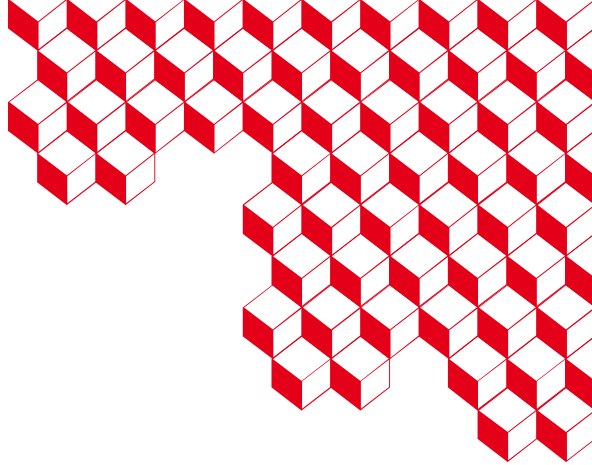


RobinHood v4 Progress Report

LAD’23 - 6th of October, 2023

Yoann VALERI, yoann.valeri@cea.fr

Commissariat à l’énergie atomique et aux énergies alternatives - www.cea.fr

Focus on RobinHood

Providing efficient and easy to use means to replicate and query any filesystem's metadata

Reminder of last year's news

- Tools in development:
  - **rbh-lustre-find**: query and filter data in a backend based on Lustre specific attributes
  - **rbh-fsevents**: update a backend based on events provided by a source
    - **CREAT**, **TIME** events and **CLOSE**
  - **rbh-find**: new filters (size, permissions, extended attributes) and actions (listing, sorting, dumping to files)

What’s new since last year ?

- **rbh-lustre-find** is near completion:
  - General tool and usage is done
  - Works like **rbh-find** -> supports its filters and additional ones for Lustre
  - Filters currently implemented: FID, HSM state and OST used
  - Usable by invoking **rbh-lfind**

Example of rbh-lfind usage

```
# content of `test_dir`
one   [0x1:0x2:0x3] (archived) [0]
two   [0x4:0x5:0x6] (released) [1,2]
three [0x7:0x8:0x9] (none)     [2,3,4]

# rbh-sync rbh:lustre:test_dir rbh:mongo:test
# rbh-lfind rbh:mongo:test -hsm-state archived
/one

# rbh-lfind rbh:mongo:test -ost 1
/two

# rbh-lfind rbh:mongo:test -fid [0x7:0x8:0x9]
/three

# rbh-lfind rbh:mongo:test -hsm-state none -or -ost 0
/one
/three

# rbh-lfind rbh:mongo:test -ost 2 -and hsm-state released
/two

# rbh-lfind rbh:mongo:test -ost 4 -or \
          ( -hsm-state archived -and -fid [0x4:0x5:0x6] )
/three
```

What's new since last year ? (2)

- **rbh-fsevents** is also near completion:
  - Lustre changelog reader/enricher done
    - All events are managed
  - Update to the database done
  - Deduplication work on-going, soon to be finished
- **rbh-fsevents [--enrich URI] [--lustre] <source> <destination>**
  - The **--lustre** is currently necessary if the source is a changelog reader, but the source will be changed in the future into a URI
  - Destination can either be a URI (for instance to update a Mongo database) or the standard output (with "**-**")

Example of rbh-fsevents usage

```
# touch test_archive
# lfs hsm_archive test_archive
# setfattr -n user.test -v 42 test_archive
# touch test_pfl
# lfs migrate -E 1k -c 2 -E -1 -c 1 test_pfl

# lfs changelog lustre-MDT0000
01 01CREAT … t=[0x200000404:0x8e:0x0] j=touch.0 … p=[0x200000404:0x8d:0x0] test_archive
02 11CLOSE … t=[0x200000404:0x8e:0x0] j=touch.0 …
03 16HSM   … 0x280 t=[0x200000404:0x8e:0x0] …
04 16HSM   … 0x280 t=[0x200000404:0x8e:0x0] j=lhsmtool_posix.0 … p=[0x200000404:0x8d:0x0]
05 16HSM   … 0x0 t=[0x200000404:0x8e:0x0] j=lhsmtool_posix.0 … p=[0x200000404:0x8d:0x0]
06 15XATTR … t=[0x200000404:0x8e:0x0] j=setfattr.0 … x=user.test
07 01CREAT … t=[0x200000404:0x94:0x0] j=touch.0 … p=[0x200000404:0x8d:0x0] test_pfl
08 11CLOSE … t=[0x200000404:0x94:0x0] j=touch.0 …
09 12LYOUT … t=[0x200000404:0x94:0x0] … p=[0x200000404:0x8f:0x0]
10 11CLOSE … t=[0x200000404:0x94:0x0] j=lfs.0 …
```

Example of rbh-fsevents usage (2)

```
# rbh-fsevents --enrich rbh:lustre:/mnt/lustre --lustre lustre-MDT0000 rbh:mongo:test
# mongo 'test' --eval 'db.entries.find({"ns.name":"test_archive"})'"
'{ "ns" : [ {"name" : "test_archive", "xattrs" : { "path" : "/test_archive" } " ],
   "xattrs" : {
        "fid" : BinData(0,"BAQAAAIAAACOAAAAAAAAAA=="),
        "hsm_archive_id" : 1, "hsm_state" : 9,
        "mdt_index" : 0, "ost" : [ 3 ],
        "user" : { "test" : BinData(0,"NDI=") } } }'

# mongo 'test' --eval 'db.entries.find({"ns.name":"test_pfl"})'"
'{ "ns" : [ {"name" : "test_pfl", "xattrs" : { "path" : "/test_pfl" } } ],
   "xattrs" : {
        "begin" : [ 0, 1048576 ], "comp_flags" : [ 16, 0 ],
        "end" : [ 1048576, -1 ], "fid" : BinData(0,"BAQAAAIAAACUAAAAAAAAAA=="),
        "flags" : 0, "gen" : 3, "mdt_index" : 0, "ost" : [ 2, 3, -1 ],
        "pattern" : [ 0, 0 ], "pool" : [ "", "" ],
        "stripe_count" : [ 2, 1 ], "stripe_size" : [ 1048576, 1048576 ] } }'
```

What's new since last year ? (3)

- Developments done for a new backend for the European project IO-SEA
- Based on a component developed in it, the Hestia object store
- Includes:
  - A new backend for **librobinhood**
  - A new tool for query and filtering, **rbh-iosea-find**

What's new since last year ? (4)

- A new retention feature
- Allows one to set an expiration date on files by using an extended attribute
- Support:
  - An absolute expiration date
  - A relative expiration date, based on the maximum between the access and modify time of the file
  - An infinite retention date
- Tied to the Lustre backend currently, but will be changed in the future
- Can be queried by using the **expired** and **expired-at** filters of **rbh-lfind**
- The **expired-at** can also check for entries that expire before or after the given date

Example of the retention feature

```
# ls -color test_dir
fileA
fileB
fileC

# now=$(date +%s)
# setfattr -n user.expires -v $((now - 30)) test_dir/fileA
# setfattr -n user.expires -v inf test_dir/fileB
# setfattr -n user.expires -v +30 test_dir/fileC

# rbh-sync rbh:lustre:test_dir rbh:mongo:test
# rbh-lfind rbh:mongo:test -expired
/fileA

# rbh-lfind rbh:mongo:test -expired-at inf
/fileB

# rbh-lfind rbh:mongo:test -expired-at -$(date +%s --date='1 hour')
/fileA
/fileC

# rbh-lfind rbh:mongo:test -expired-at +$(date +%s --date='10 seconds')
/fileC

# sleep 120
# rbh-lfind rbh:mongo:test -expired
/fileA
/fileC
```

Some early benchmarks

- 2 million inodes synchronised
- CentOS Linux release 8.4.2105
- Lustre 2.12.9, Robinhood-lustre 3.1.7
- Same material

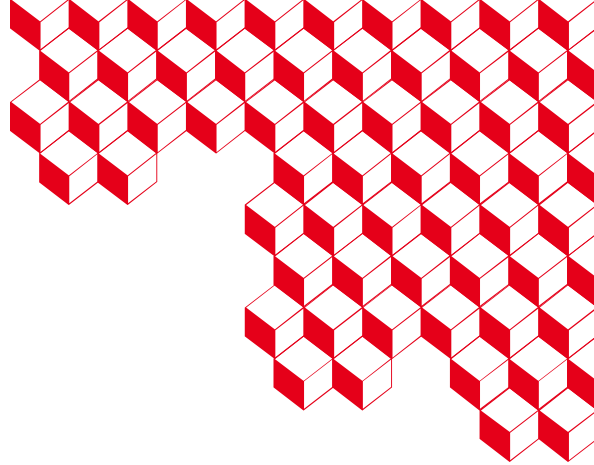
	Time (minutes)	Database size (GB)	Characteristics
Robinhood V3	24	1.58	Mono-process, 16 threads for entry process, 2 for FS scan
RobinHood V3	6	1.58	Parallelized on largest directory, fully optimized
Robinhood V4	60	0.395	Mono-process Mono-thread
RobinHood V4	27	0.395	parallelized on largest directory, mono-thread

What's next ?

- Finish the deduplication in **rbh-fsevents**
- Acknowledgement of events in **rbh-fsevents**
- Pre-production tests of the tools on our systems during 2023/2024
- The **rbh-report** tool to retrieve general information about a filesystem
- Improved synchronization by using **MPIFileUtils**
- More filters for the different **rbh-find** tools

Want to help with the development ?

- Projects have been moved on Github:
  - From one project per tool to one project for the suite
  - The address is: https://github.com/robinhood-suite/robinhood4
  - Reviews done on Gerrithub
- Some patches in review:
  - https://review.gerrithub.io/c/robinhood-suite/robinhood4/+/557610
    - First patch of the deduplication stack, documentation of the process
  - https://review.gerrithub.io/c/robinhood-suite/robinhood4/+/1167917
    - Fix a bug with **rbh-sync** branching function
- Feel free to install the suite and test it out for yourself, any feedback in appreciated!



Thanks for your attention

RobinHood v4 progress report

Yoann Valeri
Software stack developer at CEA
yoann.valeri@cea.fr

Why a version 4 for RobinHood?

	Version 3	Version 4
Scale up	SQL paradigm *MariaDB*	NoSQL paradigm *MongoDB*
Code genericity	Software specialised for Lustre filesystems	Generic tools calling specific backends
Inclusion to Linux repositories	Expert system	Library of features and applications
Code refactoring	Heavy code caused by Lustre behaviour evolution	Clean design to better correspond to current filesystems