Demo - pickleViewer + batch hadding

Matthew Citron
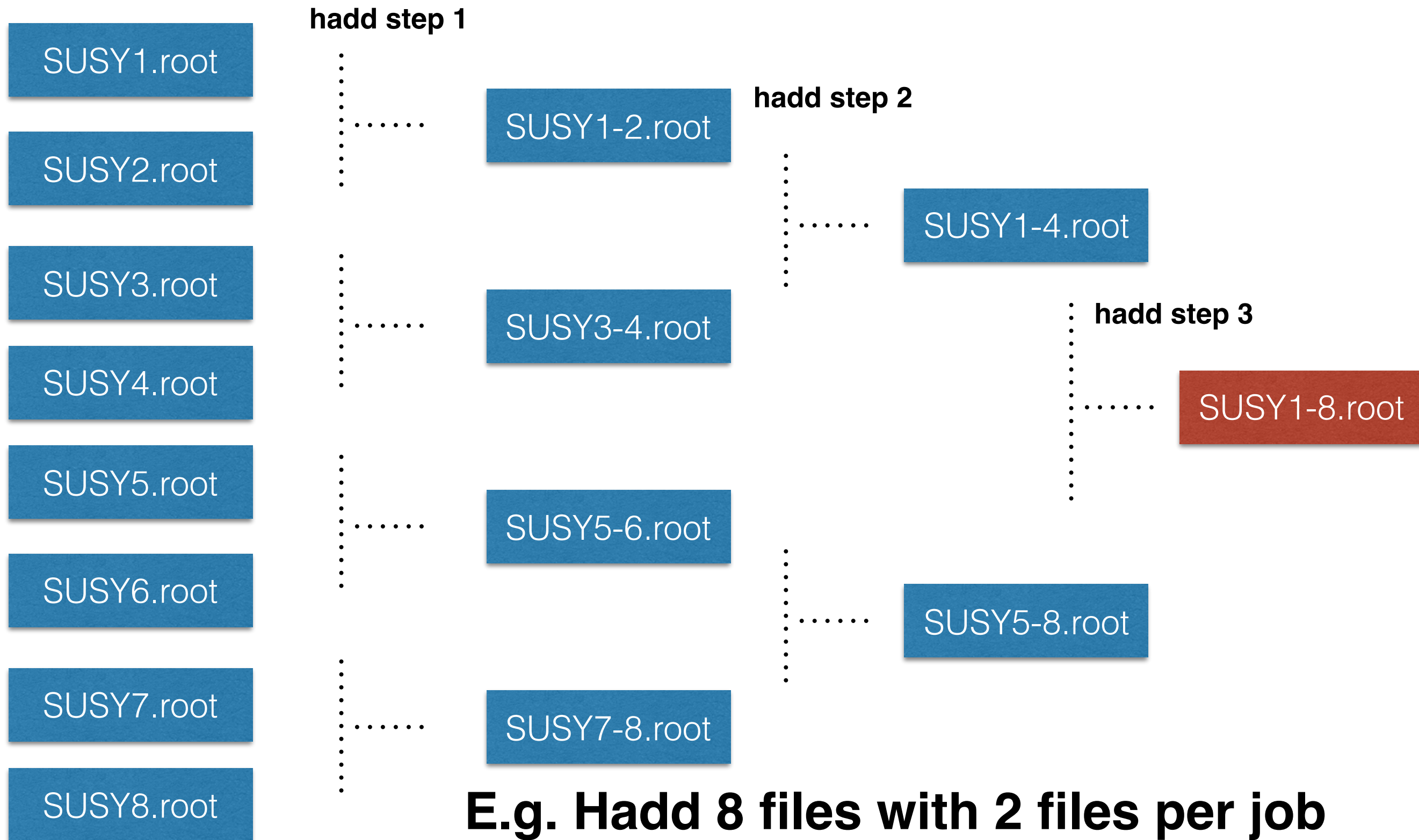
PickleViewer

- Pickle file format handy way of saving dicts/lists for many applications

  - Weights, trigger efficiencies, shopping lists, etc…

- Can be very cumbersome to look at the contents quickly

  - Often dict of list of dict of …

  - Can write lots of loops

- Useful script - ~mc909/Scripts/pickleViewer.py to recursively print pickle files to screen

  - Can twin with 'sort' bash command for sorted output

Recursive Hadding

- Often want to merge (hadd) large numbers of root files

  - E.g. Submitting many jobs to batch that needs to be merged on completion

- Use root hadd script for smaller number of files but for large numbers (> 100) can be very slow

- Useful script - ~mc909/Scripts/recursiveHadd.py allows this to be done much faster on the batch

  - Instead of hadding all at once split into N tasks hadding M files and repeat until all merged

Recursive Hadding - illustration

hadd step 1

SUSY1.root

SUSY2.root

SUSY3.root

SUSY4.root

SUSY5.root

SUSY6.root

SUSY7.root

SUSY8.root

SUSY1-2.root

hadd step 2

SUSY3-4.root

SUSY5-6.root

SUSY7-8.root

SUSY1-4.root

hadd step 3

SUSY5-8.root

SUSY1-8.root

E.g. Hadd 8 files with 2 files per job

Recursive Hadding

- Simple usage ~mc3909/Scripts/recursiveHadd.py -i <inDirWithRootFiles>

- Works by submitting each hadding stage on the batch

- All jobs submitted at once but dependant (later) jobs wait for previous stage to finish before running

- Can configure min files to hadd per jobs as well as max jobs to submit per stage (sensible defaults!)

- Well tell user number of jobs/stages to finish before submitting

- Several configurables (output name, directories etc
  - run with --help)