Clipboard meets Command Line

A Tucson JUG Lightning Talk
March 9, 2010

William Mitchell (`whm`)
Department of Computer Science
The University of Arizona

The Facts...

Editors like vi and Emacs allow one to use pipeline-friendly programs to insert or alter text in buffers.

- Able to insert output from a command

  ```
  :r!date

  C-U ALT-! ls -l $(find -name \*.as)
  ```

- Able to do in-place filtering of text

  ```
  :5,35!cat -n

  C-U ALT-| sort -n
  ```

Some 21st-century applications, especially editors in web applications, lack this basic capability.

All too common...

Ever do this?

1. Run a shell command in a terminal window
2. Use mouse to select output
3. Copy to clipboard
4. Paste into 21st-century application

Or maybe this?

1. Direct output of shell command into a file
2. Open file with an editor
3. Select all text
4. Copy to clipboard
5. Paste into 21st-century application

The clipboard provides a bridge between the rich world of pipeline-friendly programs and the impoverished 21st-century applications.

Cygwin's `/dev/clipboard`

Cygwin gives us access to the clipboard as `/dev/clipboard`...

```
% cal > /dev/clipboard
```
...paste into document...

...copy a selection in a 21ˢᵗ-century application...
```
% cut -d: -f1 < /dev/clipboard | sort | uniq -c | sort -rn
```


```
% cat $files/a4/tests/rt0.cc > /dev/clipboard
```
...paste that code into a mail message...

True story #1

IM:

    jane>   The message of the day is being truncated!
    whm>   How long is it?
    jane>   Do I really have to count?
    jane>   this many: Welcome to AnimalWatch! Explore the Dashboard and then try to
            complete all three problem sets and do a few Skill Builders.

What to do?

True story #1, continued

IM:

    jane>   The message of the day is being truncated!
    whm>   How long is it?
    jane>   Do I really have to count?
    jane>   this many: **Welcome to AnimalWatch! Explore the Dashboard and then**
            try to complete all three problem sets and do a few Skill Builders.

Select the text and let `wc` do the counting!

```
% wc < /dev/clipboard
      0      21     123
```

A handy shell script: `cb`

Platform differences:
- Cygwin has `/dev/clipboard`

- OSX has `pbpaste(1)` and `pbcopy(1)`

- X Windows has `xsel`

For convenience and portability, let's have a cross-platform script, `cb`

```
date | cb
```
(`cb` knows to put the text on the clipboard)

```
cb | wc
```
(`cb` knows to send the clipboard contents to standard output)

```
cb < file
```

```
cb > file
```

```
cb just clip this
```
(put the text "just clip this" on the clipboard)

True story #2

I'd like a countdown of the days before a deadline next Monday, to paste into a Wiki page.

```
% daysdates 3/9 3/16
Tuesday    03/09/10
Wednesday 03/10/10
Thursday   03/11/10
Friday     03/12/10        (Selected dates of interest...)
Saturday   03/13/10
Sunday     03/14/10
Monday     03/15/10
Tuesday    03/16/10
```

% cb | xfield 2 | tac | cat -n | tac
```
     5  03/10/10
     4  03/11/10
     3  03/12/10
     2  03/13/10
     1  03/14/10
```

% !! | cb
cb | xfield 2 | tac | cat -n | tac | cb   *(Note: cb on both ends!)*

True story #3

I've got this text in an assignment write-up:

> *Use* `turnin` *with the tag* `397a_4` *to submit your solutions for grading. The deliverables for this assignment are* `1000.cc, Range.h, Range.cc, get_bounds.txt, assign.h, inline.txt, answers.txt,` *and* `extra.txt.`

I want to add those file names, one per line, to a text file I'm editing.

What to do?

True story #3, continued

At hand:

> *Use* `turnin` *with the tag* `397a_4` *to submit your solutions for grading. The deliverables for this assignment are* **`1000.cc, Range.h, Range.cc, get_bounds.txt, assign.h, inline.txt, answers.txt,` *and* `extra.txt`**.

Put the text of interest on the clipboard and...

```
% cb | tr , \\012 | tr -d " "
1000.cc
Range.h
Range.cc
get_bounds.txt
assign.h
inline.txt
answers.txt
and
extra.txt
```

Looks good (almost). Put it back on the clipboard, ready to paste...
```
% !! | cb
cb | tr , \\012 | tr -d " " | cb
```

More on `cb`

http://williammitchell.blogspot.com/2008/03/fun-with-cygwins-devclipboard.html