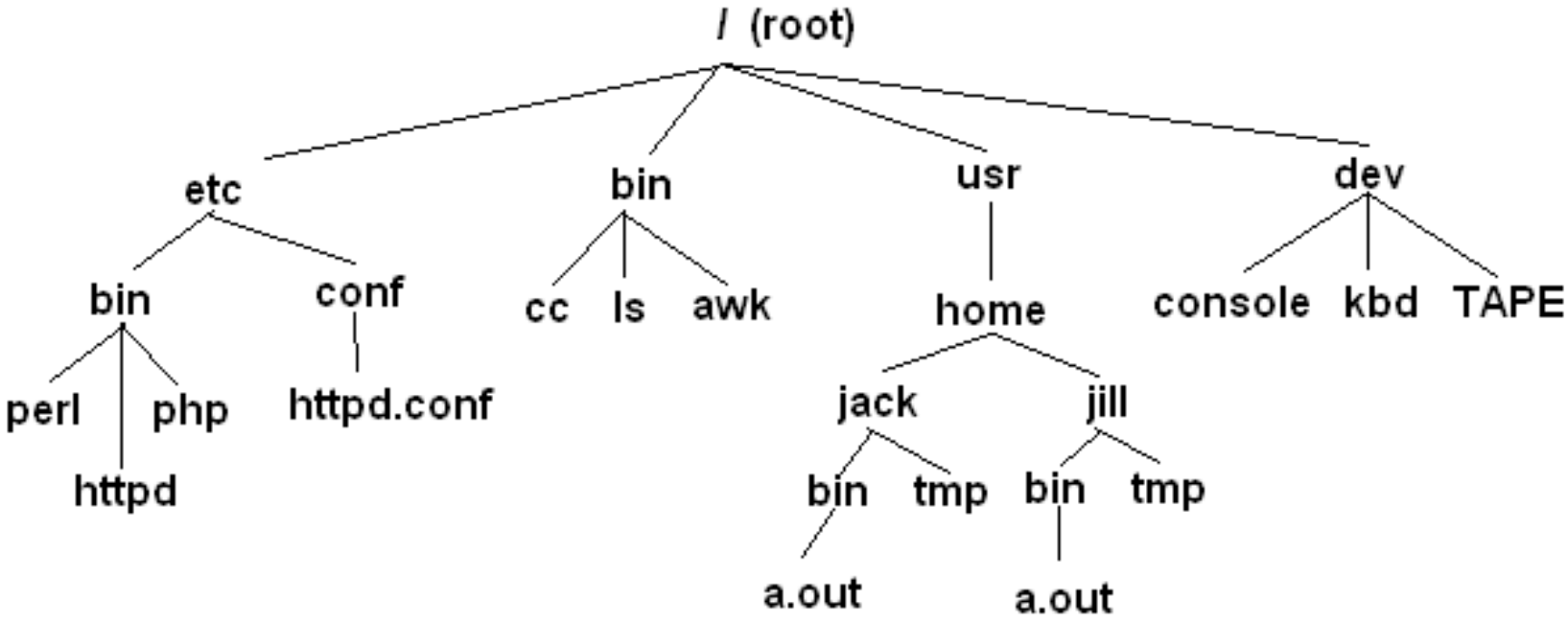
Introduction to UNIX, vol II

Kamil S Jaron, Marc Robinson-Rechavi

23.9. 2016

UNIX organisation



Variables

```
$VAR=5

$echo VAR
VAR

$echo $VAR
5
```

Environmental variables

```
$echo $USER
kjaron

# terminal prompt
$echo $PS1
\u@\h:\w$

# locations of binaries
$echo $PATH
/usr/local/sbin:/usr/local/bin:
/usr/sbin:/usr/bin:/sbin:/bin
```

Scripts

automatisation of your actions

```
$cat who_am_I.sh
#!/bin/bash

echo $USER



$bash who_am_I.sh
kjaron
```

```
$cat echo.sh
#!/bin/bash

echo $1



$bash echo.sh first second
first
```

```
$cat parameter_overview.sh
#!/bin/bash

echo $#
echo $1
echo $2
echo $*
echo $0
```

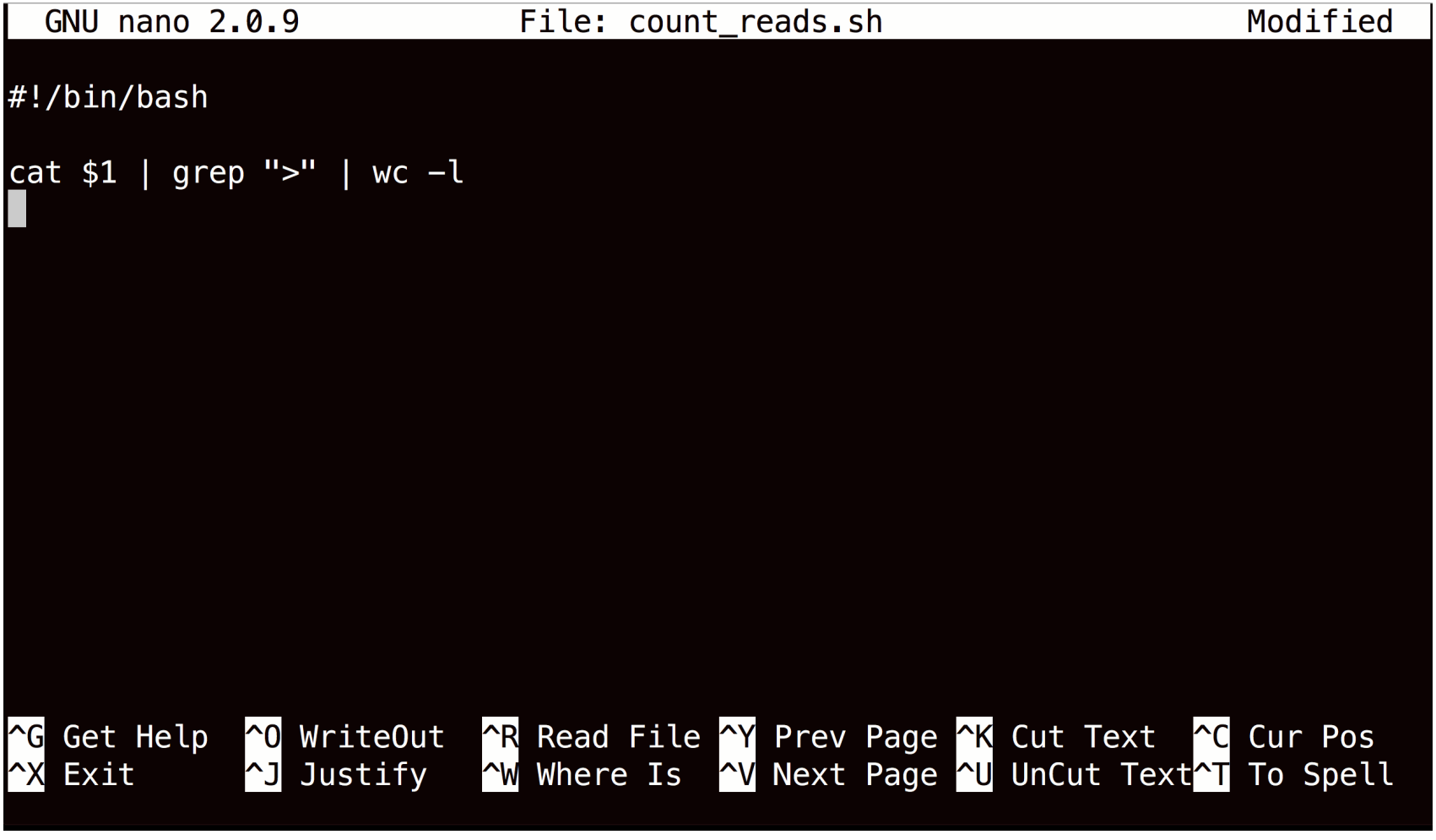
```
$bash parameter_overview.sh \
> one 2 three 4
#!/bin/bash

4
one
2
one 2 three 4
parameter_overview.sh
```

```
$bash parameter_overview.sh \
\ one 2 three 4

$#      3
$1      one
$2      2
$*      one 2 three 4
$0      parameter_overview.sh
```

Command-line texteditors

```
vi <new_name>          # create
vi <exitsing_file>     # edit

# vi is everywhere
# hopefully vim as well

nano <file>            # alternative

# common
# simpler to handle
```

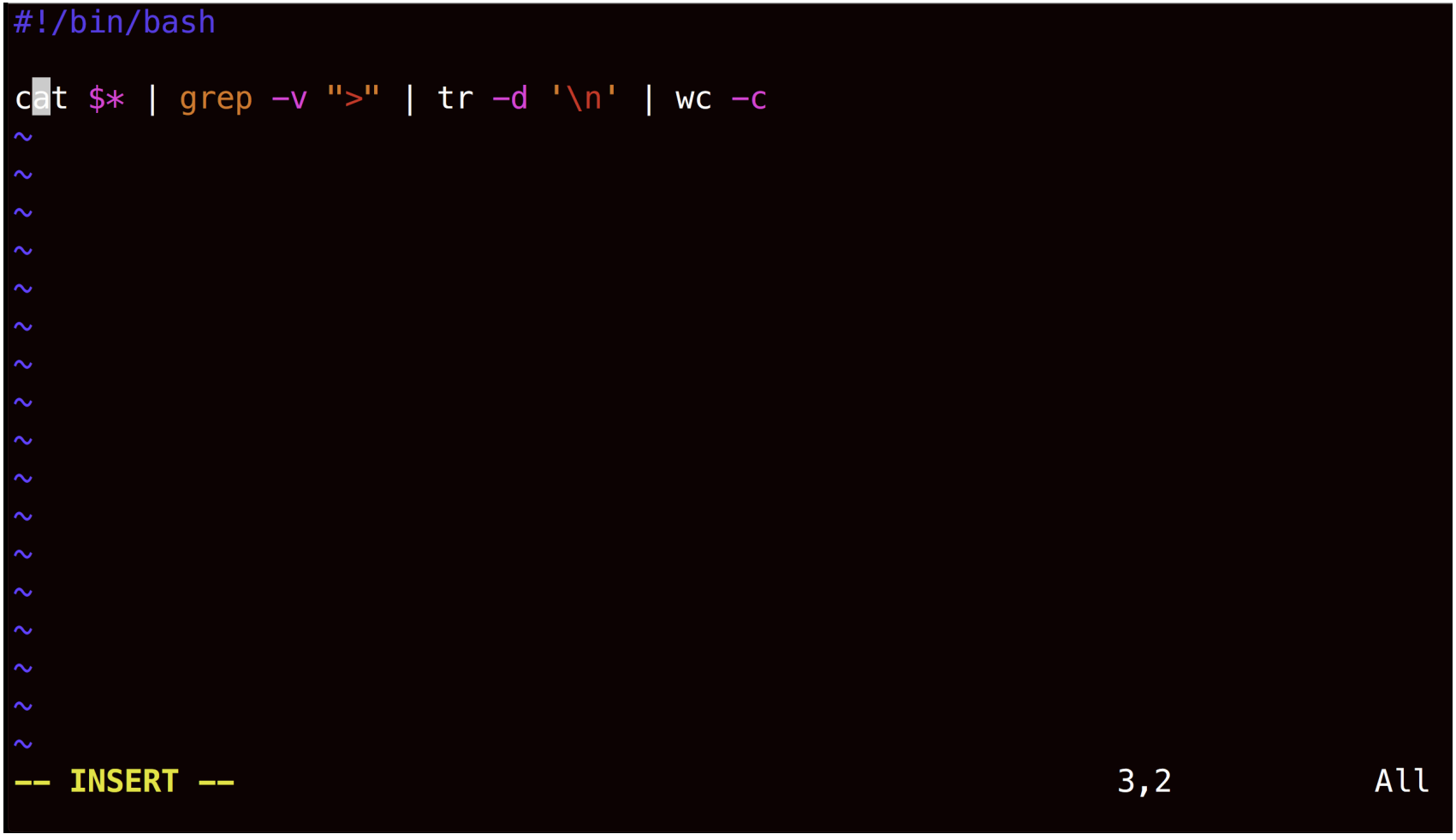
nano count_reads.sh



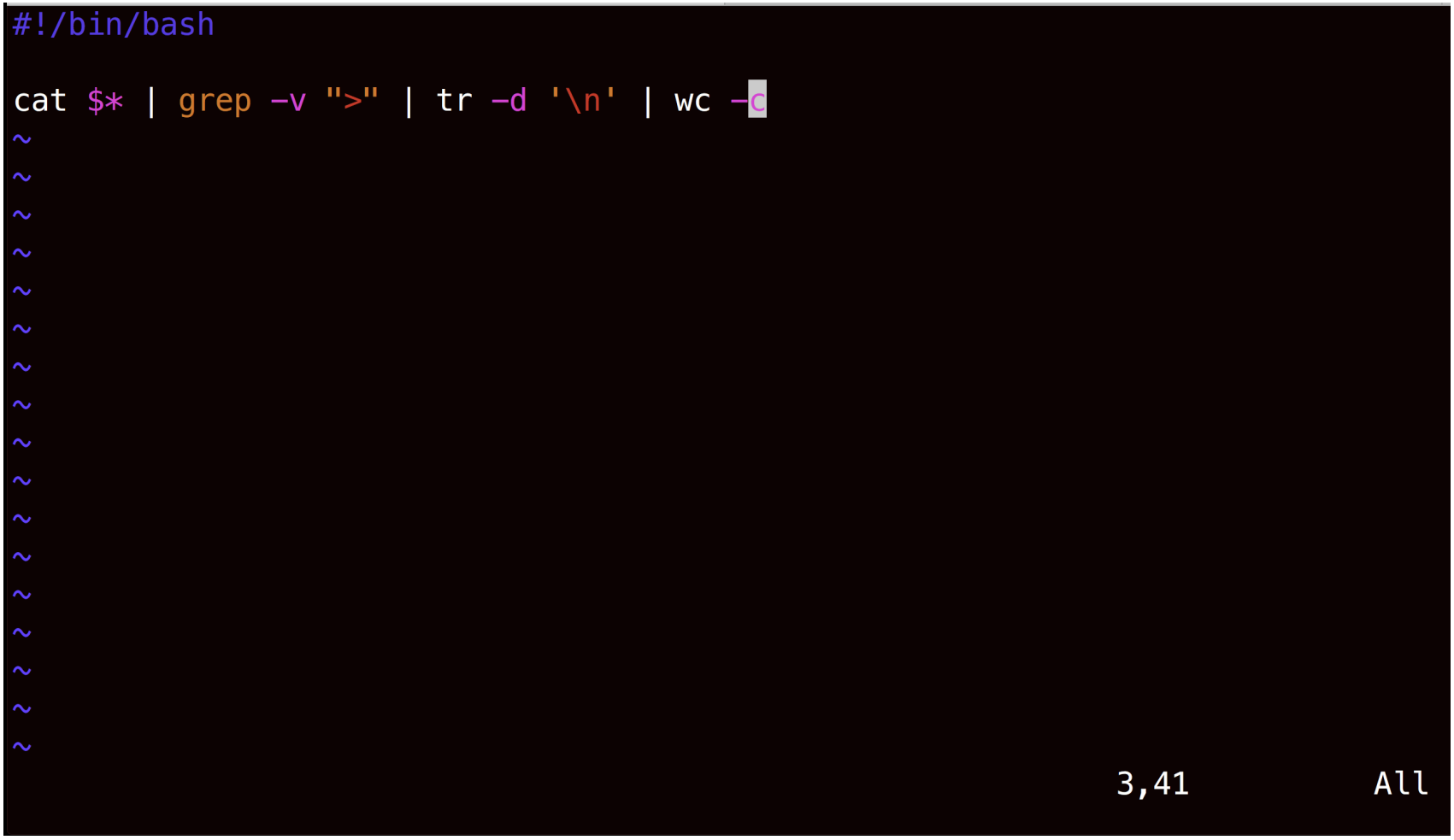
vi count_reads.sh

```
#!/bin/bash

cat $* | grep ">" | wc -l
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"count_reads.sh" 3L, 39C                                    1,1           All
```

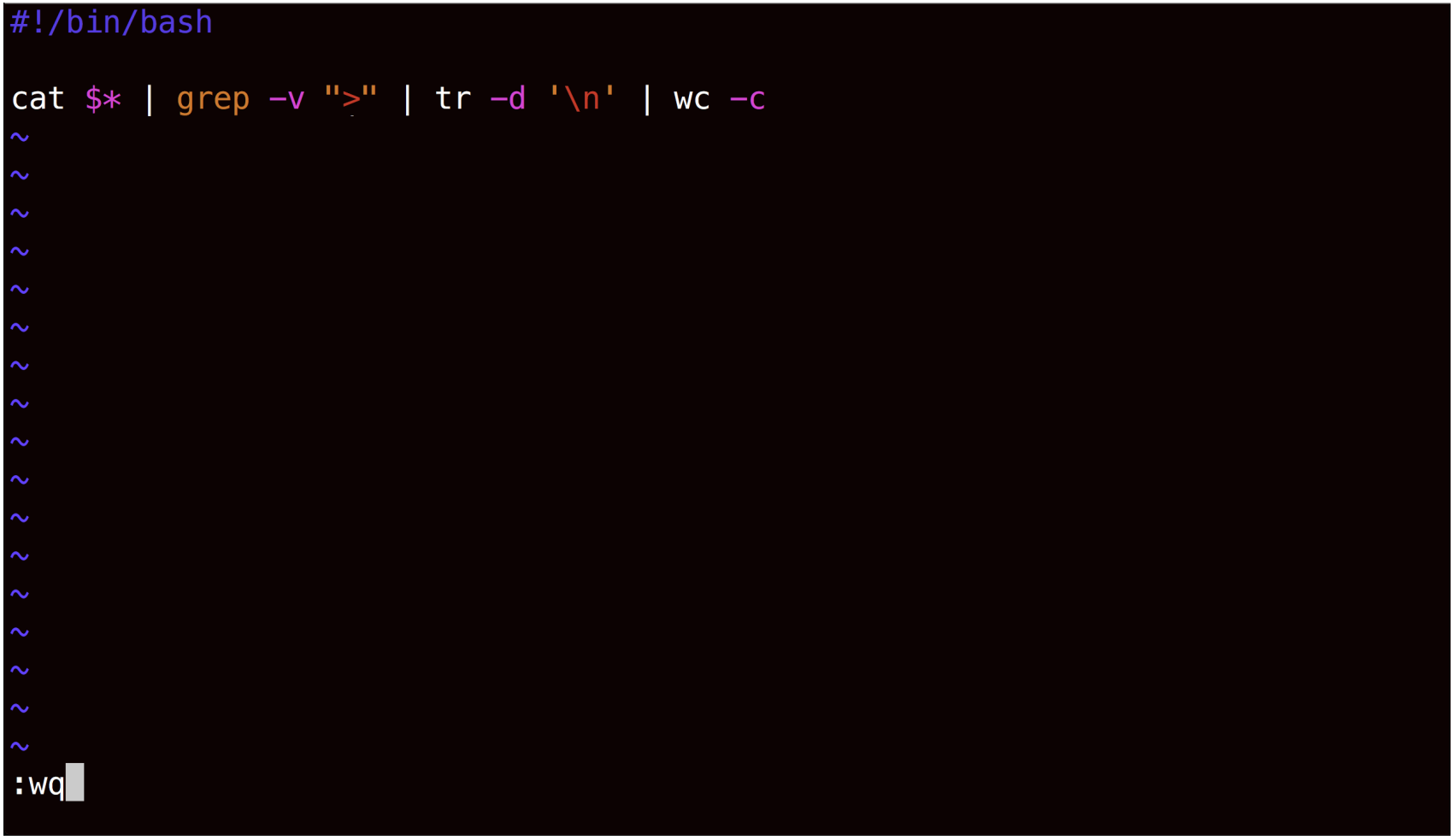
`a` or `i` enter insert mode



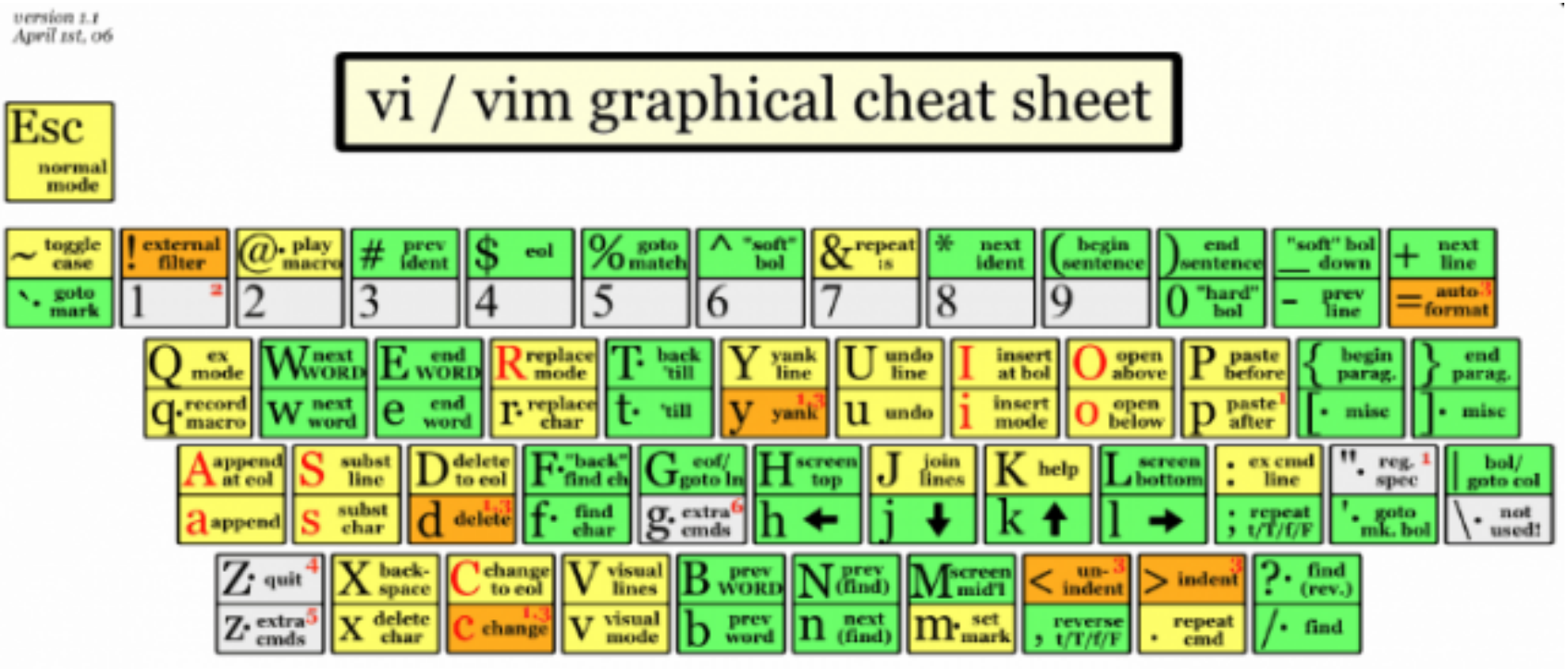
<esc> escape insert mode



:wq command write quit



Why is that monster popular?



and memes...

I Am Devloper
@iamdevloper

Following

I've been using Vim for about 2 years now, mostly because I can't figure out how to exit it.

Reply Retweet Favorite ••• More

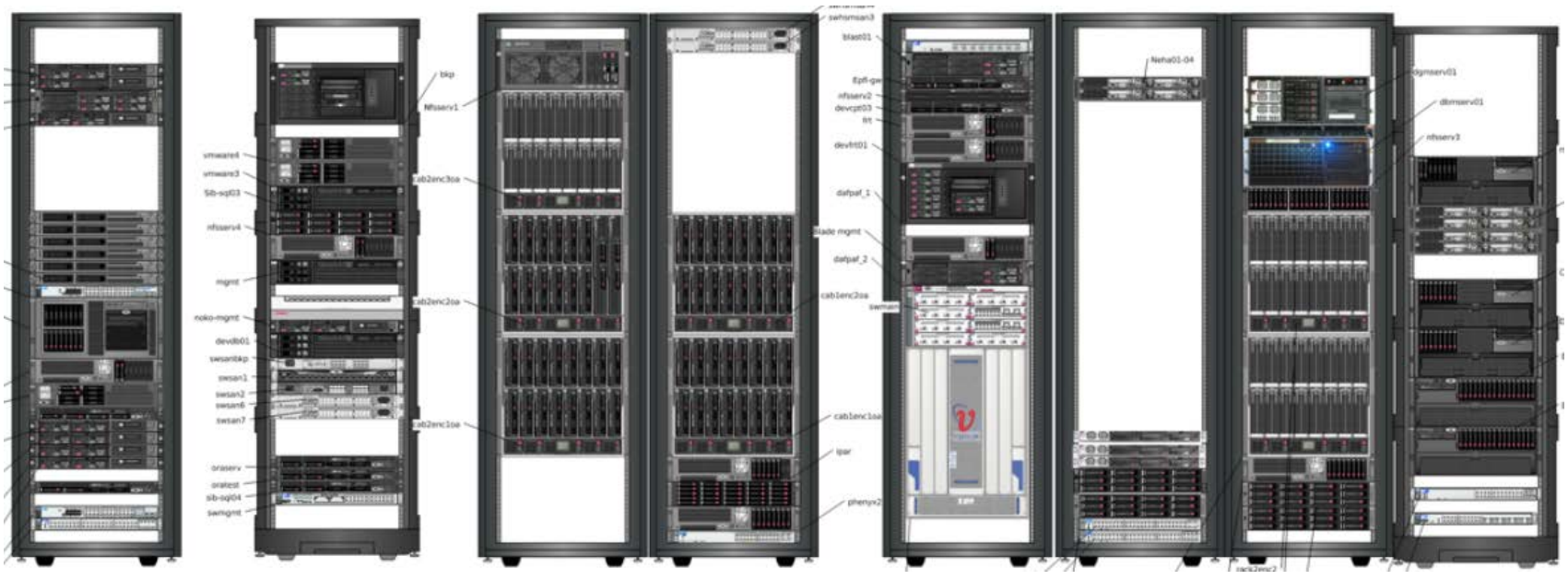
RETWEETS	FAVORITES
4,846	2,105

4:56 AM - 18 Feb 2014

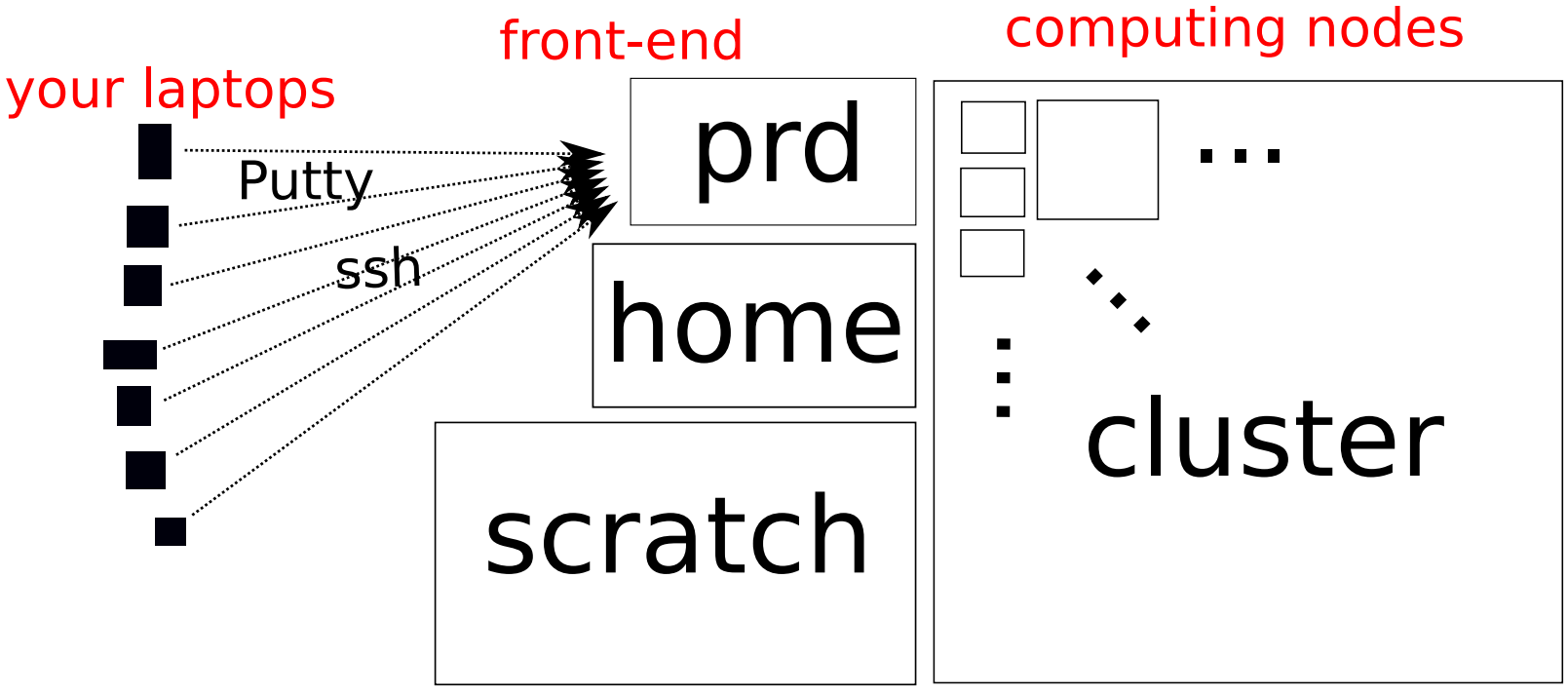
Your turn

```
vi <file>       # open file
a, i            # insert mode
<esc>           # command mode
:               # command
:w              # write
:wq             # write \& quit
:q              # quit io saving
```

Cluster computing



Cluster organisation

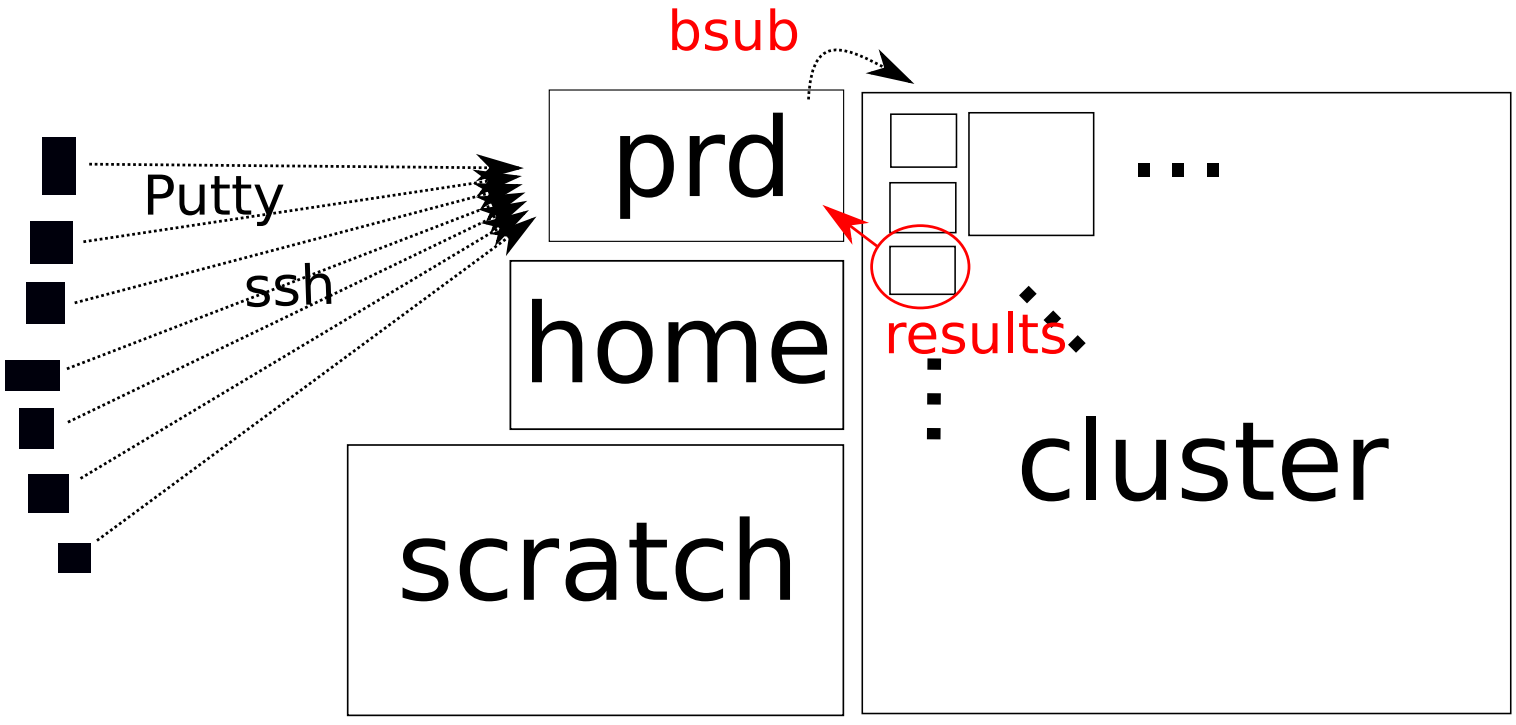


Queuing system

```
bsub          # submit a job
bjobs         # list your jobs
bkill         # kill a job
```

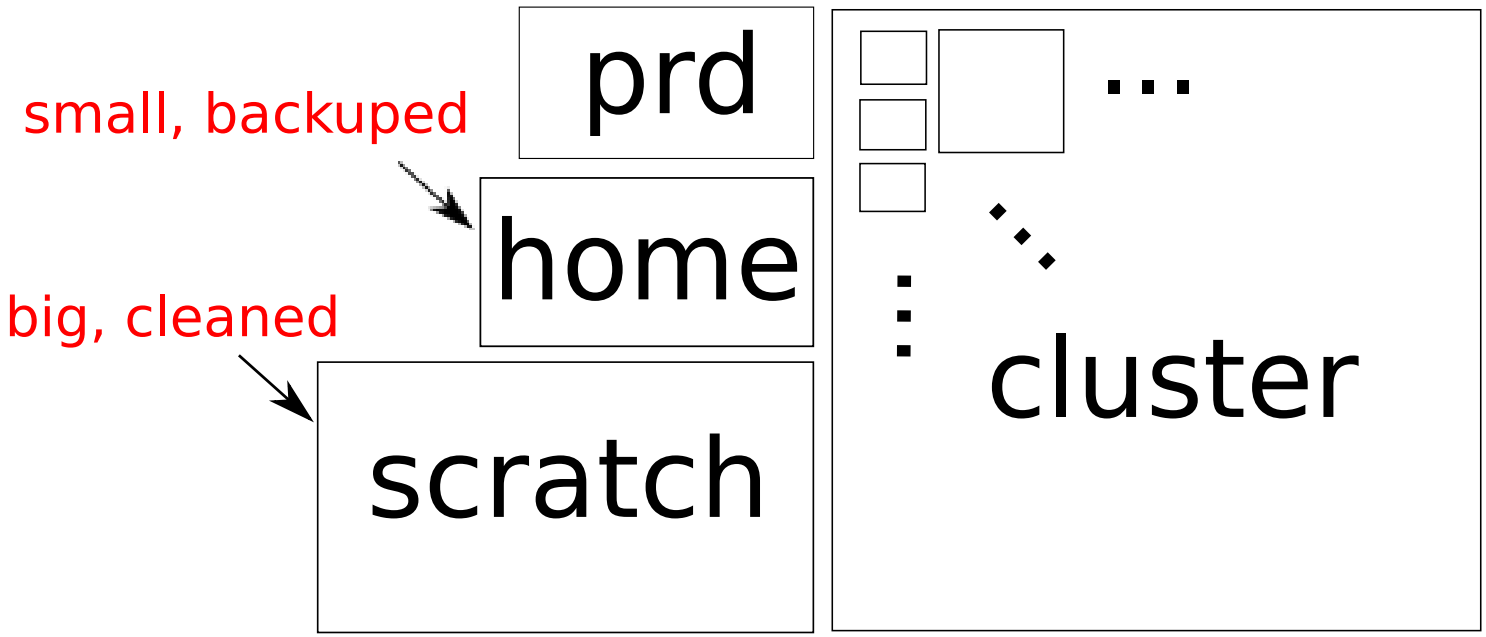
```
$bsub -q priority \
'echo "I <3 SAGE!" > message'

Job <256817> is submitted to queue <priority>.
```

```
kjaron@frt:/scratch/beegfs/monthly/kjaron/SAGE$ bsub -q priority 'echo "I <3 SAGE" > message'
Job <256825> is submitted to queue <priority>.
kjaron@frt:/scratch/beegfs/monthly/kjaron/SAGE$ bjobs
JOBID   USER    STAT  QUEUE      FROM_HOST   EXEC_HOST   JOB_NAME   SUBMIT_TIME
256825  kjaron  PEND  priority   frt                     *> message Sep 15 20:30
kjaron@frt:/scratch/beegfs/monthly/kjaron/SAGE$ bjobs
JOBID   USER    STAT  QUEUE      FROM_HOST   EXEC_HOST   JOB_NAME   SUBMIT_TIME
256825  kjaron  PEND  priority   frt                     *> message Sep 15 20:30
kjaron@frt:/scratch/beegfs/monthly/kjaron/SAGE$ bjobs
JOBID   USER    STAT  QUEUE      FROM_HOST   EXEC_HOST   JOB_NAME   SUBMIT_TIME
256825  kjaron  RUN   priority   frt         cpt214      *> message Sep 15 20:30
kjaron@frt:/scratch/beegfs/monthly/kjaron/SAGE$ bjobs
No unfinished job found
kjaron@frt:/scratch/beegfs/monthly/kjaron/SAGE$
```


bsub
prd
...
Putty
ssh
home
results
cluster
scratch

```
$bsub -q priority \
'echo "I <3 SAGE!" > message'

Job <256817> is submitted to queue <priority>.

$bkill 256817


bkill <job_id>      # generally
```

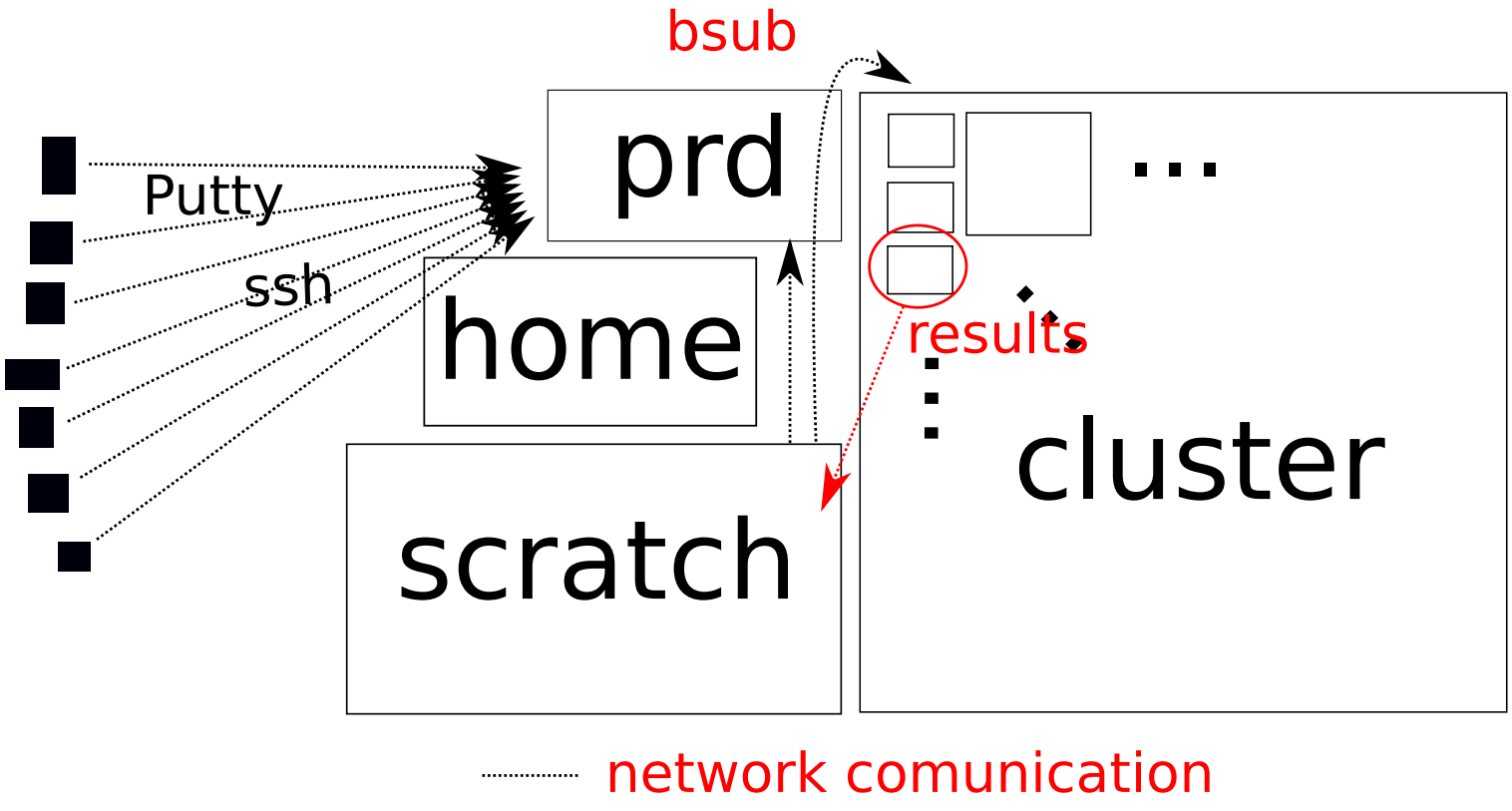
Cluster storage



```
/home/<username>                          # 2 GB quota
                                          # backuped

/scratch/beegfs/<...>/<username>  # 5 TB quota
                                          # cleaning
                    monthly               # monthly
                    weekly                # weekly
                    daily                 # daily
```

```
# scripts, logs, small tables
# do NOT bring garbage home!
/home/<username>

# your data
/scratch/beegfs/monthly/mls_2016/<username>

# meeting point
/scratch/beegfs/monthly/mls_2016/SAGE/...
```

Cluster queving for real



Module system

http://www.vital-it.ch/services/software

Module system

```
# /software/<category>/<package>/<version>

module add <category>/<package>/<version>

# to check loaded modules

module list
```

```
$module add UHTS/Quality_control/fastqc/0.11.2

# to check loaded modules
$module list
Currently Loaded Modulefiles:
  1) Development/java_jre/latest
  2) UHTS/Quality_control/fastqc/0.11.2

$fastqc -version
FastQC v0.11.2
```

bash sees the binary!

How comes?

```
$type fastqc
fastqc is hashed
(/software/UHTS/Quality_control/fastqc/0.11.2/bin/fastqc)

$echo $PATH
/software/UHTS/Quality_control/fastqc/0.11.2/bin:/software\
/Development/java_jdk/1.8.0_102/jre/bin:/mnt/common/lsf/9.\
1/linux2.6-glibc2.3-x86_64/etc:/mnt/common/lsf/9.1/linux2.\
6-glibc2.3-x86_64/bin:/software/bin:/usr/local/bin:/bin:/u\
sr/bin:/usr/local/sbin:/usr/sbin:/sbin:/home/kjaron/bin
```

```
# remove all modules
$module purge

$module list
No Modulefiles Currently Loaded.

$echo $PATH
/mnt/common/lsf/9.1/linux2.6-glibc2.3-x86_64/etc:/mnt/comm\
on/lsf/9.1/linux2.6-glibc2.3-x86_64/bin:/software/bin:/usr\
/local/bin:/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/sbin:/\
home/kjaron/bin
```

Your turn

```
bsub <parameters> '<command>'
bjobs
bkill <jid>

vit_soft -s <keyword>
vit_soft -p <package_name>
module add <path>/<package_name>
module purge
```

A + !
Bon appétit!