

org-export.org

George De Bruin

2011-07-03 Sun

Contents

1	Org-Mode Rich Export Markup	2
2	Document title is set using <code>#+TITLE:</code>	2
3	Options	2
3.1	Headings: <code>H:<n></code> to set the level that is used for sections versus lists.	2
3.2	Table of contents: <code>toc:<n></code>	2
4	Paragraph Styles and Line Breaks	3
4.1	Quotes: marked with <code>#+BEGIN_QUOTE</code> and <code>#+END_QUOTE</code>	3
4.2	Verses: marked with <code>#+BEGIN_VERSE</code> and <code>#+END_VERSE</code>	3
4.3	Centering: marked with <code>#+BEGIN_CENTER</code> and <code>#+END_CENTER</code>	3
4.4	Literal / Examples: marked with <code>#+BEGIN_EXAMPLE</code> and <code>#+END_EXAMPLE</code>	3
4.5	Source Code: marked with <code>#+BEGIN_SRC <lang></code> and <code>#+END_SRC</code>	4
4.6	Including Files: marked with <code>#+INCLUDE</code>	4
4.6.1	First parameter is the file	6
4.6.2	Second Parameter is markup type	6
4.6.3	Third parameter is for src style - indicates syntax of the language	6
5	Character Formatting	6
5.1	Emphasis / Bold is set using <code>*</code> 's	6
5.2	Italics are set using <code>/</code> 's	6
5.3	Underline is set using: <code>_</code> 's	6
5.4	Literal text is set using <code>~</code> 's	6

5.5	Strikethrough is set using +’s (highly discouraged)	6
6	Tables and Images	7
6.1	Tables can have a caption and a label (for cross referencing) .	7
6.2	Images can also have a caption and label	7
6.3	Embedded L ^A T _E X	7
7	Exporting	7
7.1	Special Options (C-c C-e t)	7
7.2	Export Dispatcher (C-c C-e)	8
8	Publishing	8
9	Settings	9

1 Org-Mode Rich Export Markup

2 Document title is set using #+TITLE:

3 Options

There are a number of options that can be set using the #+OPTIONS: directive, including:

3.1 Headings: H:<n> to set the level that is used for sections versus lists.

The default is H:2. If you want a deeper level of outline, you might set H:4.

3.2 Table of contents: toc:<n>

The table of contents is normally inserted after the title page. Setting toc:nil will instruct org-mode to not have a table of contents. toc:2 will tell org-mode to only generate a two-level table of contents.

4 Paragraph Styles and Line Breaks

- Paragraphs are normally separated by a blank line. Using `\\` will force a line break.
- There are three block paragraph styles that can be added to your documents to differentiate portions of the text.

4.1 Quotes: marked with `#+BEGIN_QUOTE` and `#+END_QUOTE`

Before God we are all equally wise - and equally foolish.
– Albert Einstein

4.2 Verses: marked with `#+BEGIN_VERSE` and `#+END_VERSE`

Join us now and share the software;
You'll be free, hackers, you'll be free.
Join us now and share the software;
You'll be free, hackers, you'll be free.

4.3 Centering: marked with `#+BEGIN_CENTER` and `#+END_CENTER`

Before God we are all equally wise - and equally foolish.
– Albert Einstein

4.4 Literal / Examples: marked with `#+BEGIN_EXAMPLE` and `#+END_EXAMPLE`

```
(setq demo-dir (file-name-directory (or load-file-name (buffer-file-name))))

(setq org-directory demo-dir)

;; Enable org-babel languages
(org-babel-do-load-languages
 'org-babel-load-languages
 '((python . t)
  (emacs-lisp . t)
  (ditaa . t)
  (sh . t)))

(global-set-key (kbd "C-c a") 'org-agenda)
```

4.5 Source Code: marked with `#+BEGIN_SRC <lang>` and `#+END_SRC`

```
(setq demo-dir (file-name-directory (or load-file-name (buffer-file-name))))

(setq org-directory demo-dir)

;; Enable org-babel languages
(org-babel-do-load-languages
 'org-babel-load-languages
 '((python . t)
  (emacs-lisp . t)
  (ditaa . t)
  (sh . t)))

(global-set-key (kbd "C-c a") 'org-agenda)
```

4.6 Including Files: marked with `#+INCLUDE`

```
;; Configure the loadpath
(setq load-path
      (append (list nil "/user/bil/emacs"
                    "/usr/local/lisplib"
                    "~/emacs"
                    "~/.emacs.d")
              load-path))

;; First all the config "libraries" that I want
;;(load-file "~/emacs.d/gdb_orgcfg.el")
(load-file "~/emacs.d/gdb_nxmlcfg.el")
(load-file "~/emacs.d/gdb_emmscfg.el")
(load-file "~/emacs.d/gdb_misccfg.el")
(load-file "~/emacs.d/gdb_bbdb.el")
(load-file "~/emacs.d/gdb_wl.el")

;; ---
;; emacs customize settings
;; ---
(custom-set-variables
```

```

;; custom-set-variables was added by Custom.
;; If you edit it by hand, you could mess it up, so be careful.
;; Your init file should contain only one such instance.
;; If there is more than one, they won't work right.
'(column-number-mode t)
'(fringe-mode 0 nil (fringe))
'(inhibit-startup-screen t)
'(scroll-bar-mode (quote right))
'(show-paren-mode t)
'(speedbar-frame-parameters (quote ((minibuffer) (width . 20) (border-width . 0) (men
'(weblogger-config-alist (quote (("CerebralRift" "http://linuxtweakers.org/xmlrpc.php

(custom-set-faces
  ;; custom-set-faces was added by Custom.
  ;; If you edit it by hand, you could mess it up, so be careful.
  ;; Your init file should contain only one such instance.
  ;; If there is more than one, they won't work right.
  '(default ((t (:inherit nil :stipple nil :background "#102e4e" :foreground "#eeeeee"
  '(background "blue")
  '(font-lock-builtin-face (((class color) (background dark)) (:foreground "Turquoise"
  '(font-lock-comment-face ((t (:foreground "MediumAquamarine"))))
  '(font-lock-constant-face (((class color) (background dark)) (:bold t :foreground "D
  '(font-lock-doc-string-face ((t (:foreground "green2"))))
  '(font-lock-function-name-face ((t (:foreground "SkyBlue"))))
  '(font-lock-keyword-face ((t (:bold t :foreground "CornflowerBlue"))))
  '(font-lock-preprocessor-face ((t (:italic nil :foreground "CornFlowerBlue"))))
  '(font-lock-reference-face ((t (:foreground "DodgerBlue"))))
  '(font-lock-string-face ((t (:foreground "LimeGreen"))))
  '(font-lock-type-face ((t (:foreground "#9290ff"))))
  '(font-lock-variable-name-face ((t (:foreground "PaleGreen"))))
  '(font-lock-warning-face (((class color) (background dark)) (:foreground "yellow" :b
  '(highlight ((t (:foreground "Black" :background "CornflowerBlue"))))
  '(list-mode-item-selected ((t (:background "gold"))))
  '(makefile-space-face ((t (:background "wheat"))))
  '(mode-line ((t (:background "Navy"))))
  '(paren-match ((t (:background "darkseagreen4"))))
  '(region ((t (:background "DarkSlateBlue"))))
  '(show-paren-match ((t (:foreground "black" :background "wheat"))))
  '(show-paren-mismatch (((class color)) (:foreground "white" :background "red"))))
  '(speedbar-button-face (((class color) (background dark)) (:foreground "green4"))))

```

```
'(speedbar-directory-face (((class color) (background dark)) (:foreground "khaki")))
'(speedbar-file-face (((class color) (background dark)) (:foreground "cyan"))))
'(speedbar-tag-face (((class color) (background dark)) (:foreground "Springgreen")))
'(vhdl-speedbar-architecture-selected-face (((class color) (background dark)) (:underline
'(vhdl-speedbar-entity-face (((class color) (background dark)) (:foreground "darkGre
'(vhdl-speedbar-entity-selected-face (((class color) (background dark)) (:underline
'(vhdl-speedbar-package-face (((class color) (background dark)) (:foreground "black"
'(vhdl-speedbar-package-selected-face (((class color) (background dark)) (:underline
'(widget-field (((class grayscale color) (background light)) (:background "DarkBlue"
(put 'downcase-region 'disabled nil)
```

4.6.1 First parameter is the file

4.6.2 Second Parameter is markup type

4.6.3 Third parameter is for src style - indicates syntax of the language

5 Character Formatting

5.1 Emphasis / Bold is set using *****'s

5.2 Italics are set using */*'s

5.3 Underline is set using: _'s

5.4 Literal text is set using `~`'s

5.5 Strikethrough is set using ~~+~~'s (highly discouraged)

Here's an example:

This is **bold** text. This is *italicized*. This is underlined. This is `literal` text. And, the highly discouraged ~~strikethrough~~.

Table 1: Example Table One

one	two	three
a	b	c
d	e	f



Figure 1: The Org-Mode Logo

6 Tables and Images

6.1 Tables can have a caption and a label (for cross referencing)

6.2 Images can also have a caption and label

6.3 Embedded \LaTeX

Angles are written as Greek letters α , β and γ . The mass of the sun is $M_{\text{sun}} = 1.989 \times 10^{30}$ kg. The radius of the sun is $R_{\text{sun}} = 6.96 \times 10^8$ m. If $a^2 = b$ and $b = 2$, then the solution must be either $a = +\sqrt{2}$ or $a = -\sqrt{2}$.

$$x = \sqrt{b} \tag{1}$$

7 Exporting

7.1 Special Options (C-c C-e t)

```
#+TITLE:      org-export.org
#+AUTHOR:     George De Bruin
#+EMAIL:      gdb@borges
```

```

#+DATE:      2011-07-03 Sun
#+DESCRIPTION:
#+KEYWORDS:
#+LANGUAGE:  en
#+OPTIONS:   H:3 num:t toc:t \n:nil @:t ::t |:t ^:t -:t f:t *:t <:t
#+OPTIONS:   TeX:t LaTeX:t skip:nil d:nil todo:t pri:nil tags:not-in-toc
#+INFOJS_OPT: view:nil toc:1 ltoc:t mouse:underline buttons:0 path:http://orgmode.org/
#+EXPORT_SELECT_TAGS: export
#+EXPORT_EXCLUDE_TAGS: noexport
#+LINK_UP:
#+LINK_HOME:
#+XSLT:

```

- We’ve talked about TITLE already.
- AUTHOR, EMAIL, DATE, DESCRIPTION, and KEYWORDS should be pretty standard.
- LANGUAGE follows traditional language options.
- The OPTIONS lines allow great flexibility for formatting the document.
- The rest of the lines are specific to the target of the export (like HTML or docbook).

7.2 Export Dispatcher (C-c C-e)

This is the menu with most of the commands for exporting. Allows specifying the character set to use, as well as the target for exporting.

8 Publishing

Org-Mode has the ability to set up a publishing system to export a series of linked files. The idea is to take all your org files (or a subset of them) and convert them to an appropriate format, and place the output in a specified location.

The obvious uses are:

- Using org to publish a blog
- Publishing all of your project information in a central place

- Publishing an Agenda

And, there are more uses for this feature.

The publishing feature is quite flexible, allowing you to do things like:

- Select the set of files to publish
- Set a custom CSS file for formatting the output
- Set the table of contents
- Etc.

I won't talk more about this feature. It could take a few more hours to explain all it's features.

9 Settings