Table of Contents

How to make best use of storage at CERN and other Grid sites

This page gives an overview of data storage at CERN (Castor and EOS) , advice on how to use them and especially hints about the proper usage of the "Cern Castor Default Pool" that should now only be used as an archive. It also explains how to use remote storage at Tier1s or Tier2s.

For a detailed description of the various area on EOS and best practice recommendation, see https://indico.cern.ch/event/789894/contributions/3283132/attachments/1791343/2918542/EOS_Presentation.pdf⧉

Introduction

Which storage elements exist at CERN

The storage area for LHCb data at CERN is using two technologies (Castor for data archive and EOS for disk-only storage). The various classes of storage are described in the table below:

#	Common name	Technology	Grid Storage	Tape backend	Permanent disk	Replicated on other sites	SRM space	Dirac storage element
1	CASTOR_HOME a.k.a "Castor default pool"	Castor	N	Y	N	N	N/A	N/A
2	EOS_HOME	EOS	N	N	Y	N	N/A	N/A
3	Grid user storage	EOS	Y	N	Y	Y	LHCb_USER	CERN-USER
4	LHCb data on permanent disk	EOS	Y	N	Y	Y	LHCb-EOS	CERN-DST-EOS, CERN-EOS-FAILOVER, CERN_MC-DST-EOS, CERN-HIST, CERN-FREEZER-EOS
5	Disk pool for LHCb data in front of tape	Castor	Y	Y	N	Y	LHCb-Tape	CERN-RAW CERN-RDST CERN-ARCHIVE
2	EOS WG space	EOS	N	N	Y	N	N/A	N/A

All the disk pools for "grid storage" also exist in a similar shape at T1 sites, e.g. CNAF-DST, GRIDKA-USER, PIC-RAW, etc. Note that the storages 4 and 5 are reserved for production output data, and only storages 1, 2 and 5 have write-permission for users.

CASTOR_HOME: the Castor default pool

As can be seen from the table above the "Castor default pool" (Pool #1) is the smallest of all disk pools. This pool should **not** be used for any analysis work but might be useful for archive storage of "private data". In all other cases one of the other pools of above shall be used. The namespace for this pool corresponds to the $CASTOR_HOME environment variable. Please note that confidential data (e.g. PC backups - which would contain at least password files) should **not** be uploaded in CASTOR, since access controls are rather weak, and the data is not encrypted.

As of February 2014, the `rootd` access protocol (a.k.a. `castor:` which also is the default when no protocol is specified) has been discontinued. It is therefore mandatory to use the `xrootd` protocol. The form of the PFN should mandatorily be:

```
root:://castorlhcb.cern.ch//castor/cern.ch/.....
```

The `xrootd` protocol can be used for copying a file to the local directory using the command `xrdcp` that therefore supersedes `rfcp`.

Most important: the default pool is not keeping files permanently on disk, therefore there may be a huge penalty for accessing files that need to be recalled from tape, making its usage in jobs or interactive analysis

most inefficient.

Castor service classes

In principle, most users don't need any longer to access files on Castor as this is mostly an archive facility. The only datasets that are relevant are the RAW files and the FULL.DST files, whose usage is highly discouraged for regular users. Development of e.g. stripping lines should used the saved disk replicas of FULL.DST and RAW files specified in the stripping twiki.

For completeness we describe here however the usage of service classes in Castor (the only one left being lhcbtape)

A "castor service class" is a hint to the castor system which disk pool to use for accessing the files - in principle a given file can be stored on several pools, and in particular there is no mapping between the namespace and the disk pool. If no service class is given for the access to the file the "default pool" will be used, even if the file is already available on a different pool. Therefore it is always essential to provide the proper service class for file access. For ROOT access, the castor service class is described by the option "svcClass=" in the access string, e.g. the file below is in service class "lhcbdisk". Note that one should never use the rfio: protocol any longer, neither to not specify a protocol (which in ROOT defaults to rfio:), but the following URL (a.k.a. PFN) should be used (as generated by the Dirac tools):

```
root://castorlhcb.cern.ch//castor/cern.ch/grid/lhcb/LHCb/Collision11/RAW/FULL/123456/123456_00000
```

The following two access modes are highly discouraged and LHCbDirac tools should be used instead for transferring files:

- RFIO access (`rfcp`), please set the environment variable `STAGE_SVCCLASS`.
- access via SRM (`srmcp`, `lcg-cp`), please specify the corresponding space token (options --sst and --dst for source and destination space tokens). SRM is special in that it will look into a list of pools if no space token has been specified, but please do not rely on this.

EOS

EOS is a disk-only storage at CERN that should be used for all analysis tasks. It offers a high reliability essentially in providing double replication of the files (transparent to users). Its main access protocol is xrootd (root: protocol). Although EOS provides its own tools, we strongly encourage LHCb users to user LHCbDirac tools for getting access to data (except for private data on EOS_HOME - see the introduction to EOS_HOME in this talk☑ at the June 2014 LHCb week☑).

Use cases

The information below is only useful when not using ganga, i.e. are only relevant for interactive or LXBATCH direct job submission. ganga and Dirac deal internally with setting the proper URL for file access.

Access to analysis files on EOS

Files that were produced by "official productions" and are located on EOS. If you want to specify a PFN in your job options, please use the LHCbDirac command for generating a proper PFN.

Access to user files

Access to "grid user files"

Grid user files Pool #3 live in the namespace "/eos/lhcb/grid/lhcb/user", the access to those files can be done via the same tools as above for analysis files. See "Get the proper access..." for more details.

Access to "local user files"

Local user files live in the namespace "/castor/cern.ch/user", Pool #1 or "/eos/lhcb/user" on EOS. Those files cannot be accessed via grid tools, as the are not known to the grid environment (e.g. no entry in the FC). For these kind of files there are two possibilities

- Gridify the file <- **RECOMMENDED**, for details on how to do this see "Gridify my user files" below. This will replicate the file on CERN-USER and register it in the FC, such that it can be used as any file created on the Grid.
- If a gridification is not possible the files need to be accessed via the default pool on castor or the native URL for EOS. You should always use the root: protocol as described above. The PFN would have to be constructed "by hand", without service class specification, e.g.
  `root:/castor/cern.ch/user/a/another/mycastorfile.root` or
  `root:/eos/lhcb/user/a/another/mycastorfile.root`

Possible solutions

Write my file output to grid storage instead of Castor default

Using Ganga

The Ganga backend when submitting jobs locally to the Cern batch system (bsub) shall be set to Dirac(). So when setting up your Ganga job you may use

```
j=Job()
j.backend= Dirac()
```

which will automatically write your output files into the grid storage area. The default is to upload two replicas in one of the 7 available storage areas (CERN-USER, CNAF-USER, GRIDKA-USER, IN2P3-USER, PIC-USER, RAL-USER and SARA-USER).

"Gridify" my user files

Files on the local disk

Files that are not located in the grid storage area need to be made known to the grid tools (e.g. File Catalog (LFC)). This can be achieved with the LHCbDirac tool "dirac-dms-add-file", e.g.

```
[pcites03] /afs/cern.ch/user/r/roiser > dirac-dms-add-file /lhcb/user/r/roiser/my-grid-analysis-f
{'Failed': {},
 'Successful': {'/lhcb/user/r/roiser/my-grid-analysis-file': {'put': 9.388253927230835,
                                                     'register': 0.24701905250549316}}}
[pcites03] /afs/cern.ch/user/r/roiser >
```

Files in the Castor default pool or in EOS

For the time being files in castor default pool or in EOS that need to be put into grid storage need to be copied to the local disk before and then the recipe above needs to be applied.

Get the proper access to files in grid storage

In case you are accessing files on "grid storage" (see table above) you may use the following two commands in order to get the proper access to the files you need, starting from a generic LFN (see the list of LHCbDirac tools).

```
[localhost, Patch] ~ $ dirac-dms-lfn-replicas /lhcb/LHCb/Collision11/CHARM.MDST/00012718/0000/000
Successful :
    /lhcb/LHCb/Collision11/CHARM.MDST/00012718/0000/00012718_00000012_1.charm.mdst :
        CERN-DST-EOS : srm://srm-eoslhcb.cern.ch/eos/lhcb/LHCb/Collision11/CHARM.MDST/00012718/00
         IN2P3_M-DST : srm://ccsrm.in2p3.fr/pnfs/in2p3.fr/data/lhcb/LHCb/Collision11/CHARM.MDST/0
             PIC-DST : srm://srmlhcb.pic.es/pnfs/pic.es/data/lhcb/LHCb/Collision11/CHARM.MDST/000
            SARA-DST : srm://srm.grid.sara.nl/pnfs/grid.sara.nl/data/lhcb/LHCb/Collision11/CHARM.
```

from the list above you need to take the name of the "Dirac storage element", i.e. here CERN-DST-EOS and feed it to

```
[localhost, Patch] ~ $ dirac-dms-lfn-accessURL /lhcb/LHCb/Collision11/CHARM.MDST/00012718/0000/00
Successful :
    CERN-DST-EOS :
        /lhcb/LHCb/Collision11/CHARM.MDST/00012718/0000/00012718_00000012_1.charm.mdst : root://e
```

The output of this command provides you a file name (PFN) that can be used in your application to access the file, e.g.

```
[pcites03] /afs/cern.ch/user/r/roiser > root -b
  *********************************************
  *                                           *
  *        W E L C O M E  to  R O O T         *
  *                                           *
  *   Version   5.32/00   2 December 2011     *
  *                                           *
  *  You are welcome to visit our Web site    *
  *           http://root.cern.ch             *
  *                                           *
  *********************************************

ROOT 5.32/00 (tags/v5-32-00@42375, Dec 02 2011, 12:42:25 on linuxx8664gcc)

CINT/ROOT C/C++ Interpreter version 5.18.00, July 2, 2010
Type ? for help. Commands must be C++ statements.
Enclose multiple statements between { }.
root [0] f = TFile().Open("root://eoslhcb.cern.ch//eos/lhcb/LHCb/Collision11/CHARM.MDST/00012718/
Warning in <TClass::TClass>: no dictionary for class DataObject is available
[...]
(class TFile*)0xda676e0
root [1] f->GetSize()
(const Long64_t)5449176776
root [2]
```

Note that with the xrootd protocol (root:) you can access files located at any storage element, even outside CERN using the same commands as above (except at CNAF where the default access if via the file: protocol that is not accessible outside. In that specific case you should instruct LHCbDirac to get the xrootd PFN:

```
dirac-dms-lfn-accessURL /lhcb/LHCb/Collision11/BHADRON.MDST/00024034/0002/00024034_00024451_1.bha
Successful :
    CNAF-DST :
        /lhcb/LHCb/Collision11/BHADRON.MDST/00024034/0002/00024034_00024451_1.bhadron.mdst : root
```

This topic: Main > CastorDefaultPool
Topic revision: r12 - 2019-02-25 - ChristopheHaen