

MARS administration

Sébastien Villaume

Manuel Fuentes, Baudouin Raoult, Tiago Quintino

Second Workshop for MARS administrators

7-8 March 2016

MARS command line administration

Outline

- common usage commands
- more advanced tricks
- Open questions

marsadm

Online documentation <https://software.ecmwf.int/wiki/display/MARSAD/marsadm+command>

Usage : run a command from a UNIX shell (batch mode)

```
marsadm [-log2cerr | -log2devnull] [-command <command>] [-name <name>]
```

Or within `marsadm` after launching it:

Usage : run a command within `marsadm` interactive shell

```
[%|@] command [-option1 ] [-option2 <arg> ...] [<param1> ...] [> file | |command]
```

@command runs the command in a loop

%command runs a UNIX shell command

Note: running simply “%” launch an interactive UNIX shell within `marsadm`

.marsadmrc

- This file is read when you start the `marsadm` shell if it exists in order from:
 - `~/etc/.marsadmrc`
 - `~/marsadmrc`
 -
- You typically put in it your favorite aliases:
 - `alias psmars '@ps mars'`
 - `alias crontab '%crontab -l'`
 - `etc.`

Basic commands

Man/help command

```
man|help [command]
```

Version of the binaries

```
version [-long]
```

History of typed commands

```
history|h
```

Uptime of the system

```
uptime
```

processes

List MARS processes

```
ps [process]
```

Start/stop processes

```
start/stop (process|all)
```

Lock/unlock the system (stop cron jobs)

```
lock/unlock
```

Kill a process

```
Kill (pid|task-id|process|all)
```

Exit command line administration

```
quit|exit
```

Printing informations

Print statistics of file-systems known to MARS

```
df
```

Statistics

```
stat (-id|-path|-master|-prearc|-cache) [-expr] [-sort [-reverse]] [-accumulate -maximum [-prefilter]]
```

List layouts/hypercubes

```
list (-id|-path|-master|-prearc|-cache) [-expr] [-sort [-reverse]] [-accumulate -maximum [-prefilter]]
```

List files

```
files (-id|-path|-master|-prearc|-cache) [-expr] [-sort [-reverse]] [-accumulate -maximum [-prefilter]]  
[-multi]
```

Validate and print good/bad layouts

```
validate (-master|-prearc|-cache)
```


Printing informations (cont.)

Show a request

```
Show (transaction_id)
```

tail

```
tail -f (task_id)
```

execute a command remotely, can be used to print remote info

```
remote (process) (status|config)
```

Find and print mismatch

```
mismatch (-master|-prearc|-cache)
```

Find and print orphans

```
orphans [-prearc|-cache]
```

sensitive commands (modify/update the metadata)

Force flush to tape

```
flush (-id|-path)[-expr] [-sort [-reverse]] [-accumulate -maximum [-prefilter]] [-forget] [-max]
```

synchronize

```
sync (-prearc|-cache) (-id|-path) [-expr] [-sort [-reverse]] [-accumulate -maximum [-prefilter]]
```

Cleanup orphans

```
orphans -cleanup
```

Purge (/!\ physically deleted data/transactions)

```
purge [-prearc|-cache|-tape] [-age]
```

Config commands

Configure/update a resource

```
config (resource) (value)
```

Remote reconfig a process resource/attribute

```
Remote (process) config (resource) (value)
```

Distributed MARS commands

Jump on the node “nodeName” marsadm

```
node (nodeName)
```

Execute a remote command on a node “nodeName”

```
node (nodeName|all|others) (command ...)
```

Synchronize metadata across nodes

```
sync -cluster [-prearc|-cache] [-expr]
```

Live session!



MARS permissions, priorities and limits

MARS permissions

- MARS does not store ownership or permissions in its metadata
- However, any requests sent to MARS contains metadata useful to set some permissions
- Permissions can be listed in `$DHSHOME/etc/marsPermissions`

Syntax of a permission

Permission

```
"an explanation to be printed if the permission is triggered"  
(a boolean expression ) = (another boolean expression)
```

example:

```
permission  
    "Too many fields (>2000000)"  
    (($_verb$ == "retrieve") && (fields() > 2000000)) = 0  
  
permission  
    "only user eras can archive ERA interim"  
    (($_verb$ == archive) && ($class$ == ei) = (eras in environ(user))
```

MARS priorities

- Once a request is permitted, one can apply priorities to it
- Priorities can be listed in `$DHSHOME/etc/marsPriorities`

Syntax of a priority

```
Priority  
  "comment for the priority"  
  (a boolean expression ) = signed integer in seconds
```

- example

```
priority "$max for online data"    (media() == 0) = 10800  
priority "eras"                    (eras in environ(user)) = 36000  
priority "host33" (host33 in environ(host)) = -36000
```


MARS limits

- Apply limits
 - To a number of requests based on criteria
 - per host, per type of requests, per number of tape accessed, per number of fields, etc.
- In `$DHSHOME/etc/marsLimits`

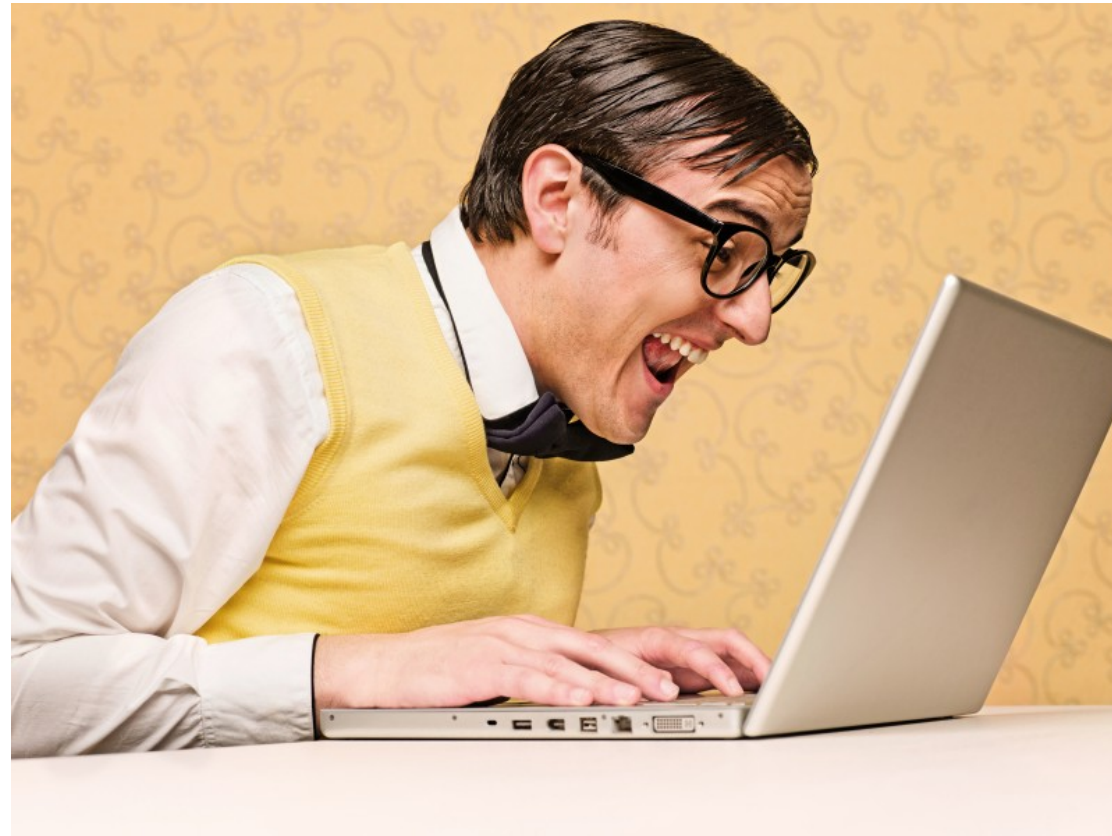
Syntax of a limits

```
limit  
  "comment for the limit"  
  (a boolean expression ) = number of requests allowed (can be 0)
```

- example:

```
limit "max 5 list requests" ($_verb$ == LIST) = 5  
limit "max 2 requests accessing more than 4 tapes" (media() > 4 && $_verb$ == RETRIEVE) = 2
```

Live session!



Special case for users

- Possibility to use `$DHSHOME/etc/userLimits` or even more fine grained `$DHSHOME/etc/userLimits.[user]`
- Same syntax than `marsLimits`
- Example in `userLimits`:

```
limit "max 5 requests" = 5
```

- Example in `userLimits.sebastien`:

```
limit "max 20 requests for sebastien" = 20
```

Questions?