

Typesetting Chess in L^AT_EX with the **skak** Package

Version 1.3

Torben Hoffmann
e-mail: Torben.Hoffmann@motorola.com

March 10, 2005

Abstract

This user guide shows how to use an enhancement to the existing package for typesetting chess in L^AT_EX (**skak**). The primary improvement over the old package is that it has become easier to typeset chess games with this new package.

Contents

1	Change History	2
2	Terms of Usage	3
3	Acknowledgements	3
4	The Old chess Package Versus the New skak Package	3
5	How to use the skak Package	4
5.1	Handling of Variations	4
5.2	Changing the Font	5
6	Installing the skak Package	5
6.1	Using the lambda Package	5
7	Future Additions/Wish-list	5
7.1	Choosing Language for mainline	5
7.2	Algebraic Notation used for Typesetting	5
7.3	Fonts	5
7.4	Optimisation of the Implementation	6
7.5	Test Suite	6
7.6	Translating PGN Files	6
7.7	Typesetting <i>e.p.</i> after en passant moves	6

1 Change History

Version 1.3

- Dirk Bächle has reworked the typesetting of chess moves:
 - A new capture symbol has been introduced to make the spacing around it better.
 - The figurine symbols have been improved so that the spacing around them are more uniform — it looks really good now, at least to my eyes.
 - A proposal for what syntax the `skak` package should handle in the future has been included in the distribution.
 - The Informator symbols “novelty”, “comment” and “various” needed another pair of curly braces in order to work right within the “mainline” and “variation” environments.
 - The check for trailing spaces in the macro “typeset@cmoves” had to be removed because it led to problems at the end of a “mainline” or “variation” environment when an additional Informator symbol was involved.
- A reference to the `pgn2ltx` tool created by Dirk Bächle has been added to the document.
- A reference to an URL describing the PGN standard has been added.

Version 1.2

- Dirk Bächle has — once again — done a great job:

- the informant symbols have been improved and now scale as they should (I have in the same go done some METAFONT clean-up associated with the symbols).
- he has also discovered that the selection of other fonts for the typesetting of moves can be done by redefining the `\skakfamily` command!
- added the `\movecomment` for flexible typesetting of comments — see the Reference Manual for details.
- The PostScript file are now 10 times smaller (for large files) and a comprehensive test of the PostScript ornaments has been created.

- All dead code has been removed from `skak.sty`.

Version 1.1

- Dirk Bächle ’s enhancements have been incorporated:
 - extension of the `\highlight` command; it now accepts X, x, O and o as optional arguments and produces a cross and a circle on the square, respectively, instead of a frame around the square.
 - unwanted generation of spaces removed.
 - a set of Informator symbols were added to the fonts and made available through commands documented in the `informator.ps` file.

- Two fixes by Ulrike Fischer have been incorporated:

- a bug in `\ParseCoordinates`.
- a problem in the fonts regarding size.
- Ulrike Fischer reported a problem with opening spaces causing the input to be ignored — this has been fixed and `test/test2.tex` verifies this.

Version 1.0

The major novelty in this release is that the documentation has been updated and a reference manual has been created.

2 Terms of Usage

This package is distributed under the terms described in the Latex Project Public Licence, i.e.,

This software is copyright but you are granted a license which gives you, the “user” of the software, legal permission to copy, distribute, and/or modify the software. However, if you modify the software and then distribute it (even just locally) you must change the name of the software to avoid confusion.

3 Acknowledgements

I would like to thank the creator of the `lambda` package, Alan Jeffrey, for making a splendid package that made this package possible.

For discussions and alpha-testing I send my thanks to Guido Governatori — some of his ideas have already been implemented; the rest? Time will show.

Thanks to Harri Haanpaa for finding a nasty bug in the castling routine.

A big thank goes to Dirk Bächle for adding the Informator symbols to the fonts and for removing a couple of nasty bugs as well as improving some of the commands.

I thank Ulrike Fischer for spotting as well as fixing a big problem with the parsing of moves and for her hints to improving the fonts.

4 The Old `chess` Package Versus the New `skak` Package

The main reason for considering an improvement of the `chess` package that Piet Tutelaers made back in 1991 is the cumbersome user interface the package has when one wants to write about a chess game and display a diagram every now and then. Typesetting the two opening moves where white and black move their kingside knights is done as follows in the `chess` package:

```
\move g1f3 g8f6
```

The package then produces a nice typesetting of these moves in figurine notation, but it is hard to keep track of what is going on because one usually uses the SAN (Short Algebraic Notation, employed in the PGN standard for typing the moves — see <http://pgn.freeservers.com/standard.txt> for details) to write down the moves of a chess game. The SAN version of the two moves above is: 1. `Nf3 Nf6`, which is much clearer to most chess players. In the `skak` package the author of a chess article is allowed to use the SAN notation as input to the command that updates the chess board. In addition to making it easier for the author to write about chess using a familiar notation it also provides an easy way to include moves generated by a chess program

such as xboard in the document—most other chess programs can also output a PGN version of a chess game and from that you can extract the SAN recording of the moves.

The skak package can also input chess board positions given in the FEN notation (also used in the PGN standard), which is also quite standard in the domain of chess programs.

Apart from a better user interface the skak chess font contains three small modifications of the font created by Piet Tutelaers: the knight now looks a bit more “youthful”, the contour of the queen has been smoothened and all the chess pieces have been shrunk such that they do not fill as much of a square as before. I find this font nicer to look at, but your milage may vary.

5 How to use the skak Package

Writing about a chess game can be done straightforward:

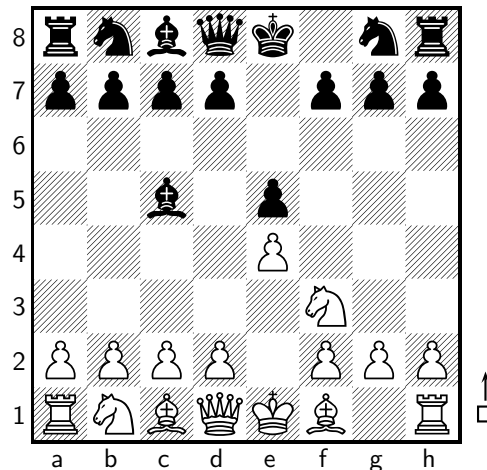
```
\newgame
```

```
\mainline{1. e4 e5 2. Nf3 Bc5}
```

starts a new game and produces the following in your document:

1 e4 e5 2 ♞f3 ♝c5

So far, so good. If you want to show the current position you just type `\[\showboard \]` in your document to get:



(the use of `\[... \]` is just to make sure the board is centered.)

This is the basic functionality of the skak package, but it offers many different ways in which one can talk about chess games. I think that a good way to show how the skak package can be used is to typeset the Fischer–Tal game from the old chess package. The result is in the file named `tuggame.ps`.

Further information about all the bells and whistles the skak package provides can be found in the *Reference Manual* where all commands of relevance are described.

5.1 Handling of Variations

The skak package does *not* support PGN variations such as

```
\mainline{12. Bb4 (12. Ng5 h6) 12... Ra8}
%gives an error
```

You have to type this as

```
\mainline{12. Bb4}
\variation{12. Ng5 h6}
\mainline{12...Ra8}
```

I do not have the nerves to update the parsing algorithm to cope with this so the best thing to do would be to craft a tool for translating PGN to \TeX (see 7.6).

5.2 Changing the Font

(New in Version 1.2)

Dirk Bächle has discovered that you can change the font used for typesetting of moves if you redefine the `\skakfamily` font.

I do not — however — know how to handle different chess fonts, so if someone could work this out I'll include that in a future version.

6 Installing the skak Package

I could not get the makefile to do the job, so if someone could help me out on this I would be very glad indeed.

Basically all you have to do is to follow the guidelines in the makefile. First you create all the required directories (stated in the `install` section of the makefile) and then you run the `make install` command. Remember to run `texhash` after you have updated your `texmf` tree.

Sorry that it is not working 100%— I just do not have the time for it right now.

If you cannot get the things working just put the files where \TeX and the other tools can find them and you will be just fine.

6.1 Using the lambda Package

The `skak` package uses the `lambda` package to do some of the hard processing necessary to allow the PGN notation as input — since it is not standard in \LaTeX distributions it is included in the distribution of the `skak` package.

7 Future Additions/Wish-list

7.1 Choosing Language for mainline

(Proposal date: mid 2002.)

It would be nice if one could choose the language for the `mainline`, `variation` and `\hidemoves` for each invocation as it allows for easier inclusion of analysis from chess programs when using a non-english language.

7.2 Algebraic Notation used for Typesetting

(Proposal date: mid 2001.)

Ingo Sander has suggested that the output of the typesetting should be the old algebraic notation where the from and to squares always are given. This requires a modification to the game engine: after each move has been made you should store the algebraic notation of the move since you have the to and from squares calculated at that moment. Changing the game engine is a bit hairy — even for me — so I'll postpone this.

7.3 Fonts

(Proposal date: dec-2002.)

Ulrike Fischer has suggested the creation PostScript version of the METAFONT font.

I do not know how to do this, but perhaps someone can help me?

Furthermore, Ulrike Fischer would like to see a more transparent fonthandling, so that it becomes easy to change to another chess-font. (Version 1.2 news: see Section 5.2.)

7.4 Optimisation of the Implementation

(Proposal date: 2-Jan-2003.)

I would like to improve some areas of the implementation since they are unnecessary slow and/or complicated, e.g., `\IsPieceName` could easily be implemented as a case statement instead of a list look-up. I have a strong feeling that this would be a lot faster than the current implementation which has a very functional programming flavour — this was very helpful during the development, but it has a tendency to be a bit inefficient.

Another great improvement would be to let the commands `\Mainline` and `\typeset@A` use the same parsing algorithm — then one could implement the long algebraic notation as suggested by Ingo Sander by letting the action taken on a move be both updating of the board *and* typesetting of the move!

7.5 Test Suite

(Proposal date: 03-Jan-2003.)

It would be a very good idea to create a test suite created using normal testing techniques. Areas of test should include:

- resolving ambiguous moves, i.e., ensuring that moves like ♖ad1 moves the right rook.
- all legal SAN moves are accepted.

7.6 Translating PGN Files

(Proposal date: 03-Jan-2003.)

It would be very nice if one had a command line tool that could translate PGN files to a TeXfile.

27-Sep-2003: Take a look at <http://pgn2ltx.sourceforge.net> for a nice helper tool.

7.7 Typesetting *e.p.* after en passant moves

Andreas Wilm asked how one should input en passant moves, especially the addition of *e.p.* after the capture.

This is not part of the SAN notation as described in the PGN standard and it requires that the improvement described in Section 7.4 is in place before it is possible to add the possibility to add the *e.p.*. It has to be this way because I do not want to support more than what is described by the PGN standard.