

1 Handling knowledge with Emacs

1.1 The Zettelkasten method

Niklas Luhmman wrote 70 books and nearly 400 scholarly articles published on a variety of subjects, including law, economy, politics, art, religion, ecology, mass media, and love.



1.2 Why?

- Improved understanding of the concepts
- Better retention and recall
- You'll find links between concepts that were not apparent before
- Intentionality - you'll be more focused when learning because you have to think and make notes
- You have one place to store all your notes - not spread across multiple platforms

1.3 The process

The Zettelkasten process has three types of notes...

1.3.1 Fleeting Notes

Incomplete ideas, memory reminders

1.3.2 Literature Notes

After reading a papers or books you need to include the references and expand the fleeting notes into more structured note.

These are notes about other's ideas.

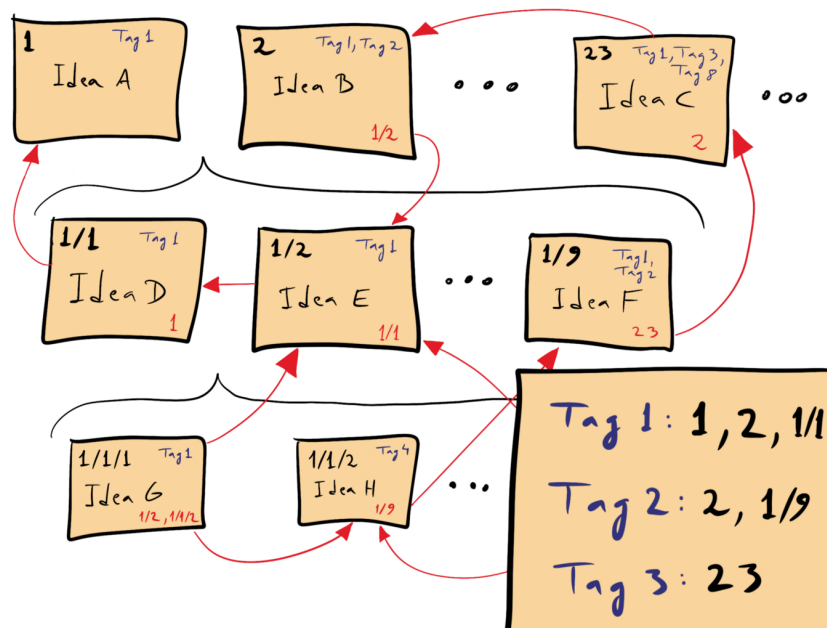
1.3.3 Permanent Notes

Once you have finished with your literature notes, you need to extract every idea/concept into separate notes.

- Write only one idea per note but be as complete as possible. This forces you to think about the content and distill it down to its core ideas.
- Write as if you are writing for somebody else.
- Show the content source.
- The note should be understood even if you don't know the context it was taken from. If you are looking at the note later, you will have forgotten the original context. The note should stand by itself.
- Be precise, clear and brief.

1.3.4 Connections

The important part here is to create connections between ideas



1.4 Implementation in Emacs

1.4.1 In init.el

```
(org
 +roam2
 +noter)
```

1.4.2 In package.el

```
(unpin! org-roam)
(unpin! org-roam-bibtex)
(package! org-roam-ui)
(package! citar)
(package! helm-bibtex)
```

1.4.3 In config.el

1. org-roam

```
(after! org-roam
 (setq org-roam-directory (file-truename "~/documents/2022/2022_01_Taller_Emacs/D
 (setq completion-ignore-case t))

(setq org-roam-capture-templates
 '(("n" "note" plain "%?"
 :target (file+head "%<%Y%m%dT%H%M%S>-${slug}.org"
 ":PROPERTIES:

:ID: %<%Y%m%dT%H%M%S>
:Time-stamp: <>
:END:
#+TITLE: ${title}
#+DATE: [%<%Y-%m-%d %H:%M:%S>]")
 :immediate-finish t
 :unnarrowed t
 :jump-to-captured t)
 ("r" "reference" plain "%?"
 :target (file+head "literature/${citekey}.org"
 ":PROPERTIES:

:ROAM_REFS: @${citekey}
```

```

:ID: %<%Y%m%dT%H%M%S>
:Time-stamp: <>
:END:
#+TITLE: ${citekey}: ${title}
#+DATE: [%<%Y-%m-%d %H:%M:%S>]
#+FILETAGS: literature

- *Author(s)* :: ${author-or-editor}
- *Title*      :: ${title}
- *Tags*       :: ${keywords}
- *URL*        :: ${url}
- *File*       :: [[file:${file}]]

* ${citekey}: Summary
:PROPERTIES:
:END:

* ${citekey}: LN
:PROPERTIES:
:END:

* Fleeting notes
:PROPERTIES:
:ROAM_EXCLUDE: t
:Custom_ID: ${citekey}
:URL: ${url}
:AUTHOR: ${author-or-editor}
:NOTER_DOCUMENT: ${file}
:NOTER_PAGE:
:END:")
      :immediate-finish t
      :unnarrowed t
      :jump-to-captured t)))

```

2. org-roam-bibtex

```

(after! citar-org
  (setq citar-bibliography '("~/Dropbox/home/documents/2022/2022_01_Taller_Emacs/I
  (setq org-cite-global-bibliography citar-bibliography)
  (setq bibtex-completion-bibliography citar-bibliography)

```

```

(setq citar-notes-paths '("~/Dropbox/home/documents/2022/2022_01_Taller_Emacs/Da
(setq org-cite-insert-processor 'citar)
(setq org-cite-follow-processor 'citar)
(setq org-cite-activate-processor 'citar)
(setq citar-open-note-function 'orb-citar-edit-note)
(setq citar-at-point-function 'embark-act)
;; Use consult-completing-read for enhanced interface.
(advice-add #'completing-read-multiple :override #'consult-completing-read-multi

(use-package! org-roam-bibtex
  :after org-roam
  :config
  ;; (require 'org-ref)
  (setq orb-preformat-keywords
    '("citekey" "title" "url" "author-or-editor" "keywords" "file"))

  (setq orb-roam-ref-format 'org-cite)

  (setq orb-process-file-keyword t
    orb-file-field-extensions '(".pdf", ".djvu")
    orb-insert-follow-link t))

```

3. org-roam-ui

```

(use-package! websocket
  :after org-roam)

(use-package! org-roam-ui
  :after org-roam
  :config
  (setq org-roam-ui-sync-theme t
    org-roam-ui-follow t
    org-roam-ui-follow-mode t
    org-roam-ui-update-on-save t
    org-roam-ui-open-on-start t))

```

1.5 Workflow

1. Include sources to Zotero: <https://www.wisdom.weizmann.ac.il/~zvika/course2015/announcements/WainerAmericanStatistician1984.pdf>

2. Export bibtex file.
3. Capture the reference.
4. Highlight the pdf (fleeting notes).
5. Create literature notes.
6. Create permanent notes.
7. Link notes.
8. Visualize

1.6 Optional

```
(after! org-roam
  (map! :leader
    (:prefix-map ("r" . "+org-roam")
      :desc "org-roam"          "l" #'org-roam-buffer-toggle
      ;; :desc "org-roam-find-index" "j" #'(lambda () (interactive) (org-roam-node-
      :desc "org-roam-node-insert" "i" #'org-roam-node-insert
      :desc "org-roam-node-find"   "f" #'org-roam-node-find
      :desc "org-roam-ref-find"    "r" #'org-roam-ref-find
      :desc "org-roam-ui-mode"     "g" #'org-roam-ui-mode
      :desc "org-roam-capture"     "c" #'org-roam-capture))

  (defun org-roam-image-name (fignumber description ext)
    (concat (org-roam-id-at-point) "-figure-" (number-to-string fignumber) "-" (clean-1
```