Introduction to NERSC Archival Storage: HPSS

Lisa Gerhardt
Nick Balthaser
Storage Systems Group

September 10, 2013
What is an archive?

• **Long-term storage of permanent records and information**
  – Often data that is no longer modified or regularly accessed
  – Storage time frame is indefinite or as long as possible
  – Archive data typically has, or may have, long-term value to the organization

• **NERSC archiving system uses HPSS (high performance storage system) software**

• **Typically used for:**
  – Long-term storage of very large raw data sets
    • Good for incremental processing
  – Long-term storage of result/processed data
  – Backups (e.g. global scratch purges)
Features of HPSS

• The NERSC archive is a Hierarchical Storage Management system (HSM)
  • NERSC archive supports parallel high-speed transfer and fast data access
• Highest performance at top level
• Lowest cost, greatest capacity at lower levels
• Migration between levels is automatic, based on policies
• HPSS responds differently than a file system

The NERSC archive is a shared multi-user system
• Shared resource, no batch system. Inefficient use affects others.
• Session limits are enforced
Using HPSS
How to Log In

• The NERSC archive uses an encrypted key for authentication
  – Key placed in ~/.netrc file at the top level of the user’s home directory on the compute platform
  – All NERSC HPSS clients use the same .netrc file
  – The key is IP specific. Must generate a new key for use outside the NERSC network.

• Archive keys can be generated in two ways
  – Automatic: NERSC auth service
    • Log into any NERSC compute platform using ssh
    • Type “hsi”
    • Enter NERSC password
  – Manual: https://nim.nersc.gov/ web site
    • Under “Actions” drop down, select “Generate HPSS Token”
    • Copy/paste content into ~/.netrc
    • chmod 600 ~/.netrc
Storing and Retrieving Files with HSI

• HSI provides a Unix-like command line interface for navigating archive files and directories
  – Standard Unix commands such as *ls*, *mkdir*, *mv*, *rm*, *chown*, *chmod*, *find*, etc. are supported

• FTP-like interface for storing and retrieving files from the archive (put/get)
  – Store from file system to archive:
    -bash-3.2$ hsi
    A:/home/n/nickb-> put myfile
    put 'myfile' : '/home/n/nickb/myfile' ( 2097152 bytes, 31445.8 KBS (cos=4))

  – Retrieve file from archive to file system:
    A:/home/n/nickb-> get myfile
    get 'myfile' : '/home/n/nickb/myfile' (2010/12/19 10:26:49 2097152 bytes, 46436.2 KBS )

  – Full pathname or rename file during transfer:
    A:/home/n/nickb-> put local_file : hpss_file
    A:/home/n/nickb-> get local_file : hpss_file
Storing and Retrieving Directories with HTAR

- HTAR stores a Unix tar-compatible bundle of files (aggregate) in the archive
  - Traverses subdirectories like tar
  - No local staging space required—aggregate stored directly into the archive

- Recommended utility for storing small files

- Some limitations
  - 5M member files
  - 64GB max member file size
  - 155/100 path/filename character limitation
  - Max archive file size* currently 20TB

- Syntax: `htar [options] <archive file> <local file|dir>

  - Store
    - `bash-3.2$ htar -cvf /home/n/nickb/mydir.tar ./mydir`
  - List
    - `bash-3.2$ htar -tvf /home/n/nickb/mydir.tar`
  - Retrieve
    - `bash-3.2$ htar -xfv /home/n/nickb/mydir.tar [file...]`

* By configuration, not an HPSS limitation
Avoiding Common Mistakes
Small Files

• Tape storage systems do not work well with large numbers of small files
  – Tape is sequential media—tapes must be mounted in drives and positioned to specific locations for IO to occur

• Mounting and positioning tapes are the slowest system activities
  – Small file retrieval incurs delays due to high volume of tape mounts and tape positioning
  – Small files stored periodically over long periods of time can be written to hundreds of tapes—especially problematic for retrieval

• Use HTAR when possible to optimize small file storage and retrieval

• Recommend file sizes in the 10s – 100s of GB
Large Directories

• Each HPSS system is backed by a single metadata server
  - Metadata is stored in a single SQL database instance
  - Every user interaction causes database activity

• Metadata-intensive operations incur delays
  - Recursive operations such as “chown –R ./*” may take longer than expected
  - Directories containing more than a few thousand files may become difficult to work with interactively

-bash-3.2$ time hsi -q ‘ls -l /home/n/nickb/tmp/testing/80k-files/’ > /dev/null 2>&1

real 20m59.374s
user   0m7.156s
sys    0m7.548s
Large Directories, continued

- hsi “ls –l” exponential delay:

![Graph showing HSI 'ls -l' performance and timing]
Long-running Transfers

- Failure prone for a variety of reasons
  - Transient network issues, planned/unplanned maintenance, etc.
- Many clients do not have capability to resume interrupted transfers
- Can affect archive internal data management (migration) performance
- Recommend keeping transfers to 24hrs or less if possible
Session Limits

• 15 concurrent session/user enforced
• Can be administratively reduced if a user is negatively affecting system usability for others
- Charging -

- **Storage and bandwidth are allocated and charged as part of a NERSC repository**
  - Exhausting an HPSS allocation is rare
  - See the NERSC web site for details
Questions, Problems, Further Reading
• Contact NERSC Consulting
  – Toll-free 800-666-3772
  – 510-486-8611, #3
  – Email consult@nersc.gov.
Further Reading

• Hands on examples at end of this talk
• NERSC Website
• HSI and HTAR man pages are installed on NERSC compute platforms
• Gleicher Enterprises Online Documentation (HSI, HTAR)
  – http://www.mgleicher.us/index.html/hsi/
  – http://www.mgleicher.us/index.html/htar/
• “HSI Best Practices for NERSC Users,” LBNL Report #LBNL-4745E
Hands-on Examples
Logging into archive: Hands-on

- Using ssh, log into any NERSC compute platform
  
  ```bash
  -bash-3.2$ ssh dtn01.nersc.gov
  ```

- Start HPSS storage client “hsi”
  
  ```bash
  -bash-3.2$ hsi
  ```

- Enter NERSC password at prompt (first time only)
  
  Generating .netrc entry...
  nickb@auth2.nersc.gov's password:

- You should now be logged into your archive home directory
  
  Username: nickb  UID: 33065  Acct: 33065(33065) Copies: 1 Firewall: off [hsi.3.4.5 Wed Jul 6 16:14:55 PDT 2011][V3.4.5_2010_01_27.01]
  A:/home/n/nickb-> quit

- Subsequent logins are now automated
Using HSI: Hands-on

- Using ssh, log into any NERSC compute platform
  ```bash
  ssh dtn01.nersc.gov
  ```
- Create a file in your home directory
  ```bash
  echo foo > abc.txt
  ```
- Start HPSS storage client “hsi”
  ```bash
  hsi
  ```
- Store file in archive
  ```bash
  hsi put abc.txt
  ```
- Retrieve file and rename
  ```bash
  hsi get abc_1.txt : abc.txt
  ```
- Compare files*
  ```bash
  sha1sum abc.txt abc_1.txt
  ```
  ```bash
  f1d2d2f924e986ac86f6f7b36c94bcd8f32beec15 abc.txt
  f1d2d2f924e986ac86f6f7b36c94bcd8f32beec15 abc_1.txt
  ```

* Note: checksums supported in the next HSI release with: ‘hsi ‘put –c on local_file : remote_file’
Using HTAR: Hands-on

- Using ssh, log into any NERSC compute platform
  - bash-3.2$ ssh dtn01.nersc.gov
- Create a subdirectory in your home directory
  - bash-3.2$ mkdir mydir
- Create a few files in the subdirectory
  - bash-3.2$ echo foo > ./mydir/a.txt
  - bash-3.2$ echo bar > ./mydir/b.txt
- Store subdirectory in archive as “mydir.tar” with HTAR
  - bash-3.2$ htar –cvf mydir.tar ./mydir
- List newly created aggregate in archive
  - bash-3.2$ htar –tvf mydir.tar
- Remove local directory and contents
  - bash-3.2$ rm –rf ./mydir
- Extract directory and files from archive
  - bash-3.2$ htar –xvf mydir.tar
National Energy Research Scientific Computing Center