

Emacs Org-mode

Bastien Guerry – bzg@gnu.org

August 27th, GNU Hackers meeting

Outline

- 1 Introduction: what is org-mode?
- 2 Incremental (live) features walkthrough
- 3 A few useful Org-mode extensions
- 4 Org-mode's documentation
- 5 Org-mode's past and future
- 6 Quick sneak into org-element.el (by Nicolas Goaziou)
- 7 Org-mode goodies

What is org-mode?

"Org-mode is for keeping notes, maintaining ToDo lists, doing project planning, and authoring with a fast and effective plain-text system."

- **A simple TODO list manager**
 - also a full-fledged organizer
- **An intuitive note taking environment**
 - also a full-fledged authoring environment

Core org-mode facts

Org-mode core facts:

- <http://orgmode.org>
- Created in **2003** by Carsten Dominik
- GNU GPLv3 or any later version
- Early integration into GNU Emacs
- Current Stable and GNU Emacs (trunk) version is **7.7**
- More than **8500** commits* since January 2008

A superactive community of users and contributors:

- **120** developers posted at least 1 commit
- **25** developers posted at least 10 commits

Org-mode design principles

Background:

- Emacs (GNU Emacs, XEmacs, Aquamacs)
- Plain text (allow flexible processing)
- Derive from `outline-mode`

Design principles:

- "Don't impose unwanted complexity"
- "95% of the features added after core implementation"
- Tries to adapt to everyone's need

Theorem (No learning curve)

You set the learning curve depending on your needs.

Manipulating structure

- 1 A simple outline file
- 2 Changing the structure of the file
- 3 Easy list manipulation
- 4 Adding checkboxes to lists
- 5 Adding links
- 6 ...

Todo and schedule tasks

- 1 A simple TODO
- 2 Updating TODO keywords
- 3 Adding logs
- 4 A simple agenda view of the TODO keywords
- 5 A TODO with a SCHEDULE timestamp
- 6 Manipulating timestamps
- 7 A simple agenda view of the SCHEDULED items
- 8 Manipulating timestamps from the agenda view
- 9 ...

Tags, Column view, clock in...

- 1 Adding tags
- 2 An agenda view of specific tags
- 3 The column view in an Org file
- 4 The column view in the agenda buffer
- 5 Adding the "effort" property
- 6 Clocking in tasks
- 7 Inserting the clocktable
- 8 Viewing a clock summary in the column view
- 9 ...

Capturing, archiving, exporting, publishing

- 1 Capturing a task (e.g. from dired)
- 2 Archiving a task (to various locations)
- 3 Exporting to HTML, L^AT_EX (pdf), ODT, etc.
- 4 Publishing a project
- 5 ...

Tables, spreadsheets, literate programming

- 1 Manipulating tables
- 2 Tables as spreadsheets
- 3 Literate programming with Babel
- 4 ...

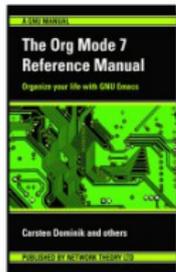
A few useful extensions

- `org-mobile.el` (for MobileOrg)
- `org-attach.el` by John Wiegley
- `org-beamer.el` by Carsten Dominik
- `org-crypt.el` by John Wiegley
- `org-habit.el` by John Wiegley
- `org-taskjuggler.el` by Christian Egli
- `org-depend.el` by Carsten Dominik
- ...

Org printed manual by Network-Theory.co.uk (Brian Gough)

NETWORK THEORY LTD - PUBLISHING FREE SOFTWARE MANUALS

[\[Home\]](#) [\[Catalogue\]](#) [\[New Titles\]](#) [\[News\]](#) [\[Site Map\]](#) [\[About\]](#) [\[Contact\]](#)



The Org Mode 7 Reference Manual (for Org version 7.3)

by Carsten Dominik and others

Paperback (6"x9"), 282 pages - New release! (now shipping - December 2010)

Retail Price: \$14.95 (£9.95 in UK)

ISBN-13: 978-1-906966-08-9

[The Book Depository - UK/Worldwide \(free shipping\)](#)

About the Book

This manual is a printed edition of the official Org reference documentation from the Org 7.3 distribution.

Org mode is a powerful system for organizing projects, tasks and notes in the Emacs editor. Major features include fast outline and table editing, TODO lists, agendas, scheduling and deadlines, cross-references and hyperlinks to arbitrary resources, as well as customisable document publishing in PDF and HTML from simple plain text markup.

Org mode stores all data in human-readable plain text files, ensuring complete portability, simple integration with other text processing tools and support for revision-tracking and synchronization using any version control system.

Org mode is free software under the GNU General Public License and can be used in Emacs on all major operating systems.

You can examine the contents of the printed book here



Org tutorials

{Back to Worg's index}

General Introductions to Org-mode

- [Google Tech Talk about Org-mode](#), a 45 minute talk given by Carsten Dominik during a visit to the University of Cologne.
- [Organizing a Scientist's Life and Work](#), a 45 minute talk given by Carsten Dominik during a visit to the University of Cologne.
- [On FLOSS Weekly Episode 136](#) (Audio+Video download available) Randal Schwartz interview with Carsten Dominik.
- [Getting Organized with Emacs Org-mode](#) by Abhijeet Chavan, an article originally published on the EmacsWiki.
- [The OrgMode tutorial](#) by David O'Toole, covering the basics of TODO lists and the agenda.
- [A 25 minute screencast](#) by Scott Jaderholm, demonstrating the basic features of Org-mode.
- [Outlining Your Notes with Org](#) by Sacha Chua covers outlining, searching and internal linking.
- [Using Emacs org-mode for GTD](#) by Charles Cave contains an introduction into Org-mode, an extensive document (see below), but here we still link to the earlier version because of its length.
- [Emacs Org-mode](#), video version of a 2-hour talk by Russell Adams at the Houston Linux Users Group.
- [In German language: Ordnung ins Chaos mit Org-Mode](#), an excellent introduction into Org-mode, published in the electronic journal Freies Magazin.
- [In French: A general introduction on Org](#) for the "lifehacking" group, by Olivier Berger.

Power users describe their setup

In the following documents, the authors describe the details of their setup, along with the workflows they use to get organized with Org-mode. While as a

Table of Contents

- General introductions to Org-mode
- Power users describe their setup
- Tutorials on specific features
 - Lists
 - Tables, Spreadsheet, Plotting
 - Dates and Times
 - Tags, Properties, and Column View
 - Remember
 - Capture
 - Agenda and search
 - Customization
 - Export and Publishing
- Special tasks
 - Natural Project Planning
 - Tracking Habits
 - Using version Control with Your org files
 - How to use jsMath with org-mode
 - Creating a Blog with Jekyll and org
 - Creating Beamer presentations
 - Creating Non-Beamer presentations
 - Keeping up with your team's tasks

Worg's facts

- Started in november 2007
- GNU Emacs + Org + git + server-side publication
- >2300 commits since then
- Text and code released under GNU GPLv3 or later
- A very long user-contributed FAQ
- Many useful tutorials, hacks, code snippets
- Maintained by Matt Lundin since January 2011

Important past changes

- November 2007: Worg setup
- January 2008: moving org.el to git
- March-April 2008: splitting org.el (v6.01)
- Using Worg/org-issues.org for tracking issues (maintained by David Maus and Nicolas Goaziou)
- New capture mechanism (a rewrite of the "remember" mechanism)
- Using patchwork for tracking patches (hosted by John Wiegley)
- Org Babel by Eric Schulte and Dan Davison in core since 7.0
- <http://orgmode.org> and the org-mode.git repository moved to a dedicated server, maintained by Jason Dunsmore
- ODT exporter by Jambunathan K. in contrib/ since Org 7.6

Important future improvements

- org-element.el
 - A *cleverer* recursive parser and a *cleaner* rendering engine.
 - A structured representation: useful when talking with organizers' APIs like redmine, bugzilla, rememberthemilk.com, etc.
- Integration of the ODT exporter into Org's core
- A cache mechanism?
A cached representation of Org files would:
 - Allow to reuse a cached representation of Org files when exporting/publishing these files to a new formats.
 - Allow to easily extract specific information from the cached files (e.g. a list of scheduled items for each on the next 12 months.)
- Tags grouping
 - Allow `#+TAGS: color: blue white red.`
- Loop over regions when scheduling, archiving, etc.

`#+TITLE: Hello World!`

`* Hello world!`

`I said it.`

```
(defun org-element-parse-region
  (beg end item structure acc)
  "Recursively parse buffer between BEG and END [...]"
  (save-restriction
    (narrow-to-region beg end)
    (goto-char beg)
    (org-skip-whitespace)
    (beginning-of-line)
    (if (eobp)
        (nreverse acc)
        (let ((element (org-element-at-point item structure)))
          (org-element-parse-region
            (org-element-get-property :end element)
            end item structure
            (cons (org-element-parse-element
                  element acc structure) acc)))))))
```

org-element.el: parsed file

```
(document
 (keyword
  (:name "TITLE" :value "Hello World!"
         :begin 1 :end 24
         :post-blank 1))
 (headline
  (:name "Hello world!" :begin 24 :end 51 :pre-blank 1
         :contents-begin 40 :contents-end 51 :level 1
         :priority nil :todo-keyword nil :todo-type nil
         :properties
         (("FILE" . "/home/guerry/test.org")
          ("BLOCKED" . "")
          ("CATEGORY" . "test"))))
 (paragraph
  (:begin 40 :end 51 :contents-begin 40
   :contents-end 51 :post-blank 0)
  "I said it.)))
```

Org T-Shirts for everyone!



Figure: Org-mode T-Shirt for everyone

Org T-Shirts – for you?



Figure: "I was on Dad and Mum Org-mode file" (Jason Dunsmore)



The Orgfather



Figure: Getting advice from the "Org Father" at FOSDEM