The Future of Org

Ihor Radchenko, Bastien Guerry

Email: yantar92@posteo.net
Matrix: @yantar92:matrix.org
Mastodon: fosstodon.org/@yantar92

Dec 7, 2024
EmacsConf 2024

Outline

Outline

Welcome Ihor as Org's maintainer!

(What really matters?)

bzg@gnu.org

(No changes need to be saved)

Outline

My step-by-step journey to Org maintenance

Background Material scientist

2015 Use Org to tame my PhD and research work (no idea about Elisp)

2017 Joined the mailing list to report a bug

https://lists.gnu.org/archive/html/emacs-orgmode/2017-12/msg00502.html

2018 Actively participating (and learning Elisp)

2019 My Org files grew enough to lag (first Emacs bug report)

https://debbugs.gnu.org/35453

I had no prior experience with Elisp, but
using Emacs and reading forums gave enough training

2020 Annoyed enough by lags (and COVID lockdown) to contribute a fix myself

`https://list.orgmode.org/87h7x9e5jo.fsf@localhost/`

≈2021 Got write access to savannah; decided to help with bug fixes in some areas

Sep 2021 Offer from Bastien to become a maintainer

2022 Helping with all the bug reports and list discussions

end of 2022 Second offer from Bastien (and unstable job situation)

2023 - Aug 2024 Working on Org full time

Sep 2024 Third maintenance offer (accepted)

Helping to fix bugs made me learn various parts of Org
Anyone interested in Org may repeat my journey regardless of experience

Priorities for Org maintenance

1. Codebase stability: fewer bugs, better APIs, long-term stability
2. Growing the Org community: new contributors, Org forums and chats
3. Support Org-related third-party packages, mobile apps, parsers
4. Org markup development
5. Strategically important new features

Active Org community and contributors are the key to get new features
A sole maintainer cannot do all the above even working full time

Outline

Modular Org

Richard Stallman <rms@gnu.org> (2020-09-16) (2020 emacs-orgmode.gnu.org maillist nolist track)
ML: Org mode Richard Stallman #saint_IGNUcius
Subject: Keep Org mode submodules in Emacs, please!
To: Russell Adams <RLAdams@AdamsInfoServ.Com>
Cc: emacs-orgmode@gnu.org, rms@gnu.org
Date: Wed, 16 Sep 2020 01:09:19 -0400
Contributor status unknown
In Org ML since 2007-11-25 (810 messages)

[[[To any NSA and FBI agents reading my email: please consider   ]]]
[[[whether defending the US Constitution against all enemies,   ]]]
[[[foreign or domestic, requires you to follow Snowden's example.]]]

Redesigning Org mode so that its secondary features are more modular,
and integrate better individually into Emacs, would be a big
improvement.

When the reorganization is finished, please keep all of the code
inside Emacs. Each piece should be in Emacs core, or in GNU ELPA,
whichever seems more convenient. Those are places we can refer to.
Please do not move any of the code to MELPA, because we will not refer
to it there.

I am not on the emacs-orgmode list. Russell, if you get this directly
but it does not come through the list, would you please forward it
there?

[5-line signature. Click/Enter to hide.]
--
Dr Richard Stallman
Chief GNUisance of the GNU Project (https://gnu.org)
Founder, Free Software Foundation (https://fsf.org)
Internet Hall-of-Famer (https://internethalloffame.org)

https://list.orgmode.org/orgmode/E1kIPh1-0001Lu-Rg@fencepost.gnu.org/

Slim down large Org libraries

> Too large and complex libraries often discourage contributions
>
> example: alphapapa's alternative to org-agenda (part of org-ql)

- Way too many Org libs are way too large
  - I plan to split large libs into smaller
  - The monsters like `org.el`, `org-agenda.el`, `org-table.el`, `org-list.el`, `ob-core.el`, `org-clock.el`, and `org-compat.el` should slim down
  - The existing 127 Org libraries will be turned into 263 (WIP!) smaller libraries

WIP: `https://git.sr.ht/~yantar92/org-mode/log/feature/refactor-deeps-v2`

Upstream generic Org libraries [tentative list]

> Upstreamed libs will benefit from Emacs dev's oversight

- `org-capture`: ... is originally forked from remember.el
  https://lists.gnu.org/archive/html/emacs-orgmode/2023-12/msg00412.html
- `org-protocol`: Processing external input into Emacs
  from command line, from clicked URLs as URL handler, etc.
  Unlike running commands directly (which is unsafe), `org-protocol`
  can be used for safe interaction with external processes
- `org-persist`: generic cache API
- `org-fold-core`: Generic folding API
- `org-read-date`: Interactive date/time selection

- `org-preview-image` and `org-preview-latex`: Image preview
  Does not have to be limited to Org mode buffers
  https://youtu.be/u44X_th6_oY?t=199 (credit: Karthik Chikmagal)
  https://list.orgmode.org/orgmode/87edbhljr7.fsf@hyperspace/
- `org-diary-lib`: Org extensions for diary-lib.el
- `org-clock-notify`: System notifications
- `org-idle`: Advanced idle time tracking
  not just for Emacs idleness, but for system idle time as well
- `org-duration`: Manipulate durations like 1d 3h 5min

- `org-agenda-undo`: undoing changes in multiple buffers at once
  This is the way undo works in agenda buffers, undoing changes in
  multiple headlines changed from agenda, even when they live in
  different Org files.
- `org-time`: Extra functions to work with Emacs time data
- `org-timer`: (almost) generic timer for Emacs
- `org-track-markers`: Keeping relative positions of markers when
  cutting/pasting text in buffers
  This is the way agenda does not get broken by cutting/pasting/refiling
  the underlying headings around.

- org-macs: many small helper functions
- (maybe) ox, org-element-ast: Org exporter can be generalized
  The core idea of Org export does not rely upon the details of Org
  syntax. It might use arbitrary parse trees (say, from tree-sitter) as
  input

  https://yhetil.org/emacs-devel/87frngx4fx.fsf@localhost/

- (maybe) ol: Org links can already be used outside Org mode
  Can we do better?
- (maybe) org-table-formula: Org spreadsheets
  Org spreadsheets are not inherently tied to Org markup and might be
  used more generally

Use modern Emacs APIs and libraries

Org covers many built-in Emacs features, but not always the newer ones

Recently added:

- `yank-media`, `dnd`: Drag and drop + pasting from system clipboard
  Credit: Visuwesh

  https://list.orgmode.org/orgmode/87jzsintv0.fsf@gmail.com/

To be added:

- `transient.el`: better replacement for Org menus (export, agenda, tempo, etc.)
  Org has an old, ad-hoc, limited, self-written menu system. And it has nothing to do with Org itself.
  Initial WIP: Tor-björn Claesson

  https://list.orgmode.org/orgmode/8734m28l9a.fsf@gmail.com/

Use modern Emacs APIs and libraries (contd.)

To be added (contd.):

- `compat.el`: Backwards compatibility library for Emacs
  Org has self-written `org-compat.el`. `compat.el` is strictly better
  when we need compatibility wrappers for older Emacs versions.

  https://list.orgmode.org/orgmode/87v8ks6rhf.fsf@localhost/

- `track-changes.el`: New library to track edits in buffer
  Now, Org uses a self-written implementation

Wish list:

- `context-menu-mode`: Context menu support
  This is a relatively new addition to Emacs. Should be supported by all
  the major modes.

- `touch-screen`: Touch screen support
  This is very important feature for Android.

- `thingatpt`: Generic parser/syntax queries

Improve Org parser APIs

Org uses two approaches to parsing simultaneously:

1. Proper parser (`org-element`)
2. Ad-hoc approximate regexp-based parser

We need to modify the parser to support all the internal use cases:

1. Generalize Org AST to be used in `ox.el` and `org-element.el` consistently

   done in `org-element-ast.el`

2. The parser should work in real time (fast)
   - `org-element-cache` is enabled by default since Org 9.6
   - `org-element-cache` is yet to support caching objects (markup vs. paragraphs and above)

Improve Org parser APIs (contd.)

We need to modify the parser to support all the internal use cases (contd.):

3. Fontification should use the parser

   https://list.orgmode.org/orgmode/87ee7c9quk.fsf@localhost/

4. Wish: Editing Org files should use parser
   We may need something akin `org-ml`

   https://github.com/ndwarshuis/org-ml

> org-element APIs should eventually be used everywhere

Improve Org babel APIs

- Org babel backends currently use ad-hoc, sometimes undocumented API
  - Backends have to use specific function names
  - Header arg conventions are often undocumented
  - Some APIs are not always reliable (async evaluation)
  - Babel backend documentation lives on WORG (Org wiki)
    https://orgmode.org/worg/org-contrib/babel/languages/index.html
- `ob-core` should use more `ox.el`-like backend definition style
- WORG documentation must eventually move to the manual
- Async API should be generalized and fixed
  Credit: Bruno Barbier

  https://list.orgmode.org/orgmode/65df0895.df0a0220.a68c8.0966@mx.google.com/

Outline

org-contrib

> Org mode is large. Some specialized parts are hard to maintain and require knowing very specialized specs/languages.

- Since Org 9.5, `org-contrib` is no longer a part of Org mode

  https://list.orgmode.org/orgmode/87a6qnixnr.fsf@gnu.org/

- We moved a number of specialized Org libraries to `org-contrib`

  https://list.orgmode.org/orgmode/87bl9rq29m.fsf@gnu.org/

  - `ob-vala`, `ob-shen`, `ob-ledger`, `ob-picolisp`, `ob-mscgen`, `ob-[h]ledger`, `ob-io`, `ob-ebnf`, `ob-coq`, `ob-asymptote`, `ob-abc`, `ob-J`

- Org itself has a fair bit of specialized libs: `org-feed`, `org-mobile`, various `ob-*` languages, various `ox-*` exporters

> If you know and use specialized libs
> from `org-contrib` or Org itself, please do volunteer

- `org-contrib`, as for now, is a collection of mostly abandoned libraries
- Org core team is still maintaining them, but on the minimal level
- In addition, we are open to provide help to third-party packages that can no longer be maintained
  - `https://orgmode.org/worg/org-orphanage.html` has a list of libraries in a search for new maintainers
  - We are also a part of Jonas Bernoulli's (tarsuis) Emacs orphanage `https://github.com/emacsorphanage/`

If you are an author/maintainer of an Org-related library and unable to provide continuous maintenance, feel free to write to Org mailing list.

Mobile apps and parsers

Org markup is understood by a number of mobile and web apps
(https://orgmode.org/tools.html):

- Orgzly revived (Android): https://www.orgzlyrevived.com/
- MobileOrg (iOS): https://mobileorg.github.io/features/
- Orgro (Android, iOS): https://orgro.org/
- Organice (Web): https://organice.200ok.ch/
- Org Note (Android, Web):
  https://github.com/Artawower/orgnote
- Logseq (Android, Web, Desktop): https://logseq.com/

We cannot directly contribute to non-Elisp development,
but we do want to unify the Org markup and avoid Markdown fate with
multiple syntax flavors

- There are existing Org parsers for Pico Lisp, Common Lisp, NodeJS, Python, Perl, Rust, JavaScript, Dart, Go, Julia, Clojure, Tree-sitter, OCaml, Haskell, and Typescript

  `https://orgmode.org/tools.html`

- They often implement a subset of Org syntax

- We provide reference Org syntax spec in
  `https://orgmode.org/worg/org-syntax.html`

- Eventually, our spec will be submitted as IETF RFC

  `https://list.orgmode.org/orgmode/871rjhha8t.fsf@gmail.com/`

- I also hope to implement a reference set of tests as a benchmark and parser development aid

  `https://list.orgmode.org/orgmode/87fsqzi4tw.fsf@localhost/`

Outline

Long-standing syntax problems

Before we fix Org syntax blueprint (IETF), we should resolve the major inconsistencies and problems.

1. Org mode does not always parse Org syntax consistently
   - Some parts of Org use a slightly different parser (regexp-based)
   - For example, numeric priorities in headings are not treated consistently
     https://list.orgmode.org/orgmode/CANDZv_N394TYMy30-KuEQV+eZOFEkm4GSjTrkUtDViK_-DkX-Q@mail.gmail.com/

2. Org syntax a number of edge cases that should better be addressed
   - Intra-word markup is awkward in Org
     https://list.orgmode.org/orgmode/4897bc60-b74f-ccfd-e13e-9b89a1194fdf@mailbox.org/
     https://list.orgmode.org/orgmode/87fw46rwd6.fsf@selune.samsung.net/
   - We often recommend zero-width space as a kind of escape character, but many people dislike it
     https://list.orgmode.org/orgmode/87ilw5yhv3.fsf@posteo.net/

2. Org syntax a number of edge cases that should better be addressed (contd.)
   - =/+_ are tricky when used inside verbatim markup beside spaces
     https://list.orgmode.org/orgmode/87v8vng70x.fsf@web.de/
   - Edge cases with double blank lines inside verbatim blocks
     https://list.orgmode.org/orgmode/87mtbilzhm.fsf@localhost/

3. Inlinetask syntax is not great both from human and technical point of view. It should be re-designed
   https://list.orgmode.org/orgmode/87fs47h2x4.fsf@localhost/

Some features have been requested many times
or can solve the existing problems

- Time zone support in timestamps
  the syntax has been decided; we need patches

  https://list.orgmode.org/87tu063ox2.fsf@localhost/

- Better repeater specifications in timestamps

  https://list.orgmode.org/877cxp1fbx.fsf@localhost/

- Custom syntax elements (aka inline special blocks)
  Credit: Juan Manuel Macías

  https://list.orgmode.org/875xwqj4tl.fsf@localhost/

- Multi-line cells in tables (we have no good ideas here; please help)

  https://list.orgmode.org/orgmode/877d0ia7vd.fsf@mat.ucm.es/

Outline

New features I hope to see in Org

1. Asynchronous LaTeX (and non-LaTeX) previews (WIP)
   Credit: Timothy Chapman (tecosaur/TEC), Karthik Chikmagalur
   (karthink)

   https://list.orgmode.org/orgmode/87lek2up0w.fsf@tec.tecosaur.net

2. Make `org-ql` part of Org core (WIP)
   Credit: Adam Porter (alphapapa)

   https://github.com/alphapapa/org-ql/issues/409

3. Multipage export (WIP)
   Credit: Orm Finnendahl

   https://list.orgmode.org/orgmode/ZoUdiTfbYqzPwTiX@orm-t14s/

4. Wish: Multi-language support
   - exporting multi-language documents
   - translation workflows
5. Wish: Collaborative editing
   - tracking changes
   - foreign comments
   - import from non-Org formats (with comments)

If you also want to see these features, please consider contributing

Outline

Org community forums → Org mailing list

- All the new features and ideas are not coming from nowhere
- Most of the ideas are first proposed and discussed on various forums/chats
  - Mailing list: `https://list.orgmode.org/`
  - Reddit: `https://reddit.com/r/orgmode/`
  - Mastodon: `https://fosstodon.org/tags/orgmode`
  - IRC/Matrix: `https://web.libera.chat/#org-mode`, `https://matrix.to/#/%23org-mode:matrix.org`
  - Meetups: `https://emacslife.com/calendar/`
  - ... and non-English, like
    - `https://emacs-china.org/`
    - `https://t.me/emacs_ru`

Eventually, we want all the serious discussions to happen on the mailing list
see `https://orgmode.org/worg/org-mailing-list.html`
... and all the non-trivial write-ups to end up in WORG (Org wiki)
see `https://orgmode.org/worg/`

Org mailing list → world

- Not everyone is interested reading the <u>whole</u> Org mailing list
- There are news feeds exposing important topics discussed on the list
  - via Woof! (news feeds in HTML, RSS, ORG, JSON, and MD)
    - `https://tracker.orgmode.org/news`
      announcements, blog-like posts
    - `https://tracker.orgmode.org/requests`
      feature requests, important discussions
  - weekly `https://sachachua.com/blog/category/emacs-news/`
    Credit: Sacha Chua
  - Woof! RSS → chatbots
    - there is experimental bot posting news and discussions in #org-mode
      Matrix room
    - wish: similar bots for reddit, mastodon, maybe IRC
- We are also looking for ideas to improve CSS styles in
  `https://list.orgmode.org/` and `https://orgmode.org/worg/`
  - wish: mailing list archives could have forum-like look&feel
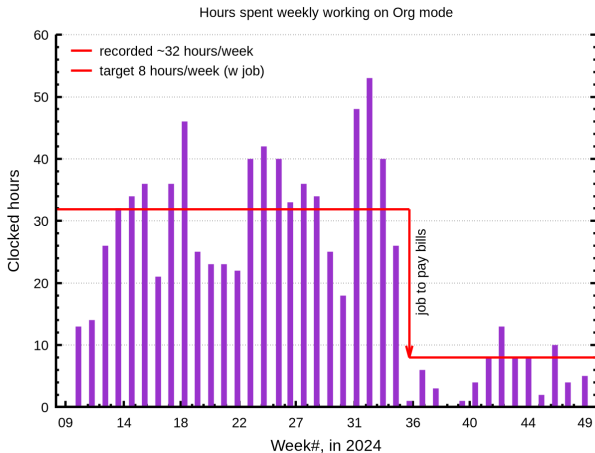
Contribute ideas!

> Unlike emacs-devel, Org mailing list is not restricted
> to Org development. It is more like forum.

- Anyone can post on the mailing list, without registration
- If you have any ideas, questions, or just rant about Org mode, feel free to post
- We'd also love to add new interesting blog posts to WORG topic collections

Outline

How much can a single person do?

Hours spent weekly working on Org mode

There is no way I can handle everything myself,
even if I work full time on Org

Contribute code!

> We badly need regular contributors

- In addition to all the listed plans&ideas, I myself now have 422 bug-related tasks, 859 feature ideas, and 894 ideas about Org improvements
  - Too many ideas, too little man-hours
- We need help fixing bugs and maintaining specialized libraries
- We need help implementing new features
- If anything, I need people who can help me if life makes unexpected turns (who knows what happens in 1, 2, 5 years from now)
- Among all types of contributions we prefer code

Why contribute?

- The absolute majority of new features are added by people who need them
- Even the commonly requested features!

Got a feature you wish existed? Do not wait for others to contribute it!
Write to mailing list, and we can help to get started.

- My job as a maintainer is guiding people through the Org code, if needed
- I will also check the code, point out mistakes, and suggest improvements
- Even Elisp beginners can contribute simple things under guidance
  All we need is:
  1. Your interest in specific new feature or area of Org
  2. Your free time spent contributing
  3. Readiness to learn

You should not be afraid to be wrong

- Not comfortable with patches?
  - git repos with changes are ok
  - even modified versions of files are ok
  - We will help as needed
- Not comfortable with emails?
  - there is #org-mode IRC and Matrix chats where you can first discuss rough ideas. You can ping me (yantar92) there

    `https://web.libera.chat/#org-mode`

    `https://matrix.to/#/%23org-mode:matrix.org`
  - We have monthly meetup where you can ask by voice

    `https://orgmode.org/worg/orgmeetup.html`

If you have any doubts, just ask. We always welcome new contributors

Benefits for code contributors

- **Contributing to Libre software can help with CV**
  - Personally written pet projects are a known good proof of skills
  - Contributing to community package can add more
    1. It is a proof of an ability to work in a team
    2. It is a proof of an ability to understand large codebases
    3. It is a proof of an ability to follow project standards

Imagine you are the main programmer on the team, and you are hiring a colleague. You have a bunch of resumes on your desk. Most of them contain puffy words and perhaps basic portfolios, half-finished simple programs that nobody uses, if anything. The last resume, unlike the others, shows contributions to well-established, real-world production code-bases. What is more, you can readily check both the patches and the discussions, in one click, to see how the person talks, to machines and to people. Which CV would you, as the expert, prefer?

People are struggling to differentiate their offering, and this is how they can do it. The fact it is not seen much in the wild is a plus! And it is understandable why, as it is not easy to contribute to, say, a complex C++ code base. Emacs is unique in this regard, and Org inherits the advantage.

by Rudolf Adamkovič,
the author of MathJax 3+ support in HTML export

- Thanks to many kind users, Org mode receives a noticeable amount of donations

  https://liberapay.com/org-mode/

- We share these donations among the interested regular contributors
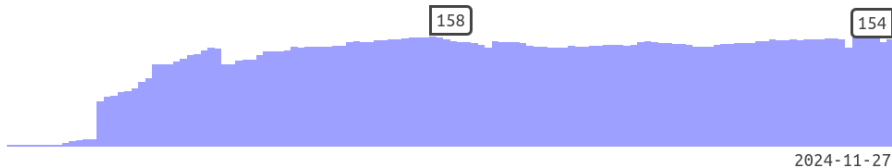
  https://liberapay.com/org-mode/income/

- If you become a regular Org contributor, you can also claim the share

org-mode joined 2 years ago.

View income history

Income Per Week (in Euro)

Contributing as non-programmer

- WORG needs more love
  - a number of articles is outdated
  - CSS styles can be improved
    - table of contents popup is not ideal on small screens
    - it would be nice to have dark/light themes and ability to switch
    - it would be nice to have some kind of "modern" theme as an option
  - more tutorials or links to good tutorials will be welcome

    `https://orgmode.org/worg/org-tutorials/index.html`

- Would be nice to have more screencasts like in
  `https://orgmode.org/features.html`
- Org manual is sometimes too technical (it is written by devs who are a bit too familiar with the code)
  - This is where input from ordinary users is invaluable

    `https://list.orgmode.org/orgmode/87y1rat98w.fsf@localhost/`

Got no free time, but still want to help?

- If you cannot help by code, text, or ideas, consider donating
- Donations do help a lot
  1. It increases motivation to do mundane tasks, like fixing bugs and writing tests
  2. It affects the amount of free time spent working on Org
  3. For a while, it was even the only source of income for me personally

Donations (both the group setup, and what people donate to me individually) have increased my motivation to work on Org, even as my time is split in more directions. It's a clear signal that the hours that are put in are valued, and ultimately that makes me feel more upbeat about the work, particularly when it's in the "bug squashing slog" phase as the LaTeX preview revamp is.

Aside from the direct impact, there are a bunch of indirect/secondary but positive aspects to it too: it also makes me feel that the community really values the work put into open-source projects like this.

by Timothy Chapman (tecosaur/TEC),
co-author of async LaTeX preview feature,
the maintainer of ox-html and org-plot

We use libre website to get donations:
`https://liberapay.com/org-mode/`

Thank you!

- EmacsConf page of this talk:
  https://emacsconf.org/2024/talks/org-update/
- How to contribute:
  https://orgmode.org/worg/org-contribute.html
- Support Org mode: https://liberapay.com/org-mode/