

LHCb gfal Python code

Notes from Christophe Haen

List of gfal2 calls used by LHCb

```
bring_online
bring_online_poll
checksum
filecopy
getxattr
listdir
listxattr
mkdir_rec
release
rmdir
set_opt_boolean
stat
transfer_parameters
unlink
```

Protocol

Changing the protocol from SRM to XRootD has no side-effects, providing that the protocol behaves the same. The URL is constructed in real time, using a configuration file. If we have to tell DIRAC that from now on, it should rely on the xroot protocol, it is one line in the configuration file for us. But of course, that is providing the protocol exposes the same functionalities, with the same interface.

See below the Python/gfal2 calls used by LHCb to stat a file, stage it, and monitor the staging status.

Stat a file

```
gfal2_stat.py
```

```
import gfal2
```

```
fileURL = 'srm://srm-lhcb.cern.ch:8443/srm/managerv2?SFN=/castor
/cern.ch/grid/lhcb/data/2018/RAW/FULL/LHCb/COLLISION18/209383
/209383_0000001095.raw'
```

```
ctx = gfal2.creat_context()
```

```

ctx.set_opt_boolean('BDII', 'ENABLE', False)
ctx.set_opt_boolean('GRIDFTP_PLUGIN', 'SESSION_REUSE', False)
ctx.set_opt_boolean('GRIDFTP_PLUGIN', 'IPV6', True)
ctx.set_opt_integer('SRM_PLUGIN', 'OPERATION_TIMEOUT', 100)
ctx.set_opt_string('SRM_PLUGIN', 'SPACETOKENDESC', 'LHCb-Tape')
ctx.set_opt_integer('SRM_PLUGIN', 'REQUEST_LIFETIME', 3600)
ctx.set_opt_string_list('SRM_PLUGIN', 'TURL_PROTOCOLS', ['gsiftp'])
print "stat"
print ctx.stat(fileURL)
print "checksum"
print ctx.checksum( fileURL, 'ADLER32')
print "status"
print ctx.getxattr( fileURL, 'user.status')

```

Result

```

stat
uid: 45
gid: 46
mode: 100644
size: 5242935904
nlink: 1
ino: 0
ctime: 1528143170
atime: 0
mtime: 1528143170

checksum
0e59c8c5
status
NEARLINE

```

Stage a file

```

gfal2_prestage.py

```

```

import gfal2

fileURL = 'srm://srm-lhcb.cern.ch:8443/srm/managerv2?SFN=/castor
/cern.ch/grid/lhcb/data/2018/RAW/FULL/LHCb/COLLISION18/209383
/209383_0000001095.raw'
lifetime = 86400
timeout = 43200
async = True

ctx = gfal2.creat_context()

```

```

ctx.set_opt_boolean('BDII', 'ENABLE', False)
ctx.set_opt_boolean('GRIDFTP_PLUGIN', 'SESSION_REUSE', False)
ctx.set_opt_boolean('GRIDFTP_PLUGIN', 'IPV6', True)
ctx.set_opt_integer('SRM_PLUGIN', 'OPERATION_TIMEOUT', 100)
ctx.set_opt_string('SRM_PLUGIN', 'SPACETOKENDESC', 'LHCb-Tape')
ctx.set_opt_integer('SRM_PLUGIN', 'REQUEST_LIFETIME', 3600)
ctx.set_opt_string_list('SRM_PLUGIN', 'TURL_PROTOCOLS', ['gsiftp'])
print ctx.bring_online(fileURL, lifetime, timeout, async)

```

Result

```
(0, '53277390')
```

Monitor the Staging Status

gfal2_stagingStatus.py

```

import gfal2

fileURL = 'srm://srm-lhcb.cern.ch:8443/srm/managerv2?SFN=/castor
/cern.ch/grid/lhcb/data/2018/RAW/FULL/LHCb/COLLISION18/209383
/209383_0000001095.raw'

# output from the bring_online command
token = '53277390'

ctx = gfal2.creat_context()

ctx.set_opt_boolean('BDII', 'ENABLE', False)
ctx.set_opt_boolean('GRIDFTP_PLUGIN', 'SESSION_REUSE', False)
ctx.set_opt_boolean('GRIDFTP_PLUGIN', 'IPV6', True)
ctx.set_opt_integer('SRM_PLUGIN', 'OPERATION_TIMEOUT', 100)
ctx.set_opt_string('SRM_PLUGIN', 'SPACETOKENDESC', 'LHCb-Tape')
ctx.set_opt_integer('SRM_PLUGIN', 'REQUEST_LIFETIME', 3600)
ctx.set_opt_string_list('SRM_PLUGIN', 'TURL_PROTOCOLS', ['gsiftp'])
print ctx.bring_online_poll(fileURL, token)

```

Result

0 if not staged, 1 if staged