

ScreenScript HowTo Rev. 1:

These are the instructions for installing and using ScreenScript on your Linux machine. If you have found this document somewhere other than at the Joseph R. Demers ScreenScript page, then you may want to go there first to get more information (<http://www.jrdemers.com/screenscript/screenscript.html>).

I wrote my first screenplay using Microsoft Word and it was extremely painful. I spent more time formatting text than I spent writing. This had to change! When I wrote my second screenplay I decided to switch to Latex (<http://www.latex-project.org/>). Latex is a type-setting program that allows you to write and not worry about the formatting. After you are done writing, the file is compiled and it produces a publication quality document based on a template file. The purpose of this document is to explain how to use ScreenScript, not how to use Linux (<http://www.linux.org>), or Latex (<http://www.latex-project.org/>).

1.0 First Rules

With that said, there are a couple of rules in Linux that if you follow will save you some headaches. First, no spaces in the directory or file name - use underscores.

For instance the file name,

`c:/home directory/The Hidden Agenda`

will not compile, but

`c:/home_directory/The_Hidden_Agenda`

will.

Second, Capital Letters Matter! To Linux (and therefore Latex),

`The_Hidden_Agenda`

is a different file from

`the_hidden_agenda`

Same with the directories!

Those are my two “heads-up” before you start working.

2.0 Download and Install

The first thing to do is download the screenscript.cls file to your computer. It is a simple text file, but I zipped it to make the transfer easier. After it is unzipped, you can edit it change it or do whatever you want! As a matter of fact, you probably will change it to tailor it to your needs. I use Mandrake Linux and my class files go into the directory:

`/usr/share/texmf/tex/latex/base/`

If you use a different brand of Linux, the class files may go somewhere else, but it doesn't matter. You can put the screenscript.cls file wherever you want since it is called from the document in the first line.

3.0 Document Syntax

For those of you who are used to a What-You-See-Is-What-You-Get type of editor (like MS Word), may it is best for me to show you what a line formatted document looks like. On the next page is the Introduction_Latex document that is on the ScreenScript website (http://www.jrdemers.com/screenscript/Introduction_Latex.pdf) before it was compiled. **Print it out and compare it to what follows!**

After the example, I will describe what the different commands mean, so don't let it intimidate you!!!

```
\documentclass[letterpaper, twoside]{/usr/share/texmf/tex/latex/base/screenscript}
\begin {document}

\title {the hidden agenda}
\author {Dr. Joseph R. Demers}
\date {July 21, 2004}
\address {2321 Bonita Dr.} {Glendale} {CA} {91028}
\phone {(626) 676-6175}
\email {jrd@jrdemers.com}

\extra {fade in:}
\scene {int} {windowed office, small technology complex} {day}
\action {IVAN MARCLOV looks out over the raging ocean from his windowed office high in the corner of the
technology complex. Dressed completely in black Marclov looks more like an assassin than a business man.}
\action {As he turns his chair towards the room, he hides the dark expression of anger with a friendly look of
incredulity. Across the desk stands a very distinguished looking SCIENTIST.}
\speech {marclov} {You are sure of this? Only ONE???)
\speechwithmidcomment {scientist} {Yes, sir.} {hands a folder to Marclov.} {Amazing isn't it. Out of the
entire human race, there was only one person that met the constraints.}
\speechwithcomment {marclov} {reading} {Yes, yes, amazing. Thank you doctor, I can see that this is very
complete. You may go.}
\action {The scientist is almost to the door before Marclov's pleasant voice stops him; he turns.}
\speechwithmidcomment {marclov} {Oh, and how is your family doing? Your daughter is as pretty as ever?}
{the surprised scientist nods} {Good.}
\action {Marclov suddenly produces a silver gun with a bulbous end and fires it at the scientist.}
\extra {blue flash:}
\action {The beam from the gun strikes the scientist in the chest and he falls to the ground barely concious.
Marclov gets up and alters a setting on the gun as he walks towards the groaning scientist.}
\action {The last thing the scientist sees is the smiling face of Marclov from above the silver gun.}
\extra {red flash, buzzing whine:}
\action {Marclov looks down at the dead scientist with satisfaction. He returns to his desk and keys a command
into his computer terminal.}
\action {A red nimbus appears around his office like a shield and then, in the center of the room, a smokey
portal appears. Suddenly, two bright yellow, goat-like eyes belonging to a DEMON-LIKE CREATURE stare
out.}
\speech {marclov} {Good news my friend. We have found a suitable candidate.}
\speech {creature} {It is about time, Marclov.}
\speechwithdoublecomment {marclov} {gestures with the gun} {Yes, I agree. And the good doctor has paid for
his tardiness.} {looking at the gun} {I must thank you for this wonderful weapon. What did you call it? A
\emph {T'nox} or something?}
\speechwithmidcomment {creature} {Something like that, yes.} {impatient} {The candidates?}
\speechwithcomment {marclov} {off-hand} {Candidate. There is only one.}
\speech {creature} {ONE???)
\speechwithmidcomment {marclov} {Yes, one... a man named...} {consults computer} {...Wilson Adam Grey.}
\extra {fast cut:}
\end {document}
```

That is the file that is required to generate the Introduction_Latex document that is on the ScreenScript website (http://www.jrdemers.com/screenscript/Introduction_Latex.pdf).

I wrote the text file using MS Word. You can use notepad or Emacs....IT DOESN'T MATTER what the text file is written with, all that matters is the class file.

3.1 The Formatting Rules

All commands begin with a \. Usually the options for the commands are placed in {}. You can look up the different Latex commands on the web or at the Latex site (<http://www.latex-project.org/>). Extra lines and returns in the text file do not matter! Notice that the commands are all lower case! This does matter.

The first command:

```
\documentclass[letterpaper, twoside]{/usr/share/texmf/tex/latex/base/screenscript}
```

tells what type of paper, if one side or two side print, and what type of class the document will be. Here is the path in {} to the screenscript class file. Notice that it is not necessary to include the extension .cls on the file name. This will actually confuse Latex.

All Latex documents begin with a

```
\begin{document}
```

and end with an

```
\end{document}
```

That is all, **NOT** your document name, just the word “document.” There are a couple of things that the screenscript class file is expecting when it goes to format your screenplay. In particular these are fairly self explanatory.

```
\title {the hidden agenda}  
\author {Dr. Joseph R. Demers}  
\date {July 21, 2004}  
\address {859 South Raymond Ave.} {Pasadena} {CA} {91105}  
\phone {(626) 304-7610}  
\email {jd@jrdemers.com}
```

NOTE: The screenscript class file automatically capitalizes some text. Pay attention to what is in caps and what is not. For instance the title is in lower case because it is automatically capitalized.

The following commands require some explanation.

The “extra” is used to add things in capital like FADE IN: or SUDDENLY and it puts an extra line above and below. It expects a single set of {}.

so \extra {fade in;} will result in

FADE IN:

The “scene” is used to put the scene into the standard format for screenplays. It expects three sets of {}.
so \scene {int} {windowed office, small technology complex} {day} will result in

INT – WINDOWED OFFICE, SMALL TECHNOLOGY COMPLEX - DAY

There is also a scenewithcomment command (not shown in the example) which puts the scene into standard format for screenplays but adds a comment. It expects four sets of {}.

so `\scene {int} {windowed office, small technology complex} {day} {titles begin}` will result in

INT – WINDOWED OFFICE, SMALL TECHNOLOGY COMPLEX – DAY – TITLES BEGIN

In the example, the “action” is used to call the shot or any scene information. It expects a single set of {}.

`\action {IVAN MARCLOV looks out over the raging ocean from his windowed office high in the corner of the technology complex. Dressed completely in black Marclov looks more like an assassin than a business man.}` will result in

IVAN MARCLOV looks out over the raging ocean from his windowed office high in the corner of the technology complex. Dressed completely in black Marclov looks more like an assassin than a business man.

Notice that upper case, lower case are transparent with the action command. Also, it adds a line after completion, so if you do two action commands in a row

`\action {IVAN MARCLOV looks out over the raging ocean from his windowed office high in the corner of the technology complex. Dressed completely in black Marclov looks more like an assassin than a business man.}`
`\action {As he turns his chair towards the room, he hides the dark expression of anger with a friendly look of incredulity. Across the desk stands a very distinguished looking SCIENTIST.}`
it looks like this:

IVAN MARCLOV looks out over the raging ocean from his windowed office high in the corner of the technology complex. Dressed completely in black Marclov looks more like an assassin than a business man.

As he turns his chair towards the room, he hides the dark expression of anger with a friendly look of incredulity. Across the desk stands a very distinguished looking SCIENTIST.

Now for the more complex commands, the speech commands. These are complex because each expects a different number of arguments and if one is left out, the compilation looks funny. The first is just the “speech” command. It expects two sets of {}. The first is the character in lower case and the second is what he says
`\speech {marclov} {You are sure of this? Only ONE???` this results in:

MARCLOV
You are sure of this? Only ONE???

Notice that the placement is correct!

What if you want to add a comment? The speechwithcomment needs 3 sets of {}

`\speechwithcomment {marclov} {reading} {Yes, yes, amazing. Thank you doctor, I can see that this is very complete. You may go.}` results in:

MARCLOV
(reading)
Yes, yes, amazing. Thank you doctor, I can see that this is very complete. You may go.

Notice that the character is properly capitalized and that the comment is put into () as it should be.

Or a comment in the middle? The `speechwithmidcomment` needs 4 sets of `{}`
`\speechwithmidcomment {scientist} {Yes, sir.} {hands a folder to Marclov.} {Amazing isn't it. Out of the entire human race, there was only one person that met the constraints.}` results in

SCIENTIST

Yes, sir.
(hands a folder to Marclov)
Amazing, isn't it. Out of the entire human race, there was only one person that met the constraints.

Or two comments? The `speechwithdoublecomment` needs 5 sets of `{}`
`\speechwithdoublecomment {marclov} {gestures with the gun} {Yes, I agree. And the good doctor has paid for his tardiness.} {looking at the gun} {I must thank you for this wonderful weapon. What did you call it? A \emph {T'nox} or something?}`

MARCLOV

(gestures with silver gun)
Yes, I agree. And the good doctor has paid for his tardiness.
(looks at silver gun)
I must thank you for this wonderful weapon. What did you call it? A *T'nox* or something?

Notice that there is another command in this one `/emph{T'nox}`. The `emph` command is a Latex command for emphasis (italicizes). Another one commonly used in screen writing is `/underline{}`.

Now, the next several commands do not have examples in the introduction, so don't look for them there.

There is `speechwithdoublemidcomment` which requires 6 sets of `{}`.
`\speechwithdoublemidcomment {phillip} {No, I insist, this is a horribly dismal place. All this sunshine...it is just unhealthy.} {gestures at clear sky} {No true society could develop under these conditions....} {gesturing at people} {...I mean what is the motivation of these people to do anything except wander around all day?}`

PHILLIP

No, I insist, this is a horribly dismal place. All this sunshine....it is just unhealthy.
(gestures at clear sky)
No true society could develop under These conditions....
(gesturing at people)
...I mean what is the motivation of these people to do anything except wander around all day?

The `speechwithmidspace` adds a space and requires 3 sets of `{}`.

```
\speechwithmidspace {ichan} {You are as fast as a snake, but still he avoids your attack.} {Why is this?}
```

ICHAN

You are as fast as a snake, but still he
avoids your attack.

Why is this?

Finally, `speechwithdoublespace` requires 4 sets of `{}`.

```
\speechwithdoublespace {grey} {I mean....I mean....I see you drive by yesterday.} {I get hit by a car....people  
have been chasing me...they said...} {...they said they wanted me to meet you.}
```

GREY

I mean....I mean....I see you drive by
yesterday.

I get hit by a car....people have been
chasing me...they said...

...they said they wanted me to meet
you.

OK, that is all of the commands!

4.0 How to Compile

Compiling means taking the text file and converting it into a `.dvi` file. This is a relative to the more common `.pdf` file and most `.dvi` readers have an export to `.pdf` option for conversion. The formatting has been set so that it results in the proper margins when the file is saved exported to a `.pdf` file and printed double sided.

In Linux, open a terminal window and then change into the directory where your text file is. Then all that is necessary is to type the following command (leave off quotes): “`latex your_file_name_here`” and then a you will see a bunch of commands fly by. If there is an error, which there will be if you have forgotten an “`{`” or used one of the speech comments with the wrong number of arguments, then the compilation will stop and ask for clarification. Usually it will tell you where the error is.

If you get no error, then, using the file browser, look for the new `your_file_name.dvi` file that has appeared in the directory. Don’t like it? Then edit the original text file a recompile.

SHOOT! You mean I need to type in the latex command again to recompile??? No, just hit the up arrow on the keyboard to cycle through the previous terminal commands. When you see the “`latex your_file_name_here`” command, hit enter and away it goes!

I **strongly** recommend that you compile the introduction above since we know that it will compile free of errors before you start on your own work.

5.0 Me

My e-mail address is jd@jrdemers.com. Please try to research any problems that you have before contacting me. As I get more feedback I will update this howto with the info.

Good luck! And if you write the next block buster....

JD