

# FIM (Fbi IMproved)

## ein Bildbetrachter für Linux-Enthusiasten, den Neulingen erklärt

Michele MARTONE

Augsburger Linux-Infotag 2023  
29.04.2023



*Junger Hase*

Albrecht Dürer, 1502 (von Wikipedia)

- ▶ ich nutzte `fbi` (*Linux framebuffer imageviewer*) von Gerd Hoffmann
- ▶ ich wollte VIM-artige `h` `j` `k` `l` *Navigationstasten*

## FIM: *Fbi IMproved*



*Wolpertinger*

Rainer Zenzs Überarbeitung von Dürers "Junger Hase" (von Wikipedia)

fim begann als fbi-Fork, inspiriert von:

- ▶ dem VIM Texteditor
- ▶ dem Mutt E-Mail-Programm
- ▶ Shell-Sprachen

# mutt E-Mail-Programm

```
q:Quit d:Del u:Undel s:Save m:Mail r:Reply R:Group ?:Help
17644 N 20210916@19:10 Michele Martone michelemartone@users.sourceforge 2.0K
17645 Ns 20210916@20:39 Paul Gevers elbrus@debian.org 3.2K
17646 Ns 20210916@20:39 Paul Gevers elbrus@debian.org 3.2K
17647 N 20210916@18:03 Debian Bug Trac owner@bugs.debian.org 0.8K
17648 N 20210916@17:58 Michele Martone michelemartone@users.sourceforge 1.3K
17649 N 20210916@17:58 Michele Martone michelemartone@users.sourceforge 1.3K
17650 Ns 20210916@13:58 Paul Gevers elbrus@debian.org 2.6K
17651 Ns 20210916@13:58 Paul Gevers elbrus@debian.org 2.6K
17652 N + 20210916@10:30 Debian Bug Trac owner@bugs.debian.org 0.8K
17653 N 20210916@10:27 Michele Martone michelemartone@users.sourceforge 0.2K
17654 NsF 20210916@10:26 To Paul Gevers michele.martone@posteo.de 5.5K
17655 NsF 20210916@10:26 To Paul Gevers michele.martone@posteo.de 5.5K
17656 NsF 20210916@10:26 To Paul Gevers michele.martone@posteo.de 5.5K
17657 Ns 20210916@11:53 Paul Gevers elbrus@debian.org 6.0K
17658 Ns 20210916@11:53 Paul Gevers elbrus@debian.org 6.0K
17659 N 20210916@07:33 Debian Bug Trac owner@bugs.debian.org 0.8K
17660 Ns 20210916@07:20 Michele Martone michelemartone@users.sourceforge 4.8K
17661 Ns 20210916@07:20 Michele Martone michelemartone@users.sourceforge 4.8K
17662 Ns 20210916@08:54 Paul Gevers elbrus@debian.org 12K
17663 Ns 20210916@08:54 Paul Gevers elbrus@debian.org 12K
17664 N 20210916@00:51 Debian Bug Trac owner@bugs.debian.org 0.8K
17665 N 20210916@00:40 Michele Martone michelemartone@users.sourceforge 2.0K
17666 N 20210916@00:40 Michele Martone michelemartone@users.sourceforge 2.0K
17667 Ns 20210915@22:12 Paul Gevers elbrus@debian.org 4.7K Bug#994426: octave-sparsersb: flaky
17668 N 20210915@19:24 Dmitriy Traytel traytel@di.ku.dk 0.8K [TYPES/announcement] PhD position at the
17670 N + 20210915@12:22 ACM TechNews technews-editor@acm.org 115K ACM TechNews, Wednesda
17671 N + 20210915@10:25 Manning Publica mkt@manning.com 92K New at Manning! Save up to 40% sitewi
17672 N 20210915@14:51 Vincenzo De Mai hpc-announce@mcs.anl.gov 4.0K [hpc-announce] Call fo
---Mutt: [Msgs:77500/79181 New:78738 Flag:22 Post:13 3093M]---(reverse-threads/date)-----
```

# vim (Vi IMproved) Texteditor

```
To: "19. Augsburg Linux-Infotag" <root@luga.de>  
From: Michele Martone <michelemartone@users.sourceforge.net>  
Subject: Re: 19. Augsburg Linux-Infotag: Programm ist online.  
In-Reply-To: <168044737036.207651.14793612260229213105@dingbuns.luga.de>
```

On 20230402@14:56, 19. Augsburg Linux-Infotag wrote:

> Liebe Vortragende und Standbetreuer!

>

> Wir haben das Programm des 19. Augsburg Linux-Infotages online gestellt.

> Ihr findest dieses auf der Webseite des Linux-Infotags unter <https://www.luga.de/LIT-2023/> .

>

> Bitte meldet Euch unter [lit-2023@luga.de](mailto:lit-2023@luga.de), falls es noch Korrekturbedarf gibt  
> oder ihr sonst weitere Fragen habt.

>

> Wir werden uns vor dem Linux-Infotag noch mit weiteren Informationen zum Ablauf  
> melden und freuen uns bereits, Euch dann an der Veranstaltung begrüßen zu dürfen.

>

> Viele Grüße!

>

> Das Orga-Team des 19. Augsburg Linux-Infotages

Toll!

-


-

```
.mutt/tmp/mutt-laptop-1000-9296-1551449151162669755 @laptop[+] 84=0x54 l:22,c:1 All
```

## grep: Suche nach einem regulären Ausdruck 🔍


```
user@computer ~ $ echo muster | grep 'muster' # ganzes Wort
muster
user@computer ~ $ echo ausdruck | grep 'muster' # ganzes Wort
user@computer ~ $ echo ausdruck | grep '[muster]' # einzelne Buchstaben
ausdruck
user@computer ~ $ echo ausdruck | grep '[aus]' # einzelne Buchstaben
ausdruck
user@computer ~ $ echo ausdruck | grep 'aus.*k' # aus IRGENDETWAS k
ausdruck
user@computer ~ $
```


## ein Unix-Werkzeug

 eine Aufgabe: Bilder zeigen

>\_ Kommandozeilenorientiert

 Konfigurationsdateien `/etc/fimrc`, `$HOME/.fimrc`

 Reguläre Ausdrücke, stdin, stdout, scripting

 vollständig interoperabel

*there's more than one way to do it (siehe Perl und C++)*

**Wohlgemerkt:**

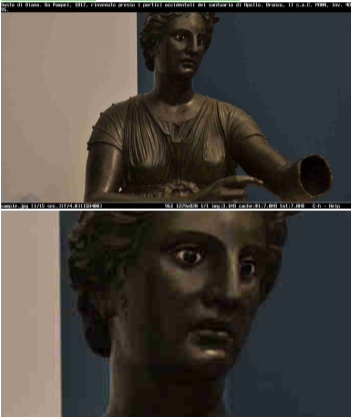
ein Bildbetrachter ist *nicht unbedingt* ein Bildbearbeitungsprogramm

zumindest nicht hier

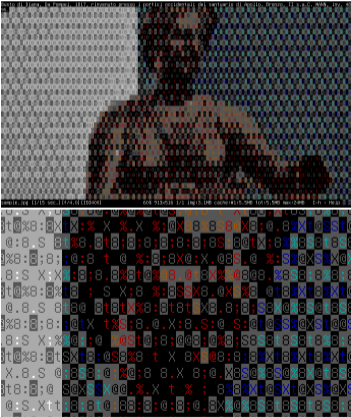
# ein universeller Bildbetrachter

- ▶ minimalistische Benutzeroberfläche
- ▶ einheitliches *look and feel*
- ▶ verschiedene Videomodi

-o fb / -o sdl



-o ca



-o aa







## respektiert langsamere Hardware

- ▶ caching (vermeidet wiederholtes Laden)
- ▶ background prefetch (Laden im Hintergrund)
- ▶ ssh- und screen- freundlich

## Grundlagen des Aufrufens >\_


 `$ fim file.jpg` *Dateiname nicht wichtig – Inhalt (Format) schon*

 `$ fim dir/` *Filtrierung nach Dateiname*


 `$ fim -R dir/` *Rekursion*


 `$ fim -B dir/` *Rekursion, lädt aber im Hintergrund*

und im Window Manager?


 `$ fim --cd-and-readdir..` *wechsle zum Verzeichnis, reihe hiesige Dateien ein*


## und im Window Manager?


 `$ fim --cd-and-readdir..` *wechsle zum Verzeichnis, reihe hiesige Dateien ein*

 `$ fim --cd-and-readdir -a ...` *auto Skalierung*


## und im Window Manager?


 `$ fim --cd-and-readdir..` *wechsle zum Verzeichnis, reihe hiesige Dateien ein*


 `$ fim --cd-and-readdir -a ...` *auto Skalierung*

 `$ fim --cd-and-readdir -a --` *Ende der Optionen*

## und im Window Manager?

 `$ fim --cd-and-readdir..` *wechsle zum Verzeichnis, reihe hiesige Dateien ein*


 `$ fim --cd-and-readdir -a ...` *auto Skalierung*


 `$ fim --cd-and-readdir -a --` *Ende der Optionen*

 `$ fim --cd-and-readdir -a -- files...` *empfohlene Kombination*


## und im Window Manager?

 `$ fim --cd-and-readdir..` *wechsle zum Verzeichnis, reihe hiesige Dateien ein*

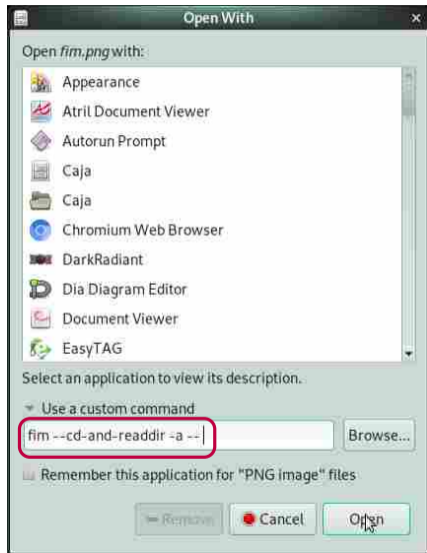
 `$ fim --cd-and-readdir -a ...` *auto Skalierung*

 `$ fim --cd-and-readdir -a --` *Ende der Optionen*

 `$ fim --cd-and-readdir -a -- files...` *empfohlene Kombination*

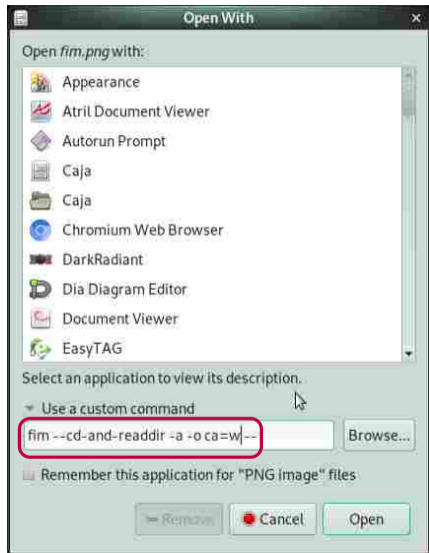
 `$ fim --cd-and-readdir -a -- -a` *öffne Datei -a*

# Beispiel von `vim --cd-and-readdir -a -- fim.png`





Beispiel von `vim --cd-and-readdir -a -o ca=w -- vim.png`



## tastaturorientierte interaktive Anwendung

Zum Skalieren und um sich “herum zu bewegen”, wie erwartbar:

`+`, `-`, `Left`, `n`, `p`, `r`, `PageDown`, ...

## zu Aktionen zugeordnete Tasten in der Sprache FIMs

- ▶ `+` `magnify`
- ▶ `-` `reduce`
- ▶ `Left` `pan_left`
- ▶ `n` `next`
- ▶ `p` `prev`
- ▶ `r` `rotate90`
- ▶ `Del` `list 'pop';reload`
- ▶ `PageDown` `if(_display_console==0){next;}else{scd;}`

# Befehle "leben" im `:` Kommandozeilenmodus

- ▶ `:` aktiviert Kommandozeilenmodus
- ▶ `Tab` löst Autovervollständigung aus



- ▶ derselbe Satzbau wie Konfigurationsdateien und Skriptdateien

# Satzbauelemente von FIM

- ▶ Befehle
- ▶ Aliase (einstellbar)
- ▶ Variablen (einstellbar)
- ▶ if- und while- Blöcke
- ▶ abkürzende Ausdrücke und Sätze

## wie ich `fim` anwende 🔑

- ▶ täglich
  - ▶ interaktiv
  - ▶ Verwaltung von Bilddateisammlungen
- ▶ manchmal
  - ▶ besondere Funktionen
  - ▶ die interne Kommandozeile
- ▶ selten
  - ▶ ändere ich die Konfigurationsdatei
  - ▶ schreibe ich Shell-Skripts oder Funktionen um `fim`

im Mittelpunkt dieses Vortrages ♪

## bestimmende Merkmale von FIM zeigen

- ▶ das ist kein Tutorial
- ▶ nicht erschöpfend
- ▶ für sprachliche Aspekte, siehe meinen FOSDEM'23 Vortrag (<https://fosdem.org/2023/schedule/event/fim/>)

## Kurzformen vom `goto` Befehl

- ▶ `:` aktiviere Kommandozeile
- ▶ `:` `3` `goto 3`
- ▶ `:` `^` `goto '^'` *wechsle zur ersten Datei mit Zirkumflex*
- ▶ `:` `$` `goto '$'` *wechsle zur letzten Datei*



## Kurzformen vom `scale` Befehl

- ▶ `:` `-10%` `scale '-10%'`     *verkleinere um ein Zehntel*
- ▶ `:` `+10%` `scale '+10%'`     *vergrößere um ein Zehntel*
- ▶ `:` `*0.9` ...
- ▶ `:` `*1.1` ...

## Tastenaktionen per Kommandozeile ausführen

- ▶ `$ fim . -k $KeySym` *KeySyms sind in man fimrc verzeichnet*

## interaktive Tasten über `-k`

- ▶ `$ fim . -k r` *dreh einmal um 90 Grad*
- ▶ `$ fim . -k Del` *entferne erstes Bild aus der Liste*
- ▶ `$ fim . -k C-h` *frag Hilfe ab*


## Tastenkombinationen mit `-K`

- ▶ `$ fim . -K rA` *drehe (rotate) and autoskaliere*

stelle eine Ziffer vor eine Taste, um zu wiederholen 

Ziffern haben eine besondere Nutzung:

▶ `$ fim . -K 2r` *drehe zweimal*

wiederhole die letzte Aktion mit  danach  
(`recording 'repeat_last'`)

vergrößere einmal, und noch einmal:

▶ `$ fim . -K ++`

▶ `$ fim . -K +.`

▶ `$ fim . -K +1.`

vergrößere einmal, und noch *dreimal*:

▶ `$ fim . -K +3.`

▶ `$ fim . -K ++++`

# ohne Übertreiben

☐ wiederhole lediglich die Aktion der *letzten Taste*

▶ `fim . -K Z+n3.`

vs

▶ `fim . -K Z+nZ+nZ+nZ+n`

▶ Für komplexere Aktionen, passe `~/ .fimrc` an:

```
1 bind 'S' 'sleep 1;magnify;next;';
```

und

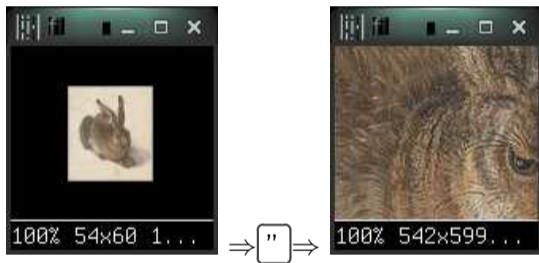
▶ `fim . -K 4S`

▶ `fim . -K S3.`

## zeige gleichnamige Varianten einer Datei

```
$ fim ~/lr/ --load-shadow ~/hr/
```

bei ~/lr/**a.jpg**, drucke " um Bild ~/hr/**a.jpg** zu laden und zeigen



```
1 fim -- -N -osdl=128 lr/ --load-shadow hr/
```

zeige "thumbnails", danach lade gleichnamige Dateien in Hochoauflösung



wechsle wiederholt zwischen zwei Ansichten mit `'` (Apostroph)  $\rightleftharpoons$

- ▶ soll Positionierung und Skalierung behalten



```
1 fim -q -c'_want_wm_caption_status=" "' -N -osdl=400:200 duerer_1502_hare.jpg  
zenz_2005_wolperdinger.jpg -= "-K+kkn+kk"
```

- ▶ *Shortlisting*:

`Enter` wählt aus, ``` (Backtick) begrenzt die Dateiliste zur eingewählten Dateien

## conversion pipelines

- ▶ auf manchen Dateinamenserweiterungen (SVG, DIA, XCF, ...)
- ▶ ruf einen externen Konverter auf (`inkscape`, ...)
- ▶ `-X` deaktiviert das

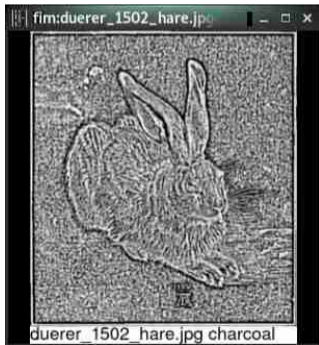


```
1 fim -q -osdl=240 /usr/share/inkscape/branding/tux.svg
```

## personalisierte *conversion pipelines*

- ▶ jedes Bild filtern vor dem Zeigen
- ▶ vertraut mit `find -exec 'cmd {};' ?`
- ▶ (`{}` werden durch den Dateinamen des Bildes ersetzt)

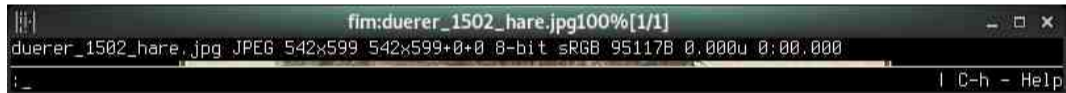
```
fim --pread-cmd="convert {} -charcoal 5 -pointsize 36 label:'{} charcoal' -append ppm:-" *
```



# personalisierte Aktionen mit Shellbefehlen

`:` `! "identify" i:_filename`

```
1 fim -N -q duerer_1502_hare.jpg -osdl=800:40 -K':! "identify" i:_filename' -K '' -=
```



```
fim:duerer_1502_hare.jpg100%[1/1]
duerer_1502_hare.jpg JPEG 542x599 542x599+0+0 8-bit sRGB 95117B 0.000u 0:00.000
|_
| C-h - Help
```

## kann wirklich nichts editieren?

Warnung: gefährlich 

```
:alias "remove_exif_tags" "!'exiftool' '-all=' '--exif:DateTimeOriginal'  
i:_filename" # lösche EXIF tags
```

# lade **Dateinamen** aus einer **Dateinamenlistendatei**

# lade Dateinamen aus einer Dateinamenlistendatei<sup>1</sup>

<sup>1</sup>Kennt Ihr Rhabarberbarbara <https://www.youtube.com/watch?v=gG62zay3kck> schon?



## lade Dateinamen aus einer Dateinamensliste

```
$ cat objectors.list
```

```
1 daniel.jpg  
2 edward.jpg  
3 chelsea.jpg
```

```
$ fim -L objectors.list
```



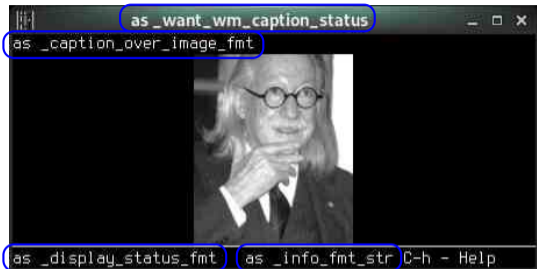
## Beschreibungsdatei mit bilddateibezogenen `i:_comment`-Variablen

```
$ cat intellectuals.desc
```

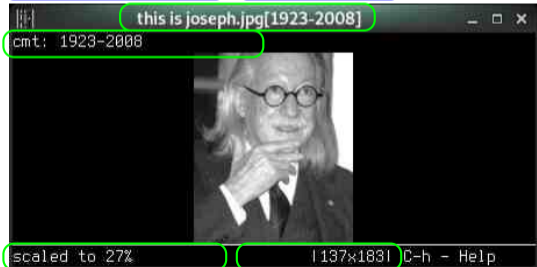
```
1 ivan.jpg          1926-2002
2 joseph.jpg        1923-2008
3 noam.jpg           1928-
```

```
$ fim -D intellectuals.desc -L intellectuals.list
```

# Nutzung von `i:_comment` in *expandos*, für Überschriften



```
1 _caption_over_image=2;  
2 _info_fmt_str="as _info_fmt_str";  
3 _display_status_fmt="as _display_status_fmt";  
4 _want_wm_caption_status="as _want_wm_caption_status";  
5 _caption_over_image_fmt="as _caption_over_image_fmt";
```



```
1 _caption_over_image=2;  
2 _info_fmt_str="|%wx%h|";  
3 _display_status_fmt="scaled to %p%";  
4 _want_wm_caption_status="this is %N %k";  
5 _caption_over_image_fmt="%?_comment?cmt: %:_comment:?";
```

## i:variablen (per-Bild) zuweisen und *erben*

```
$ cat programmers.desc
```

```
1  #!fim:category=businessman
2  bill.jpg
3  #!fim:category=activist
4  aaron.jpg      RIP
5  asbesto.jpg   https://museo.freaknet.org/
6  bram.jpg      https://iccf.nl/
7  richard.jpg   https://fsf.org/
```

## shebang (#!fim:) modifiers und Expansion von @Variablen

```
$ cat activists.desc
```

```
1 #!fim: ^=she
2 #!fim: activity=programs useful tools
3 alexandra.jpg @activity
4 greta.jpg     blasts leaders over greenwashing
5 #!fim: ^=he
6 bram.jpg      @activity
```

## wechsle zur Datei mit der passenden Beschreibung

- ▶ `:` `goto '/green/'`
- ▶ `/green` *Suchzeile, als alternative Kurzform*
- ▶ `?green` *ähnlich, aber nach hinten*



```
1 fim -c'_want_wm_caption_status="%k"' -N -Dactivists.desc -Lactivists.list -osdl=400:200 -K/green -K ''
```

(Bild: Wikipedia)

## wechsle zur Datei mit anderen `i:category` oder `i:_comment`

```
: goto '+category|_comment+'
```

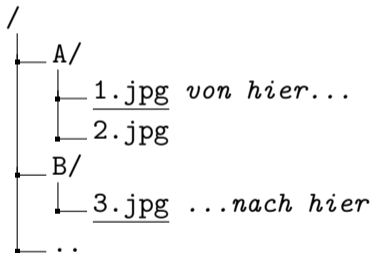
```
$ cat programmers.desc
```

```
1  #!fim:category=businessman
2  bill.jpg
3  #!fim:category=activist
4  aaron.jpg          RIP
5  asbesto.jpg
6  bram.jpg
7  richard.jpg
```

## wechsle zur Datei in dem *nächsten* Verzeichnis

1. `$ fim A/1.jpg A/2.jpg B/3.jpg`

2. `: goto '+/S'`



im Fall mehrerer Suchparameter wird nur ein Wechsel geschehen

`:` `goto '+category|_comment+' '+/S' ...` *siehe help goto für mehr*



# mutt und limit



```
q:Quit d:Del u:Undel s:Save m:Mail r:Reply R:Group ?:Help
 97 N L 20230419@17:09 To calcul@liste claire.mouton@creatis.insa-lyo 9.8K [calcul]
456 N L 20230406@08:43 To calcul celine.acary-robert@univ-greno 22K [calcul]
 557 N L 20230331@14:19 To calcul celine.acary-robert@univ-greno 23K [calcul]
 852 N L 20230322@14:47 To calcul celine.acary-robert@univ-greno 19K [calcul]
33564 N L 20200608@14:09 Cc cocci@system Markus.Elfring@web.de 1.4K Re: [Co
33565 N L 20200608@14:09 Cc cocci@system Markus.Elfring@web.de 1.4K Re: [Co
33566 N L 20200608@13:21 Cc cocci@system julia.lawall@inria.fr 1.7K Re: [Co
33569 N L 20200608@12:31 Cc cocci@system julia.lawall@inria.fr 1.7K Re: [Co
33570 N L 20200608@12:27 Cc Coccinelle julia.lawall@inria.fr 2.7K Re: [Co
33585 N L 20200607@23:23 Cc Coccinelle julia.lawall@inria.fr 1.2K Re: [Co
33588 N L 20200607@22:12 Cc Coccinelle julia.lawall@inria.fr 3.3K Re: [Co
42081 N L 20191113@05:57 Cc calcul Damien.MATEO@cnrs.fr 3.4K Re: [ca
43828 N L 20191014@11:47 To calcul@liste karim.hasnaoui@idris.fr 1.7K [calcul]
50709 N L 20190420@10:05 Cc Coccinelle julia.lawall@lip6.fr 0.7K Re: [Co
50711 N L 20190420@09:35 Cc Coccinelle julia.lawall@lip6.fr 2.5K Re: [Co
54333 N L 20181211@13:11 To calcul@liste remy.mozul@umontpellier.fr 198K [calcul]
63487 N L 20180129@14:54 To calcul@liste franck.thollard@univ-grenoble- 73K [calcul]
64012 N L 20180116@11:31 To calcul cyrille.bonamy@legi.cnrs.fr 14K [calcul]
68606 N L 20170724@20:55 To calcul@liste Karim.Ramage@ipsl.polytechniqu 368K [calcul]
70874 N L 20170510@09:59 To calcul@liste loic.gouarin@math.u-psud.fr 2.5K [calcul]
---Mutt. [Msgs.20/79181 New.78738 Flag.22 Post.13 3093M]---(reverse-date/date)-(all)
Limit to messages matching: ~l ~s pyth ~f fr ~t de
```



# vorübergehend Dateiliste verkürzen ✂



```
limit 'category' 'activist'
```



(Bilder: Wikipedia)

```
1 #!fim:category=businessman
2 bill.jpg
3 #!fim:category=activist
4 aaron.jpg      RIP
5 asbesto.jpg
6 bram.jpg
7 richard.jpg
```

## begrenze die Dateiliste mit *limit* auf andere Parameter

- ▶ `ls -l | limit '~='` *Dateien mit selbem Grundname (basename)*
- ▶ `ls -l | limit '~d'` *Dateien vom selben Tag*
- ▶ `ls -l | limit '~z'` *Dateien mit selber Größe*
- ▶ ...

# Zusammenfassung

FIMs Nutzungsgrundkonzept:

- ▶ oft benutzte Aktionen als **Aliase** herunterschreiben
- ▶ **Tasten** zu beliebten Aliasen mit **bind** zuordnen
- ▶ ändere **~/.fimrc** mit diesen Aliasen
- ▶ nur auswendig lernen, was der täglichen Benutzung dient

# das Ende

- ▶ im Netz: <http://www.nongnu.org/fbi-improved>
- ▶ in der Distro:
  - ▶ `$ sudo apt-get install fim` *Debian & family*
  - ▶ oder andere Distro, sei es NixOS, Gentoo, arch, ...
- ▶ siehe `man fim` für Optionen, und `man fimrc` für die Sprache
- ▶ viel Spaß mit fim! (Feedback jeder Art willkommen!)

## Übrigens



Für mehr über `fims Sprache`, siehe meinen FOSDEM'23 Vortrag (<https://fosdem.org/2023/schedule/event/fim/>)

## extra: *amateur files recovery*

```
$ fim --offset 2:9 dump.bin
```

*suche image signature in Bytes-Spannweite (offset)*

```
$ fim -C _seek_magic=GIF89a dump.bin
```

*lade nur nach passendem signature*

```
1 00000000: 6f66 6673 6574 0a47 4946 3839 610f 000f  offset.GIF89a...
2 00000010: 00b3 0e00 ffea 0045 4545 0000 00ff ce00  ....EEE.....
3 00000020: ffc9 00ff b400 fe9d 00ff fe93 fffd 13ff  ....
4 00000030: ffff ffff c733 3333 ffff ebff e500 0000  ....333.....
5 00000040: 0000 0000 21f9 0401 0000 0e00 2c00 0000  ....!.....
6 00000050: 000f 000f 0000 045b d049 196a 9dd8 55a6  ....[.I.j..U.
7 00000060: ce19 1716 7008 029c 8311 4c41 2708 27f0  ....p....LA'.'.
8 00000070: a6ab 7698 710e 6a0a 0e9f 3f82 2a70 03c6  ..v.q.j...?.*p..
9 00000080: 8285 ca21 c704 1092 01d3 a209 583c 2b03  ...!......X<+.
10 00000090: d96b bb15 ae02 5941 623c 1610 769a c1a0  .k....YAb<..v...
11 000000a0: c135 d358 6a82 5cfe 1605 0af8 c230 c3b2  .5.Xj.\.....0..
12 000000b0: d426 1100 3b                .&..;
```

p.s. das ist keine gute Steganografie ;-)