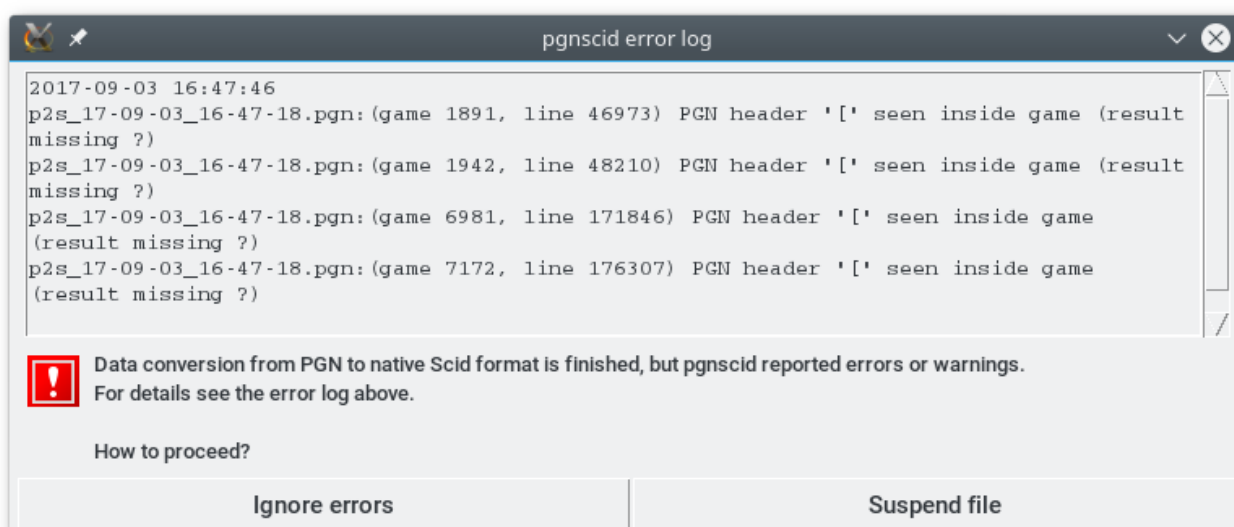
Note

Depending on the kind and number of errors displayed in the error log window, choosing <Ignore errors> might end up in a corrupt database when you merge the resulting Scid file with an existing Scid database. One of the most common error is a missing game result. In this case you can ignore this error without the risk of corrupting your database. You might add the game result in your Scid database afterwards.

Ignore errors: *pgn2scid* ignores all errors displayed in the '*pgnscid* error log' window and appends the log files content to a master log file called 'pgn2scid_error.log'. This master log file, which collects all cumulated *pgnscid* warnings or errors across all sessions so far, resides in the applications root directory. If there's no such file, *pgn2scid* creates it. The initial error log file (NOT the master log file) is deleted. Finally *pgn2scid* goes on to the next PGN file or to the next selected option.

Suspend file: *pgn2scid* deletes all formerly created *Scid* files based on the faulty PGN file. It then moves the faulty PGN file to the folder 'suspended_pgn_files', located in the applications root folder. If this folder doesn't exist, *pgn2scid* creates it. It then adds the log files content to the master log file. The initial log file is deleted. Afterwards *pgn2scid* goes on to the next PGN file or to the next selected option.

When you have suspended a faulty PGN file, you can always read the master log file which gives you hints about where in this PGN file you have to look for the mentioned errors. You can then open the PGN file with a text editor (it's just an ASCII file), correct any errors and move the file back to the applications working directory to see if it now passes the conversion process without errors.

Note

To correct errors in a PGN file you need to have a good understanding of the PGN file format and the algebraic chess notation, otherwise you could make things even worse!

10. The 'png2scid_error.log' file

As mentioned in section 9.2 *pgn2scid* runs an error log file called 'pgn2scid_error.log', which is located in the applications root directory. It contains all cumulated conversion errors as well as occurred network errors in a chronological order with the latest entries at the end of this file. Each entry begins with a timestamp, followed by the actual error or warning message. In case of pgnscid conversion errors the users' reaction (ignore error / suspend file) is also logged. If the user suspends a faulty PGN file, the full path to this file is given. In the image below you can see an excerpt from this logfile, opened in a text editor.

```
14  2017-09-05 18:13:29
15  p2s_17-09-05_17-50-44.pgn:(game 1167, line 28594) PGN header '[' seen inside game (result missing ?)
16  p2s_17-09-05_17-50-44.pgn: error(s) IGNORED by user!
17
18  2017-09-05 18:13:32
19  p2s_17-09-05_18-13-12.pgn:(game 1167, line 28594) PGN header '[' seen inside game (result missing ?)
20  p2s_17-09-05_18-13-12.pgn: error(s) IGNORED by user!
21
22  2017-09-19 13:04:30
23  p2s_17-09-19_13-02-54.pgn:(game 1167, line 28594) PGN header '[' seen inside game (result missing ?)
24  p2s_17-09-19_13-02-54.pgn:(game 2684, line 64877) Error reading move in game Rabindra, KO - Lochan,
    Kumar Das, 2017: O-O
25  p2s_17-09-19_13-02-54.pgn: file SUSPENDED by user!
26  Suspended file moved to /home/akr/bin/pgn2scid/suspended_pgn_files
27
28  2017-09-19 16:50:31
29  NETWORK ERROR - Downloading file 'twic1190g.zip' from TWIC server: timed out
30
31  2017-09-22 18:48:05
32  NETWORK ERROR - Request information from TWIC server: [Errno -2] Name or service not known
```

11. Invoke 'scmerge' to merge Scid files with an existing database

You have to select this option if you want to merge two or more *Scid* databases, either created by ***pgn2scid*** (please refer to section 9 for more details) or from any other source. In most cases, the other database to merge, specified in 'Select database' (see section 11.3) is your 'main database', containing all the games you have collected so far. ***pgn2scid*** uses the external command line tool *scmerge* to perform this operation.

As with *pgnscid*, *scmerge* is executed in the background, so you don't have to deal with a command line interface. Actually, due to its internal function, *scmerge* can't merge *Scid* databases directly. Instead it always creates a third database, consisting of the two databases you want to merge. You don't have to worry about this because ***pgn2scid*** has a build in 'workaround' addressing this issue so at the end of the merge operation you will always and only have your existing database, containing your already collected games plus all the new games. There are some file operations involved here hence it's important to BACKUP your database regularly.

Note

It's advisable to merge a larger number of PGN files (described in section 8 of this manual) before converting them to the native *Scid* format, particularly if you finally want to merge them with an existing Scid database because it minimises file operations.

15

11.1 Create a ZIP compressed copy of the existing database before merging

Before starting database merging *pgn2scid* can create a zipped copy of your existing database. This is NO REPLACEMENT of a real backup, of course. However, it can help to recover your database quickly if something goes wrong during the merging process or when you realised that, despite the warning message, you mistakenly merged a defective Scid file. When you select this option, *pgn2scid* ZIP compresses all database files, namely the .si4, the .sg4 and the .sn4 file into one ZIP file and writes it into the 'scid_db_copy' folder, located in the applications root directory. If this directory doesn't exist, *pgn2scid* creates it first.

Similar to the PGN merge operation described in section 8 a timestamp format is used to safe the zipped *Scid* database copy. Let's assume your existing *Scid* database is named 'my_chessgames', the ZIP file will be named as 'my_chessgames_YY-MM-DD_HH-MM-SS.zip'.

Note

If there are no Scid files in your working directory to merge, the merging process is skipped, hence there's actually no need to create a ZIP compressed copy of your existing database. In this case *pgn2scid* asks you whether you would like to create such a copy anyway.

11.2 Delete remaining Scid files after merging

After merging database files successfully *pgn2scid* can delete the remaining *Scid* files because they are not necessary anymore.

When you don't select this option because you would like to keep these files, *pgn2scid* moves them to the folder 'scid_files', located in the applications root directory. If this folder doesn't exist, *pgn2scid* creates it.

11.3 Select database

Here you have to select an already existing database to which you want to add other *Scid* files to, either created by *pgn2scid* or from other sources. If 'Invoke *scmerge* to merge Scid files with an existing database' is selected and there's no database given here, a warning message pops up informing you that you have to select a *Scid* database first.

12. START

When *pgn2scid* is configured as desired (or the last successful configuration has been read out of the 'pgn2scid.ini' file) you can press the <START> button. *pgn2scid* then processes all selected options from top to bottom if possible. Within the status window (see section 1 for more information) you can follow the steps currently performed by *pgn2scid*.

After completing all tasks successfully, *pgn2scid* writes the current set of options as well as the selected working and database path to a configuration file called 'pgn2scid.ini'. This file is located in the applications root directory.

Note

When you start *pgn2scid* it reads the configuration file 'pgn2scid.ini'. If this file is corrupt, *pgn2scid* displays an appropriate warning message. Just click on <OK> then. There's no further action required on your part, because *pgn2scid* always creates a new and valid configuration file based on your last settings as soon as all selected options pass without errors. The only consequence is that your last settings might be lost and you might have to reconfigure the program.

13. Exit

Exits *pgn2scid*.