

#+TITLE: OSH principles for a \TeX nician

#+AUTHOR: Marcin Borkowski

#+INSTITUTE: \TeX AM scientific circle

#+DATE: 2013-04-18

#+TITLE: OSH principles for a TeXnician

#+AUTHOR: Marcin Borkowski

#+INSTITUTE: TEXUM scientific circle

#+DATE: 2013-04-18

Remark

For the sake of today's presentation, I will use the terms "work environment" and "text editor" interchangeably.

#+TITLE: OSH principles for a TeXnician

#+AUTHOR: Marcin Borkowski

#+INSTITUTE: TeXAM scientific circle

#+DATE: 2013-04-18

Remark

For the sake of today's presentation, I will use the terms "work environment" and "text editor" interchangeably.

In other words, I omit the problems of choosing your chair, keyboard, OS or PDF viewer.

#+TITLE: OSH principles for a TeXnician

#+AUTHOR: Marcin Borkowski

#+INSTITUTE: TeXAM scientific circle

#+DATE: 2013-04-18

Remark

For the sake of today's presentation, I will use the terms "work environment" and "text editor" interchangeably.

In other words, I omit the problems of choosing your chair, keyboard, OS or PDF viewer.

It doesn't mean they are irrelevant!

Disclaimer

As experience shows, the issue of choosing your text editor is *very delicate*.

Disclaimer

As experience shows, the issue of choosing your text editor is *very delicate*.

Therefore, at the very beginning let me emphasize that I will present *my personal opinions* on that matter.

Disclaimer

As experience shows, the issue of choosing your text editor is *very delicate*.

Therefore, at the very beginning let me emphasize that I will present *my personal opinions* on that matter.

That they are *right*, is another story. ;-)

Timetable

- * Introduction
- * Facts and myths
- * How to choose
- * Basic requirements
- * Typical errors
- * Example tools

- * Introduction
- ** Before I'll tell anything
- ** What I won't tell
- ** What I will tell

* Facts and myths

* How to choose

* Basic requirements

* Typical errors

* Example tools

Before I'll tell anything...

...I'm going to ask you a few questions.

Before I'll tell anything...

...I'm going to ask you a few questions.

- Does anybody here do anything connected with programming?

Before I'll tell anything...

...I'm going to ask you a few questions.

- Does anybody here do anything connected with programming?
- Does anybody use a text editor *older than him/herself*?

Before I'll tell anything...

...I'm going to ask you a few questions.

- Does anybody here do anything connected with programming?
- Does anybody use a text editor *older than him/herself*?
- Does the name of that editor have more than 4 letters?

Before I'll tell anything...

...I'm going to ask you a few questions.

- Does anybody here do anything connected with programming?
- Does anybody use a text editor *older than him/herself*?
- Does the name of that editor have more than 4 letters?

Overall, I think text editing is a solved problem.

Text editors are like wine. The older the better.

You do not want a shiny new text editor. You want the text editor that has been around long enough and used by enough geeks that every conceivable pattern of manipulating symbols on the screen has been thought of, and crystallized into a re-usable pack.

(see <http://blog.vivekhaldar.com/post/31970017734>)

What I won't tell

I won't tell anybody what editor you should use.

What I won't tell

I won't tell anybody what editor you should use.

Emacs rulez!

What I won't tell

I won't tell anybody what editor you should use.

Nor what editor to avoid.

What I won't tell

I won't tell anybody what editor you should use.

Nor what editor to avoid.

Vim sucks...

What I won't tell

I won't tell anybody what editor you should use.

Nor what editor to avoid.

Neither will I tell *how* to use the chosen editor

What I won't tell

I won't tell anybody what editor you should use.

Nor what editor to avoid.

Neither will I tell *how* to use the chosen editor
- for that, you have the user's manual!

What I will tell

I will tell you:

- what you should take into consideration when choosing your (A)T_EX work environment,

What I will tell

I will tell you:

- what you should (in my opinion!) take into consideration when choosing your (A)T_EX work environment,

What I will tell

I will tell you:

- what you should (in my opinion!) take into consideration when choosing your (A)T_EX work environment,
- what I use

What I will tell

I will tell you:

- what you should (in my opinion!) take into consideration when choosing your (A)T_EX work environment,
- what I use - but YMMV,

What I will tell

I will tell you:

- what you should (in my opinion!) take into consideration when choosing your (A)T_EX work environment,
- what I use - but YMMV,
- what does the usage of two example editors look like.

* Introduction

* Facts and myths

** On “ease of use”

** On effectiveness

* How to choose

* Basic requirements

* Typical errors

* Example tools

On “ease of use”

“Easy to use” usually means

On “ease of use”

“Easy to use” usually means “ineffective”.

On “ease of use”

“Easy to use” usually means “ineffective”.

- drop-down menus

On “ease of use”

“Easy to use” usually means “ineffective”.

- drop-down menus
- toolbars

On “ease of use”

“Easy to use” usually means “ineffective”.

- drop-down menus
- toolbars
- symbol panel

On “ease of use”

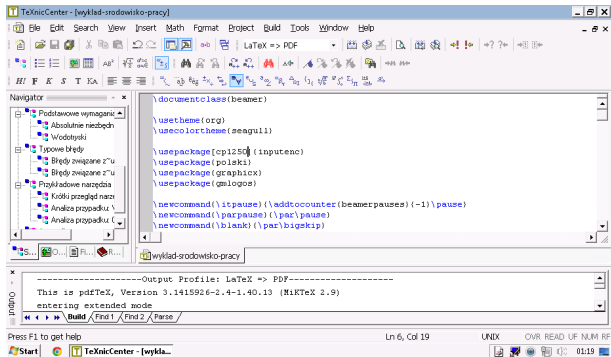
“Easy to use” usually means “ineffective”.

- drop-down menus
- toolbars
- symbol panel
- structure window

On “ease of use”

“Easy to use” usually means “ineffective”.

- drop-down menus
- toolbars
- symbol panel
- structure window



On effectiveness

“Effective” usually means

On effectiveness

“Effective” usually means “requiring some time investment”.

On effectiveness

“Effective” usually means “requiring (at the beginning) some time investment”.

On effectiveness

“Effective” usually means “requiring (at the beginning) some time investment”.

Some features of effective programs:

On effectiveness

“Effective” usually means “requiring (at the beginning) some time investment”.

Some features of effective programs:

- no drop-down menus

On effectiveness

“Effective” usually means “requiring (at the beginning) some time investment”.

Some features of effective programs:

- no drop-down menus, which are distractive, slow and take up precious screen real estate,

On effectiveness

“Effective” usually means “requiring (at the beginning) some time investment”.

Some features of effective programs:

- no drop-down menus, which are distractive, slow and take up precious screen real estate,
- no toolbars

On effectiveness

“Effective” usually means “requiring (at the beginning) some time investment”.

Some features of effective programs:

- no drop-down menus, which are distractive, slow and take up precious screen real estate,
- no toolbars, which are distractive, slow and take up precious screen real estate,

On effectiveness

“Effective” usually means “requiring (at the beginning) some time investment”.

Some features of effective programs:

- no drop-down menus, which are distractive, slow and take up precious screen real estate,
- no toolbars, which are distractive, slow and take up precious screen real estate,
- no symbol panel

On effectiveness

“Effective” usually means “requiring (at the beginning) some time investment”.

Some features of effective programs:

- no drop-down menus, which are distractive, slow and take up precious screen real estate,
- no toolbars, which are distractive, slow and take up precious screen real estate,
- no symbol panel, which is distractive, slow and takes up precious screen real estate,

On effectiveness

“Effective” usually means “requiring (at the beginning) some time investment”.

Some features of effective programs:

- no drop-down menus, which are distractive, slow and take up precious screen real estate,
- no toolbars, which are distractive, slow and take up precious screen real estate,
- no symbol panel, which is distractive, slow and takes up precious screen real estate,
- no structure window

On effectiveness

“Effective” usually means “requiring (at the beginning) some time investment”.

Some features of effective programs:

- no drop-down menus, which are distractive, slow and take up precious screen real estate,
- no toolbars, which are distractive, slow and take up precious screen real estate,
- no symbol panel, which is distractive, slow and takes up precious screen real estate,
- no structure window, which is distractive, slow and takes up precious screen real estate,

On effectiveness

“Effective” usually means “requiring (at the beginning) some time investment”.

Some features of effective programs:

- no drop-down menus, which are distractive, slow and take up precious screen real estate,
- no toolbars, which are distractive, slow and take up precious screen real estate,
- no symbol panel, which is distractive, slow and takes up precious screen real estate,
- no structure window, which is distractive, slow and takes up precious screen real estate,
- usable without a mouse

On effectiveness

“Effective” usually means “requiring (at the beginning) some time investment”.

Some features of effective programs:

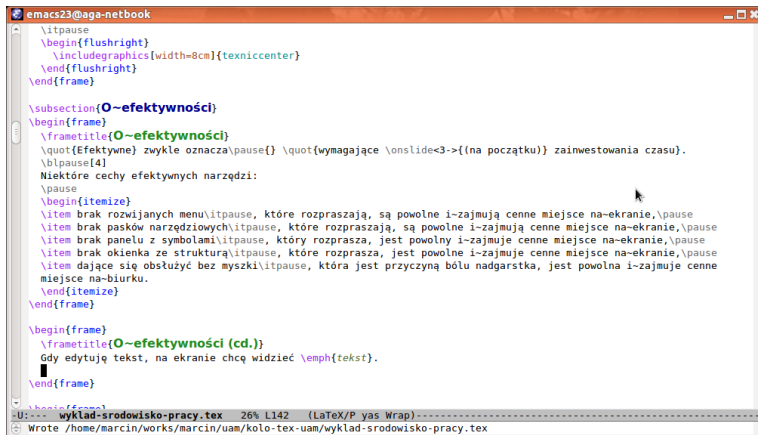
- no drop-down menus, which are distractive, slow and take up precious screen real estate,
- no toolbars, which are distractive, slow and take up precious screen real estate,
- no symbol panel, which is distractive, slow and takes up precious screen real estate,
- no structure window, which is distractive, slow and takes up precious screen real estate,
- usable without a mouse, which is slow and takes up precious desktop real estate.

On effectiveness (cont.)

When I'm editing text, I want to see *text*.

On effectiveness (cont.)

When I'm editing text, I want to see *text*.



```
emacs23@aga-netbook
\itpause
\begin{flushright}
\includegraphics[width=8cm]{texniccenter}
\end{flushright}
\end{frame}

\subsection{O~efektywności}
\begin{frame}
\frametitle{O~efektywności}
\quot{Efektywne} zwykle oznacza\pause{} \quot{wymagające \onslide<3->{(na początku)} zainwestowania czasu}.
\blpause[4]
Niektóre cechy efektywnych narzędzi:
\pause
\begin{itemize}
\item brak rozwijanych menu\itpause, które rozpraszają, są powolne i-zajmują cenne miejsce na-ekranie,\pause
\item brak pasków narzędziowych\itpause, które rozpraszają, są powolne i-zajmują cenne miejsce na-ekranie,\pause
\item brak panelu z symbolami\itpause, który rozprasza, jest powolny i-zajmuje cenne miejsce na-ekranie,\pause
\item brak okienka ze strukturą\itpause, które rozprasza, jest powolne i-zajmuje cenne miejsce na-ekranie,\pause
\item dające się obsłużyć bez myszki\itpause, która jest przyczyną bólu nadgarstka, jest powolna i-zajmuje cenne
miejsce na-biurku.
\end{itemize}
\end{frame}

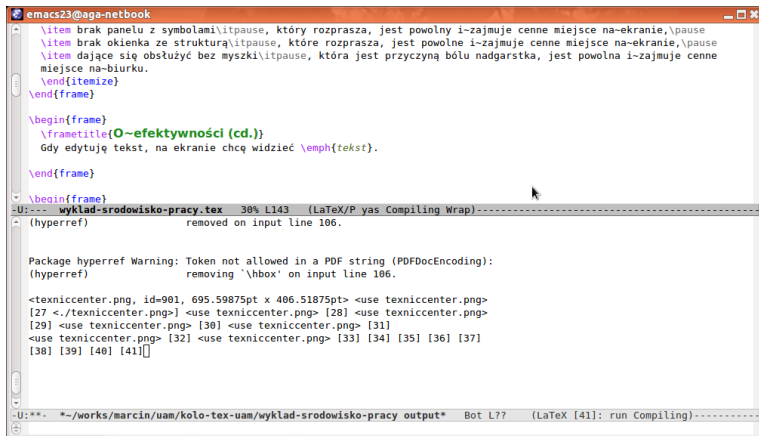
\begin{frame}
\frametitle{O~efektywności (cd.)}
Gdy edytuję tekst, na ekranie chcę widzieć \emph{tekst}.
\end{frame}

\begin{frame}
\end{frame}

-U:--- wykład-srodowisko-pracy.tex 26% L142 (LaTeX/P yas Wrap)-----
Wrote /home/marcin/works/marcin/uam/kolo-tex-uam/wyklad-srodowisko-pracy.tex
```

On effectiveness (cont.)

When I'm editing text, I want to see *much* text.



```
emacs23@aga-netbook
\item brak panelu z symbolami\itpause, który rozprasza, jest powolny i-zajmuje cenne miejsce na-ekranie,\pause
\item brak okienka ze strukturą\itpause, które rozprasza, jest powolne i-zajmuje cenne miejsce na-ekranie,\pause
\item dające się obsłużyć bez myszki\itpause, która jest przyczyną bólu nadgarstka, jest powolna i-zajmuje cenne
miejsce na-biurku.
\end{itemize}
\end{frame}

\begin{frame}
\frametitle{O~efektywności (cd.)}
Gdy edytuję tekst, na ekranie chcę widzieć \emph{tekst}.
\end{frame}

\begin{frame}
\end{frame}

-U:-- wyklad-srodowisko-pracy.tex 30% L143 (LaTeX/P yas Compiling Wrap)-----
(hyperref) removed on input line 106.

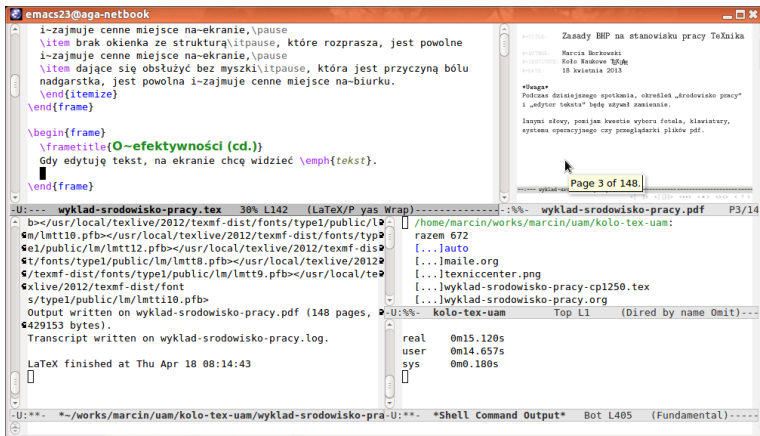
Package hyperref Warning: Token not allowed in a PDF string (PDFDocEncoding):
(hyperref) removing '\hbox' on input line 106.

<texniccenter.png, id=901, 695.59875pt x 406.51875pt> <use texniccenter.png>
[27 <./texniccenter.png>] <use texniccenter.png> [28] <use texniccenter.png>
[29] <use texniccenter.png> [30] <use texniccenter.png> [31]
<use texniccenter.png> [32] <use texniccenter.png> [33] [34] [35] [36] [37]
[38] [39] [40] [41]□

-U:**- *-~/works/marcin/uam/koło-tex-uam/wykład-srodowisko-pracy output* Bot L?? (LaTeX [41]: run Compiling)-----
```

On effectiveness (cont.)

When I'm editing text, I want to see *a lot of text!*



On effectiveness (cont.)

Some other features of effective tools:

- mature

On effectiveness (cont.)

Some other features of effective tools:

- mature
- stable

On effectiveness (cont.)

Some other features of effective tools:

- mature
- stable
- customizable

On effectiveness (cont.)

Some other features of effective tools:

- mature
- stable
- customizable

“People talk about getting used to a new editor, but over time, it is precisely the opposite that should happen - the editor should get used to us.”

(see <http://blog.vivekhaldar.com/post/31970017734>)

On effectiveness (cont.)

Some other features of effective tools:

- mature
- stable
- customizable

“People talk about getting used to a new editor, but over time, it is precisely the opposite that should happen - the editor should get used to us.”

(see <http://blog.vivekhaldar.com/post/31970017734>)

- “Easy things should be easy and hard things should be possible”

On effectiveness (cont.)

Some other features of effective tools:

- mature
- stable
- customizable

“People talk about getting used to a new editor, but over time, it is precisely the opposite that should happen - the editor should get used to us.”

(see <http://blog.vivekhaldar.com/post/31970017734>)

- “Easy things should be easy and hard things should be possible”

Larry Wall

On effectiveness (cont.)

Some other features of effective tools:

- mature
- stable
- customizable

“People talk about getting used to a new editor, but over time, it is precisely the opposite that should happen - the editor should get used to us.”

(see <http://blog.vivekhaldar.com/post/31970017734>)

- “Easy things should be easy and hard things should be possible”

Larry Wall

(the author of one of the most
ineffective tools in existence!)

- * Introduction
- * Facts and myths
- * How to choose
 - ** The factors influencing choice
 - ** What about \TeX without \TeX ?
- * Basic requirements
- * Typical errors
- * Example tools

The factors influencing choice

The choice of the editor is (should be?) influenced by a few factors. One might want to consider...

The factors influencing choice

The choice of the editor is (should be?) influenced by a few factors. One might want to consider...

- [] what it will be used for?

The factors influencing choice

The choice of the editor is (should be?) influenced by a few factors. One might want to consider...

- [] what it will be used for?
- [] how intensive the usage will be?

The factors influencing choice

The choice of the editor is (should be?) influenced by a few factors. One might want to consider...

- [] what it will be used for?
- [] how intensive the usage will be?
- [] will it be used for \LaTeX exclusively?

The factors influencing choice

The choice of the editor is (should be?) influenced by a few factors. One might want to consider...

- [] what it will be used for?
- [] how intensive the usage will be?
- [] will it be used for (A)TeX exclusively?
- [] whether the tool has a good community?

The factors influencing choice

The choice of the editor is (should be?) influenced by a few factors. One might want to consider...

- [] what it will be used for?
- [] how intensive the usage will be?
- [] will it be used for (A)TeX exclusively?
- [] whether the tool has a good community?
- [] is it *fun* to use?

The factors influencing choice

The choice of the editor is (should be?) influenced by a few factors. One might want to consider...

- what it will be used for?

Factors... : Usage patterns of text editors

One might naïvely think that a text editor is used to edit text.

Factors... : Usage patterns of text editors

One might naïvely think that a text editor is used to edit text.

Completely wrong.

Factors... : Usage patterns of text editors

One might naïvely think that a text editor is used to edit text.

Completely wrong.

A text editor might be used to

Factors... : Usage patterns of text editors

One might naïvely think that a text editor is used to edit text.

Completely wrong.

A text editor might be used to

- *write* texts,

Factors... : Usage patterns of text editors

One might naïvely think that a text editor is used to edit text.

Completely wrong.

A text editor might be used to

- *write* texts,
- *modify* texts,

Factors... : Usage patterns of text editors

One might naïvely think that a text editor is used to edit text.

Completely wrong.

A text editor might be used to

- *write* texts,
- *modify* texts,
- *read* texts,

Factors... : Usage patterns of text editors

One might naïvely think that a text editor is used to edit text.

Completely wrong.

A text editor might be used to

- *write* texts,
- *modify* texts,
- *read* texts,
- and a lot of other things.

Factors... : Usage patterns of text editors

One might naïvely think that a text editor is used to edit text.

Completely wrong.

A text editor might be used to

- *write* texts,
- *modify* texts,
- *read* texts,
- and a lot of other things.

Basically, every editor lets you write text quite conveniently...

Factors... : Usage patterns of text editors

One might naïvely think that a text editor is used to edit text.

Completely wrong.

A text editor might be used to

- *write* texts,
- *modify* texts,
- *read* texts,
- and a lot of other things.

Basically, every editor lets you write text quite conveniently...

...but only some of them support *modifying* it!

The factors influencing choice

The choice of the editor is (should be?) influenced by a few factors. One might want to consider...

- what it will be used for?
- how intensive the usage will be?

Factors... : Usage intensity

We will use a different tool for a one-shot job,
and a different one if we are going to need it every day
for the next several (dozen?) years.

Factors... : Usage intensity

We will use a different tool for a one-shot job,
and a different one if we are going to need it every day
for the next several (dozen?) years.

In the former case - and only then - “easy to use” tools
are suitable.

The factors influencing choice

The choice of the editor is (should be?) influenced by a few factors. One might want to consider...

- what it will be used for?
- how intensive the usage will be?
- will it be used for \LaTeX exclusively?

Factors... : Universality of the editor

If we predict that we are going to program (or even write in something different than \TeX) - there's no point in learning a tool which won't help us then.

Factors... : Universality of the editor

If we predict that we are going to program (or even write in something different than \TeX) - there's no point in learning a tool which won't help us then.

There are enough *all-purpose* editors which are really good at editing \LaTeX documents.

The factors influencing choice

The choice of the editor is (should be?) influenced by a few factors. One might want to consider...

- what it will be used for?
- how intensive the usage will be?
- will it be used for ~~(A)~~TeX exclusively?
- whether the tool has a good community?

Factors... : Community

Before we invest time and effort to master a new tool,
it may be wise to ask ourselves a question
whether we will be on our own in case of problems...

Factors... : Community

Before we invest time and effort to master a new tool, it may be wise to ask ourselves a question whether we will be on our own in case of problems...

- Is the tool a fad or a well-established one?

Factors... : Community

Before we invest time and effort to master a new tool, it may be wise to ask ourselves a question whether we will be on our own in case of problems...

- Is the tool a fad or a well-established one?
- Is it being developed?

Factors... : Community

Before we invest time and effort to master a new tool, it may be wise to ask ourselves a question whether we will be on our own in case of problems...

- Is the tool a fad or a well-established one?
- Is it being developed?
- Is there a place to ask when we have a problem (fora, mailing lists, StackOverflow...)?

The factors influencing choice

The choice of the editor is (should be?) influenced by a few factors. One might want to consider...

- [] what it will be used for?
- [] how intensive the usage will be?
- [] will it be used for (A)TeX exclusively?
- [] whether the tool has a good community?
- [X] is it *fun* to use?

Factors... : Fun factor

If we are going to spend hundreds of man-hours
working with some tool,

Factors... : Fun factor

If we are going to spend hundreds of man-hours working with some tool, it had better not be a torment to use.

What about \TeX without \TeX ?

It may be a good idea to consider using another tool, generating a \LaTeX file automatically, instead of \LaTeX itself.

What about \TeX without \TeX ?

It may be a good idea to consider using another tool, generating a \LaTeX file automatically, instead of \LaTeX itself.

There are quite a few such tools, including:

What about \TeX without \TeX ?

It may be a good idea to consider using another tool, generating a (\LaTeX) file automatically, instead of (\LaTeX) itself.

There are quite a few such tools, including:

- LyX,

What about \TeX without \TeX ?

It may be a good idea to consider using another tool, generating a (\LaTeX) file automatically, instead of (\LaTeX) itself.

There are quite a few such tools, including:

- LyX,
- pandoc,

What about \TeX without \TeX ?

It may be a good idea to consider using another tool, generating a $(\text{A})\TeX$ file automatically, instead of $(\text{A})\TeX$ itself.

There are quite a few such tools, including:

- LyX,
- pandoc,
- Org-mode.

TeX without TeX: LyX

“LyX is a document processor that encourages an approach to writing based on the structure of your documents (WYSIWYM) and not simply their appearance (WYSIWYG).”

TeX without TeX: LyX

“LyX is a document processor that encourages an approach to writing based on the structure of your documents (WYSIWYM) and not simply their appearance (WYSIWYG).”

In other words, LyX gives us:

TeX without TeX: LyX

“LyX is a document processor that encourages an approach to writing based on the structure of your documents (WYSIWYM) and not simply their appearance (WYSIWYG).”

In other words, LyX gives us:

- a simple (“mouse-driven”) UI,

TeX without TeX: LyX

“LyX is a document processor that encourages an approach to writing based on the structure of your documents (WYSIWYM) and not simply their appearance (WYSIWYG).”

In other words, LyX gives us:

- a simple (“mouse-driven”) UI,
- semantic tagging,

\TeX without \TeX : LyX

“LyX is a document processor that encourages an approach to writing based on the structure of your documents (WYSIWYM) and not simply their appearance (WYSIWYG).”

In other words, LyX gives us:

- a simple (“mouse-driven”) UI,
- semantic tagging,
- \TeX export.

TeX without TeX: pandoc

“If you need to convert files from one markup format into another, pandoc is your swiss-army knife.”

TeX without TeX: pandoc

“If you need to convert files from one markup format into another, pandoc is your swiss-army knife.”

In other words, pandoc gives us:

TeX without TeX: pandoc

“If you need to convert files from one markup format into another, pandoc is your swiss-army knife.”

In other words, pandoc gives us:

- conversion of Markdown, reST (attention Pythonistas!), MediaWiki (and a few other formats)...

\TeX without \TeX : pandoc

“If you need to convert files from one markup format into another, pandoc is your swiss-army knife.”

In other words, pandoc gives us:

- conversion of Markdown, reST (attention Pythonistas!), MediaWiki (and a few other formats)...
- ...to HTML, EPUB, \TeX , docx, odt (and a few other formats).

TeX without TeX: Org-mode

“Org-mode is for keeping notes, maintaining TODO lists, planning projects, and authoring documents with a fast and effective plain-text system.”

TeX without TeX: Org-mode

“Org-mode is for keeping notes, maintaining TODO lists, planning projects, and authoring documents with a fast and effective plain-text system.”

In other words, Org-mode gives us:

TeX without TeX: Org-mode

“Org-mode is for keeping notes, maintaining TODO lists, planning projects, and authoring documents with a fast and effective plain-text system.”

In other words, Org-mode gives us:

- conversion of plain text files to ~~TeX~~, HTML, odt (and a few other formats),

TeX without TeX: Org-mode

“Org-mode is for keeping notes, maintaining TODO lists, planning projects, and authoring documents with a fast and effective plain-text system.”

In other words, Org-mode gives us:

- conversion of plain text files to ~~TeX~~TeX, HTML, odt (and a few other formats),
- an environment for note-taking and time management (and a few other things),

TeX without TeX: Org-mode

“Org-mode is for keeping notes, maintaining TODO lists, planning projects, and authoring documents with a fast and effective plain-text system.”

In other words, Org-mode gives us:

- conversion of plain text files to ~~TeX~~TeX, HTML, odt (and a few other formats),
- an environment for note-taking and time management (and a few other things),
- whose basics can be learned within a few minutes.

TEX without TEX: Org-mode (cd.)

“At its core, Org-mode is a simple outliner for note-taking and list management. You can learn the basics for using it in five minutes. This may be all you need, and Org-mode will not impose more complex features on you.”

(from the previous version of the <http://orgmode.org> site)

TEX without TEX: Org-mode (cd.)

“At its core, Org-mode is a simple outliner for note-taking and list management. You can learn the basics for using it in five minutes. This may be all you need, and Org-mode will not impose more complex features on you.”

(from the previous version of the <http://orgmode.org> site)

“That’s right... because org-mode is just a collection of lisp running in an editor. It cannot impose more complex features on you. The genius of org-mode is that you will eventually impose more complex features on yourself.”

(from <http://mph.puddingbowl.org>)

- * Introduction
- * Facts and myths
- * How to choose
- * **Basic requirements**
- ** Absolutely indispensable features**
- ** Bells and whistles**
- * Typical errors
- * Example tools

Absolutely indispensable features

Remark

As usually, the list below (and the next one, too) is my subjective choice. YMMV.

Absolutely indispensable features

Remark

As usually, the list below (and the next one, too) is my subjective choice. YMMV.

- free (as in free beer)

Absolutely indispensable features

Remark

As usually, the list below (and the next one, too) is my subjective choice. YMMV.

- free (as in free beer)
- able to run \TeX and a viewer

Absolutely indispensable features

Remark

As usually, the list below (and the next one, too) is my subjective choice. YMMV.

- free (as in free beer)
- able to run \TeX and a viewer
- syntax highlighting

Absolutely indispensable features

Remark

As usually, the list below (and the next one, too) is my subjective choice. YMMV.

- free (as in free beer)
- able to run \TeX and a viewer
- syntax highlighting
- easy navigation

Absolutely indispensable features

Remark

As usually, the list below (and the next one, too) is my subjective choice. YMMV.

- free (as in free beer)
- able to run \TeX and a viewer
- syntax highlighting
- easy navigation
- autocompletion of commands, environments and labels

Absolutely indispensable features

Remark

As usually, the list below (and the next one, too) is my subjective choice. YMMV.

- free (as in free beer)
- able to run \TeX and a viewer
- syntax highlighting
- easy navigation
- autocompletion of commands, environments and labels
- nice icon

Bells and whistles

- getting to the spot where an error occurred

Bells and whistles

- getting to the spot where an error occurred
- help with entering commands, environments and symbols

Bells and whistles

- getting to the spot where an error occurred
- help with entering commands, environments and symbols
- partial compilation

Bells and whistles

- getting to the spot where an error occurred
- help with entering commands, environments and symbols
- partial compilation
- multi-file projects

Bells and whistles

- getting to the spot where an error occurred
- help with entering commands, environments and symbols
- partial compilation
- multi-file projects
- *snippets* and *templates*

Bells and whistles

- getting to the spot where an error occurred
- help with entering commands, environments and symbols
- partial compilation
- multi-file projects
- *snippets* and *templates*
- spellchecking

Bells and whistles

- getting to the spot where an error occurred
- help with entering commands, environments and symbols
- partial compilation
- multi-file projects
- *snippets* and *templates*
- spellchecking
- commenting out parts of the document

Bells and whistles

- getting to the spot where an error occurred
- help with entering commands, environments and symbols
- partial compilation
- multi-file projects
- *snippets* and *templates*
- spellchecking
- commenting out parts of the document
- support for VCSs

Bells and whistles

- getting to the spot where an error occurred
- help with entering commands, environments and symbols
- partial compilation
- multi-file projects
- *snippets* and *templates*
- spellchecking
- commenting out parts of the document
- support for VCSs
- highlighting and insertion of braces

Bells and whistles

- getting to the spot where an error occurred
- help with entering commands, environments and symbols
- partial compilation
- multi-file projects
- *snippets* and *templates*
- spellchecking
- commenting out parts of the document
- support for VCSs
- highlighting and insertion of braces
- syncTeX/follow focus

Bells and whistles

- getting to the spot where an error occurred
- help with entering commands, environments and symbols
- partial compilation
- multi-file projects
- *snippets* and *templates*
- spellchecking
- commenting out parts of the document
- support for VCSs
- highlighting and insertion of braces
- syncTeX/follow focus
- deletion of temporary files

Bells and whistles

- getting to the spot where an error occurred
- help with entering commands, environments and symbols
- partial compilation
- multi-file projects
- *snippets* and *templates*
- spellchecking
- commenting out parts of the document
- support for VCSs
- highlighting and insertion of braces
- syncTeX/follow focus
- deletion of temporary files
- code indenting

- * Introduction
- * Facts and myths
- * How to choose
- * Basic requirements
- * Typical errors
 - ** Errors connected with using \LaTeX
 - ** Errors connected with using a computer
- * Example tools

Errors connected with using \LaTeX

\LaTeX users - especially beginners, but not only - tend to commit some typical errors. Here is a (subjective) list of a few of them, connected with choosing and using the editor.

Errors connected with using (A)T_EX

(A)T_EX users - especially beginners, but not only - tend to commit some typical errors. Here is a (subjective) list of a few of them, connected with choosing and using the editor.

- using too primitive an editor,

Errors connected with using (A)T_EX

(A)T_EX users - especially beginners, but not only - tend to commit some typical errors. Here is a (subjective) list of a few of them, connected with choosing and using the editor.

- using too primitive an editor,
- using strange notation for diacritics,

Errors connected with using (A)T_EX

(A)T_EX users - especially beginners, but not only - tend to commit some typical errors. Here is a (subjective) list of a few of them, connected with choosing and using the editor.

- using too primitive an editor,
- using strange notation for diacritics,
- using dvi files

Errors connected with using (A)T_EX

(A)T_EX users - especially beginners, but not only - tend to commit some typical errors. Here is a (subjective) list of a few of them, connected with choosing and using the editor.

- using too primitive an editor,
- using strange notation for diacritics,
- using dvi files (unless you have good reasons!),

Errors connected with using (A)T_EX

(A)T_EX users - especially beginners, but not only - tend to commit some typical errors. Here is a (subjective) list of a few of them, connected with choosing and using the editor.

- using too primitive an editor,
- using strange notation for diacritics,
- using dvi files (unless you have good reasons!),
- using Adobe Reader

Errors connected with using \LaTeX

\LaTeX users - especially beginners, but not only - tend to commit some typical errors. Here is a (subjective) list of a few of them, connected with choosing and using the editor.

- using too primitive an editor,
- using strange notation for diacritics,
- using dvi files (unless you have good reasons!),
- using Adobe Reader (with the default configuration!).

Errors connected with using a computer

(A) \TeX users - especially beginners, but not only - tend to commit some typical errors. Here is a (subjective) list of a few of them, connected with computer usage in general.

Errors connected with using a computer

(A) \TeX users - especially beginners, but not only - tend to commit some typical errors. Here is a (subjective) list of a few of them, connected with computer usage in general.

- using spaces and funny characters in filenames,

Errors connected with using a computer

(A) \TeX users - especially beginners, but not only - tend to commit some typical errors. Here is a (subjective) list of a few of them, connected with computer usage in general.

- using spaces and funny characters in filenames,
- not using a coherent system of naming files and placing them in the folder/directory structure,

Errors connected with using a computer

(A) \TeX users - especially beginners, but not only - tend to commit some typical errors. Here is a (subjective) list of a few of them, connected with computer usage in general.

- using spaces and funny characters in filenames,
- not using a coherent system of naming files and placing them in the folder/directory structure,
- not using VCSs (especially with co-authored documents),

Errors connected with using a computer

(A)T_EX users - especially beginners, but not only - tend to commit some typical errors. Here is a (subjective) list of a few of them, connected with computer usage in general.

- using spaces and funny characters in filenames,
- not using a coherent system of naming files and placing them in the folder/directory structure,
- not using VCSs (especially with co-authored documents),
- not making backups,

Errors connected with using a computer

(A) \TeX users - especially beginners, but not only - tend to commit some typical errors. Here is a (subjective) list of a few of them, connected with computer usage in general.

- using spaces and funny characters in filenames,
- not using a coherent system of naming files and placing them in the folder/directory structure,
- not using VCSs (especially with co-authored documents),
- not making backups,
- not learning to touchtype.

- * Introduction
- * Facts and myths
- * How to choose
- * Basic requirements
- * Typical errors
- * Example tools
 - ** A short survey of tools
 - ** Case study: \TeX works
 - ** Case study: GNU Emacs

A short survey of tools

Here is a list of popular (hence maybe good) editors supporting editing of \LaTeX files.

A short survey of tools

Here is a list of popular (hence maybe good) editors supporting editing of \LaTeX files.

- Emacs
- Vim
- Kile
- WinEDT (unfortunately, it costs real \$\$\$)
- LEd (unfortunately, no longer developed)
- LyX
- TeXmaker
- TeXnicCenter
- Gummi

see <http://tex.stackexchange.com/questions/339>

Case study: T_EXworks

T_EXworks is a relatively young editor, aimed at editing *exclusively* (A)T_EX files. Its advantages are (among others): integrated pdf viewer with syncTeX, elegant UI, extensibility (through plugins), document templates. It is worth mentioning that T_EXworks is installed by default with the two most popular T_EX distros and that it is supported by the T_EX Users Group.

Case study: GNU Emacs

GNU Emacs is one of the oldest still used text editors. It is an all-purpose editor, used mainly (but not only) for programming. Its advantages are (among others): unmatched customizability, feature-richness (there are lots of both built-in features and add-ons), maturity and an active community.

Case study: GNU Emacs

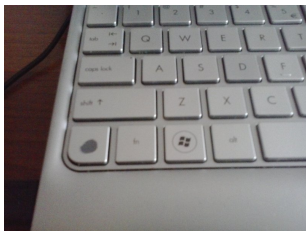
GNU Emacs is one of the oldest still used text editors. It is an all-purpose editor, used mainly (but not only) for programming. Its advantages are (among others): unmatched customizability, feature-richness (there are lots of both built-in features and add-ons), maturity and an active community.

Unfortunately, Emacs does have disadvantages...

Case study: GNU Emacs

GNU Emacs is one of the oldest still used text editors. It is an all-purpose editor, used mainly (but not only) for programming. Its advantages are (among others): unmatched customizability, feature-richness (there are lots of both built-in features and add-ons), maturity and an active community.

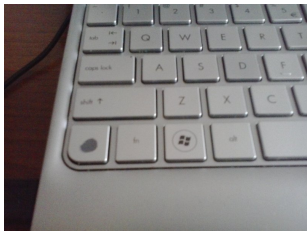
Unfortunately, Emacs does have disadvantages...



Case study: GNU Emacs

GNU Emacs is one of the oldest still used text editors. It is an all-purpose editor, used mainly (but not only) for programming. Its advantages are (among others): unmatched customizability, feature-richness (there are lots of both built-in features and add-ons), maturity and an active community.

Unfortunately, Emacs does have disadvantages...



Thank you
for your attention!