



Proposal for a medium term plan for DM Clients

Alejandro Alvarez Ayllon *on behalf of the DM
Clients developers*
IT/SDC

09/10/2013



Overview

- About GFAL2
- Use cases and alternatives
- Status summary
- Proposal
 - Actions to be taken
 - Release medium

About GFAL2

- GFAL2 is a tool-kit for grid and cloud data operations
 - Will replace gfal
- Provides a single API for different protocols and storages
- Addresses shortcomings in GFAL
 - Error handling
 - Difficult maintainability and extensibility
 - We need to reduce workload
- Reduces number of dependencies
 - You install what you need
- Extensible
 - Plugin based. Isolation!

About GFAL2

- POSIX-like API
- Getting, putting and third party copy all in one call
- Session reuse (SRM, GridFTP, HTTPS)
- Python wrappers
- Interaction with the infosys is **entirely optional**

About GFAL2

- Currently used by FTS3
 - Reliable!
- Mind that there is no complete API compatibility!

About GFAL2-Utills

- Set of CLI based on GFAL2
 - Written in Python
- Will substitute lcg-util
- Better error logging
 - No more cryptic “invalid parameter” messages
- Not released yet
 - First release date to be decided
 - Probably within a month
 - Beta testers welcome!

About GFAL2-Utills

- As it is based on GFAL2, there is no need to interact with the Infosys
 - Still can be done if SURL resolution is needed
 - But even without a BDII, GFAL2 will try to “guess” the endpoint if a full SURL is not given

Extremely summarized experiments use cases

- CMS does not use LFC
- ATLAS and LHCb use LFC but **not** through gfal/lcg-util
 - Deprecation of LFC interaction in gfal2-utils seems safe
 - To Be Verified!
- Do we still provide everything needed?

A note on the deprecation of LFC

- Special consideration of LFC is what is going to be deprecated
 - Copy & Register a file *à la* lcg-cr
 - And all catalog-specific CLI (lcg-aa, lcg-lg...)
- It will be treated as any other protocol
 - gfal-ls lfn://host/path will work

Alternatives - CLI

Lcg-util	GFAL2-Util
lcg-cp	gfal-copy
lcg-ls	gfal-ls
lcg-del	gfal-rm
lcg-lr	No CLI available, API is there
lcg-get-checksum	gfal-sum
lcg-geturls, lcg-gt	Not available, can be done
lcg-stmd	Not available (?)
lcg-aa, lcg-cr, lcg-la, lcg-lg... and other catalog related cli	Not available
-	gfal-save, gfal-cat

Alternatives – Python API

GFAL	GFAL2	C API*
gfal_ls	Listdir	-
gfal_turlsfurls	-	gfal2_getxattr
gfal_deletesurls	Unlink	-
gfal_removedir	Rmdir	-
gfal_pin	-	gfal2_bring_online
gfal_prestage	-	gfal2_bring_online
gfal_prestagestatus	-	gfal2_bring_online_poll
gfal_release	-	gfal2_release
lcg_cp4	filecopy	-

(*) Not in the Python API yet, but they will be exposed

In summary

- Deprecation of LFC in gfal2-util is safe
- Pretty much all the needed functionality is there
 - One missing bit (lcg-stmd) that needs to be addressed
 - What else could be needed related to space management?
- So we propose...

Proposal

- Freeze immediately the addition of new features in gfal / lcg-util
 - No tickets open requesting any, so it should be ok
- Obsolete them in favor of gfal2 / gfal2-util
 - Any functionality request will go there
- Only **critical** bug-fixes until the **31st October 2014**
 - Meaning data corruption and/or loss, security compromises, etc...
- Help and support for migration will be kept indefinitely

Proposal

- Release medium for both
 - EPEL5 and EPEL6 repositories
 - After the proposed obsolescence date, gfal /lcgutil will **be removed** from EPEL
 - Regular flow
 - Development -> *RC (Optional)* -> EPEL Testing -> EPEL Stable
 - All packages will be removed from EMI2 and EMI3 repositories
 - Clients will be available in the Application Area

Notes

- This is a **proposal**
 - Can be discussed and refined
- However, gfal / lcg-util will eventually be deprecated
 - Need to reduce workload after EMI finished
- If you have any suggestion, question and/or complain, please, do contact us
 - lcgutil-devel@cern.ch
- All this is documented
 - <https://svnweb.cern.ch/trac/lcgutil/wiki/MediumTermProposal>

Questions?