

Introduction to Unix for beginners

Ciprian-Virgil Gerstenberger

UiT The Arctic University of Norway

26.-30.06.2017 Tromsø



Introduction to Unix

Why command line?

Commands

And then?

Introduction to Unix

Why command line?

Commands

And then?

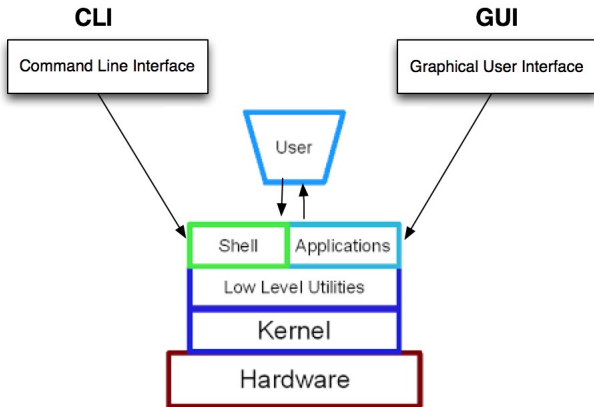
Introduction to Unix

Why command line?

Commands

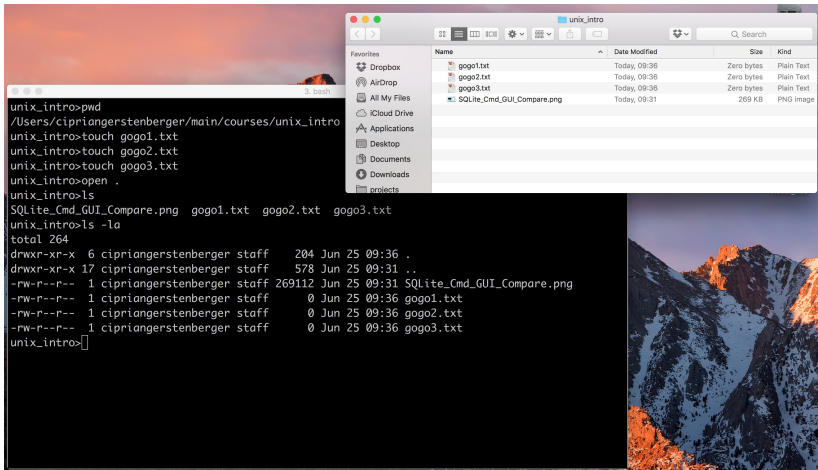
And then?

Human-Computer Interaction



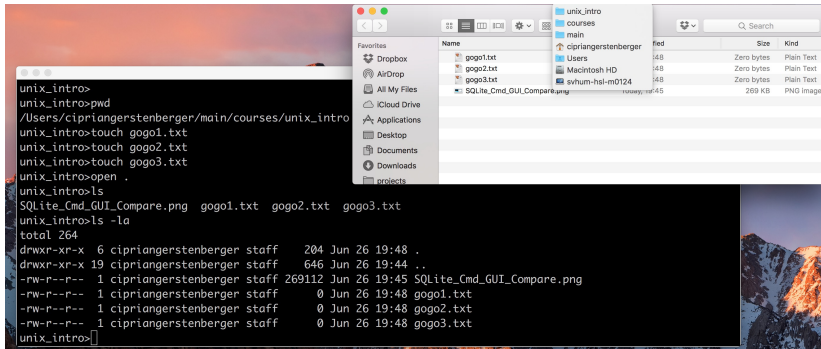
CLI vs. GUI

- ▷ location: **pwd** print working directory
- ▷ content: **ls** list directory contents



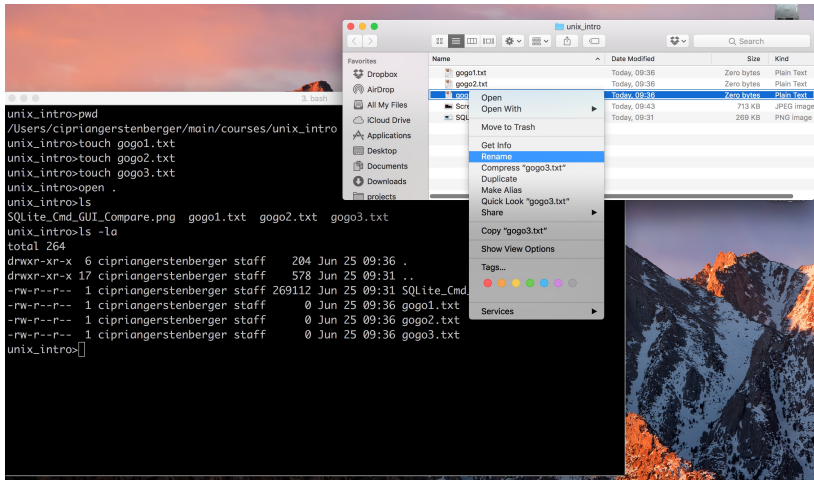
CLI vs. GUI

- ▶ location via GUI: **click on the current directory icon**
- ▶ content via GUI: **default**



CLI vs. GUI

▶ GUI: rename



CLI vs. GUI

▷ GUI: **gogo3.txt** → **blabla.txt**

The image displays a comparison between a Command Line Interface (CLI) and a Graphical User Interface (GUI) for file management. On the left, a terminal window shows the following commands and output:

```

unix_intro>pwd
/Users/cipriangerstenberger/main/courses/unix_intro
unix_intro>ls
SQLite_Cmd_GUI_Compare.png blabla.txt gogo1.txt gogo2.txt
unix_intro>ls -la
total 264
drwxr-xr-x  6 cipriangerstenberger  staff   204 Jun 25 09:49 .
drwxr-xr-x 19 cipriangerstenberger  staff   646 Jun 25 09:50 ..
-rw-r--r--  1 cipriangerstenberger  staff 269112 Jun 25 09:31 SQLite_Cmd_GUI_Compare.png
-rw-r--r--  1 cipriangerstenberger  staff    0 Jun 25 09:36 blabla.txt
-rw-r--r--  1 cipriangerstenberger  staff    0 Jun 25 09:36 gogo1.txt
-rw-r--r--  1 cipriangerstenberger  staff    0 Jun 25 09:36 gogo2.txt
unix_intro>

```

On the right, a file manager window titled 'unix_intro' shows a graphical view of the same directory. The files are listed in a table:

Name	Date Modified	Size	Kind
blabla.txt	Today, 09:36	Zero bytes	Plain Text
gogo1.txt	Today, 09:36	Zero bytes	Plain Text
gogo2.txt	Today, 09:36	Zero bytes	Plain Text
SQLite_Cmd_GUI_Compare.png	Today, 09:31	269 KB	PNG image

CLI vs. GUI

- ▷ CLI move: **mv gogo2.txt whatevername.txt**

The image shows a side-by-side comparison of a file system state. On the left is a terminal window with the following output:

```

unix_intro>pwd
/Users/cipriangerstenberger/main/courses/unix_intro
unix_intro>ls -la
total 264
drwxr-xr-x  6 cipriangerstenberger  staff   204 Jun 25 13:19 .
drwxr-xr-x 34 cipriangerstenberger  staff  1156 Jun 25 13:19 ..
-rw-r--r--  1 cipriangerstenberger  staff 269112 Jun 25 09:31 SQLite_Cmd_GUI_Compare.png
-rw-r--r--  1 cipriangerstenberger  staff   0 Jun 25 09:36 blabla.txt
-rw-r--r--  1 cipriangerstenberger  staff   0 Jun 25 09:36 gogol.txt
-rw-r--r--  1 cipriangerstenberger  staff   0 Jun 25 09:36 gogo2.txt
unix_intro>mv gogo2.txt whatever_name.txt
'gogo2.txt' -> 'whatever_name.txt'
unix_intro>ls
SQLite_Cmd_GUI_Compare.png blabla.txt gogol.txt whatever_name.txt
unix_intro>ls -la
total 264
drwxr-xr-x  6 cipriangerstenberger  staff   204 Jun 25 13:22 .
drwxr-xr-x 34 cipriangerstenberger  staff  1156 Jun 25 13:19 ..
-rw-r--r--  1 cipriangerstenberger  staff 269112 Jun 25 09:31 SQLite_Cmd_GUI_Compare.png
-rw-r--r--  1 cipriangerstenberger  staff   0 Jun 25 09:36 blabla.txt
-rw-r--r--  1 cipriangerstenberger  staff   0 Jun 25 09:36 gogol.txt
-rw-r--r--  1 cipriangerstenberger  staff   0 Jun 25 09:36 whatever_name.txt
unix_intro:~

```

On the right is a macOS Finder window titled 'unix_intro' showing the same directory contents in a graphical table view:

Name	Date Modified	Size	Kind
blabla.txt	Today, 09:36	Zero bytes	Plain Text
gogol.txt	Today, 09:36	Zero bytes	Plain Text
SQLite_Cmd_GUI_Compare.png	Today, 09:31	269 KB	PNG image
whatever_name.txt	Today, 09:36	Zero bytes	Plain Text

CLI vs. GUI: SQL

The image shows a side-by-side comparison of SQL execution. On the left is a Command Prompt window running SQLite, and on the right is the SQLite Database Browser GUI.

CLI Window (Left):

```

C:\Windows\system32\cmd.exe - sqlite3.exe f:/database/sqlite/GIS_test.db3
0!Line
1!Point
2!Raster
sqlite> insert into gis(ft_type) values ('polygon');
sqlite> select * from gis;
ft_id|ft_type
0!Line
1!Point
2!Raster
3!polygon
sqlite> insert into gis(ft_type) values ('line2');
sqlite> insert into gis(ft_type) values ('multiline');
sqlite> insert into gis(ft_type) values ('DEM');
sqlite> insert into gis(ft_type) values ('3DPolygon');
sqlite> select * from gis;
ft_id|ft_type
0!Line
1!Point
2!Raster
3!polygon
4!line2
5!multiline
6!DEM
7!3DPolygon
sqlite>
  
```

GUI Window (Right): SQLite Database Browser - F:/Database/SQLite/GIS_test.db3

Table: gis

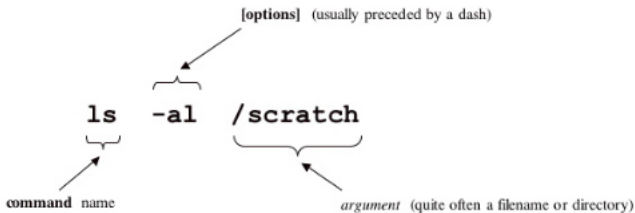
ft id	ft type
1	0 Line
2	1 Point
3	2 Raster
4	3 polygon
5	4 line2
6	5 multiline
7	6 DEM
8	7 3DPolygon
9	8

The GUI also shows a right-hand pane with "Import" and "Export" buttons, and a status bar at the bottom indicating "Type of data currently in cell: Text / Number" and "1 char".

Why command line?

- ▶ xfst, hfst, or vislcg3 do not have GUI
- ▶ rich collection of Unix text processing utilities

Basic syntax



▷ default argument:

ls .

ls

"Deictic elements" and paths

Three deictic elements

- ▶ current directory `.` mother directory `..` home directory `~`

Relative paths

```

6. bash
unix_intro>ls ./img
cli_vs_gui_00.jpg      cli_vs_gui_rename_03.jpg    cmd-anatomy.jpg  shell.png
cli_vs_gui_01.jpg      cli_vs_gui_rename_mv_04.jpg hci_01.gif       uit_new.png
cli_vs_gui_rename_02.jpg cli_vs_gui_sql.jpg         hci_01.png
unix_intro>
unix_intro>ls ../../xtdoc
LICENSE.txt  apachesetup  dicts  divvun  divvun.org  gtuit  gtuit2  sd2  techdoc
unix_intro>
unix_intro>ls ~/main/tools/alignment-tools
README.txt  europarl  python-tca2
unix_intro>

```

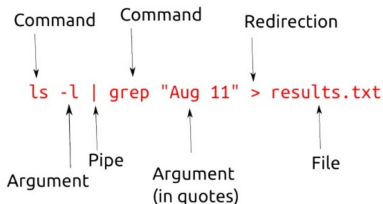
Absolute paths

```

6. bash
unix_intro>ls /Users/cipriangerstenberger/main/courses/unix_intro/img
cli_vs_gui_00.jpg      cli_vs_gui_rename_03.jpg    cmd-anatomy.jpg  shell.png
cli_vs_gui_01.jpg      cli_vs_gui_rename_mv_04.jpg hci_01.gif       uit_new.png
cli_vs_gui_rename_02.jpg cli_vs_gui_sql.jpg         hci_01.png
unix_intro>
unix_intro>ls /Users/cipriangerstenberger/main/xtdoc
LICENSE.txt  apachesetup  dicts  divvun  divvun.org  gtuit  gtuit2  sd2  techdoc
unix_intro>
unix_intro>ls /Users/cipriangerstenberger/main/tools/alignment-tools
README.txt  europarl  python-tca2
unix_intro>

```

Chaining commands, IO redirection



- ▷ pipe: output of cmdA is input of cmdB **cmdA | cmdB**
- ▷ IO redirection:
 - read from file **cmdA < file.txt**
 - write to file **cmdA > file.txt**
 - append to file **cmdA >> file.txt**

Man pages (1)

Command

- man

Purpose

- Gives you the manual page for a given command

Example

```
$ man pwd

pwd(1)
NAME
    pwd - print working directory name
SYNOPSIS
    pwd
DESCRIPTION
    pwd prints the path name of the working (current)
    directory.  pwd is both an explicit command (invoked as
    /usr/bin/pwd), as well as a builtin
```

Man pages (2)

```

unix_intro>man man

```

```

6. bash
man(1) man(1)

NAME
    man - format and display the on-line manual pages

SYNOPSIS
    man [-acdfFhkKtW] [--path] [-m system] [-p string] [-C config_file]
    [-M pathlist] [-P pager] [-B browser] [-H htmlpager] [-S section_list]
    [section] name ...

DESCRIPTION
    man formats and displays the on-line manual pages.  If you specify sec-
    tion, man only looks in that section of the manual.  name is normally
    the name of the manual page, which is typically the name of a command,
    function, or file.  However, if name contains a slash (/) then man
    interprets it as a file specification, so that you can do man ./foo.5
    or even man /cd/foo/bar.1.gz.

    See below for a description of where man looks for the manual page
    files.

OPTIONS
    -C config_file
        Specify the configuration file to use; the default is /pri-

```

Man pages (3)

```

6. sh
TIPS
If you add the line

    (global-set-key [(f1)] (lambda () (interactive) (manual-entry (current-word))))

to your .emacs file, then hitting F1 will give you the man page for the
library call at the current cursor position.

To get a plain text version of a man page, without backspaces and
underscores, try

    # man foo | col -b > foo.mantxt

AUTHOR
John W. Eaton was the original author of man. Zeyd M. Ben-Halim
released man 1.2, and Andries Brouwer followed up with versions 1.3
thru 1.5p. Federico Lucifredi <flucifredi@acm.org> is the current
maintainer.

SEE ALSO
apropos(1), whatis(1), less(1), groff(1), man.conf(5).

September 19, 2005                                man(1)
(END)
```

man vislcg3

```
unix_intro>man vislcg3
unix_intro>
```

```
vislcg3(1) vislcg3(1)
NAME
    vislcg3 - This application is part of ( cg3 )

    This tool is part of the CG-3 constraint grammar system: http://visl.sdu.dk/cg3.html.

SYNOPSIS
    vislcg3 [OPTIONS]

DESCRIPTION
    vislcg3 is the grammar parser and disambiguator in the VISL constraint grammar system. It chooses between
    different analyses (or "readings") of a word delivered by a morphological analyser based on a set of gram-
    matical rules.

    By default, vislcg3 reads input from stdin and writes output to stdout.

OPTIONS
    -h, --help
        shows this help

    -?, --?
        shows this help
```

which vislcg3

```
unix_intro>ls
SQLite_Cmd_GUI_Compare.png  blabla.txt  gogo1.txt  whatever_name.txt
unix_intro>which vislcg3
/opt/local/bin/vislcg3
unix_intro>|
```

- ▶ absolute path of the installed programm

apropos vislcg3

```

6. bash
unix_intro>apropos visl
cg-comp(1) - This application is part of ( cg3 ) This tool is part of the CG-3 constraint grammar system: http://v
isl.sdu.dk/cg3.html
cg-conv(1) - This application is part of ( cg3 ) This tool is part of the CG-3 constraint grammar system: http://v
isl.sdu.dk/cg3.html
cg-proc(1) - This application is part of ( cg3 ) This tool is part of the CG-3 constraint grammar system: http://v
isl.sdu.dk/cg3.html
cg3-autobin.pl(1), cg-autobin.pl(1) - This application is part of ( cg3 ) This tool is part of the CG-3 constraint gramme
m: http://visl.sdu.dk/cg3.html
vislcg3(1) - This application is part of ( cg3 ) This tool is part of the CG-3 constraint grammar system: http://v
isl.sdu.dk/cg3.html
unix_intro>apropos vislcg3
vislcg3(1) - This application is part of ( cg3 ) This tool is part of the CG-3 constraint grammar system: http://v
isl.sdu.dk/cg3.html
unix_intro>

```

▶ the more specific search the more specific results

man lookup, apropos lookup

```

6. bash
unix_intro>ls
SQLite_Cmd_GUI_Compare.png  blabla.txt  gogo1.txt  whatever_name.txt
unix_intro>man lookup
No manual entry for lookup
unix_intro>

```

```

6. sh
unix_intro>
unix_intro>apropos lookup
Config::Extensions(3) - hash lookup of which core extensions were built
hfst-lookup(1) - =perform transducer lookup (apply)
hfst-optimized-lookup(1) - =Usage: hfst-optimized-lookup [OPTIONS] TRANSDUCER
hfst-pmatch(1) - =perform matching/lookup on text streams
hfst-tokenize(1) - =perform matching/lookup on text streams
CURLINFO_NAMELOOKUP_TIME(3) - get the name lookup time
Config::Extensions(3pm) - hash lookup of which core extensions were built
IP::Authority(3pm) - fast lookup of authority by IP address
IP::Country(3pm) - fast lookup of country codes from IP addresses
IP::Country::Fast(3pm) - fast lookup of country codes by IP address
IP::Country::Medium(3pm) - cached lookup of country codes by IP address and domain name
IP::Country::Slow(3pm) - cached lookup of country codes by domain name and IP address
ModPerl::MethodLookup(3pm) - -- Lookup mod_perl modules, objects and methods
Net::DNS::Resolver::Recurse(3pm) - Perform recursive dns lookups
SSL_want(3ssl), SSL_want_nothing(3ssl), SSL_want_read(3ssl), SSL_want_write(3ssl), SSL_want_x509_lookup(3ssl) - obtain state
information TLS/SSL I/O operation
Tcl_GetIndexFromObj(3tcl), Tcl_GetIndexFromObjStruct(3tcl) - lookup string in table of keywords
X509_NAME_get_index_by_NID(3ssl), X509_NAME_get_index_by_OBJ(3ssl), X509_NAME_get_entry(3ssl), X509_NAME_entry_count(3ssl),
AME_get_text_by_NID(3ssl), X509_NAME_get_text_by_OBJ(3ssl) - X509_NAME lookup and enumeration functions
dig(1) - DNS lookup utility

```

<TAB><TAB> and -h(elp) to the rescue

```

unix_intro>
unix_intro>look
look      lookbib      lookup      lookup.perl  lookup2cg
unix_intro>lookup
*** Calling problem. Help:  lookup -h
unix_intro>lookup -h
USAGES:
  lookup -h
              (HELP)
  lookup -v
              (version)
... | lookup <lexicon_file> [ options ] | ...
              a single lexical fst defined explicitly
... | lookup -l <language> [ options ] | ...
              default fst-s and strategies for a given language
... | lookup -f <lookup_script> [ options ] | ...
              one or more strategies - lists of cascaded fst-s
OPTIONS:
  [-flags <LOOKUP_FLAGS-values>]
  The environment variable LOOKUP_FLAGS can also be specified.
  [-flags <yourflags>] > LOOKUP_FLAGS > default values (see below).
  c - print all comments;
  n - at [-a 1]: result net for the lookup of every word;
  k# - at [-a 1]: checknet if less then # states;

```

```

unix_intro>which lookup
/Users/cipriangerstenberger/local/bin/lookup
unix_intro>

```

Environment variables

```
unix_intro>echo $LOOKUP
lookup -q -flags mbTT
unix_intro>
unix_intro>echo $GTHOME
/Users/cipriangerstenberger/main
unix_intro>
```

```
export GTHOME=/Users/cipriangerstenberger/main
```

- ▶ declared in ~/.profile or ~/.bashrc

Searching string patterns in files: grep

```

unix_korpus>grep 'ara' *txt
kivi.txt:Heidän isänsä, joka oli ankan innokas metsämies, kohtasi hänen
kivi.txt:SIMEONI. Hyvin tietysti. Sentähden on parasta mennä ottamaan se sauna ja
kivi.txt:peloittavan ankara mies. Liioinkin laiskalukijoille oli hän armoton,
kivi.txt:lukemaan.--Muistellen näitä seikkoja, istuessaan kotona avarassa tuvassa
kivi.txt:SIMEONI. Jumala paratkoon! hillitön, villitty on elämämme ollut aina
kivi.txt:JUHANI. Parastani tahdon koettaa. Mutta kun vaan tottelisitte ilman
kivi.txt:könistämistä ja ruoskaa! Mutta parastani tahdon koettaa.
kivi.txt:TIMO. Viinaa vaan, ei sen kalliimpaa tavaraa. Niin, mistähän meille
kivi.txt:kohden maamme onnea ja parasta. Hänenpä uljaasta, väsymättömästä
kivi.txt:mutta aivan avaralle kuitenkin ei ulettunut hänen emännyytensä valta.
kivi.txt:mittasi tavarat sekä väkensä että karjansa tarpeiksi ja maksoi niin
kivi.txt:vilinätä Jukolan avarassa tuvassa, koska lapsia ankara joukko
kivi.txt:Impivaaran kantoisella aholla, komeroisen, kohisevan vuoren alla. Ja
sme_vaalit2012.txt:Veahkadatregistarguovddáš ja magistráhta dollet jienastanvuogiatvuoharegistar, mas leat

```

- ▶ **wildcards:** "*", ".", "?" (quantification in regular expressions)
- ▶ **masking characters:** "\" (quote vs. use)

Using flags: `grep -i -color="auto"`

```

unix_korpus>ls
eng_vaalit2012.txt kivi.txt rus_vaalit2012.txt smn_nouns.lexc sms_nouns.lexc
fin_vaalit2012.txt kursplana.jspwiki sme_vaalit2012.txt smn_vaalit2012.txt
unix_korpus>g 'ara' *txt
kivi.txt:Heidän isäänsä, joka oli ankara innokas metsämies, kohtasi hänen
kivi.txt:SIMEONI. Hyvin tietysti. Sentähden on parasta mennä ottamaan se sauna ja
kivi.txt:peloittavan ankara mies. Liioinkin laiskalukijoille oli hän armoton,
kivi.txt:lukemaan.--Muistellen näitä seikkoja, istuessaan kotona avaassa tuvassa
kivi.txt:SIMEONI. Jumala paratsoon! hillitön, villitty on elämämme ollut aina
kivi.txt:JUHANI. Parastani tahdon koettaa. Mutta kun vaan tottelisitte ilman
kivi.txt:könistämistä ja ruoskaa! Mutta parastani tahdon koettaa.
kivi.txt:TIMO. Viinaa vaan, ei sen kalliimpaa tavaraa. Niin, mistähän meille
kivi.txt:kohden maamme onnea ja parasta. Hänenpä uljaasta, väsymättömästä
kivi.txt:mutta aivan avaralle kuitenkin ei ulettunut hänen emännyytensä valta.
kivi.txt:mittasi tavarat sekä väkensä että karjansa tarpeiksi ja maksoi niin
kivi.txt:vilinätä Jukolan avaassa tuvassa, koska lapsia ankara joukko
kivi.txt:Impivaaran kantoisella aholla, komeroisen, kohisevan vuoren alla. Ja
sme_vaalit2012.txt:Veahkadatregistarguovddáš ja magistráhta dollet jienastanvuogatvuoharegistara, mas leat
unix_korpus>

```

- ▷ better output
- ▷ What does `g` mean?

Aliasing

- ▶ **save time by typing less:**
compare to environment variable declaration

```

6. bash
unix_intro>alias g
alias g='grep -i --color='\''auto'\''
unix_intro>alias v
alias v='vim'
unix_intro>

```

```

alias smeik="pushd $GTHOME/langs/sme; make clean GTLANG=sme; make GTLANG=sme; make abbr GTLANG=sme; popd"

alias d="cd .."
alias l="less"
alias g="grep -i --color='auto'"
alias s="sort"
alias snr="sort -nr"
alias u="uniq"
alias uc="uniq -c"
alias xl="xmllint -noout"
alias t="sluclsnr"
alias tl="sluclsnr|l"
alias z="cut -d '>' -f2|cut -d '<' -f1"
alias zz="cut -d '<' -f2|cut -d '>' -f2"
alias ql="qlmanage -p"

alias rm='rm -iv'
alias cp='cp -iv'
alias cp_brutal='cp -v'
alias mv='mv -iv'
alias v='vim'
alias y='yes'

```

More aliasing: usma, usme

- ▷ **u=upper side**
- ▷ **sma=South Saami**

```

6. lookup
unix_intro>alias usma
alias usma='$LOOKUP $GTHOME/langs/sma/src/analyser-gt-desc.xfst'
unix_intro>usma
aajhtsedh
aajhtsedh      aajhtsedh+V+TV+Der/d+V+ConNeg
aajhtsedh      aajhtsedh+V+TV+Der/d+V+Ind+Prs+Sg2
aajhtsedh      aajhtsedh+V+TV+Der/d+V+Imprt+Sg2
aajhtsedh      aajhtsedh+V+TV+Inf

```


More aliasing: usme

```
6. bash
unix_intro>alias usme
alias usme='$LOOKUP $GTHOME/langs/sme/src/analyser-gt-desc.xfst'
unix_intro>usme
ERROR in 'LOOKUP' :
Lexicon file '/Users/cipriangerstenberger/main/langs/sme/src/analyser-gt-desc.xfst' could not be found or opened
unix_intro>|
```

⇒ **Finally an error!**

Coping with errors: usme

- ▶ understand it
- ▶ correct it

```
6. bash
unix_intro>alias usme
alias usme='$LOOKUP $GTHOME/langs/sme/src/analyser-gt-desc.xfst'
unix_intro>usme
ERROR in 'LOOKUP' :
Lexicon file '/Users/cipriangerstenberger/main/langs/sme/src/analyser-gt-desc.xfst' could not be found or opened
unix_intro>|
```

⇒ generate the analyser-gt-desc.xfst file

Coping with errors: vislcg3

```
unix_intro>
unix_intro>vis
vis      vislcg3 visudo
unix_intro>vislcg3
Error: No grammar specified - cannot continue!
Usage: vislcg3 [OPTIONS]

Options:
  -h, --help                shows this help
  -?, --?                   shows this help
  -V, --version             prints copyright and version information
  --min-binary-revision    prints the minimum usable binary grammar revision
  -g, --grammar             specifies the grammar file to use for disambiguation
  --grammar-out            writes the compiled grammar in textual form to a file
  --grammar-bin            writes the compiled grammar in binary form to a file
  --grammar-only           only compiles the grammar; implies --verbose
  --ordered                (will in future allow full ordered matching)
```

⇒ give the grammar file as argument to vislcg3

Quick reference: cheat sheet 1

Bash Commands	Bash Variables	Command Lists
<code>uname -a</code> Show system and kernel	<code>env</code> Show environment variables	<code>cmd1 ; cmd2</code> Run <code>cmd1</code> then <code>cmd2</code>
<code>head -n1 /etc/issue</code> Show distribution	<code>echo \$NAME</code> Output value of <code>\$NAME</code> variable	<code>cmd1 && cmd2</code> Run <code>cmd2</code> if <code>cmd1</code> is successful
<code>mount</code> Show mounted filesystems	<code>export NAME=value</code> Set <code>\$NAME</code> to value	<code>cmd1 cmd2</code> Run <code>cmd2</code> if <code>cmd1</code> is not successful
<code>date</code> Show system date	<code>\$PATH</code> Executable search path	<code>cmd &</code> Run <code>cmd</code> in a subshell
<code>uptime</code> Show uptime	<code>\$HOME</code> Home directory	
<code>whoami</code> Show your username	<code>\$SHELL</code> Current shell	
<code>man command</code> Show manual for command		
Bash Shortcuts	IO Redirection	Directory Operations
<code>CTRL-c</code> Stop current command	<code>cmd < file</code> Input of <code>cmd</code> from file	<code>pwd</code> Show current directory
<code>CTRL-z</code> Sleep program	<code>cmd1 <(cmd2)</code> Output of <code>cmd2</code> as file input to <code>cmd1</code>	<code>mkdir dir</code> Make directory <code>dir</code>
<code>CTRL-a</code> Go to start of line	<code>cmd > file</code> Standard output (stdout) of <code>cmd</code> to file	<code>cd dir</code> Change directory to <code>dir</code>
<code>CTRL-e</code> Go to end of line	<code>cmd > devnull</code> Discard stdout of <code>cmd</code>	<code>cd ..</code> Go up a directory
<code>CTRL-u</code> Cut from start of line	<code>cmd >> file</code> Append stdout to file	<code>ls</code> List files
<code>CTRL-k</code> Cut to end of line	<code>cmd 2> file</code> Error output (stderr) of <code>cmd</code> to file	
<code>CTRL-r</code> Search history	<code>cmd 1-&2</code> stdout to same place as <code>stderr</code>	ls Options
<code>!!</code> Repeat last command	<code>cmd 2-&1</code> stderr to same place as <code>stdout</code>	<code>-a</code> Show all (including hidden)
<code>!abc</code> Run last command starting with <code>abc</code>	<code>cmd &> file</code> Every output of <code>cmd</code> to file	<code>-R</code> Recursive list
<code>!abcp</code> Print last command starting with <code>abc</code>	<code>cmd</code> refers to a command.	<code>-r</code> Reverse order
<code>!\$</code> Last argument of previous command		<code>-t</code> Sort by last modified
<code>ALT-.</code> Last argument of previous command	Pipes	<code>-S</code> Sort by file size
<code>!*</code> All arguments of previous command	<code>cmd1 cmd2</code> stdout of <code>cmd1</code> to <code>cmd2</code>	<code>-l</code> Long listing format
<code>*abc^123</code> Run previous command, replacing <code>abc</code> with <code>123</code>	<code>cmd1 & cmd2</code> stderr of <code>cmd1</code> to <code>cmd2</code>	<code>-1</code> One file per line
		<code>-m</code> Comma-separated output
		<code>-O</code> Quoted output
		Search Files
		<code>grep pattern files</code> Search for pattern in files
		<code>grep -i</code> Case insensitive search
		<code>grep -r</code> Recursive search
		<code>grep -v</code> Inverted search
		<code>grep -o</code> Show matched part of file only
		<code>find /dir-name name*</code> Find files starting with name in dir

Quick reference: cheat sheet 2

File Commands	System Info
<p>ls - directory listing ls -al - formatted listing with hidden files cd <i>dir</i> - change directory to <i>dir</i> cd - change to home pwd - show current directory mkdir <i>dir</i> - create a directory <i>dir</i> rm <i>file</i> - delete <i>file</i> rm -r <i>dir</i> - delete directory <i>dir</i> rm -f <i>file</i> - force remove <i>file</i> rm -rf <i>dir</i> - force remove directory <i>dir</i> * cp <i>file1 file2</i> - copy <i>file1</i> to <i>file2</i> cp -r <i>dir1 dir2</i> - copy <i>dir1</i> to <i>dir2</i>; create <i>dir2</i> if it doesn't exist mv <i>file1 file2</i> - rename or move <i>file1</i> to <i>file2</i> if <i>file2</i> is an existing directory, moves <i>file1</i> into directory <i>file2</i> ln -s <i>file link</i> - create symbolic link <i>link</i> to <i>file</i> touch <i>file</i> - create or update <i>file</i> cat > <i>file</i> - places standard input into <i>file</i> more <i>file</i> - output the contents of <i>file</i> head <i>file</i> - output the first 10 lines of <i>file</i> tail <i>file</i> - output the last 10 lines of <i>file</i> tail -f <i>file</i> - output the contents of <i>file</i> as it grows, starting with the last 10 lines</p>	<p>date - show the current date and time cal - show this month's calendar uptime - show current uptime w - display who is online whoami - who you are logged in as finger <i>user</i> - display information about <i>user</i> uname -a - show kernel information cat /proc/cpuinfo - cpu information cat /proc/meminfo - memory information man <i>command</i> - show the manual for <i>command</i> df - show disk usage du - show directory space usage free - show memory and swap usage whereis <i>app</i> - show possible locations of <i>app</i> which <i>app</i> - show which <i>app</i> will be run by default</p>
Process Management	Compression
<p>ps - display your currently active processes top - display all running processes kill <i>pid</i> - kill process id <i>pid</i> killall <i>proc</i> - kill all processes named <i>proc</i> * bg - lists stopped or background jobs; resume a stopped job in the background fg - brings the most recent job to foreground fg <i>n</i> - brings job <i>n</i> to the foreground</p>	<p>tar cf <i>file.tar files</i> - create a tar named <i>file.tar</i> containing <i>files</i> tar xf <i>file.tar</i> - extract the files from <i>file.tar</i> tar czf <i>file.tar.gz files</i> - create a tar with Gzip compression tar xzf <i>file.tar.gz</i> - extract a tar using Gzip tar cjf <i>file.tar.bz2</i> - create a tar with Bzip2 compression tar xjf <i>file.tar.bz2</i> - extract a tar using Bzip2 gzip <i>file</i> - compresses <i>file</i> and renames it to <i>file.gz</i> gzip -d <i>file.gz</i> - decompresses <i>file.gz</i> back to <i>file</i></p>
File Permissions	Network
<p>chmod <i>octal file</i> - change the permissions of <i>file</i> to <i>octal</i>, which can be found separately for user, group, and world by adding:</p> <ul style="list-style-type: none"> 4 - read (r) 2 - write (w) 1 - execute (x) <p>Examples: chmod 777 - read, write, execute for all chmod 755 - rwx for owner, rx for group and world For more options, see man chmod.</p>	<p>ping <i>host</i> - ping <i>host</i> and output results whois <i>domain</i> - get whois information for <i>domain</i> dig <i>domain</i> - get DNS information for <i>domain</i> dig -x <i>host</i> - reverse lookup <i>host</i> wget <i>file</i> - download <i>file</i> wget -c <i>file</i> - continue a stopped download</p>
SSH	Installation
<p>ssh <i>user@host</i> - connect to <i>host</i> as <i>user</i> ssh -p <i>port user@host</i> - connect to <i>host</i> on port <i>port</i> as <i>user</i> ssh-copy-id <i>user@host</i> - add your key to <i>host</i> for <i>user</i> to enable a keyed or passwordless login</p>	<p>Install from source: ./configure make make install dpkg -i <i>pkg.deb</i> - install a package (Debian) rpm -Uvh <i>pkg.rpm</i> - install a package (RPM)</p>
Searching	Shortcuts
<p>grep <i>pattern files</i> - search for <i>pattern</i> in <i>files</i> grep -r <i>pattern dir</i> - search recursively for</p>	<p>Ctrl+C - halts the current command Ctrl+Z - stops the current command, resume with fg in the foreground or bg in the background Ctrl+D - log out of current session, similar to exit Ctrl+W - erases one word in the current line Ctrl+U - erases the whole line Ctrl+R - type to bring up a recent command !! - repeats the last command</p>

Quick reference: cheat sheet 3

File commands

```
ls          Directory listing
ls -al     Formatted listing with hidden files
cd dir     Change directory to dir
cd         Change to home
pwd        Show current directory
mkdir dir  Create a directory
rm file    Delete file
rm -r dir  Delete directory dir
Force remove file
rm -f file
rm -rf dir Force remove directory dir
cp file1 file2 Copy file1 to file2
cp -r dir1 dir2 Copy dir1 to dir2, create dir2 if it doesn't exist
mv file1 file2 Rename or move file1 to file2, if file2 is an existing directory, moves file1 into directory. file2
ln -s file link Create symbolic link link to file
touch file Create or update file
cat > file Places standard input into file
more file Output the contents of file
head file Output the first 10 lines of file
tail file Output the last 10 lines of file
tail -f file Output the contents of file as it grows, starting with the last 10 lines
```

Process Management

```
ps          display all currently active processes
top         display all running processes
kill pid   kill process id pid
killall proc kill all processes named proc *
bg         lists stopped or background jobs; resume a stopped job in the background
fg         brings the most recent job to the foreground
fg a       brings job a to the foreground
```

File Permissions

chmod octal file change the permissions of file to octal, which can be found separately for user, group, and world by adding:

- 4 – read (r)
- 2 – write (w)
- 1 – execute (x)

Examples:
chmod 777 – read, write, execute for all
chmod 755 – rwx for owner, rx for group and world. For more options, see [man chmod](#).

SSH

```
ssh user@host connect to host as user
ssh -p port user@host connect to host on port port as user
ssh-copy-id user@host add your key to host for user to enable a keyed or passwordless login
```

Searching

```
grep pattern files search for pattern in files
grep -r pattern dir search recursively for pattern in dir
command | grep pattern search for pattern in the output of command
locate file find all instances of file
```

System info

```
date        show the current date and time
cal         show this month's calendar
uptime     show current uptime
w          display who is online
whoami     who you are logged in as
finger user display information about user
uname -a   show kernel information
cat /proc /cpuinfo cpu information
cat /proc /meminfo memory information
man command show the manual for command
du         show disk usage
du -sh     show directory space usage
df         show memory and swap usage
whereis app show possible locations of app
which app  show which app will be run by default
```

Compression

```
tar cf file.tar files create a tar named file.tar containing files
tar xf file.tar files extract the files from file.tar
tar czf file.tar.gz files create a tar with Gzip compression
tar xzf file.tar.gz extract a tar using Gzip
tar cJf file.tar.bz2 create a tar with Bzip2 compression
tar xJf file.tar.bz2 extract a tar using Bzip2
gzip file compresses file and renames it to file.gz
gzip -d file.gz decompresses file.gz back to file
```

Network

```
ping host ping host and output results
whois domain get whois information for domain
dig domain get DNS information for domain
dig -x host reverse lookup host
wget file download file
wget -c file continue a stopped download
```

Installation

Install from source:

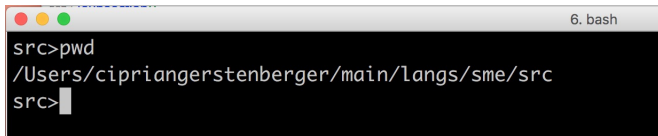
```
./configure
make
make install
dpkg -i pkg.deb install a package (Debian)
rpm -Uvh pkg.rpm install a package (RPM)
```

Shortcuts

```
Ctrl+C     halts the current command
Ctrl+Z     stops the current command, resume with fg in the foreground or bg in the background
Ctrl+D     log out of current session, similar to exit
Ctrl+W     erases one word in the current line
Ctrl+U     erases the whole line
Ctrl+R     type to bring up a recent command
!!         repeats the last command
!          log out of current session
*         use with extreme caution
```

Where am I now?

- ▶ **pwd** print working directory

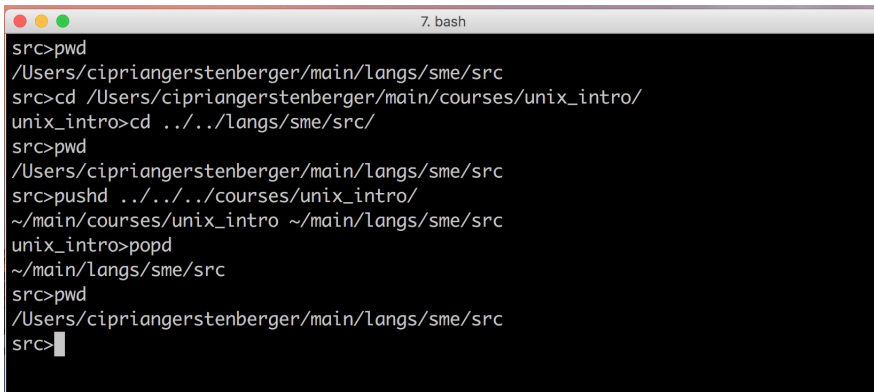


```
src>pwd
/Users/cipriangerstenberger/main/langs/sme/src
src>
```

A terminal window titled "6. bash" with a dark background. The prompt "src>" is followed by the command "pwd". The output is the full path "/Users/cipriangerstenberger/main/langs/sme/src". The prompt "src>" is followed by a white cursor bar.

How to navigate between directories?

- ▶ change directory: **cd PATH-TO-DIRECTORY-B**
- ▶ push directory: **pushd PATH-TO-DIRECTORY-B**
- ▶ pop directory: **popd**



```
7. bash
src>pwd
/Users/cipriangerstenberger/main/langs/sme/src
src>cd /Users/cipriangerstenberger/main/courses/unix_intro/
unix_intro>cd ../../langs/sme/src/
src>pwd
/Users/cipriangerstenberger/main/langs/sme/src
src>pushd ../../../../courses/unix_intro/
~/main/courses/unix_intro ~/main/langs/sme/src
unix_intro>popd
~/main/langs/sme/src
src>pwd
/Users/cipriangerstenberger/main/langs/sme/src
src>
```


Push it as far as you want!

```
7. bash
src>pushd ../../../../courses/unix_intro/
~/main/courses/unix_intro ~/main/langs/sme/src
unix_intro>pushd ~
~ ~/main/courses/unix_intro ~/main/langs/sme/src
~>pushd main/plan/
~/main/plan ~ ~/main/courses/unix_intro ~/main/langs/sme/src
plan>popd
~ ~/main/courses/unix_intro ~/main/langs/sme/src
~>popd
~/main/courses/unix_intro ~/main/langs/sme/src
unix_intro>popd
~/main/langs/sme/src
src>popd
-bash: popd: directory stack empty
src>|
```

▷ stack: **FILO (first in last out)**

Practice, practice, practice

- ▶ Which command type do you need?
- ▶ Where to find information about it?
- ▶ How to interpret error messages?

**You are the Hercule Poirot or
the Miss Marple of a Unix mystery!**