DE LA RECHERCHE À L'INDUSTRIE

www.cea.fr

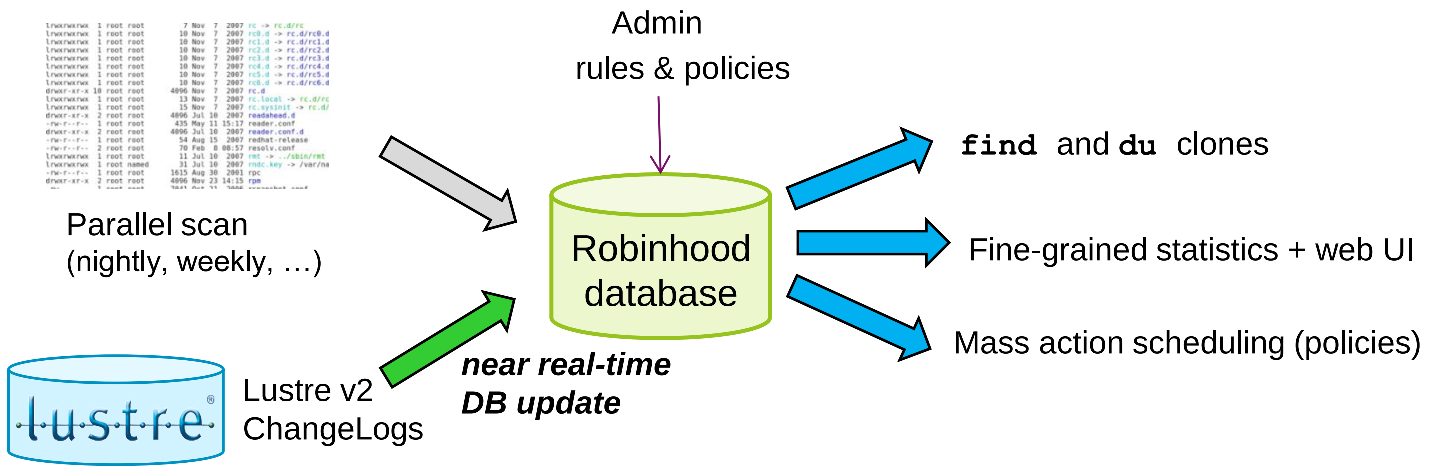
ROBINHOOD POLICY ENGINE

Aurélien DEGREMONT
Thomas LEIBOVICI

CEA/DAM

LUSTRE USER GROUP 2013
16-18 APRIL 2013

ROBINHOOD: BIG PICTURE



```
lrwxrwxrwx  1 root root        7 Nov  7  2007 rc -> rc.d/rc
lrwxrwxrwx  1 root root       10 Nov  7  2007 rc0.d -> rc.d/rc0.d
lrwxrwxrwx  1 root root       10 Nov  7  2007 rc1.d -> rc.d/rc1.d
lrwxrwxrwx  1 root root       10 Nov  7  2007 rc2.d -> rc.d/rc2.d
lrwxrwxrwx  1 root root       10 Nov  7  2007 rc3.d -> rc.d/rc3.d
lrwxrwxrwx  1 root root       10 Nov  7  2007 rc4.d -> rc.d/rc4.d
lrwxrwxrwx  1 root root       10 Nov  7  2007 rc5.d -> rc.d/rc5.d
lrwxrwxrwx  1 root root       10 Nov  7  2007 rc6.d -> rc.d/rc6.d
drwxr-xr-x 10 root root     4096 Nov  7  2007 rc.d
lrwxrwxrwx  1 root root       13 Nov  7  2007 rc.local -> rc.d/rc
lrwxrwxrwx  1 root root       15 Nov  7  2007 rc.sysinit -> rc.d/
drwxr-xr-x  2 root root     4096 Jul 10  2007 readahead.d
-rw-r--r--  1 root root      435 May 11 15:17 reader.conf
drwxr-xr-x  2 root root     4096 Jul 10  2007 reader.conf.d
-rw-r--r--  1 root root       54 Aug 15  2007 redhat-release
-rw-r--r--  2 root root       70 Feb  8 08:57 resolv.conf
lrwxrwxrwx  1 root root       11 Jul 10  2007 rmt -> ../sbin/rmt
lrwxrwxrwx  1 root named      31 Jul 10  2007 rndc.key -> /var/na
-rw-r--r--  1 root root     1615 Aug 30  2001 rpc
drwxr-xr-x  2 root root     4096 Nov 23 14:15 rpm
```

Scan sometimes (or never), query at will

1. Fill the database
2. Apply rules
3. Query at will for:
    - Searches
    - Statistics
    - Actions

DE LA RECHERCHE À L'INDUSTRIE

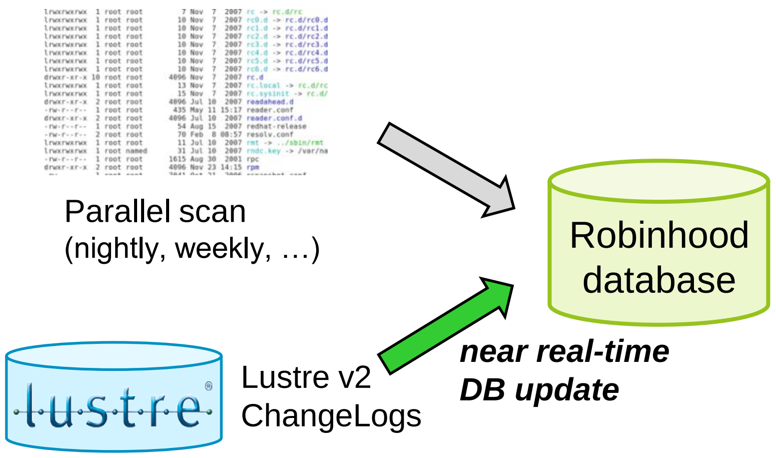
cea

ANALYZING FILESYSTEM

Feeding the database

- Robinhood information and actions are based upon the database data.
  - Robinhood supports MySQL as backend.

- Database could be filled using:
  - Parallel filesystem scan.
    - For Lustre 1.8 or any POSIX filesystem.
  - Reading Lustre Changelog
    - For Lustre 2.x
    - Only an initial scan is needed.



```
lrwxrwxrwx  1 root root        7 Nov  7  2007 rc -> rc.d/rc
lrwxrwxrwx  1 root root       10 Nov  7  2007 rc0.d -> rc.d/rc0.d
lrwxrwxrwx  1 root root       10 Nov  7  2007 rc1.d -> rc.d/rc1.d
lrwxrwxrwx  1 root root       10 Nov  7  2007 rc2.d -> rc.d/rc2.d
lrwxrwxrwx  1 root root       10 Nov  7  2007 rc3.d -> rc.d/rc3.d
lrwxrwxrwx  1 root root       10 Nov  7  2007 rc4.d -> rc.d/rc4.d
lrwxrwxrwx  1 root root       10 Nov  7  2007 rc5.d -> rc.d/rc5.d
lrwxrwxrwx  1 root root       10 Nov  7  2007 rc6.d -> rc.d/rc6.d
drwxr-xr-x 10 root root     4096 Nov  7  2007 rc.d
lrwxrwxrwx  1 root root       13 Nov  7  2007 rc.local -> rc.d/rc
lrwxrwxrwx  1 root root       15 Nov  7  2007 rc.sysinit -> rc.d/
drwxr-xr-x  2 root root     4096 Jul 10  2007 readahead.d
-rw-r--r--  1 root root      435 May 11 15:17 reader.conf
drwxr-xr-x  2 root root     4096 Jul 10  2007 reader.conf.d
-rw-r--r--  1 root root       54 Aug 15  2007 redhat-release
-rw-r--r--  2 root root       70 Feb  8 08:57 resolv.conf
lrwxrwxrwx  1 root root       11 Jul 10  2007 rmt -> ../sbin/rmt
lrwxrwxrwx  1 root named      31 Jul 10  2007 rndc.key -> /var/na
-rw-r--r--  1 root root     1615 Aug 30  2001 rpc
drwxr-xr-x  2 root root     4096 Nov 23 14:15 rpm
```

RBH-FIND, RBH-DU

Fast *find* and *du* clones

- Query Robinhood DB instead of performing POSIX namespace scan

➔ faster!

```
> rbh-find [path] -user "foo*" -size +1G -ost 4
```

➢ 20sec for 40M entries

- Enhanced *du* :
  - Detailed stats (by type…)
  - Can filter by user

```
> rbh-du -sH /fs/dir -u foo --details
/fs/dir
        symlink count:30777,   size:1.0M, spc_used:9.1M
        dir     count:598024,  size:2.4G, spc_used:2.4G
        file    count:3093601, size:3.2T, spc_used:2.9T
```

DE LA RECHERCHE À L'INDUSTRIE

cea

ROBINHOOD: FINE-GRAINED STATISTICS

Usage statistics

- Per user, per group, per type, …
  - Possibly split user usage by group

```
> rbh-report -u foo -S

user   ,     group,    type,      count,   spc_used,   avg_size
foo    ,   proj001,     dir,          2,    8.00 KB,    4.00 KB
foo    ,       gr1,     dir,      74441,  291.64 MB,    4.01 KB
foo    ,       gr1,    file,     422367,   71.01 GB,  335.54 KB
foo    ,       gr1, symlink,       1418,    1.35 MB,         46
foo    ,   proj002,    file,          2,    1.27 GB,  651.43 MB

Total: 498230 entries, 77918785024 bytes used (72.57 GB)
```

ROBINHOOD: FINE-GRAINED STATISTICS

Top users and groups

- Sorted by volume, object count, avg file size…

```
> rbh-report --top-users --by-count
rank, user       ,    spc_used,      count,   avg_size
   1, john       ,   423.23 GB,    1599881,   275.30 KB
   2, paul       ,   292.91 GB,     954153,   330.98 KB
   3, mike       ,    65.37 GB,     543169,   130.98 KB
…
```

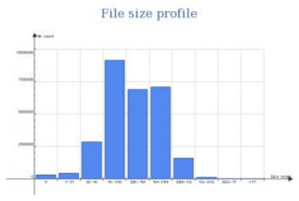
Top directories

- Sorted by object count, avg file size…

```
> rbh-report --top-dirs --by-count
rank,                path, dircount,   avgsize, user, group,  last_mod
   1, /hpss/foo1/dir1,    24832,   2.62 GB, foo1, gr59,  2013/03/11 17:13:45
   2, /hpss/foo2/dir3,    20484, 339.88 MB, foo2,  g03,  2013/02/03 06:59:05
   3, /hpss/bar2/dir4,    19484, 543.82 MB, bar2,  g03,  2012/05/28 12:45:26
…
```

ROBINHOOD: FINE-GRAINED STATISTICS

File size profile



- Global or for a given user/group:

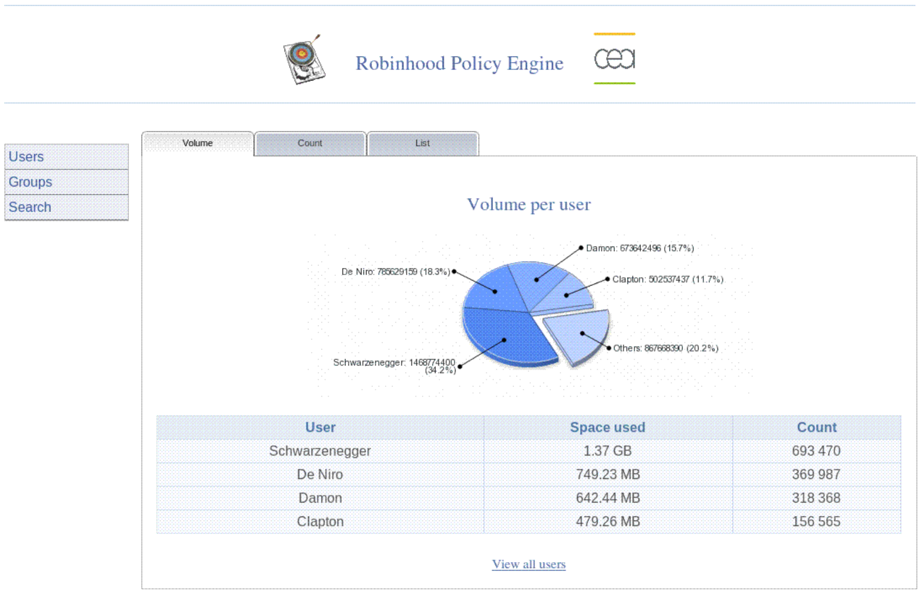
```
> rbh-report –u foo --sz-prof
user, type, count,  spc_used, avg_size,    0,  1~31, 32~1K-,  1K~31K,  32K~1M-,   1M~31M,  32M~1G-,   1G~31G,  32G~1T-,   +1T
foo , file, 62091,  58.50 TB,  1.09 GB,   40,    44,    233,     548,     1879,     5820,     8004,     1835,      178,     0
```

- User, group, directories can also be sorted by the ratio of file in a given range:

```
> rbh-report --top-users --by-szratio=1..31M

rank, user     , ratio(1..31M)
1, john        ,    91.30%
2, perrez      ,    87.64%
3, matthiew    ,    85.76%
4, vladimir    ,    78.50%
5, gino        ,    77.02%
…
```

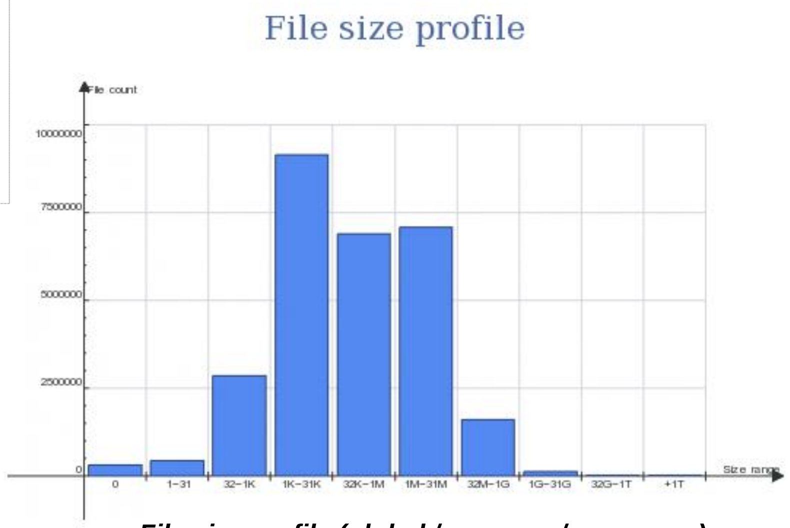
WEB INTERFACE

Web UI


Robinhood Policy Engine
Users
Groups
Search
Volume
Count
List
Volume per user
Damon: 673642496 (15.7%)
Clapton: 502537437 (11.7%)
De Niro: 785629159 (18.3%)
Others: 867668390 (20.2%)
Schwarzenegger: 1468774400 (34.2%)
View all users

User	Space used	Count
Schwarzenegger	1.37 GB	693 470
De Niro	749.23 MB	369 987
Damon	642.44 MB	318 368
Clapton	479.26 MB	156 565

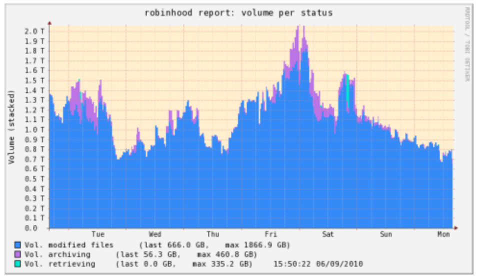
Usage stats (per user, per group)



File size profile (global / per user / per group)

Mass action scheduling on filesystem entries

- Admin-defined rules
- Build-in policies:
  - Purge
  - Directory removal
  - Deferred removal (undelete)
  - Archiving
  - HSM: schedule 'archive' and 'release' actions
- Policy definition:
  - Flexible and highly customizable
  - Attribute-based
  - Using fileclass definitions
- Example:



```
fileclass BigLogFiles {
    definition { type == file and size > 100MB
                     and (path == /fs/logdir/*
                            or name == *.log) }
 …
}
```

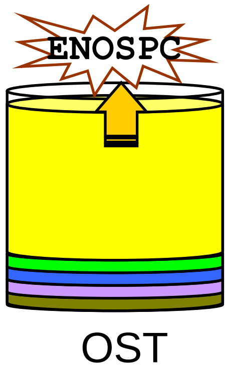
```
purge_policies {
    ignore_fileclass = my_fileclass;

    policy purge_logs {
         target_fileclass = BigLogFiles;
         condition { last_mod > 15d }
    }
}
```

PURGE POLICIES

Use cases

- Managing file lifetime, cleaning tmp data
  - Cleanup after a code run (lifetime: few hours)
  - Cleanup after a simulation (lifetime: several months)
  - Clean old krb tickets, logs, core dumps, crash dumps, …

- Avoid "ENOSPC" errors caused by full OSTs
  - Admin defines high/low OST usage thresholds and purge policy rules
  - Robinhood monitors free space per OST
    - ➔ Applies purge policies on OSTs that exceed high threshold

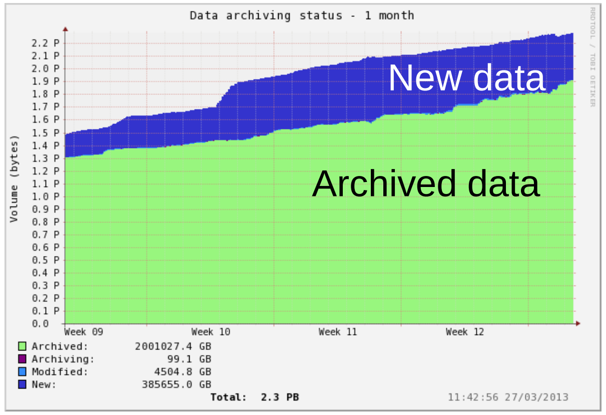


- Disk space fair-share
  - Admin defines max usage per user or group
  - Robinhood monitors usage per user/group
    - ➔ Applies purge policies for user/group over their limit

ARCHIVING POLICIES

An alternative to ‘rsync’

- No need to scan each time you want to archive data
- Policy-driven:
  - Can skip some kind of files
  - Can control the delay before archiving a file
- Fine control of copy streams:
  - number of simultaneous copies in parallel
  - max files/hour, max volume/hour
- Robinhood detects file modification and maintains file status in its DB
  - report available with current status of files (new/archived/modified…)



Allows undelete

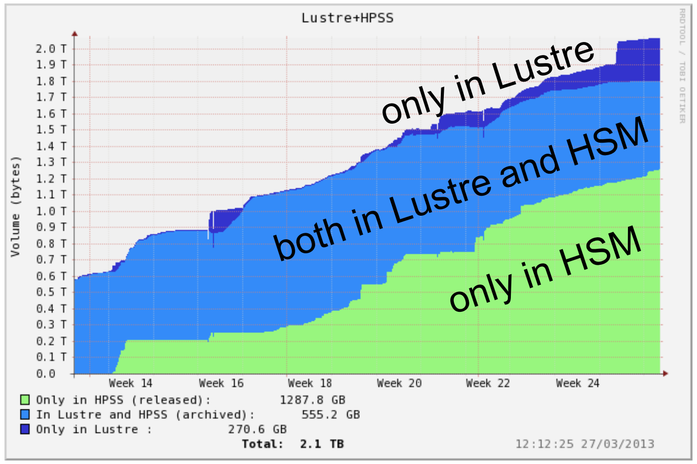
- Removed files in Lustre are not immediately removed in backup space
  ➔ Configurable delay before cleaning the backup copy
  - Can undelete a file during this delay

ROBINHOOD AND LUSTRE-HSM

Lustre-HSM support

- **Archiving** policies to schedule copy from Lustre to the HSM
- Purge policies to **release disk space** in Lustre OSTs when needed (file remains visible in Lustre for users)
- GC of deleted files
- **Undelete**
- **Disaster recovery**: to rebuild a Lustre filesystem from the archive

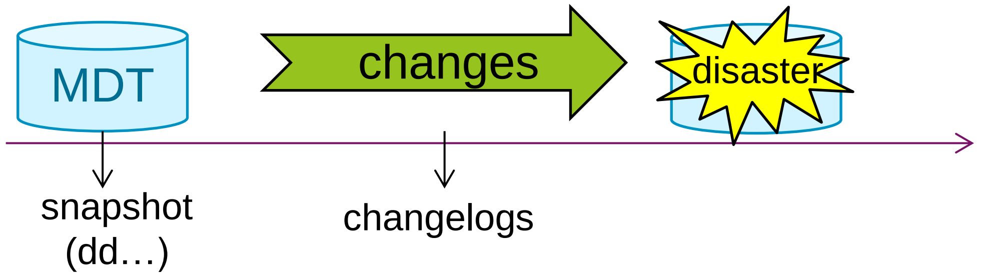
- Aware of 'HSM' specific changelog records:
  - To keep track of file status



WORK IN PROGRESS

MDS disaster recovery

- Scenario:



  - 1) Restore MDT snapshot
  - 2) Replay changes between snapshot and disaster (rbh-diff / rbh-apply)

New policies

- **Rebalance** files between OSTs / **migrate** files between OST pools
- **Generic policies** to schedule any kind of action, e.g.
  - Datascrubbing (to detect silent corruption)
  - MD scrubbing (FS consistency check)
  - Run any command on FS entries…

FUTURE WORK

Distributed database

- For now: 1 single MySQL database
- Need to distribute the database:
  - To handle higher MD rate (DNE)
  - To manage xx billions of entries

Turn it to a framework

- API to extract customized reports
- All components as dynamic customizable modules:
  - Policy criteria
  - Policies
  - Stats
  - Filesystem backend, DB backend…

ROBINHOOD WEBSITE & SUPPORT

Last release: robinhood 2.4.2

Project home: http://robinhood.sourceforge.net

Mailing lists:

robinhood-news@lists.sourceforge.net

robinhood-support@lists.sourceforge.net

robinhood-devel@lists.sourceforge.net