

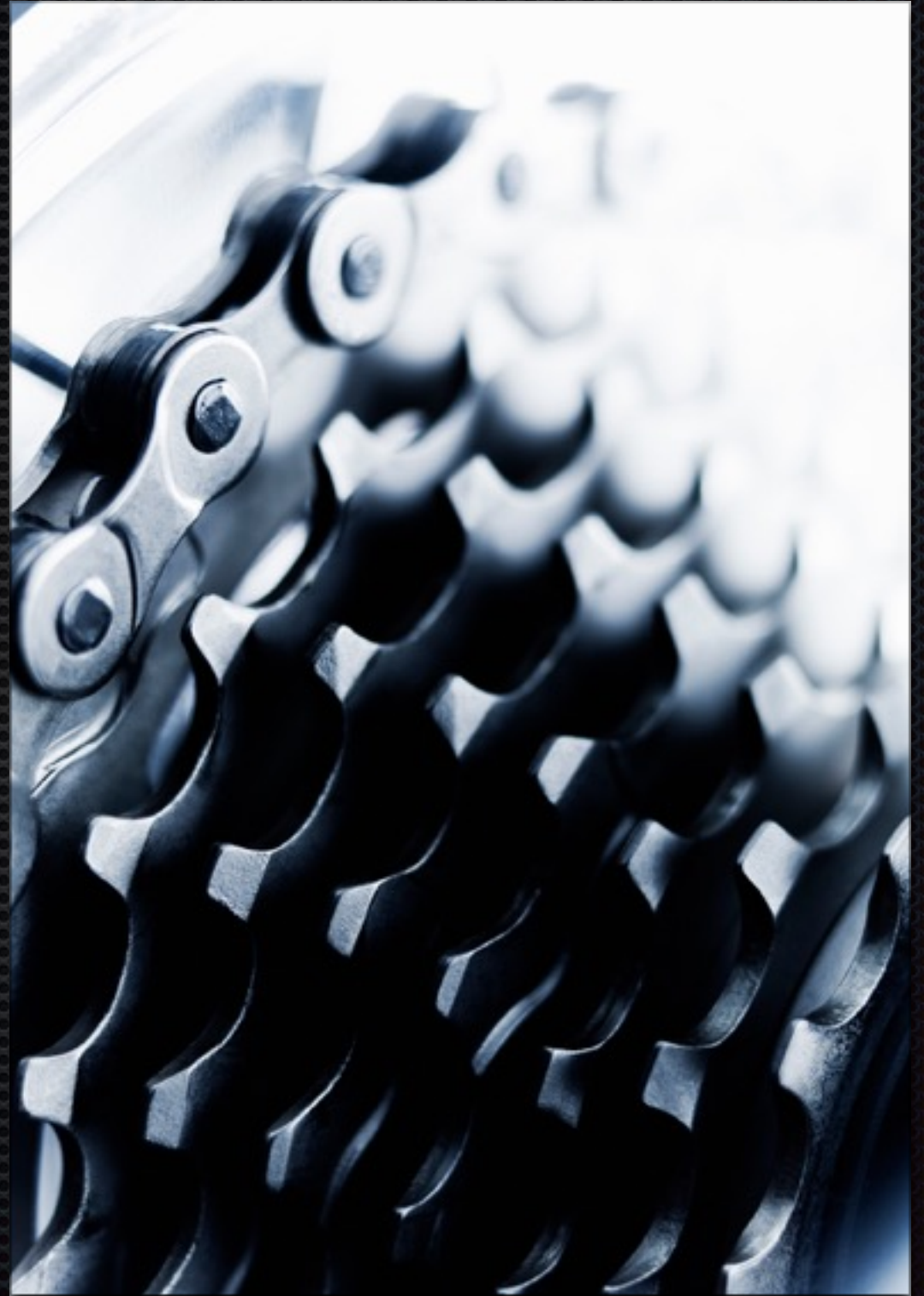
Technologies For Data Manipulation And Analysis

There's More than One Way to Groom a Cat(alog)

Presented by Annie Glerum and Kathryn Lybarger
OLAC Preconference October 26, 2017

Tools

Command Line



Command line

- ✧ Unix / linux / Mac terminal
 - ✧ bash
 - ✧ Now available in Windows 10
- ✧ “shell access” or “server access”
- ✧ Common set of tools available
 - ✧ Others may be installed

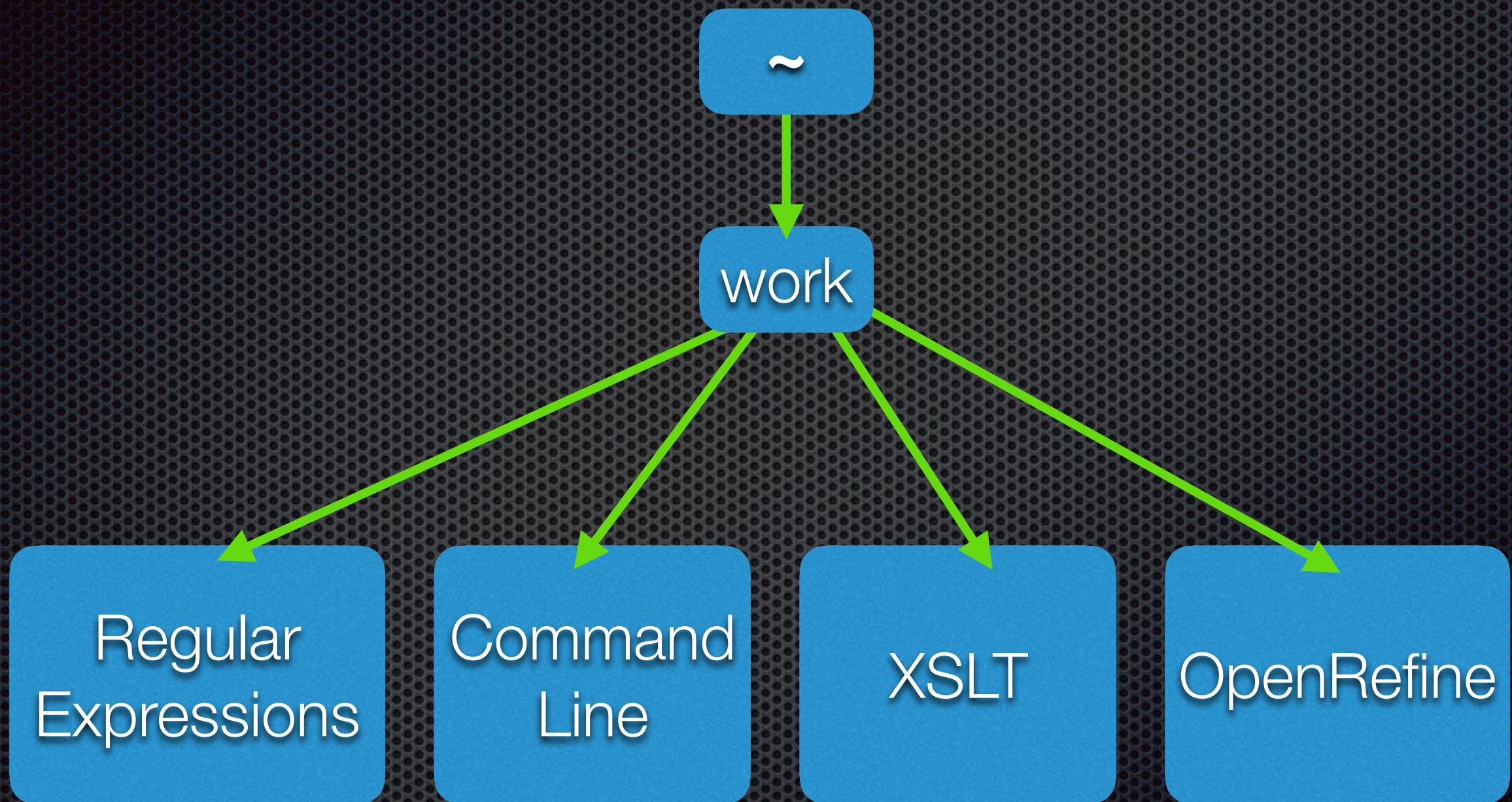
Follow along on the web

- ✦ Visit <https://olac2017.zemkat.org/>
- ✦ Login with your Google account
- ✦ Start your server
- ✦ Open a terminal

Command structure

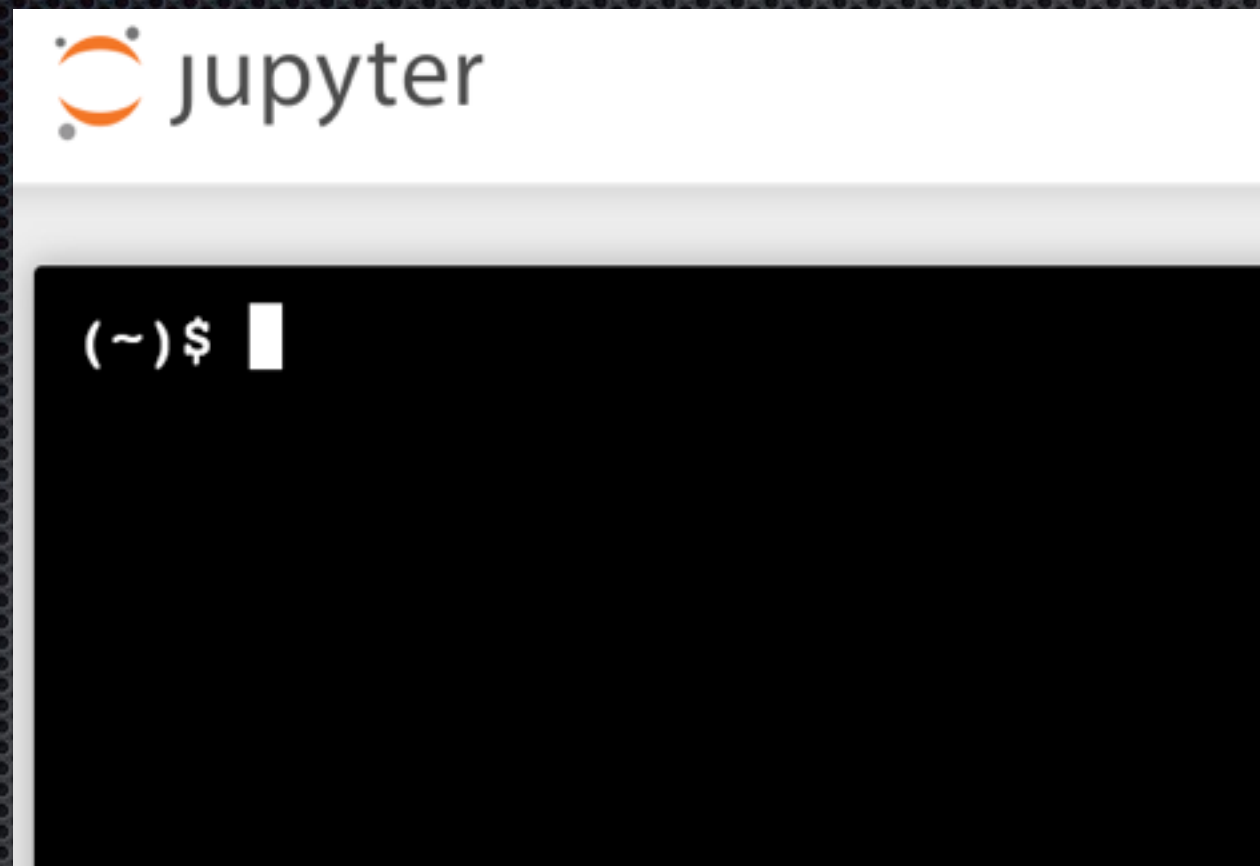
- ✧ `convert -size 320x90 image.png image.jpg`
- ✧ `command`
 - ✧ what program are you running?
- ✧ `switches`
 - ✧ how should the program behave?
- ✧ `arguments`
 - ✧ what (files?) are you running the program on?

Directory structure



~/work/XSLT

The Prompt



- \$ — a common prompt, ready for your input
- (~) — present working directory

Command: `ls`

list directory contents

- ✦ Try this:
 - ✦ `ls`
- ✦ What do you see?
- ✦ What do the colors mean?

Command: `cat` print file to screen

- ✦ Short for `concatenate`
 - ✦ (can be used for many other things)
- ✦ Try this:
 - ✦ `cat README`

Command: `cd` change directory

- ✧ Try this:
 - ✧ `cd work`
- ✧ What is your present working directory now?
- ✧ What files or directories are in there?
- ✧ (How do you get back to where you were?)

Special symbol: ..

dot dot = parent directory

- ✧ Try this:
 - ✧ `cd ..`
- ✧ What is your present working directory now?
- ✧ How do you get back into the `work` directory?

Activity: Treasure hunt

- ✦ Change to the `work` directory
- ✦ Change to the `CommandLine` directory
- ✦ Change to the `treasure_hunt` directory
- ✦ Many subdirectories have a `treasure` file, but is there really treasure there? Look in the file to find out!
- ✦ (Let me know when you've found the treasure!)

Treasure hunt debrief

- ✦ Did you hunt through the directory tree in any particular order?
- ✦ Is there a better way to find the treasure?

Hunting through MARC

- ✧ Under the CommandLine directory, go to the MARC directory
- ✧ Look at `LC1000bibsA.mrc`
- ✧ What does the file look like?
 - ✧ Can you read it?
 - ✧ Would it be easy to edit?

Software: YAZ

- ✧ Try this:

- ✧ `yaz-marcdump -i marc -o line`
`LC1000bibbsA.mrc`

- ✧ Is that better? Also, try this:

- ✧ `yaz-marcdump -i marc -o line`
`LC1000bibbsA.mrc > LC1000bibbs.mrk`

I/O redirection

- ✦ `> filename`

- ✦ Following your command with this means don't put it on the screen, send it to that file

- ✦ `| program`

- ✦ Following your command with this means don't put it on the screen, use as input for next program
- ✦ This vertical bar is called a "pipe"

Commands: head, tail, less

- ✧ Try these:
 - ✧ `head LC1000bibsA.mrk` — view the top ten lines
 - ✧ `tail LC1000bibsA.mrk` — view the last ten lines
 - ✧ `less LC1000bibsA.mrk` — view the file a screen at a time
- ✧ But what if you're looking for something specific?
 - ✧ Books with “science” in the title?

Regular expression

- ✦ What regular expression would match lines from this file with “science” in the title?

Regular expression

- ✦ What regular expression would match lines from this file with “science” in the title?
- ✦ `/^245.*science/`

Command: **grep**

print lines matching a pattern

- ✧ “global regular expression print”
- ✧ print the lines from a file that match a regular expression
- ✧ works best with line-based files
- ✧ Try this:
 - ✧ **grep** “^245.*science” LC1000bibsA.mrk

Command: `grep -i` case insensitive `grep`

- ✦ The original regex was only matching lowercase “science”, so how about Science, SCIENCE, or SCieNCe?
- ✦ Try this:
 - ✦ `grep -i “^245.*science” LC1000bibsA.mrk`
- ✦ (Did that find more?)

Command: `wc`

count lines, words, bytes

- ✧ Try this:

- ✧ `grep "^245.*science" LC1000bibsA.mrk | wc -l`

- ✧ `grep -i "^245.*science" LC1000bibsA.mrk | wc -l`

- ✧ How many did you find?

Command: `grep -l` which file matches

- ✦ We can use `grep` on binary MARC, we probably just don't want to see the output
- ✦ Try this:
 - ✦ `grep -i "the flowery kingdom" *.mrc`
 - ✦ `grep -il "the flowery kingdom" *.mrc`

How many MARC records in this file don't have links?

- ✦ How can we answer this?
- ✦ How can we answer this with **grep**?

Shell scripts

```
Go away or I will  
replace you with  
a very small  
shell script.
```

- ✦ A sequence of commands you type on the command line can be saved for later use as a “shell script”
- ✦ Try this:
 - ✦ `look_for_missing.sh` `OpenTextbooks.mrc`

Command: `python` run a python program

- ✦ pymarc — python library for working with MARC
- ✦ Look at `find_missing.py`
- ✦ Try this:
 - ✦ `python find_missing.py`
`OpenTextbooks.mrc`
- ✦ Did it work?

Command: **nano** a text editor

- ✧ Try this:
 - ✧ **nano** find_missing.py
- ✧ To save the file:
 - ✧ ^o — save
 - ✧ <enter>
 - ✧ ^x — to exit

Command: man

- ✧ Manual, “man pages”
- ✧ Should exist for most common commands
- ✧ Tells you:
 - ✧ What does the program do?
 - ✧ What switches/arguments does it take?
- ✧ `man cat`

Mashcat

- ✦ A loose organization of catalogers and coders working together for better communication
- ✦ Webinars, twitter chats, in-person conferences
- ✦ Slack channel: bash