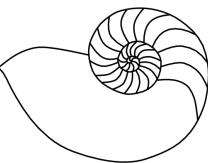UNIVERSITY OF MANITOBA | Computer Science
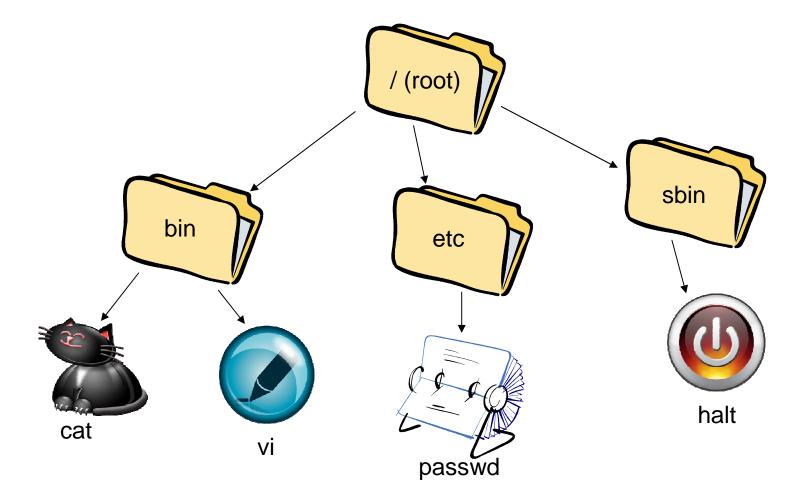
UNIX File System Hard & Symbolic Links RTFM: ln(1) & rm(1)

Gilbert Detillieux
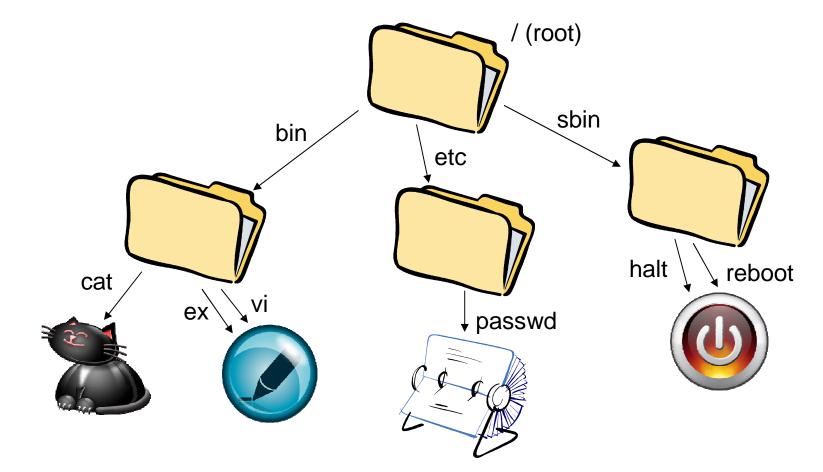
October 13, 2009
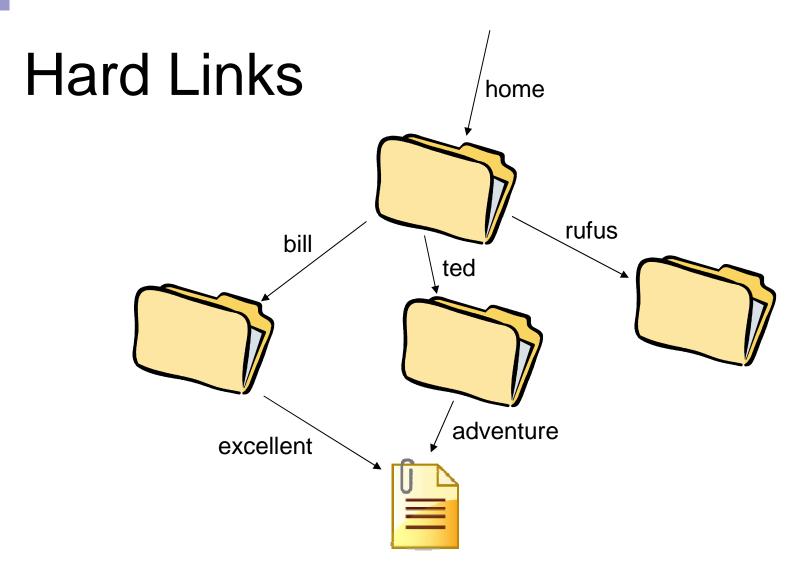MUUG Meeting

Traditional File Tree Structure

UNIX Directory/File Structure

Hard Links


```
ln /home/ted/adventure excellent          rm adventure
```

Directory Links

20	"."
2	".."
21	"bill"
22	"ted"

2	d
…	…
20	d
21	d
22	d

Inode list

22	"."
20	".."

Directory Data

Hard Links - Features

- No replication of file data

- Only space used is extra directory entry

- All hard links to a file are equivalent
  - Can't distinguish original from new links

- Removal of extra links leaves file content

- Removal of last link causes file deletion

Hard Links - Limitations

- No arbitrary links to directories
  - Only special links "." and ".." allowed
  - Need to avoid loops in tree
- Can't link to other file systems
  - Only inodes within same file system

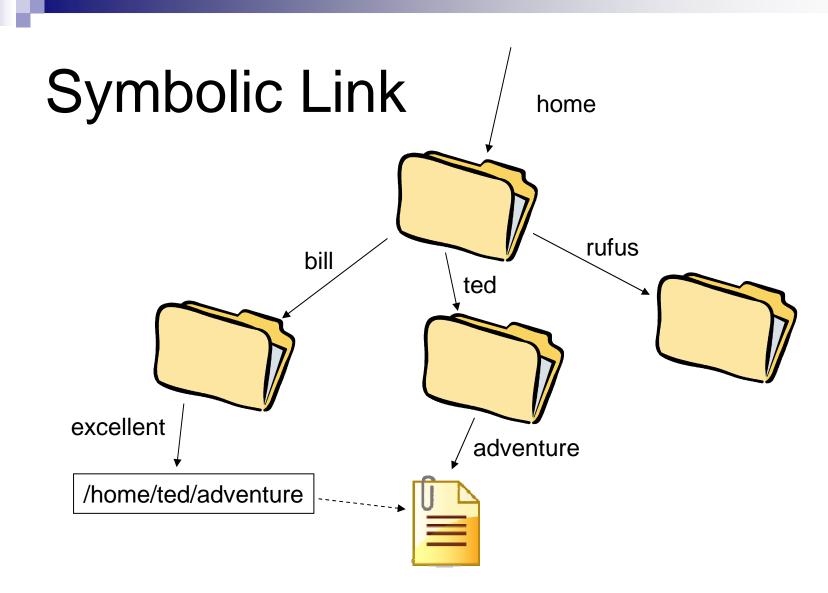# Symbolic Link

home

bill

ted

rufus

excellent

adventure

/home/ted/adventure

```
ln -s /home/ted/adventure excellent      rm adventure
```

Symbolic Links - Features

- No replication of file data
- Space used is extra directory entry **plus** inode & data block(s) for path name
- Symlinks can point to arbitrary path
  - Relative or absolute (fully qualified) path
- Creation/removal of symlinks don't affect target
- Removal of target leaves dangling link(s)

LN(1) Man Page

- ## NAME
  - □ ln - make links between files
- ## SYNOPSIS
  - □ **ln [-fns**]... *TARGET LINK_NAME*
  - □ **ln [-fns**]... *TARGET*
  - □ **ln [-fns**]... *TARGET... DIRECTORY*