

Chi sono?

- Studente Phd, LAAS - CNRS (Toulouse)
- Lavoro sul silicio
- Faccio parte del Mittelab

Il Mittelab?

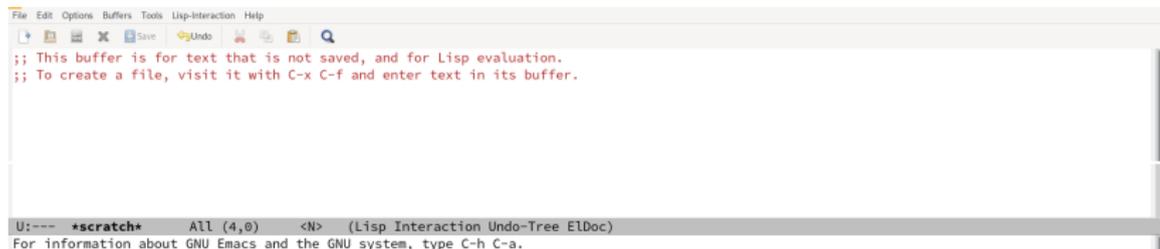
Hackerspace dove facciamo cose e vediamo gente

Siamo qui per parlare di orgmode

- emacs
- sapacemacs
- orgmode

Cos'è Emacs?

Un text editor



```
File Edit Options Buffers Tools Lisp-Interaction Help
;; This buffer is for text that is not saved, and for Lisp evaluation.
;; To create a file, visit it with C-x C-f and enter text in its buffer.

U:--- *scratch* All (4,0) <N> (Lisp Interaction Undo-Tree ELDoc)
For information about GNU Emacs and the GNU system, type C-h C-a.
```

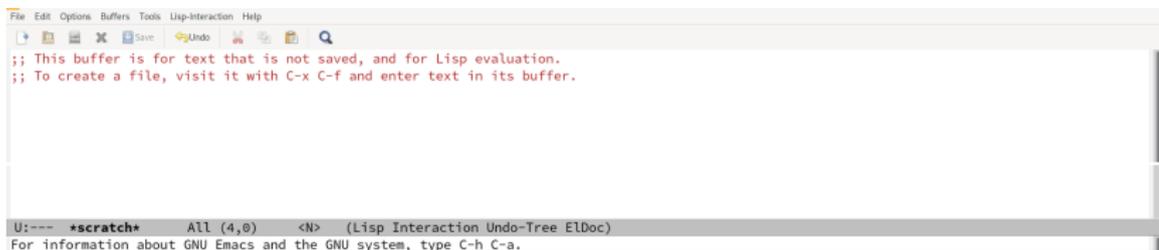
Con molte estensioni...

- mail client
- news reader
- debugger
- calendario
- uno psicologo
- project planner

Praticamente può diventare un sistema operativo

Cos'è Emacs?

Un text editor



Con molte estensioni...

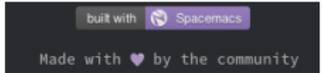
- mail client
- news reader
- debugger
- calendario
- uno psicologo
- project planner

Praticamente può diventare un sistema operativo

Cos'è Spacemacs?



Una distribuzione di Emacs!



Cos'è Spacemacs?

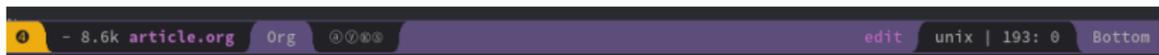


- Vim nativo
- mnemonico
- consistente

built with  Spacemacs

Made with ❤️ by the community

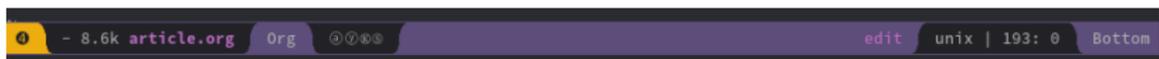
Buffer, Major, Minor



Da sinistra a destra:

- 1 Buffer
- 2 stato del buffer (- *)
- 3 dimensione e nome
- 4 major mode
- 5 minor modes
- 6 Scopo della finestra
- 7 file encoding
- 8 posizione

Org come major mode



Orgmode in questo caso è la major mode.

Come può essere attivata?

- Aprendo un file *.org
- 2 x SPC, org-mode

A cosa serve Orgmode?

- ◎ Le mie note
- A cosa serve Orgmode?
 - A gestire outline
 - A prendere appunti
 - Come agenda
 - Come time tracking
 - Come todolist
 - A fare presentazioni
 - A gestire blog

~

- * Le mie note
- ** A cosa serve Orgmode?
 - A gestire outline
 - A prendere appunti
 - Come agenda
 - Come time tracking
 - Come todolist
 - A fare presentazioni
 - A gestire blog

Installare orgmode

SPC

```
c → +compile/comments  
e → +errors  
f → +Files  
g → +git/versions-control  
h → +help  
i → +insertion
```

Installare orgmode

SPC + f

```
B → Find bookmark in file tree      R → Rename...  
D → Delete...                       S → Save all  
E → Open using sudo...              T → Focus current file in file tree  
F → helm-find-files                 M-t → Focus tag in file tree  
J → open-junk-file                  ESC → +prefix  
L → helm-locate                      e → +Emacs/Spacemacs
```

Installare orgmode

SPC + f + e

```
c → Recompile packages      D → Diff. with dotfile template
d → Open Spacemacs dotfile  E → Refresh env. variables
e → Open ".spacemacs.env"   R → Reload configuration
i → Open Emacs "init.el"    U → Update packages...
l → helm-locate-library     C-e → Recreate env. variables file
v → Copy Spacemacs version
```

Installare orgmode

SPC + f + e + d

dotspacemacs/layers:

```
(python :variables
  python-backend 'lsp ;;anaconda
  python-test-runner 'pytest
  python-formatter 'yapf
  python-fill-column 99
  python-enable-yapf-format-on-save t)
;;python-pipenv-activate t)

emacs-lisp
git
docker
org
markdown
graphviz
treemacs
```

Le basi

Prendere appunti in outline

```
◎ Il mio progetto
  ○ Introduzione
    Org-mode is a document editing, formatting, and organizing mode, designed for
    notes, planning, and authoring within the free software text editor Emacs
  ○ Un po di codice...
  ○ Budget...
```

- I titoli sono sequenze di *
- Elenchi puntati sono composti di -
- Elenchi numerati sono composti come #.
- I titoli con ... alla fine sono sezioni chiuse
- Il numero di sotto sezioni varia a piacere
- Ogni sezione si può chiudere/aprire (TAB sul titolo)

Le basi

Prendere appunti in outline

```
⊙ Il mio progetto
  ○ Introduzione
    Org-mode is a document editing, formatting, and organizing mode, designed for notes, planning, and authoring within the free software text editor Emacs
  ○ Un po di codice...
  ○ Budget...
```

Dopo un tab:

```
⊙ Il mio progetto
  ○ Introduzione...
  ○ Un po di codice...
  ○ Budget...
~
~
```

Il codice

Talvolta è utile inserire del codice per commentarlo

```
Ⓞ Il mio progetto
  Ⓞ Introduzione...
    Ⓞ Un po di codice
      #+begin_src python
        import os

        def _a_function(s):
            return 'func out: ' + s

        print(_a_function('hello world'))
      #+end_src
    Ⓞ Budget...
```

Editandolo in un buffer con l'autocompletamento (, +
)

Il codice

magari non anche con lsp attivo...

```
Edit, then exit with ', c' or abort with ', k'
• 1 import os
  2
• 3 def _a_function(s):
  4     return 'func out: ' + s
  5
  6 print(_a_function('hello world'))
~
~
~
~
```

* 94 *Org Src org.org[python]* Python •1 •3

alla fine (, + c)

Il codice

E automaticamente reinserirlo nel testo

```
● Il mio progetto
  ● Introduzione...
  ○ Un po di codice
    #+begin_src python
      import os

      def _a_function(s):
          return 'func out: ' + s

      print(_a_function('hello world'))
    #+end_src
  ○ Budget...
```

Tabelle (, + t + n)

Ovviamente spreadsheet!

```

◉ Il mio progetto
  ◉ Introduzione...
  ◉ Un po di codice...
  ◉ Budget

```

codice prodotto	prezzo unità	unità	totale
A	.4	6	2.4
B	3.7	8	29.6
totale			32.

```

#+TBLFM: @2$>..@-1$>=$2*$3::@>$>=vsum(@2..@-1)

```

con formule incluse

Sconsiglio di scrivere formule complesse, anche se lisp è supportato
Le tabelle sono plottabili in gnuplot

Links (, + i + l)

```
o Link
  interni
  =[[Introduzione][interni]]=
  esterni
  =[[file:screen.png][esterni]]=
  ssh
  =[[\ssh:server:pathtofile][ssh]]=
  Siti web
  =[[https://www.wikipedia.org/][Siti web]]=
  citazioni:
  cite:Test_19
```

- Le citazioni supportano file bib standard
- I link esterni possono essere verso qualsiasi cosa, si può linkare a punti specifici

Equazioni

o Formule in tex

$$\sum_{n=1}^{\infty} \frac{1}{n^2} = \frac{\pi^2}{6}$$

```
\begin{equation*}
\sum_{n=1}^{\infty} \frac{1}{n^2} = \frac{\pi^2}{6}
\end{equation*}
```

Editandolo in un buffer con l'autocompletamento (, + ')

Equazioni

```
edit, then exit with ', c' or abort with ', k'  
\begin{equation*}  
\sum_{n=1}^{\infty} \frac{1}{n^2} = \frac{\pi^2}{6}  
\end{equation*}  
~  
~
```

(, + c) per chiudere.

Il tutto in un file di testo

```

* Il mio progetto
** Introduzione
  Org-mode is a document editing, formatting, and organizing mode, designed for
  notes, planning, and authoring within the free software text editor Emacs
** Un po di codice
  #+begin_src python
    import os

    def _a_function(s):
      return 'func out: ' + s

    print(_a_function('hello world'))
  #+end_src
** Budget
  | codice prodotto | prezzo unità | unità | totale |
  |-----+-----+-----+-----|
  | A                |          .4 |    6 |    2.4 |
  | B                |          3.7 |    8 |   29.6 |
  |-----+-----+-----+-----|
  | totale           |              |      |   32.  |
#+TBLFM: @2$>..@-1$>=$2*$3::@>$>=vsum(@2..@-1)

** Link
  [[Introduzione][interni]]
  =[[Introduzione][interni]]=
  [[file:screen.png][esterni]]
  =[[file:screen.png][esterni]]=
  [[\ssh:server:pathtofile][ssh]]
  =[[\ssh:server:pathtofile][ssh]]=
  [[https://www.wikipedia.org/][Siti web]]
  =[[https://www.wikipedia.org/][Siti web]]=
  citazioni:
  cite:Test_19
** Formule in tex

```

Organizzare i progetti

Disclaimer

C'è una sostanziale differenza fra i tools offerti da orgmode e come poi uno decide di usarli. L'idea di fondo è che ognuno si dovrebbe personalizzare la propria installazione in base al proprio workflow

Io mostrerò esempi abbastanza generici che cercano di stare nel mezzo mostrando un possibile workflow che utilizza i principali tools.

Organizzare i progetti

projectile → (, + p + o) apre il TODOs.org file del repo ¹

```
#+Title: Un titolo di esempio
#+FILETAGS: :un_tag_per_il_file:

☉ Issue 1 link alla issue...
☉ Issue 2...
```

Ogni progetto è visto come una main outline.

Ad ogni progetto sono associate delle azioni che ruotano nella sequenza

TOD0, NEXT — DONE, CANCELLED ²

¹git ma non solo

²GTD

Il contenuto di un progetto

```
#+Title: Un titolo di esempio
#+FILETAGS: :un_tag_per_il_file:

◎ Issue 1 link alla issue
  ○ DONE leggere la issue
    CLOSED: [2020-05-21 Thu 18:14]
  ○ NEXT riprodurre la issue [1/3]
    - [X] installare il codice
    - [ ] copiare il db
    - [ ] girare il codice
  ○ TODU aggiornare la issue su gitlab
  ○ TODU fix se riproducibile, chiudo altrimenti
```

DONE : quando concludiamo un azione si aggiorna un date token

NEXT : la prossima azione

TODU : cose da fare dopo

Il contenuto di un progetto

```
◎ Issue 1 link alla issue  
○ DONE leggere la issue  
  CLOSED: [2020-05-21 Thu 18:14]  
○ NEXT riprodurre la issue [3/3]  
  - [X] installare il codice  
  - [X] copiare il db  
  - [X] girare il codice  
○ TODU aggiornare la issue su gitlab  
○ TODU fix se riproducibile, chiudo altrimenti
```

- []: elenchi di sotto operazioni (, + ,) per cambiare stato
[/]: contatore vuoto (, + ,) per aggiornare il contenuto

Chiudere un azione (, + L)

```
⦿ Issue 1 link alla issue
  ○ DONE leggere la issue
    CLOSED: [2020-05-21 Thu 18:14]
  ○ DONE riprodurre la issue [3/3]
    CLOSED: [2020-05-21 Thu 18:19]
    - [X] installare il codice
    - [X] copiare il db
    - [X] girare il codice
  ○ TODO aggiornare la issue su gitlab
  ○ TODO fix se riproducibile, chiudo altrimenti
```

per tornare indietro u oppure (, + H)

Cambiare arbitrariamente (, + T + T)

```
4 * 477 todos.org Org
{ [t] TODO [n] NEXT [d] DONE }
{ [w] WAITING [c] CANCELLED [m] MEETING }
5 * 105 *Org todo* Fundamental
```

Gli stati possono essere globali o per file o per progetto e sono arbitrari.

Azioni al cambiamento

Cambiare lo stato può triggerare un azione arbitraria

```
● Issue 1 link alla issue
  ○ DONE leggere la issue
    CLOSED: [2020-05-21 Thu 18:14]
  ○ DONE riprodurre la issue [3/3]
    CLOSED: [2020-05-21 Thu 18:19]
    - [X] installare il codice
    - [X] copiare il db
    - [X] girare il codice
  ✖ CANCELLED aggiornare la issue su gitlab
    CLOSED: [2020-05-21 Thu 18:21]
    - State "CANCELLED" from "TODO" [2020-05-21 Thu 18:21] \\
      Luca ha aggiornato la issue
  ○ TODO fix se riproducibile, chiudo altrimenti
```

Ad esempio salvare un commento con dei metadati.

Cosa faccio ora? (, + T + t)

```
① Issue 1 link alla issue...  
② TODO fix se riproducibile, chiudo altrimenti...  
③ Issue 2  
④ NEXT [#A] contattare marco marco@gmail.com...
```

e questo ci evidenzia solo le azioni da fare

Altre cose supportate

```
◎ Issue 1 link alla issue...
◎ Issue 2
○ NEXT [#A] contattare marco marco@gmail.com
  DEADLINE: <2020-05-24 Sun>
```

- Deadline (, + d + d)
- priorità (, + p + [a,b,c])
- Schedule (, + d + s)

E questo ci porta all'agenda

Integrare i progetti con l'agenda (ctrl-c + [])

- Dire all'agenda che il file esiste. (ctrl-c + [])
- Aprire l'agenda. (ctrl-c + a)

```
Press key for an agenda command:
-----
a  Agenda for current week or day
t  List of all TODO entries
m  Match a TAGS/PROP/TODO query
s  Search for keywords
/  Multi-occur
?  Find :FLAGGED: entries
*  Toggle sticky agenda views

n  Agenda and all TODOs

<  Buffer, subtree/region restriction
>  Remove restriction
e  Export agenda views
T  Entries with special TODO kwd
M  Like m, but only TODO entries
S  Like s, but only TODO entries
C  Configure custom agenda commands
#  List stuck projects (!=configure)
```

TODO entries

(t): Cosa devo fare dopo?

```
Global list of TODO items of type: ALL
Press 'N r' (e.g. '0 r') to search again: (0)[ALL] (1)TODO (2)NEXT (3)DONE (4)WAITING
(5)CANCELLED (6)MEETING
todos:      NEXT [#A] contattare marco marco@gmail.com :un_tag_per_il_file::
TODos:      WAITING [#A] :Si:Srf:Pot::
TODos:      NEXT [#A] :Si:Srf:Pot::
TODos:      NEXT [#A] :Si:Srf:Pot::
TODos:      TODO [#A] :Si:Srf:Pot:
todos:      TODO fix se riproducibile, chiudo altrimenti :un_tag_per_il_file::
TODos:      NEXT :Articol:
TODos:      NEXT :PANNA:code::
```

Questa funzione fa una regexp su tutti i file, ora sono ordinate per priorità

Da real agenda

(a): Il programma per la settimana?

```
Monday    18 May 2020 W21
Tuesday   19 May 2020
Wednesday 20 May 2020
Thursday  21 May 2020
           8:00..... -----
           10:00..... -----
           12:00..... -----
           14:00..... -----
           16:00..... -----
           18:00..... -----
           19:31..... now  - - - - -
           20:00..... -----
todos:    21:00-23:00 MEETING 42                               :un_tag_per_il_file::
todos:  In 3 d.: NEXT [#A] contattare marco marco@gmail.com :un_tag_per_il_file::
recurring_tsk:Sched. 5x: TODO pulire la cucina
Friday    22 May 2020
Saturday  23 May 2020
recurring_tsk:Scheduled: TODO pulire la cucina
recurring_tsk:TODO pulire il bagno
Sunday    24 May 2020
todos:  Deadline: NEXT [#A] contattare marco marco@gmail.com :un_tag_per_il_file::
```

- Sono stati inseriti anche dei task ricorrenti
- E inserito un meeting

Navigare l'agenda

- h,j,k,l ovviamente
- Alt + SPC

```
Org-agenda transient state
-----
Headline      Visit entry      Filter            Date              Toggle mode       View              Clock             Other
-----
[ht] set status [SPC] in other window [ft] by tag       [ds] schedule     [tf] follow       [vd] day          [ci] in           [gr] reload
[hk] kill       [TAB] & go to location [fr] refine by tag [ds] un-schedule [tl] log          [vu] week         [co] out          [.] go to today
[hr] refill    [RET] & del other windows [fc] by category [dd] set deadline [ta] archive     [vt] fortnight  [cq] cancel      [gd] go to date
[hA] archive   [o] link           [fh] by top headline [dd] remove deadline [tr] clock report [vn] month        [cj] jump
[hi] set tags  [fx] by regexp      [dt] timestamp     [*] do later      [ti] clock issues [vy] year
[hp] set priority [fd] delete all filters [-] do earlier     [td] diaries     [vn] next span   [vp] prev span
                                     [vr] reset
```

Navigare l'agenda

Alt + SPC

```
Org-agenda transient state
-----
Headline      Visit entry      Filter      Date      Toggle mode      View      Clock      Other
-----
[ht] set status [SPC] in other window [ft] by tag [ds] schedule [tf] follow [vd] day [cI] in [gr] reload
[hk] kill [TAB] & go to location [fr] refine by tag [ds] un-schedule [tl] log [vw] week [cO] out [.] go to today
[hr] refile [RET] & del other windows [fc] by category [dd] set deadline [ta] archive [vt] fortnight [cq] cancel [gd] go to date
[hA] archive [o] link [fh] by top headline [dd] remove deadline [tr] clock report [vn] month [cj] jump
[h:] set tags [fx] by regexp [dt] timestamp [dt] timestamp [ti] clock issues [vy] year
[hp] set priority [fd] delete all filters [+] do later [td] diaries [vn] next span
[-] do earlier [-] do earlier [vp] prev span
[vr] reset
```

Headline

```
-----
[ht] set status
[hk] kill
[hr] refile
[hA] archive
[h:] set tags
[hp] set priority
```

Date

```
-----
[ds] schedule
[DS] un-schedule
[dd] set deadline
[DD] remove deadline
[dt] timestamp
[+] do later
[-] do earlier
```

Il meeting

non è altro che un time stamp

```
◎ MEETING 42  
<2020-05-21 Thu 21:00-23:00>
```

(, + d + t) 9pm+2

Time tracking

Accessibile più o meno ovunque in Emacs,

- (, + C) in orgmode
- (SPC + a + o + c) ovunque
- (ALT + SPC + c) in agenda

```
c → org-clock-cancel          o → org-clock-out
d → org-clock-display         p → org-pomodoro
e → org-evaluate-time-range   r → org-resolve-clocks
g → org-clock-goto           I → org-clock-in-last
i → org-clock-in             R → org-clock-report
j → org-clock-jump-to-current-clock
```

Aprire e chiudere un timer

Clock-in e clock-out:

```
o DONE task1
  CLOSED: [2020-05-21 Thu 16:00]
  :LOGBOOK:
  CLOCK: [2020-05-21 Thu 15:30]--[2020-05-21 Thu 16:00] => 0:30
  CLOCK: [2020-05-21 Thu 14:00]--[2020-05-21 Thu 15:00] => 1:00
  :END:
o DONE task2
  CLOSED: [2020-05-21 Thu 19:34]
  :LOGBOOK:
  CLOCK: [2020-05-21 Thu 16:33]--[2020-05-21 Thu 19:33] => 3:00
  :END:
```

è tutto testo, se è sbagliato basta editarlo e premere (, + ,)

Il report finale

```
◎ Issue 3
#+BEGIN: clocktable :scope subtree :maxlevel 2
#+CAPTION: Clock summary at [2020-05-21 Thu 19:35]
|  Headline      | Time      |      |
|-----+-----+-----|
| *Total time*  | *4:30*   |      |
|-----+-----+-----|
| Issue 3       | 4:30     |      |
| \_ task1      |          | 1:30 |
| \_ task2      |          | 3:00 |
#+END:
*◎ DONE task1...
  ◎ DONE task2...
```

Conclusioni

- Orgmode è un gran pezzo di software
- Fornisce tools per organizzarci
- ma sta a noi decidere come combinarli
- Spacemacs lo rende intuitivo per chi arriva da VIM
- c'è molto di più di quello che ho detto.³

³Ho detto che orgmode fa anche presentazioni esportando in beamer?

Ringraziamenti software

Un po' di software che mi hanno aiutato in questo lavoro

- L^AT_EX
- maim
- pdfpc

Referenze

- sachar's blog
- What is orgmode
- What are the advantages and disadvantages of Workflow compared to Org-Mode?
- Un'idea di utilizzo (commerciale)
- Un'idea di utilizzo più legata al codice