National Aeronautics and Space Administration



Shift New Features

Dec. 12th, 2018

NASA Advanced Supercomputing Division

www.nasa.gov



3 Slide Shift Overview

Motivation for Shift



- Let scientists focus on science without wading through documentation on transfer best practices
  - Specify transfers in simplest, naive fashion
    - Source and destination
- Provide tool to perform transfer as if scientist were HPC expert
  - Choose appropriate tools and optimize for best performance
  - Fully utilize available resources without starving other users
  - Manage files appropriately by file system type to ensure efficient access by later and/or behind the scenes processes

Shift's Three Main Capabilities

- Copying files (local/LAN/WAN)
  - **pfe%** shiftc /file1 /file2
  - **pfe%** shiftc /file1 lfe:/file2
  - **your_host%** sup shiftc /file1 pfe:/file2
- Synchronizing directories
  - **pfe%** shiftc --sync /dir1 lfe:/parent_dir1
- Creating/extracting tar files
  - **pfe%** shiftc --create-tar /dir1 /dir1.tar
  - **pfe%** shiftc --extract-tar /dir1.tar /dir1

More Information on Basic Usage

- Knowledge base article
  - http://www.nas.nasa.gov/hecc/support/kb/entry/300
- Man page
  - "man shiftc" on any HEC front-end
- Past webinars (slightly old, but mostly correct)
  - http://www.nas.nasa.gov/hecc/support/past_webinars.html
    - Simple Automated File Transfers Using SUP and Shift
    - Advanced Features of the Shift Automated File Transfer Tool
    - Simplifying and Optimizing Your Data Transfers



New Features

Silent Corruption Detection



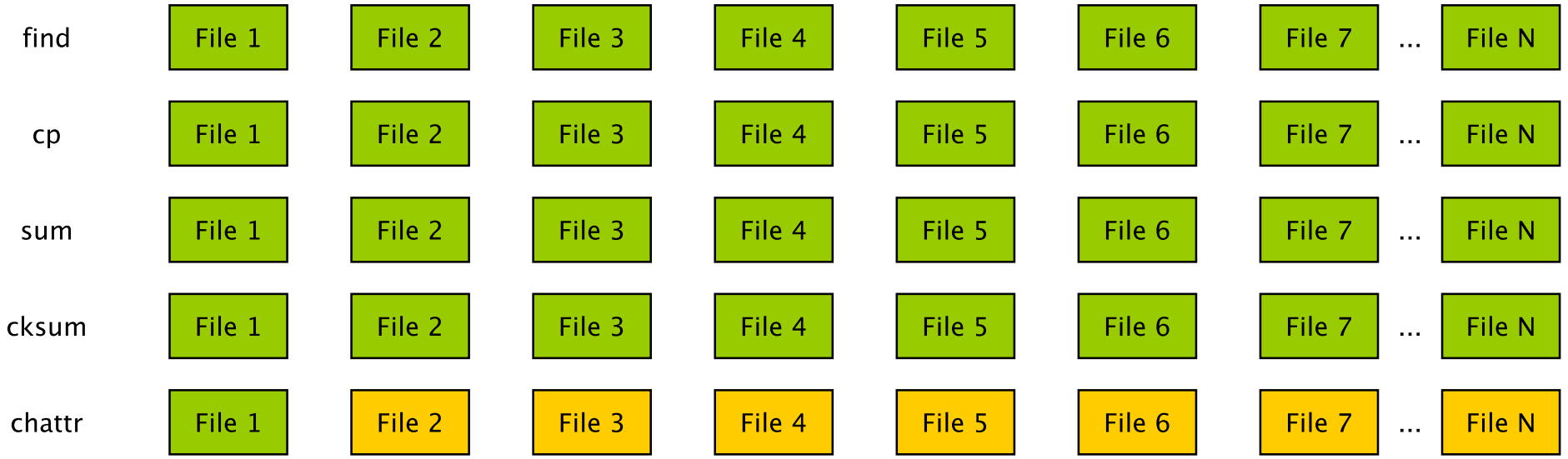
- Shift uses checksums to ensure data arrives intact
- Previously, no way to detect if file contents have changed while sitting on disk
- Now, Shift compares final checksums of previously transferred files with newly computed checksums
- If checksum differs but modification time the same, alert is added to transfer with affected byte ranges
  - "alert" is new transfer state used for info beyond scope of Shift
  - Currently, 1 GB granularity
  - Informational only - no way for Shift to act on silent corruption
- Checksum information can be queried
  - shiftc --last-sum /file
  - shiftc --last-sum --index-tar /file.tar file.in_tar

Overlapping Data Processing With Transfer

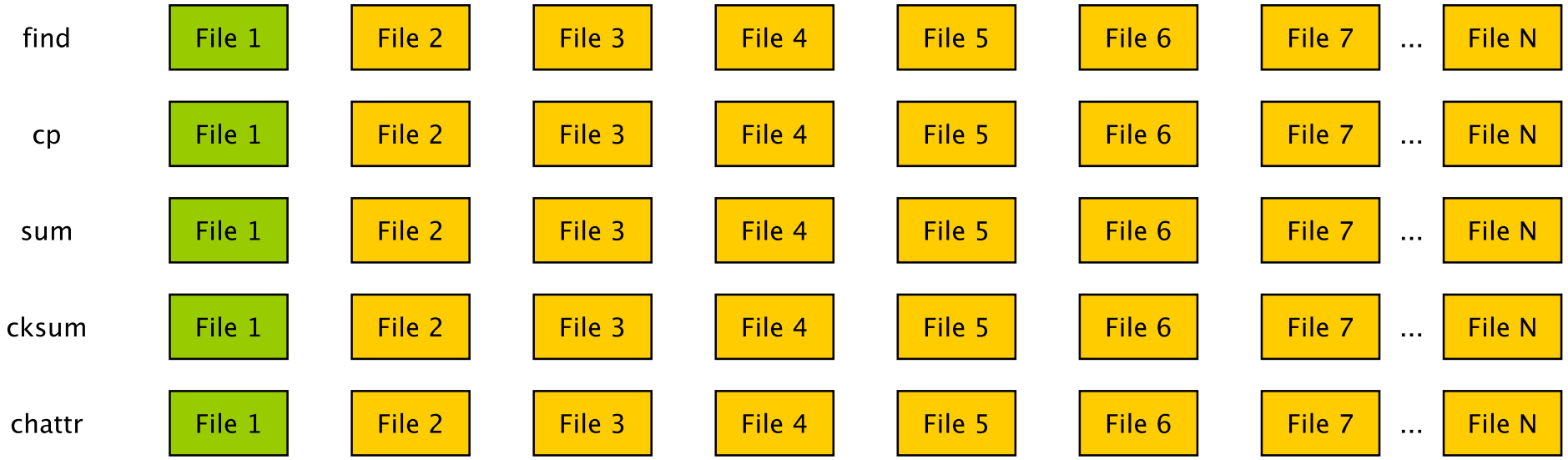


- During transfer of large files, may want to start processing completed files while others in progress
- The --pipeline option produces verified files sooner
  - Akin to CPU pipeline where different file stages being overlapped on different files to effectively produce one batch of verified files after each “cycle”
  - Can also think of as depth-first vs. breadth-first
- Parallel Shift clients prioritize different file stages

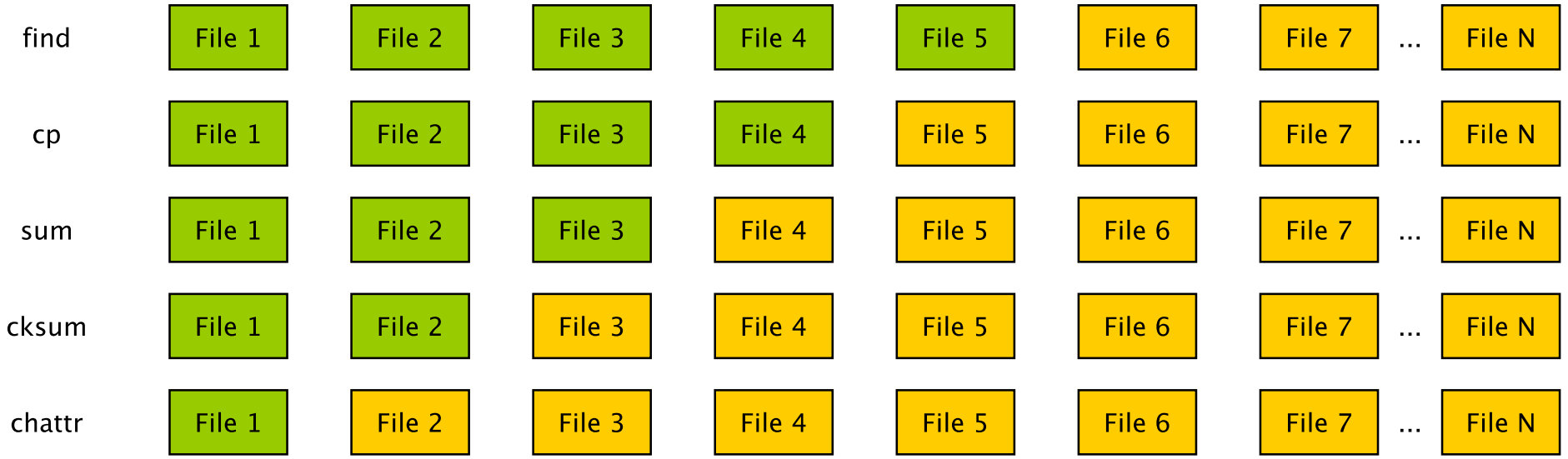
First Verified File (Default)



First Verified File (Serial --pipeline)



First Verified File (Parallel --pipeline)



Overlapping Data Processing With Transfer (cont.)



- Finding files that are done
  - shiftc --status=csv --id=N --state=done --search=chattr
- Danger of doing this
  - Files are cached in various places on various systems
  - More likely to hit cache during sum when read earlier
  - Uncaught bit flip during src read may be propagated to sum
- Built-in mitigations
  - Files are now fadvise'd out of client cache
  - Different stages processed on different clients if possible
  - Configurable size/time at which files likely out of cache

Monitor

- Previously, could run --status repeatedly to track transfer
- Now, can use --monitor to get live real-time status
  - Status is ”real-time” at the granularity Shift tracks it
  - Does not show partial progress of individual file batches
- Monitor all running transfers
  - shiftc --monitor
- Monitor specific transfer
  - shiftc --monitor --id=N
- Monitor transfer while waiting
  - shiftc --monitor --wait /src /dst
  - Akin to what you would see when you run “scp /src /dst”

Dynamic Striping



- Previously, could allocate one stripe per unit of size (still the default)
- Now, can define arbitrary expressions for stripe count, size, and pool based on source file name, size, and existing stripe size/count
- Expressions can use arbitrary perl with special constants
  - NM/SZ/SC/SS for src name/size/stripe count/stripe size
  - --stripe=count_expr::size_expr::pool_expr
- For example, dynamic stripe count based on src size
  - --stripe='SZ < 10g ? 1 : (SZ < 64g ? 1g : int(sqrt(SZ)))'
- Note, Lustre is getting smarter with progressive file layouts so striping not as critical

More Flexible --older/--newer



- Previously, --older and --newer based only on modification time (still the default)
- Now, can optionally specify arbitrary combinations of atime, mtime, and ctime and their negations
  - Format: [acmACM]+(l[acmACM]+)*
- Examples
  - File created or modified before Aug. 15, 2018
    - --older='mlc:8/15/18'
  - File accessed after 4pm today but not modified
    - --newer='aM:16:00'

Faster Initialization



- Previously
  - Some file attributes cannot be manipulated by perl core
  - Separate programs to get/set file information
    - Striping, ACLs, extended attributes, preallocation, fadvise
  - Batch operations not efficiently supported so required one call per program per file
- Now
  - Custom C binary “shift-bin”
  - One call total per file batch for all get/set operations
  - Orders of magnitude faster
    - Just one aspect of initialization
    - Traversing file system still takes time

Selective Email Status

- By default, emails are sent for the first instance of any state combination during a transfer
  - States include alert, done, error, run, stop, throttle, warn
- Previously, could disable email completely
- Now, can exclude emails for subset of states
- For example, do not send status regarding warnings or throttling
  - --no-mail=warn,throttle

Selective Attribute Preservation



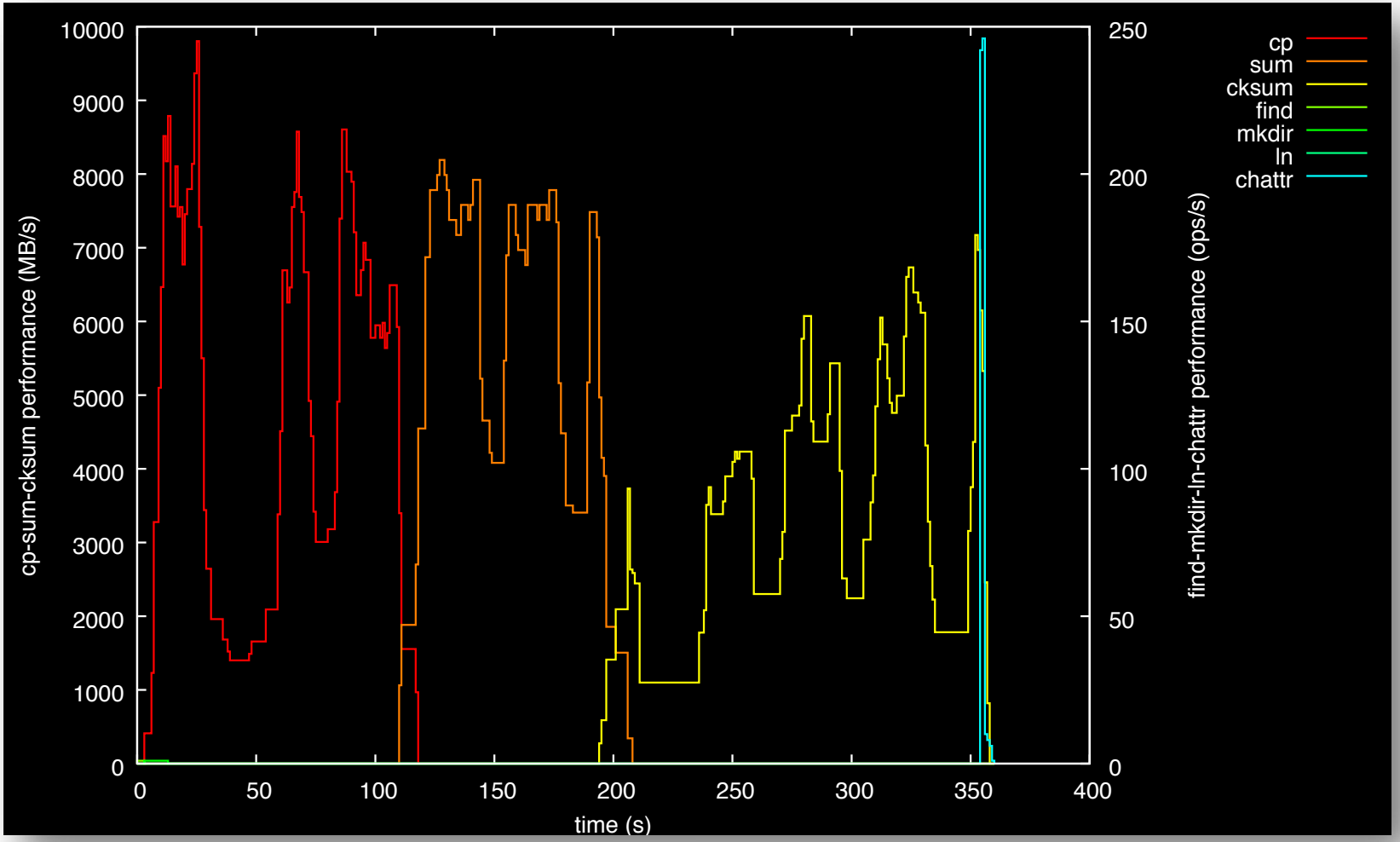
- By default, Shift attempts to preserve all possible file attributes
  - Attributes include acl, mode, owner, stripe, time, xattr
- Previously, could disable preservation completely
- Now, can exclude subset of attributes from preservation
- For example, do not preserve ACLs or extended attributes
  - --no-preserve=acl,xattr
- Can speed up initialization and final chattr stage
  - Less benefit after shift-bin optimization

Plotting Performance

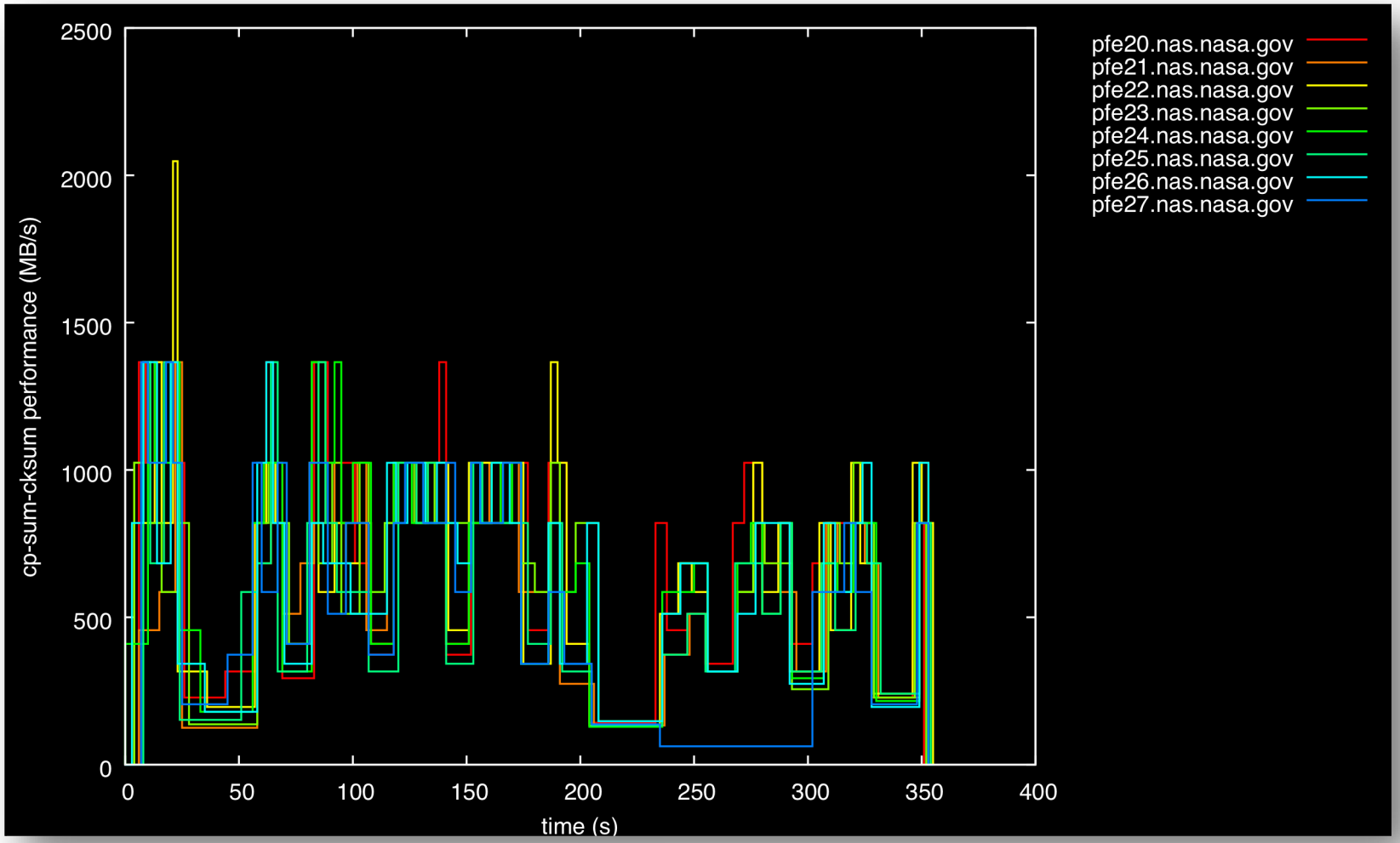


- Rate given in status a composite with many factors
  - Difficult to see how individual stages/hosts/etc. performing
- The --plot option produces output suitable for gnuplot
  - Requires gnuplot >= 5.0 with pdf terminal (need pkgsrc version on pfe)
- Examples
  - Plot all transfers
    - shiftc --plot lgnuplot >all.pdf
  - Plot single transfer
    - shiftc --plot --id=N lgnuplot >id.pdf
  - Plot single transfer by host
    - shiftc –plot=host --id=N lgnuplot >id_host.pdf

Plotting Single Transfer



Plotting Single Transfer By Host



Ignoring Unrecoverable Errors



- Many errors are not possible for Shift to correct
  - e.g. source file/dir removed
- Such errors may inhibit other operations from completing successfully
- Previously, there was no way to recover from this
- Now, new option --restart=ignore
  - Transfer proceeds as if files with errors were successful
  - New errors still tracked normally
    - Can use --restart=ignore as many times as needed
- Best to wait until transfer ends in ”error” state
  - i.e. don’t --stop immediately after first error and restart
- Make sure you are not ignoring something important!

Other Additions



- Force overwrite of read-only files using --force
- Now overwrites dst symlinks
- fish-tcp transport encrypts data when using --secure
  - You must have IO::Socket::SSL perl module installed
  - Performance will be decreased
- Better support of user@ syntax
  - Previously chopped off, now utilized in remote transports
- Many fixes and improvements behind the scenes

Questions?



- Always feel free to contact me directly
  - If something not working properly
    - Some bugs would never have been found without users reporting them
  - If there are features/behavior you would like to see
    - Several features exist because users requested them
  - If you have general usage questions
  - Best is email: paul.kolano@nasa.gov