

Emacs and Org-Mode

George De Bruin (License: CC BY-SA)

July 5th, 2011

Outline

1 Introduction

2 Emacs

3 Org-Mode

4 Final Notes

5 Resources

6 Q & A

Objectives

- Proselytizing with the objective of indoctrinating **you** to *The Church of Emacs*
- Proselytizing for Org-Mode as a stepping stone for conversion to *The Church of Emacs*
- Last, but not least: advancing my plans to take over the world!!!

... But seriously

- Emacs isn't for everyone, but just dismissing it without a good reason is a bad idea.
- Org-Mode may not be for everyone, but it has so many uses you may want it.
- Taking over the world isn't necessary: fighting for free software and personal rights is sufficient.

Why Choose Emacs?

- Free Software vs. Open Source: Free Software doesn't compromise for expedience of acceptance
- One of the first and best known Free Software applications: development started in the 1970's. Emacs is still actively maintained and developed.
- Stability: Emacs has established its core so well that stability is almost a given. Not that there can't be bugs, but you rarely see Emacs turn up in vulnerability lists
- The Power of Plain Text: the most accessible format for storing information is text. Plain text has been around for 60+ years
- More than just a text editor

Emacs Philosophy

Note: These are my opinions only. I don't claim to have insight into the Emacs development team, or any of the Emacs users.

- A Few Jokes

- Emacs: great operating system with a crappy editor
- Eight Megs And Constantly Swapping
- Emacs Manuals Are Cryptic and Surreal
- Erasing Minds Allows Complete Submission

- Emacs View

- Text is everywhere: web pages, source code, documentation, metadata (tags), databases, spreadsheets...
- Doing one thing and doing it better
- Building on a platform

Emacs Concepts

- Editing: text is edited in buffers, then written to disk
- Modes: in Emacs a mode is an enhancement to the editing environment. Typical enhancements: auto-indenting, colorization, enhanced movement, etc.
- Enhancements: most written in LISP. Emacs is a LISP “machine” in it’s own right

Emacs Enhancements

- Language Modes: XML, C, C++, Shell Scripting, PHP, Java, TCL, HTML, Lua, and Ascii Doc, T_EX, L^AT_EX, Groff to name a few
- Communication: GNUS, Rmail, Wanderlust, MH-E, Mew, VM, ERC, weblogger, etc.
- Games: Go, Hangman, ChessMode, MineSweep, Sudoku, SokobanMode, etc.
- Editing: Yasnippets, StateRestore, Remember, FlySpell, Org-Mode
- Other: Calendar, EMMS, SuperCite, BBDB (aka the Insidious Big Brother Database)

What is Org-Mode

- The motto: Your Life In Plain Text
- Structured Editor: aka outline editor
- **TODO** / Task Lists
- Project Planning
- Programming Enhancement
- Publishing Platform
- The ubiquitous: and more...

Why Use Org-Mode

- Brings organization and structure to your work
- Integrates into your work environment
- Extremely flexible and extensible
- Always accessible: even without Emacs

Why To Not Use Org-Mode

- Un-indexed plain text files can cause performance issues
- Specific "syntax" can break
- Not (easily) used in multiple user scenarios ¹
- Emacs does have a learning curve
- By extension, learning all the ins and outs of org-mode is tied to Emacs knowledge ²

¹You can use version control (like git or subversion), or possibly an extension called Worg. However, these topics are beyond the scope of this presentation.

²OTOH - all tools have learning curves, and the advantages of Emacs and org-mode may outweigh the learning curve.

Document Structure

- Outlines: Org is based on Outline mode
- Headlines: How to typeset Org tree headlines
- Visibility cycling: Showing and hiding - greatly simplified
- Motion: Navigating the Outline Tree
- Structure editing: Changing sequence and level of headlines
- Sparse trees: Matches embedded in context
- Plain lists: Additional structure within an entry
- Footnotes: How footnotes are defined in Org's syntax
- Tables: How to format a table

Hyperlinks

- Link format: How links in Org are formatted
- Internal links: Links to other places in the current file
- External links: URL-like links to the world
- Handling links: Creating, inserting and following
- Targeted links: Point at a location in a file

TODO Items

- Using TODO states: Setting and switching states
- Multi-state workflows: More than just on/off
- Progress logging: Dates and notes for progress
- Priorities: Some things are more important than others
- Breaking down tasks: Splitting a task into manageable pieces
- Checkboxes: Tick-off lists
- Closing items: When was this entry marked DONE?
- Tracking TODO state changes: When did the status change?

Dates and Times

- Timestamps: Assigning a time to a tree entry
- Creating timestamps: Commands which insert timestamps
- Deadlines and scheduling: Planning your work
- Clocking work time: Tracking how long you spend on a task

Tags and Properties

- Tag inheritance: Tags use the tree structure of the outline
- Setting tags: How to assign tags to a headline
- Tag searches: Searching for combinations of tags
- Properties: How to add properties

Capture - Refile - Archive

- Setting up a capture location: Where notes will be stored
- Capture templates: Define the outline of different note types
- Using capture: Commands to invoke and terminate capture
- Refiling notes: Moving a tree from one place to another
- Archiving: What to do with finished projects

Agenda Views

- Agenda files: Files being searched for agenda information
- Agenda dispatcher: Keyboard access to agenda views
- Built-in agenda views: What is available out of the box?
- Weekly/daily agenda: The calendar page with current tasks
- Global TODO list: All unfinished action items
- Matching tags and properties: Structured information with fine-tuned search
- Timeline: Time-sorted view for single file
- Search view: Find entries by searching for text
- Agenda commands: Remote editing of Org trees
- Custom agenda views: Defining special searches and views

Markup for rich export

- Document title: Where the title is taken from
- Headings and sections: The document structure as seen by the exporter
- Table of contents: The if and where of the table of contents
- Paragraphs: Paragraphs
- Emphasis and monospace: Bold, italic, etc.
- Comment lines: What will **not** be exported
- Images and tables: Tables and Images will be included
- Literal examples: Source code examples with special formatting
- Include files: Include additional files into a document
- Embedded \LaTeX : \LaTeX can be freely used inside Org documents
- Exporting & Publishing

Working With Source Code

- Structure of Code Blocks
- Editing source code
- Evaluating code blocks
- Extracting source code
- Library of Babel
- Header Arguments

Not (Completely) Covered

- Completion: M-TAB knows what you need
- Clean view: Getting rid of leading stars in the outline
- MobileOrg: Org-mode on Android & iPhone
- Reading RSS Feeds
- Reminders
- Worg: Wiki-Org collaborative web editing
- Limiting Task Time
- Habit Tracking
- Publishing Projects
- Babel Library
- Table Math / Spreadsheet Functions

Org-Mode Updates

Org-Mode is in constant development, however the version shipping with Emacs is not constantly refreshed. A recent installation of Debian installed version 6.33 of Org-Mode as part of the Emacs package. Debian has 7.05 in repos. Current version of Org-Mode is 7.55, which is available in the testing repo.

This is a light presentation, based on the 40 page Org Guide. The actual Org-Mode manual is over 200 pages long. Almost every feature I've covered has a lot more information in the manual. Additional information is available on Worg, and a few user contributed customization documents are extremely extensive.

Emacs

- Emacs Jokes:
<http://www.gnu.org/fun/jokes/gnuemacs.acro.exp.html>
- Emacs: <http://www.gnu.org/software/emacs/>
- Emacs Wiki: <http://www.emacswiki.org/>
- Emacs Manual:
http://www.gnu.org/software/emacs/manual/html_node/elisp/index.html

Lisp Info:

- Programming in Emacs Lisp:
http://www.gnu.org/software/emacs/emacs-lisp-intro/html_node/index.html
- Emacs Lisp Reference:
http://www.gnu.org/software/emacs/manual/html_node/elisp/index.html

Org-Mode

- Org-Mode Website: <http://orgmode.org/>
- Org-Mode Tutorials:
<http://orgmode.org/worg/org-tutorials/index.html>
- **Advanced Org-Mode Config:**
<http://doc.norang.ca/org-mode.html>
- Worg: <http://orgmode.org/worg/>
- MobileOrg: <http://mobileorg.ncogni.to/>

Presentation

The following tools were used to prepare this presentation:

- Emacs 23.3
- Org-Mode 7.55
- L^AT_EX
- Beamer Class

Need any more be said?

Q & A

Questions and Answers