Data Deduplication

Jeff Deifik

jeff@jdeifik.com

- Block based - requires custom file system

- File based - can be file system independent

- Primary storage - requires custom file system
  If one copy is modified, need to modify it, not all the copies

- Secondary storage - can be file system independent Ok for backup, read-only media - not good for general purpose use

- In-line - requires lots of cpu cycles, show, custom software

- Post-process - requires less cpu cycles, very portable

- I wrote a file based deduplicator, using post-processing, suitable for secondary storage. Available at jdeifik.com, GPL 2

gather data on all files - names, size, inode #

sort data by file size

for each group of data of same size files

unique list by inode #

compare each file against all the rest of the files

if they are the same then

figure out which file has a higher inode count then

more = file with more inode count

less = file with less inode count

- try to rename less to tmp
- try to link less to more
- try to remove less
- log the results

- Can work on a single directory tree (with the --one option)
- Can work on an active directory tree and a readonly tree
- Can work in verbose mode
- Can start with biggest files or smallest files
- Can ignore files smaller than a specified size (with the --size # option)