

Gfal/lcg-util -> Gfal2/gfal2-util

Cristina Aiftimiei
EGI.eu

- DMC & their support calendar
- gfal2/gfal-util
- lcg-util -> gfal-util
- Retirement plan for gfal/lcg-util



DataManagement clients

Project	Description
LCG Util	lcg_utils was used to create Grid equivalents of UNIX copy commands and file registration that can span two storage systems and catalogues of choice
gfaFS	gfaFS is a FUSE module for GFAL 2.0. gfaFS is able to mount any point of the CERN's stockage system like a local folder.
GFAL 2.0 PYTHON	GFAL 2.0 python Bindings
GFAL 2.0	GFAL (Grid File Access Library) is a C library providing an abstraction layer of the grid storage system complexity.
GFAL 2 utility tools	GFAL 2 utils are a group of command line tools for file manipulations with any protocol managed by gfa 2.0
GFAL	The Grid File Access Library (GFAL) is a library that offers to the user a POSIX-like interface to access data on various flavours of Storage Elements offering an SRM interface.
EMI_datalib	The EMI_datalib project provides a consolidated grid data management stack derived from the arclib and lcg_util projects.
Davix	Davix is a shared library and a set of tools for advanced file I/O with HTTP based protocols.



DMC support calendar

- Support for DPM/LFC and gfal2 is **indefinite**.
 - LFC:
 - Atlas and LHCb both intend to stop using LFC by the end of 2014
 - LFC will still be supported for other VOs - **best-effort**.
- GFAL/lcg_util - will **not be supported** after October 2014

End-of-life mode of gfal and lcgutil

Alejandro Alvar... on Fri, 05/09/2014 - 11:01

gfal and lcg-util are in end-of-life mode, which means no new features, nor new bug fixes will be done, excepting very critical ones if needed.

Support for both will definitely end the **1st of November 2014**, so we recommend to all our users to [switch to the newer gfal2 and gfal2-utils](#).

- GFAL 2
 - a tool-kit for all grid and cloud data operations
 - provides one single uniform API for different protocols and storage elements
 - gfal-util written once – to add support for an additional protocol (http, xrootd) just install a plug-in
 - plug-in based, designed to reduce no. of dependencies, extensible



GFAL2 & gfal-util (2)

- More advantages:
 - POSIX-like API
 - Getting, putting and third party copy all in one call
 - Session reuse for SRM, GridFTP and HTTPS
 - Python wrappers available
 - Interaction with the infosys is entirely optional, and not needed
- Existing plugins:
 - SRM, GridFTP
 - RFIO-Castor and RFIO-DPM
 - DCAP / GSIDCAP
 - XROOTD
 - HTTP/S and WebDAV
 - S3, Local file



lcg-util vs. gfal-util (1)

Description	lcg-util	gfal-util
Add an alias for a given GUID	lcg-aa	Deprecated
Bring SURLs online	lcg-bringonline	Not available, could be done
Copy files with no catalog involved	lcg-cp	gfal-copy
Copy and register a file	lcg-cr	gfal-copy
Delete a file / directory	lcg-del	gfal-rm
Get the file checksum	lcg-get-checksum	gfal-sum
Get the TURLs for given SURLs	lcg-getturls	gfal-xattr "user.replicas"
Get the TURL for a given SURL	lcg-gt	gfal-xattr "user.replicas"
List aliases for a given LFN/GUID	lcg-la	Deprecated
Get the GUID for a given LFN	lcg-lg	gfal-xattr "user.guid"
Lists the replicas for a given LFN	lcg-lr	gfal-xattr "user.replicas"
List information / directory	lcg-ls	gfal-ls
Remove an alias	lcg-ra	Deprecated
Copy between Ses using the catalog	lcg-rep	gfal-copy
Register a file in a catalog	lcg-rf	gfal-copy
Set a file status to done	lcg-sd	Deprecated*
Get space tokens associated to a description	lcg-stmd	gfal-xattr "spacetoken"/"spacetoken.token?\$tok"/"spacetoken.description?\$desc"
Unregister a file	lcg-uf	Deprecated
Create a directory		gfal-mkdir
Dump the standard input into a file		gfal-save
Dump a file into the standard output		gfal-cat

lcg-util vs. gfal-util (2)

- Ex: lcg-cr replaced by gfal-copy:
 - # gfal-copy file://... srm://...
 - # gfal-copy srm://... lfn://...
- No replacement for:
 - lcg-aa
 - lcg-la
 - lcg-ra
 - lcg-uf

Rationale behind deprecation

- CMS
 - Not using LFC
 - Use case based mainly on lcg-cp, lcg-ls and lcg-del
- ATLAS
 - Uses LFC, but not through lcg-utils
 - Use cases based on lcg-cp, lcg-lr, lcg-ls, lcg-stmd
- LHCb
 - Uses LFC, but not through lcg-utils
 - lcg-utils CLI used manually for debugging purposes
 - DIRAC uses Python wrappers

- gfal / lcg-util
 - **No** new features into gfal / lcg-util
 - Latest version, 1.16.0 – available in **EPEL**
 - support for critical bug-fixes until the **31st October 2014**
 - packages will be **retired** from EPEL
- **Help and support for migration into gfal2 / gfal-util will be kept indefinitely**

- Start deployment of gfal-utils
 - for other VOs to tests its functionality
 - Only on SL6 for the moment
 - SL5 version not production-ready
- Clear document gfal-util commands as replacement of lcg-util ones
 - Understand if there are missing functionalities