Assignment 3

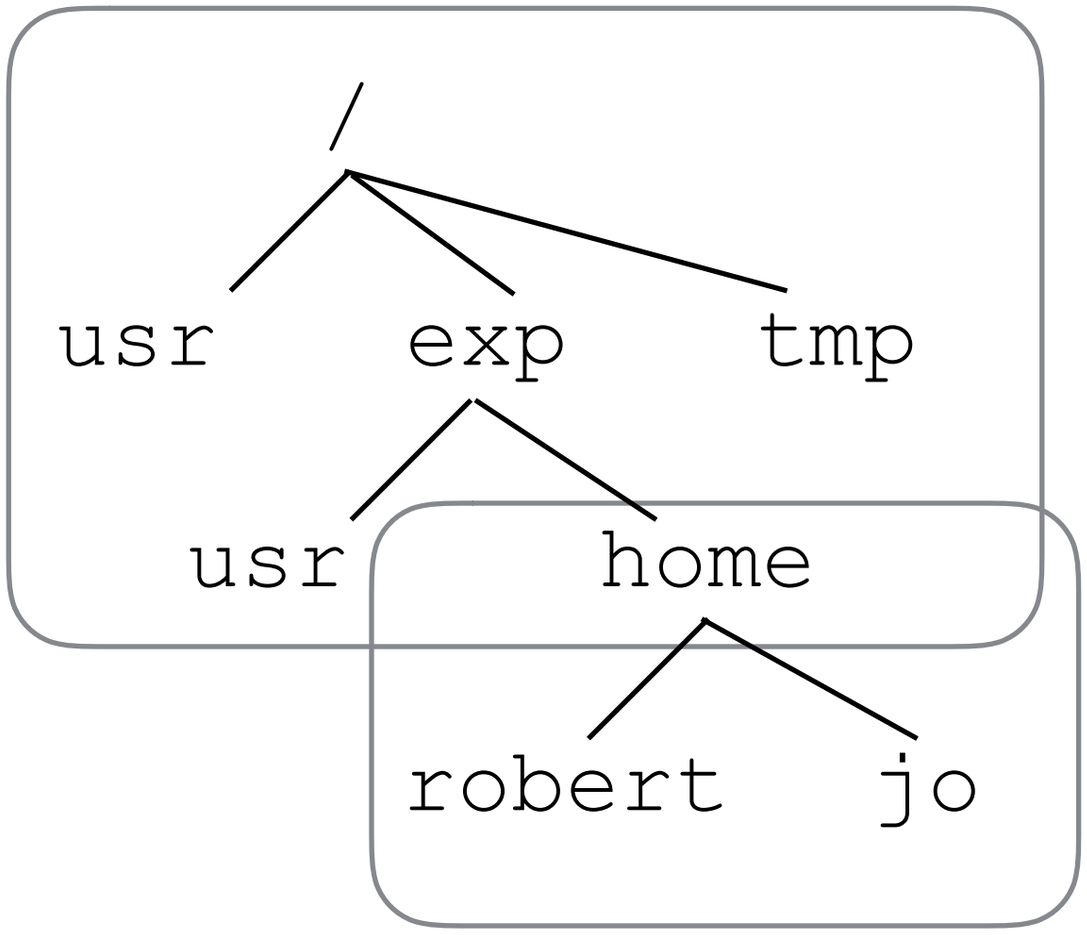
User space file systems

File system in USEr space - FUSE

- Linux has a library which allows file systems to be written and used without root privileges - libfuse.
- The library is used in conjunction with a kernel module which provides the privileged operations.
- The user's file system gets mounted on to an existing directory.

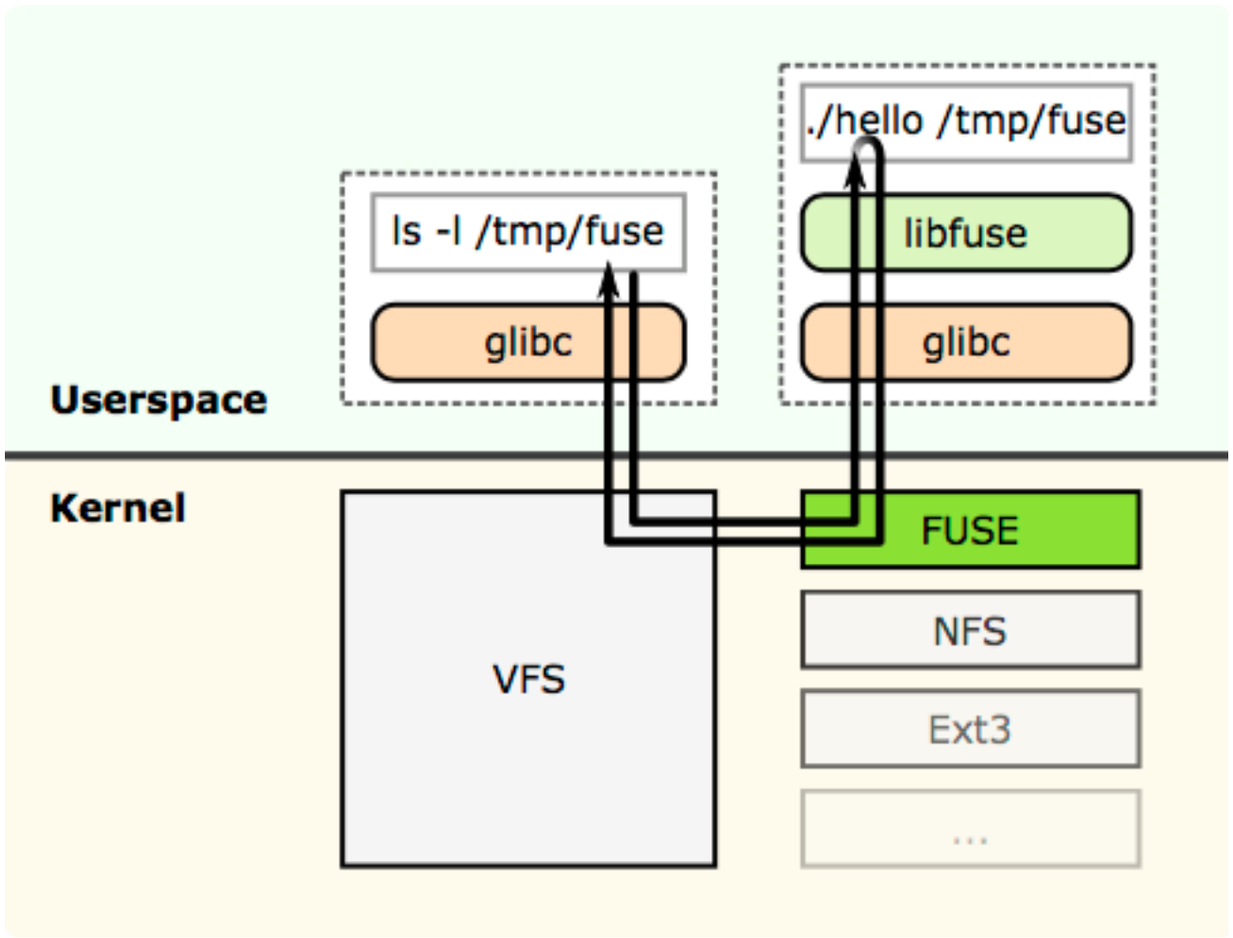
Mounting a file system

- A file system (in Unix speak) can be thought of as a device or partition which can be connected into the standard hierarchical file system starting from the root "/".



Mounting a file system

- We need a mount point (a directory where we plant the file system).
  - in the previous slide this is `/exp/home`
- Any files which are already in the mount point directory then disappear.
- They get replaced by the files/directories in the new file system.



How it works

from Wikipedia

`./hello` is mounted on `/tmp/fuse`

All the operations on `/tmp/fuse` and its files can be handled by user level code.

Setting up for this assignment

- You need a recent version of Ubuntu (actually can use other OSs but the markers will use Ubuntu in the labs as in A2). Doesn't run on Windows 10 subsystem for Linux.
- Download fuse.py and versionfs.py from the A3 files section on Canvas.
  - fuse.py is from https://github.com/terencehonles/fusepy
  - you can also get examples from the github site

OMG Python

- Much (much) simpler to do this assignment in Python.
- If you don't know how to do something in Python just ask on the forum, or one of the tutors/lecturers, or on StackOverflow.
- fuse.py is written using Python 2. The standard python on new Ubuntu distributions is now Python 3. You may need to install Python 2:
  - sudo apt install python

versionfs.py

- This neatly enapsulates the methods you may need to modify (not all of them).
- There are two sorts of methods
  1. filesystem methods which deal with directories and files as a whole
  2. file methods which deal with the contents of files

Part1

- Ensure fuse.py, versionfs.py are in the same directory.
- In this directory make a directory called "mount"
- Run "python2 versionfs.py mount"
- In a separate terminal window create files in the mount directory and observe the output in the versionfs.py window
- To stop "fusermount -u mount"

Part2

- Modify versionfs.py to create a file system which maintains previous versions of files when you save a modified version.
- You will also have to provide some extra programs (can be in any language which runs on Ubuntu in the labs) to manipulate the versions
- e.g. listversions filename
  - lists the versions of that file
- e.g catversion filename version#
  - display the contents of a version of filename on the screen