

# A L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> class for Draft Screenplays

by  
John Pate

This paper describes class `screenplay`  
version v1.487, released 2006/09/27

## 1 Introduction

Every year *The Academy of Motion Picture Arts and Sciences* runs a competition for screenplay writers, the *Nicholl Fellowships in Screenwriting*. Full-length feature film screenplays are submitted (for a fee). You can win money but, more importantly, placing in the competition will make agents take notice. The *Academy* recommends a particular format which this `class` file attempts to duplicate, using a `.pdf` available from the Academy website as the model. This format has various functions, amongst which is a gate-keeping function to show you have at least a first idea. It also gives readers a better chance of assessing how long, in filmic terms, your screenplay would be. Formatted along the lines recommended, you should aim to have your screenplay in the region of 120 pages long. Although they say A4 paper is acceptable, it is widely held that you severely handicap yourself if you don't use American 8.5" by 11" Letter size paper. I have a ream on my lap as I type: 20 pound, 106 bright, acid free. My sister brought it from America for me—it's very expensive to buy outside the US and not easy to find. However, if all else fails you can have a stationer cut down A3 80gsm paper for you. It won't be the same weight but it's unlikely anyone will notice. One might imagine it's at the photocopy stage that dealing with A4 in America becomes a nuisance—one more reason to put your magnum opus to the bottom of the pile.

Of course, if you didn't know this already you likely need to do some serious reading and research *before* you start writing. I guess trying to understand what's going on here is a start.

The format specifications are pretty straightforward. You'll have to believe me when I say it wasn't quite so straightforward for me to start from a minimal understanding of T<sub>E</sub>X and L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> to a usable `.dtx` file.

This is an attempt to turn T<sub>E</sub>X into a typewriter emulator. A crazy idea—which is why it appealed to me, obviously. Fortunately I'm easily entertained.

I hope `screenplay.dtx` proves of use (and instruction) to others as it has to me.

Anyhoo, on with the show...

## 2 Getting started

You should have:

- `README`—plain text file
- `screenplay.dtx`—the source file
- `screenplay.ins`—driver file
- `screenplay.pdf`—the file I assume you’re reading right now
- `test.pdf`—format test `.pdf` result
- `COPYING`—GNU General Public License

If not, get `screenplay.zip` via <http://dvc.org.uk/sacrific.txt/screenplay.zip>

You use the command,

```
> latex screenplay.ins
to generate from screenplay.dtx...
```

- `screenplay.cls`—class file
- `example.tex`—example (very) mini-screenplay
- `test.tex`—format test source file

Once you’ve unpacked the bundle and generated the `class` file do,

```
> latex test
```

to generate the `test.dvi` you can compare with the `test.pdf`.

Make sure everything in the printing chain can deal properly with *letter paper*. Print out your version of `test` and the provided `test.pdf` (on letter paper of course!) and compare them. The two printouts should be dimensionally identical. If you have an old, or broken, version of `dvips` you’ll need to fix it before you proceed further.

## 3 Some example bits of markup

Although `screenplay.cls` builds on the standard `TEX` document class `article`, it is not intended you pass it any options (except for the release date specification, see below). The formatting required is minimalist. `screenplay.cls` will issue an error message if you use any options.

```
\documentclass{screenplay}[2006/09/27]
```

is your first command, you can give the optional date argument in case of any earlier copies of `screenplay.cls` lying around. I’ll try not to break things retro-actively with later revisions but this can’t be guaranteed. Per this revision, anything marked with `[2006/09/23]` or later will produce the expected results.

Your second command is, of course...

```
\begin{document}
```

The first typesetting command, in the body of the document, must be the one to build the title page. For the *Nicholl Competition* this is simply,

`\nicholl{the title of your screenplay}`

for other formats, please refer to **The Coding** section or `example.tex`.

The next command is always `\fadein` to produce the words FADE IN: left justified.

Next would typically be a slugline,

`\intslug[day]{Johnny's House}`

which would produce...

INT. JOHNNY'S HOUSE    DAY

I made the time of day optional because my screenplay had stuff happening in space where there is no time of day. You will likely *always* use either *day* or *night*. (Refer to the *Academy's* information on formatting.) Likewise there's an `\extslug[]{} for EXT`.

After a slugline you would typically have some description. This is simply ordinary text with paragraph breaks as per usual.

When you get to dialogue, you have the `dialogue` environment.

`\begin{dialogue}{Johnny}` After some description you  
would typically have    some dialogue. `\end{dialogue}`

Directions (short) can be put inside dialogue text either as an optional argument to the dialogue environment, `\begin{dialogue}[shouting]{Johnny}` which will result in parenthesized text below the dialogue name, or the `\paren{}` command can be used to put parentheses into the blocks of dialogue text.

Do not put any blank lines in dialogue text.

So you wander on with your screenplay requiring little in the way of L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>'s spiffy powers until you reach the end of the final scene. Personally I use **Elvis** with a selection of macros in my `.exrc` to save me having to type the commands I've invented.

The command at the end of the final scene is always `\fadeout` to produce the words FADE OUT: flushright.

Then it's simply a case of `\theend` and `\end{document}`.

The only tricky parts of the formatting are deciding where to put *pagebreaks* and hunting down places where *linebreaks*... `\` ... will help T<sub>E</sub>X out with formatting. You may need to insert linebreaks manually even where 'Overfull `\hbox`' warnings are absent. You'll know when it is necessary by spotting extra vertical whitespace in the document. `example.tex` demonstrates this, read through the raw `.tex` file and try modifying it as indicated. It's likely you will have to put a quite a few linebreaks manually into dialogue text due to the restricted margins.

**Do not put in pagebreaks until you have really, really, really finished editing.**

Putting in pagebreaks will likely involve using the `\dialbreak` command, though for description text `\pagebreak` is all that is required.

The `\dialbreak` command needs to know the character name and any optional directions as required.

If you've used the `\titleover` environment you may need the `\titbreak` command (more tea, vicar?). However, mostly for TITLE OVER: you'll be doing a date to accommodate flashbacks or flashforwards, I expect, so use the `\centretitle` command as in

`\centretitle{December 21, 2012}`.

Yes, that's the English (or for that matter French and Canadian) "centre" not "center" in a stunning break from tradition.

For further enlightenment, you should scrutinise the raw text of `example.tex` and peruse **The Coding** section of this document.

You can see a completed script marked-up via L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> and formatted with `screenplay.cls` at <http://dvc.org.uk/sacrific.txt/>. There is also, at time of writing, a “work-in-progress” script that has had manual linebreaks inserted but not manual pagebreaks.

## 4 Specs from the Academy

Be sure to read and inwardly digest, <http://www.oscars.org/nicholl/format.html> and view, <http://www.oscars.org/nicholl/script.pdf>, the model document.

- Letter paper 8.5”x11.0”  
Courier 12pt, 10 pitch
- **Page**  
Top margin: 1.0”  
Bottom margin: 0.5”-1.5”  
Left margin: 1.5”  
Right margin: 1.0”
- **Dialogue**  
tab NAME to: 4.2”  
Left margin: 3.0”  
Right margin: 2.5”  
*Parenthetical*  
Left margin: 3.7”  
Right margin: 3.3”
- **Numbering**  
top right, Arabic  
Top margin: 0.5”  
Right margin: 0.75”

INT/EXT. SLUG LINE DESCRIPTION DAY/NIGHT  
(with some kind of extra spacing before time of day)

## 5 The Coding

`screenplay.cls` v1.487 was tested with...

- pdf<sub>E</sub>T<sub>E</sub>X 3.141592-1.21a-2.2 (WEB2C 7.5.4)
- dvips(k) 5.95a
- kpathsea version 3.5.4
- article 2004/02/16 v1.4f

IMHO, it's worth reading through the comments, there's some useful stuff for you in there.

We are a L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> thang...

```
1 \NeedsTeXFormat{LaTeX2e}[2003/12/01]
```

Say it like it is...

```
2 \ProvidesClass{screenplay}[2006/09/27 v1.487
```

```
3 LaTeX2e Draft Screenplay document class
```

```
4 (C)2006 John Pate
```

```
5 released under the GNU General Public License]
```

Whinge if the user is being a bozo...

```
6 \DeclareOption*{%
```

```
7     \PackageError{screenplay}{Unwanted option '\CurrentOption'}{%
```

```
8     {There are no options available for class screenplay v1.487}}{}
```

```
9 \ProcessOptions\relax
```

ifthen needed for testing for optional args...

```
10 \RequirePackage{ifthen}
```

Use `article` class for its base functionality, *must* be American letter paper *not* A4, `screenplay.cls` eats any papersize options and ignores them...

```
11 \LoadClass[12pt,letterpaper,onecolumn,oneside]{article}[2004/02/16]
```

`geometry` package makes dvips/pdf do the right thang for letter paper...

```
12 \RequirePackage[dvips,letterpaper]{geometry}
```

...and make sure your printer driver knows letter. Yes, you *must* use American 8.5"x11" letter paper—stationers in the UK call it 'quarto' paper, see...

<http://www.gwydir.demon.co.uk/jo/units/print.htm>

```
13 \setlength{\paperheight}{11in}
```

```
14 \setlength{\paperwidth}{8.5in}
```

*Must* have Courier as font (may cause warnings about font stuff, ignore)...

```
15 \RequirePackage{courier}
```

```
16 \renewcommand*\familydefault{\ttdefault}
```

Can load hyphenation ability for non-standard font...

```
17 %\RequirePackage[htt]{hyphenat}
```

but *don't*, you should *not* hyphenate for Nicholl format you want this...

```
18 \raggedright
```

(you'll have to nuke `\raggedright` wherever found to test out hyphenation)

Page Nums Arabic top right...

```
19 \pagenumbering{arabic}
```

```
20 \pagestyle{myheadings}
```

CR/LF Line spacing / pitch—it looks to me the standard L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> courier font is correct pitch, 12pt line-spacing is what's in `script.pdf`...

```
21 \setlength{\baselineskip}{12pt plus 0pt minus 0pt}
```

and this `\baselinestretch` appears to emulate `script.pdf` closely...

```
22 \renewcommand*{\baselinestretch}{0.8}
```

adjust space above `\minipage` in dialogue to emulate description linebreaks, must be adjusted if `\baselinestretch` is altered...

```
23 \newcommand{\dialfix}{\vspace{3pt}}
```

skip one line between paras...

```
24 \setlength{\parskip}{\baselineskip}
```

no paragraph indent...

```
25 \setlength{\parindent}{0in}
```

**Margins/tabs** See specs, fudge factors needed to resemble the ‘correct’ layout, take a ruler to it and `script.pdf`...  
left margin 1.5”, default 1” +

```
26 \setlength{\evensidemargin}{0.5in}
```

```
27 \setlength{\oddsidemargin}{0.5in}
```

top margin (0.5” to number, 1” to body)...

```
28 \setlength{\topmargin}{-0.5in}
```

```
29 \setlength{\headsep}{0.4in}
```

```
30 \setlength{\topskip}{0in}
```

11” - (1” top margin) - (0.5” bottom) =

```
31 \setlength{\textheight}{9.5in}
```

...bottom can go to 1.5” for page breaking purposes.

6” width for description text to give 1” right margin, +0.1” fudge...

```
32 \setlength{\textwidth}{6.1in}
```

Extra inter-word spacing in sluglines...

```
33 \newcommand{\slugspace}{\hspace{2.0em plus 0em minus 0em}}
```

...could do it various other ways...

```
34 %\newcommand{\slugspace}{ - }
```

```
35 %\newcommand{\slugspace}{\quad}
```

### Tabbing / spacing dialogues

Width of dialogue text, 3” with 0.2” fudge...

```
36 \newcommand{\dialwidth}{3.2in}
```

3” from left edge of page (1.5” left margin + 1.5”)...

```
37 \newcommand{\dialtab}{\hspace*{1.5in}}
```

**name**, tab in 4.2” (margin + `\dialtab` + `\dialnametab`) from left edge of page...

```
38 \newcommand{\dialnametab}{\hspace*{1.2in}}
```

parenthetical 1.5” with 0.2” fudge...

```
39 \newcommand{\parenwidth}{1.7in}
```

parenthetical, tab in 3.7” (margin + `\dialtab` + `\parentab`) from left edge of page...

```
40 \newcommand{\parentab}{\hspace{0.7in}}
```

fix for formatting of multi-line parenthetical (I really should look at the `TeX` source sometime because I don’t know why I had to do this.)...

```
41 \newcommand{\parenkludge}{\dialfix}
```

AFAICS floats not involved...

```
42 \renewcommand*{\textfraction}{1}
43 \renewcommand*{\topfraction}{0}
44 \renewcommand*{\bottomfraction}{0}
```

## Title Page

Use one of `\nicholl`, `\titleauthor`, `\titleextra` to taste. This must be the first command after `\begin{document}`.

`\nicholl` `\nicholl{title of your magnum opus}` for Nicholl Competition—just the title, not bolded...

```
45 \newcommand{\nicholl}[1]{
46 \thispagestyle{empty}
47 \vspace*{2.25in}
48 \begin{center}\MakeUppercase{#1}\end{center}
49 \newpage\setcounter{page}{1}
50 }%nicholl
```

Now the ability to make the the type of cover pages for Draft Scripts submitted to agents, `\titleauthor` being of the form suggested in `script.pdf`.

You may have to adjust the `\titlesepspace` if you put in a lot of text but if you do there's probably too much...

```
51 \newcommand{\titlesepspace}{\vspace{3.75in plus 0in minus 1.0in}}
```

there's also a the space occupied by the optional copyright year...

```
52 \newcommand{\copyspace}{\vspace*{2ex plus 0ex minus 0ex}}
```

`\titleauthor` `\titleauthor [year] {title} {author's name} {contact details\\email etc}` the copyright notice doesn't fit the spec so optional...if you put it on the web though © certainly won't hurt!

```
53 \newcommand{\titleauthor}[4][]{
```

no page number...

```
54 \thispagestyle{empty}
```

0.5" topmargin +...

```
55 \vspace*{2.25in}
```

centre and bold the title (bolding the title is probably the only embellishment you're allowed)...

```
56 \begin{center}{\textbf{\MakeUppercase{#2}}}\
```

```
57 \vspace{\baselineskip}
```

```
58 by\
```

```
59 \vspace{\baselineskip}
```

```
60 #3\
```

```
61 \vspace*{\baselineskip}
```

logic for handling optional © argument...

```
62 \ifthenelse{\not\equal{#1}{}}{
```

```
63 {
```

```
64 \scriptsize\copyright\normalsize#1
```

```
65 }{
```

```
66 \copyspace
```

```
67 }
```

```
68 \end{center}
```

`\titlesepspace` is a rubber length, try not put in excessive amounts of text though...

69 `\titlesepspace`

Details in a `\parbox 5"` out from left margin...

70 `\hspace*{4.0in}\parbox[t]{2.0in}{\raggedright#4}`

reset page numbers for actual blurb...

71 `\newpage\setcounter{page}{1}`

72 `}%titleauthor`

`\tilteextra` `\titleextra{extra blurb}{title}{author's name}{contact details\\email etc}` for title page—author details with extra blurb not recommended by `script.pdf` but by popular request, apparently some people like to put in copyright and WGA registration details

73 `\newcommand{\titleextra}[4]{`

74 `\thispagestyle{empty}`

75 `\vspace*{2.25in}`

76 `\begin{center}{\textbf{\MakeUppercase{#2}}}\`

77 `\vspace{\baselineskip}`

78 `by\`

79 `\vspace{\baselineskip}`

80 `#3\`

81 `\vspace*{\baselineskip}`

82 `\copyspace`

83 `\end{center}`

84 `\titlesepspace`

Extra details 3" out in a `\parbox, 2"` space then other details in a `\parbox 5"` out...

85 `\parbox[t]{2.0in}{\raggedright#1}\hspace*{2.0in}`

86 `\parbox[t]{2.0in}{\raggedright#4}`

87 `\newpage\setcounter{page}{1}`

88 `}%titleextra`

End title page—author details

Dialogue blocks...

`\paren` `\paren{hello world}` parenthetical within dialogue environment stage dirs... only for *very short* text (directions) within dialogue environment, the optional argument is for internal use *only*...

89 `\newcommand{\paren}[2][\]{%`

90 `#1\parentab\parbox[t]{\parenwidth}{\raggedright(#2)\parenkludge}\`

91 `}%paren`

dialogue env `\begin{dialogue}[directions]{character name} ... \end{dialogue}`

`\begin{dialogue}{character name} ... \end{dialogue}`

`\minipage` won't allow pagebreaks, see below for new manual pagebreak command for this environment. Do *not* have blank lines inside this environment. **Name** is outside `\minipage` to mitigate 'Overfull \hbox' warnings with long names...

92 `\newenvironment{dialogue}[2][]{`

93

94 `\dialtab\dialnametab\hbox{\MakeUppercase{#2}}\nopagebreak\`

95 `\dialtab\begin{minipage}[b]{\dialwidth}`



```

96 \dialfix
97 \raggedright
98 \ifthenelse{\not\equal{#1}{}}{
99 {
100 \paren[]{#1}
101 }{ }
102 }{
103 \end{minipage}
104
105 }%dialogue

\dialbreak Forcing pagebreak in dialogue,
\dialbreak{Character}
\dialbreak[directions]{Character}
(need to know character name— $\text{\LaTeX 2}_{\epsilon}$  scoping rules don't seem to like global vars, somebody
tell me otherwise) Note: only do any manual pagebreaks when you absolutely positively have killed
every muthafucka in the room!...
106 \newcommand{\dialbreak}[2][]{
107 \\\dialnametab\hbox{(MORE)}%
108 \end{dialogue}
109 \pagebreak
110 \ifthenelse{\not\equal{#1}{}}{
111 {
112 \begin{dialogue}[#1]{\MakeUppercase{#2} (CONT'D)}
113 }{
114 \begin{dialogue}{\MakeUppercase{#2} (CONT'D)}
115 }
116 }%dialbreak

Sluglines
Optional arg for night/day. You probably should not consider time of day optional when you're
writing descriptions—unless you're in a spaceship, of course...

\intslug INT. SLUGLINE DESCRIPTION DAY
\intslug[time of day]{description text}
\intslug{description text}
117 \newcommand{\intslug}[2][]{
118
119 \MakeUppercase{INT. #2}%
120 \ifthenelse{\not\equal{#1}{}}{
121 {
122 \slugspace\MakeUppercase{#1}\hspace{\fill}
123 }{ }
124 \nopagebreak
125
126
127 }%intslug

\extslug EXT. SLUGLINE DESCRIPTION DAY
\extslug[time of day]{description text}

```

```

\extslug{description text}
128 \newcommand{\extslug}[2][]{
129
130 \MakeUppercase{EXT. #2}%
131 \ifthenelse{\not\equal{#1}{}}{
132     {%
133 \slugspace\MakeUppercase{#1}\hspace{\fill}
134     }{}
135 \nopagebreak
136
137
138 }%\extslug

\intextslug Apparently people do this...
INT./EXT. SLUGLINE DESCRIPTION DAY
\intextslug[time of day]{description text}
\intextslug{description text}
139 \newcommand{\intextslug}[2][]{
140
141 \MakeUppercase{INT./EXT. #2}%
142 \ifthenelse{\not\equal{#1}{}}{
143     {%
144 \slugspace\MakeUppercase{#1}\hspace{\fill}
145     }{}
146 \nopagebreak
147
148
149 }%\intextslug

\extintslug and/or this...
EXT./INT. SLUGLINE DESCRIPTION DAY
\extintslug[time of day]{description text}
\extintslug{description text}
150 \newcommand{\extintslug}[2][]{
151
152 \MakeUppercase{EXT./INT. #2}%
153 \ifthenelse{\not\equal{#1}{}}{
154     {%
155 \slugspace\MakeUppercase{#1}\hspace{\fill}
156     }{}
157 \nopagebreak
158
159
160 }%\extintslug

Title Over

titleover env Title Over...
\begin{titleover} ... \end{titleover} as pseudo dialogue environment (think “Star Wars”)
the optional argument is for internal use only...

```

```

161 \newenvironment{titleover}[1][]{
162
163 \ifthenelse{\not\equal{#1}{}}{
164 {%true
165 TITLE OVER (#1):\\
166 }{%false
167 TITLE OVER:\\
168 }%ifthenelse
169 \vspace{\baselineskip}
170 \nopagebreak
171 \dialtab\begin{minipage}[b]{\dialwidth}
172 \raggedright
173 }{
174 \end{minipage}
175
176 }%titleover

\titbreak  pagebreak for titleover environment...
177 \newcommand{\titbreak}{
178 \\ \dialnametab\hbox{(MORE)}%
179 \end{titleover}
180 \pagebreak
181 \begin{titleover}[CONT'D]
182 }%titbreak

\centretitle  titleover for short text \centretitle {short text} most often a date...
183 \newcommand{\centretitle}[1]{
184
185
186 TITLE OVER:
187 \nopagebreak
188
189 \hspace*{\fill}{#1}\hspace{\fill}
190
191
192 }%centretitle

Other spiffy macros

\fadein  \fadein always the first thing in your script, for some reason...
193 \newcommand{\fadein}{
194
195
196 FADE IN:
197
198
199 }%fadein

\fadeout  \fadeout always the last thing in your script, for some reason...
200 \newcommand{\fadeout}{

```

```

201
202
203 \hspace*{\fill}FADE OUT:
204
205
206 }%fadeout

\intercut I don't bother with this but it's in script.pdf...
207 \newcommand{\intercut}{
208
209
210 \hspace*{\fill}INTERCUT WITH:
211
212
213 }%intercut

IMHO this stuff is useful for description ...

\pov point of view...
214 \newcommand{\pov}{P.O.V. }

\nb
215 \newcommand{\nb}{NOTE: }

\theend Yeah! This is what we've been waiting for...OK, so this is actually the last thing in the script...
216 \newcommand{\theend}{
217
218
219 \hspace*{\fill}THE END\hspace{\fill}
220
221
222 }%THE END

```

## 6 Todo

If anybody else uses it maybe they'll spot stuff that's wrong but *Johnny* says, "It works for me."

The test file `test.tex` shows `screenplay` markup applied to "FOR A FEW DAYS MORE" (<http://www.oscars.org/nicholl/script.pdf>) and the `test.pdf` file was generated from that.

A printout of the first page `script.pdf` and `test.pdf` show identical pitch and the inter-line spacing is pretty damn close. You can easily tell which is which because `test.pdf` numbers the first page, `script.pdf` starts numbering on page 2. You can suppress the first page number in your own documents with the `\thispagestyle{empty}` command if you so desire. (The cover page is how *I* want it.)

If I get any *useful* feedback I *may* alter things accordingly.

Feel free to help with documentation.

## 7 A note for our Sponsor

I don't think I'm alone when I say...

Hail Eris!  
All Hail Discordia!

## Change History

1.0.0	1.0.2
General: initial version of <code>.dtx</code> , class date [2006/09/23] . . . . . 1	General: minor code changes, revision to doc- umentation, added <code>test</code> file, <code>example</code> file, to <code>.dtx</code> , amended <code>.ins</code> . . . . . 1
1.0.1	1.0.3
General: minor code changes, revision to doc- umentation . . . . . 1	General: minor code changes, revision to doc- umentation, added change history . . . . . 1