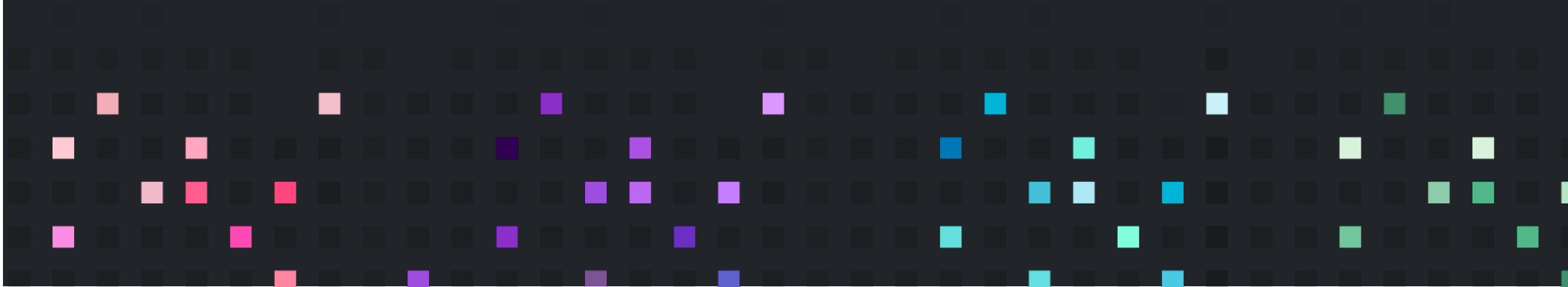


Linux konsollen

28. Maj 2024

Mikkel Holm Olsen



Kommandolinjen

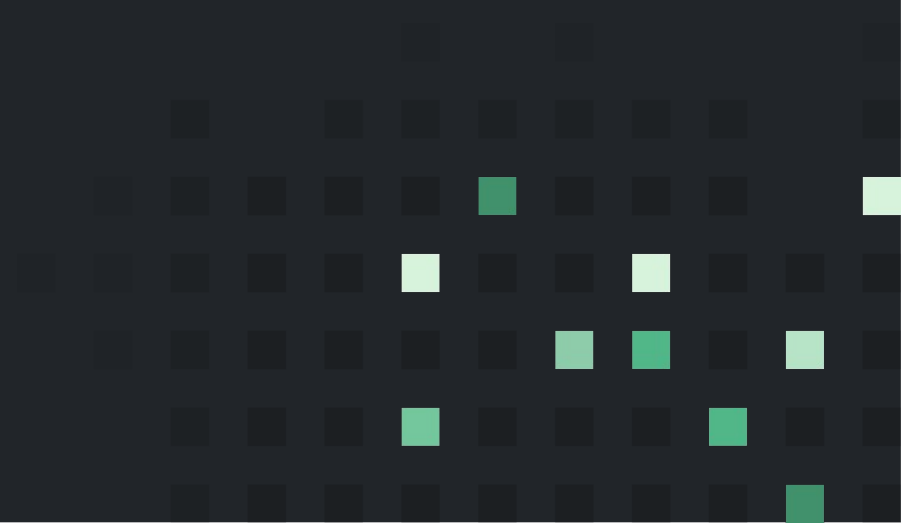
- Består af op til 3 dele
 - Kommando (shell-builtin eller executable)
 - Options (starter med - eller --)
 - Øvrige argumenter (fx filnavne eller lign.)

Kommando

- Shell built-ins
 - Fx cd, pwd, echo
- Executables (programmer)
 - Hvis der ikke angives en sti, så ledes i alle dirs nævnt i \$PATH
 - Der kan angives en absolut sti (fx /bin/bash)
 - Eller relativ (fx ./mit_program)
 - Der skal være execute-rettigheder

Kommandoer vi har brugt

- ls
- cat, touch
- grep, sort, uniq, rev
- cp, mv, mkdir, rmdir
- man
- nano



Options (optional arguments)

- De fleste programmer forventer options (der starter med '-') før andre argumenter (som fx filnavne)
- Short options er oftest med en '-', long options er med to (fx '--version')
- Short options kan som regel "bundles":
'ls -l -a' er det samme som 'ls -la'

Options - fortsat

- Desværre er der undtagelser:
 - Hvis vi fx har et filnavn, der starter med '-', så bliver programmet snydt og tror det er en option
 - For mange programmer kan man bruge '--' til at stoppe option-processering (fx 'ls -- -a')
 - Man kan også angive filen med en sti, fx 'rm ./-i'
 - Eksempel: touch -- -la ; ls *

Argumenter (non-option)

- Kan fx være filnavne (se 'man cat':
cat [OPTION]... [FILE]...)
- Kan også være andet ('./level_02 ostemad')
- Kommer ind i programmet som en liste af strenge (shell opfatter mellemrum som sep.)
- Hvis filnavn indeholder mellemrum må vi skrive "min fil", 'min fil' eller min\ fil

Shell-expansion

- Shell expander nogle udtryk, fx 'echo *'
 - * ekspanderer til alle filer i cwd (undt. dot-filer)
 - ? ekspanderer til en karakter i filnavne i cwd
 - [ace] ekspanderer til bogstaverne 'a', 'c' og 'e' i filnavne i cwd
 - {abc,def,ghi} ekspanderer til strengene 'abc', 'def' og 'ghi', behøver *ikke* at være filer
 - Der ekspanderes ikke imellem enkeltping (')

Output-redirection

- Output (std-out) kan omdirrigeres til en fil med '>', fx 'echo hej > min_flotte_fil'
- Hvis der bruges '>>' appendes (tilføjes) til filen
- Man kan også omdirrigere stderr med '2>', fx: 'find . 2>/dev/null'

Input-redirection

- Input (std-in) kan læses fra en fil i stedet for konsollen. Det gøres med '<', fx:
'rev < ost' (læser filen ost som input til rev)

Pipes

- Med en pipe forbindes std-out fra et program til std-in på et andet program, fx:
`'echo hej | rev'`
 - Her køres programmet `echo`, der udskriver `'hej'` på std-out, som sendes ind i `rev` (på std-in)
 - Efter `|`-tegnet (pipe) skal der gives et program (en executable i `$PATH`, eller med en angiven sti)

Pipes (fortsat)

- Programmet efter pipe kan få argumenter, fx options: `'ls -la | grep -i bash'`
- Man kan bruge flere pipes i samme kommandolinje, evt. afsluttet med redirection `'du -sm * | sort -nr | head -n 5 >top5.txt'`



Spørgsmål?

28. Maj 2024

Mikkel Holm Olsen