What is the bash cd history?

GNU bash can maintain a history of visited working directories. To add to this list, use **pushd** rather than **cd** to change directory. **popd** removes an entry from the list. If **pushd** always specifies a new WD, the list operates as a stack. But **pushd** $\pm n$ makes the indicated entry the CWD (the last entry is -0). The history is rotated, no entries are lost. **popd** $\pm n$ removes the indicated entry. **dirs** displays the history.

Streamlining cd history builtins

I find the default behaviour to be a bit clunky: **pushd** & **popd** do a gratuitous **dirs**, you have to enter **dirs -v** to get the entry numbers and who wants to type e.g. **pushd +2** anyway? The extensions in the next slide can all be found in dot_bashrc (part of command line tools (C_L_T))

command_line_tools extensions

All are functions, except as marked

Bash builtin	extensions
pushd	pd, pdd, up
popd	ppd
dirs	],]] (aliases)

The extensions to **pushd** & **popd** suppress the gratuitous **dirs** and treat unsigned numbers as having a leading +. Thus **pd 3** becomes **pushd +3**. (quick demo)

Saving the cd list

This is straightforward (show **svdirs**). Write the cd history to {per-window command history filename}.dirs . Error if command history filename is not per-window.
When closing a window, alias **exit_sh** does an **svdirs** then exits bash.

Restoring the cd list

.bashrc does this. So that *.bashrc* may be invoked multiple times, only restore the list if it is currently empty (has 1 entry).
(Show the code from dot_bashrc).
99: cut is a loadable builtin which some systems won't have
100: test if we should go ahead
103: use **tac** since 1ˢᵗ entry is CWD which we want to push last; saved list has line numbers we need to remove; don't push (now) nonexistent directories
107: when done, discard initial CWD