

dCache

The dCache distributed storage system and a guide for DPM migration

Michele Delli Veneri, Bernardino Spisso, Alessandra Doria



Overview

1. Getting to know dCache
 1. dCache basic components
 2. Understanding dCache topology
 3. Authentication and Authorisation
 4. Link Groups and Space Reservation
2. Which OS and why migrating from DPM
3. Installation and DPM migration guides
 1. Overview of the Installation process
 2. Guide to install dCache
 3. Overview of the Migration process
 4. Guide to migrate from DPM to dCache

dCache

Architecture I - Basic Blocks

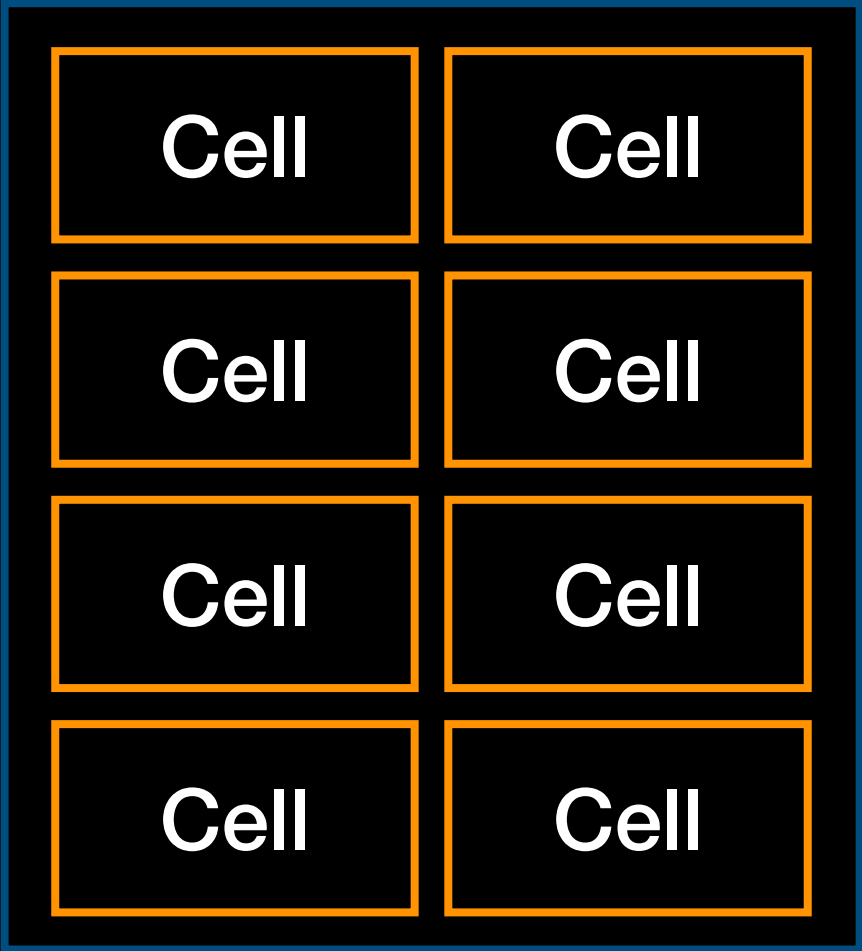


Cell

dCache basic architectural element

- Performs specific tasks;
- May rely on other cells;
- High level tasks are obtained through the interaction between multiple cells

Domain



Cell

Cell

Cell

Cell

Cell

Cell

Cell

Cell

dCache high level element

- Runs a Java Virtual Machine instance equipped to run **cells**
- Can communicate with other Domains through TCP;
- Does not share the JVM instance with other domains

dCache

Architecture II - Cells

Zookeeper

a distributed directory and coordination system. Allows domains to communicate and coordinate with each other

Pnfsmanager

manages the pNFS file system, the pNFS database and the metadata

Cleaner

periodically removes deleted file replica from the pools

Poolmanager

heart of the system. Handles read / write requests and manages the transfer of files between the user and the system

Spacemanager

Handles free space reservation. Relies upon Poolmanager and SRM

Pinmanager

Ensures the presence of file replica on disk. It is used to ensure that a certain number of replica are available

Billing

Built-in monitor system which provides an overview of activity and performance of doors and pools

GPlazma

authorisation and authentication. Credentials are collected by doors and sent to gplazma for authentication

Admin

Provides admin shell services that allow interaction with all cells within the dCache system

SRMmanager

The Storage Resource Manager, handles dynamic space allocation and file management

Transfermanager

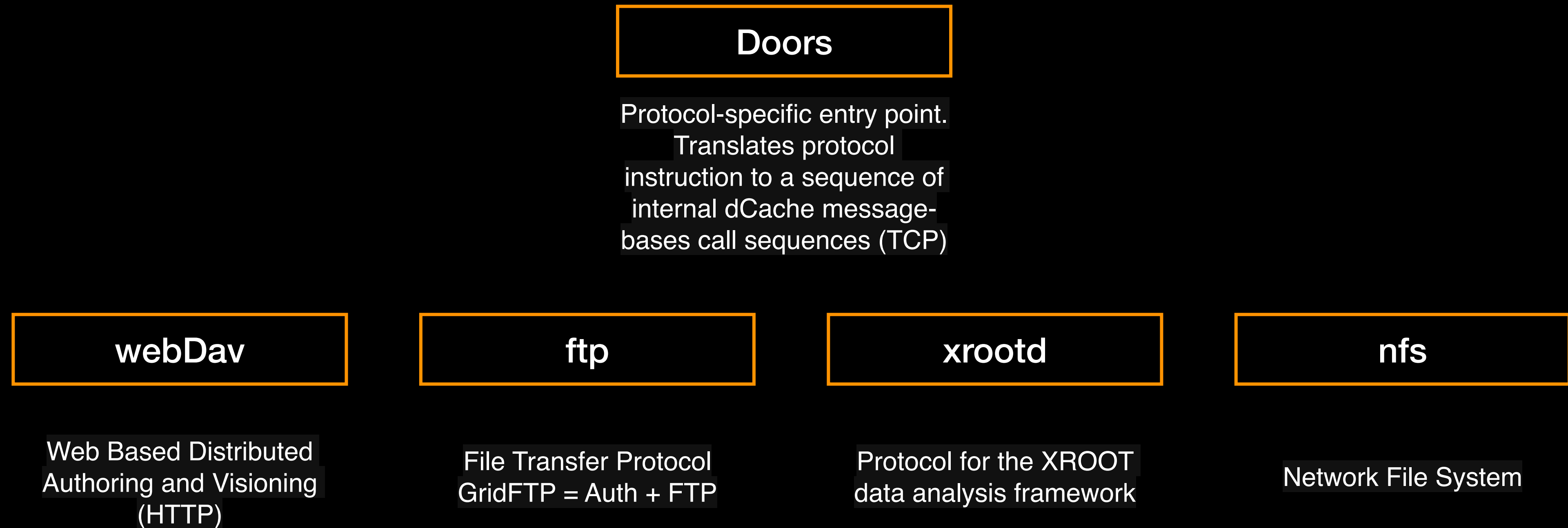
Handles 3rd party copy transfers initiated by SRM or WebDav

Pool

Handles data storage, performs checksums and data migrations

dCache

Architecture III - Communication and Transfers Cells



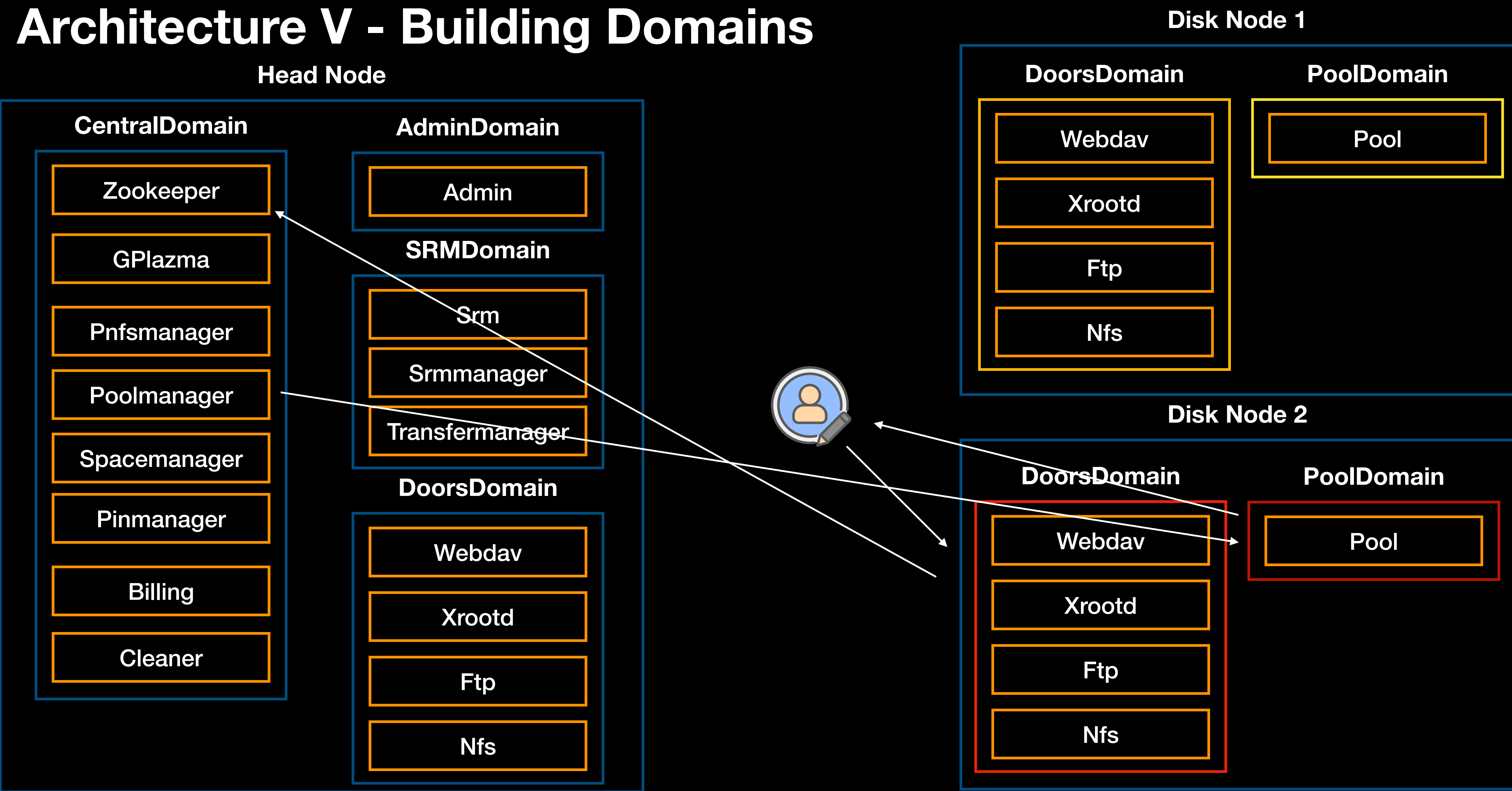
dCache

Architecture IV - Other Components

- Frontend -> Web page with monitoring services
- Namespace -> PostgreSQL server managing several databases:
 1. Chimera
 2. space manager
 3. pin manager
 4. srm

dCache

Architecture V - Building Domains



dCache

Architecture VII - Layout Files, i.e. building domains

```
# Headnode dCache layout configuration
#
=====

[centralDomain]
dcache.broker.scheme = core
[centralDomain/zookeeper]
[centralDomain/pnfsmanager]
[centralDomain/cleaner]
[centralDomain/poolmanager]
[centralDomain/spacemanager]
[centralDomain/pinmanager]
[centralDomain/billing]
[centralDomain/gplazma]
[adminDoorDomain]
[adminDoorDomain/admin]
[informationDomain]
[informationDomain/httpd]
[informationDomain/topo]
[informationDomain/info]
[informationDomain/statistics]
[informationDomain/frontend]
frontend.authn.basic=true
frontend.authn.protocol=http
frontend.authz.anonymous-
operations=READONLY
frontend.srr.public=true
[doorsDomain]
[doorsDomain/webdav]
webdav.authn.protocol = https
webdav.net.port = 443
[doorsDomain/xrootd]
xrootd.plugins=gplazma:gsi
[doorsDomain/ftp]
ftp.authn.protocol=gsi
[srmDomain]
[srmDomain/srm]
srm.net.port = 8446
[srmmanagerDomain]
[srmmanagerDomain/srmmanager]
[srmmanagerDomain/transfermanagers]
```

```
# Disknode dCache layout configuration
# =====

# disknodes needs to know how to contact central node
dcache.zookeeper.connection = t2-dpm-dome.na.infn.it:2181

[doorsDomain-${host.name}]
[doorsDomain-${host.name}/webdav]
webdav.authn.basic = true
webdav.authn.protocol = https
webdav.loginbroker.port = 443

[doorsDomain-${host.name}/xrootd]
xrootd.plugins=gplazma:gsi

[doorsDomain-${host.name}/ftp]
ftp.authn.protocol=gsi

# project staticPool on filesystem /data/t2-disk01-static
[poolsDomain_${host.name}_staticPool]
[poolsDomain_${host.name}_staticPool/pool]
pool.name=staticPool_008
pool.tags=hostname=${host.name} poolgroup=staticPool
pool.path=/data/t2-disk01-static/dcache/staticPool_008
```

```
# Disknode dCache layout configuration
# =====

# disknodes needs to know how to contact central node
dcache.zookeeper.connection = t2-dpm-
dome.na.infn.it:2181

[doorsDomain-${host.name}]
[doorsDomain-${host.name}/webdav]
webdav.authn.basic = true
webdav.authn.protocol = https
webdav.loginbroker.port = 443

[doorsDomain-${host.name}/xrootd]
xrootd.plugins=gplazma:gsi

[doorsDomain-${host.name}/ftp]
ftp.authn.protocol=gsi

# project StaticEscape on filesystem /mpatha
[poolsDomain_${host.name}_StaticEscape]
[poolsDomain_${host.name}_StaticEscape/pool]
pool.name=StaticEscape_001
pool.tags=hostname=${host.name}
poolgroup=StaticEscape
pool.path=/mpatha/dcache/StaticEscape_001

# project StaticEscape on filesystem /mpathb
[poolsDomain_${host.name}_StaticEscape]
[poolsDomain_${host.name}_StaticEscape/pool]
pool.name=StaticEscape_002
pool.tags=hostname=${host.name}
poolgroup=StaticEscape
pool.path=/mpathb/dcache/StaticEscape_002

# project StaticEscape on filesystem /mpathd
[poolsDomain_${host.name}_StaticEscape]
[poolsDomain_${host.name}_StaticEscape/pool]
pool.name=StaticEscape_004
pool.tags=hostname=${host.name}
poolgroup=StaticEscape
pool.path=/mpathd/dcache/StaticEscape_004
```

```
# project StaticEscape on filesystem /mpathe
[poolsDomain_${host.name}_StaticEscape]
[poolsDomain_${host.name}_StaticEscape/pool]
pool.name=StaticEscape_005
pool.tags=hostname=${host.name} poolgroup=StaticEscape

pool.path=/mpathe/dcache/StaticEscape_005
# project StaticEscape on filesystem /mpathc
[poolsDomain_${host.name}_StaticEscape]
[poolsDomain_${host.name}_StaticEscape/pool]
pool.name=StaticEscape_003
pool.tags=hostname=${host.name} poolgroup=StaticEscape

pool.path=/mpathc/dcache/StaticEscape_003
# project StaticEscape on filesystem /mpathf
[poolsDomain_${host.name}_StaticEscape]
[poolsDomain_${host.name}_StaticEscape/pool]
pool.name=StaticEscape_006
pool.tags=hostname=${host.name} poolgroup=StaticEscape
pool.path=/mpathf/dcache/StaticEscape_006

# project staticPool on filesystem /mpathg
[poolsDomain_${host.name}_staticPool]
[poolsDomain_${host.name}_staticPool/pool]
pool.name=staticPool_009
pool.tags=hostname=${host.name} poolgroup=staticPool
pool.path=/mpathg/dcache/staticPool_009

# project staticPool on filesystem /mpathh
[poolsDomain_${host.name}_staticPool]
[poolsDomain_${host.name}_staticPool/pool]
pool.name=staticPool_010
pool.tags=hostname=${host.name} poolgroup=staticPool
pool.path=/mpathh/dcache/staticPool_010
```


dCache

Architecture VII - Layout Files, i.e. building domains

<pre># Headnode dCache layout configuration # ===== [centralDomain] dcache.broker.scheme = core [centralDomain/zookeeper] [centralDomain/pnfsmanager] [centralDomain/cleaner] [centralDomain/poolmanager] [centralDomain/spacemanager] [centralDomain/pinmanager] [centralDomain/billing] [centralDomain/gplazma] [adminDoorDomain] [adminDoorDomain/admin] [informationDomain] [informationDomain/httpd] [informationDomain/topo] [informationDomain/info] [informationDomain/statistics] [informationDomain/frontend] frontend.authn.basic=true frontend.authn.protocol=http frontend.authz.anonymous- operations=READONLY frontend.srr.public=true [doorsDomain] [doorsDomain/webdav] webdav.authn.protocol = https webdav.net.port = 443 [doorsDomain/xrootd] xrootd.plugins=gplazma:gsi [doorsDomain/ftp] ftp.authn.protocol=gsi [srmDomain] [srmDomain/srm] srm.net.port = 8446 [srmmanagerDomain] [srmmanagerDomain/srmmanager] [srmmanagerDomain/transfermanagers]</pre>	<pre># Disknode dCache layout configuration # ===== # disknodes needs to know how to contact central node dcache.zookeeper.connection = t2-dpm-dome.na.infn.it:2181 [centralDomain/zookeeper] [doorsDomain-\${host.name}] [centralDomain/pnfsmanager] [doorsDomain-\${host.name}/webdav] [centralDomain/cleaner] [centralDomain/poolmanager] [centralDomain/spacemanager] [doorsDomain-\${host.name}/xrootd] xrootd.plugins=gplazma:gsi [centralDomain/pinmanager] [doorsDomain-\${host.name}/ftp] ftp.authn.protocol=gsi [centralDomain/gplazma] # project staticPool on filesystem /data/t2-disk01-static [poolsDomain_\${host.name}_staticPool] [poolsDomain_\${host.name}_staticPool/pool] pool.name=staticPool_008 pool.tags=hostname=\${host.name} poolgroup=staticPool pool.path=/data/t2-disk01-static/dcache/staticPool_008</pre>	<pre># Disknode dCache layout configuration # ===== # disknodes needs to know how to contact central node dcache.zookeeper.connection = t2-dpm- dome.na.infn.it:2181 [doorsDomain-\${host.name}] [doorsDomain-\${host.name}/webdav] webdav.authn.basic = true webdav.authn.protocol = https webdav.loginbroker.port = 443 [doorsDomain-\${host.name}/xrootd] xrootd.plugins=gplazma:gsi [doorsDomain-\${host.name}/ftp] ftp.authn.protocol=gsi # project StaticEscape on filesystem /mpatha [poolsDomain_\${host.name}_StaticEscape] [poolsDomain_\${host.name}_StaticEscape/pool] pool.name=StaticEscape_001 pool.tags=hostname=\${host.name} poolgroup=StaticEscape pool.path=/mpatha/dcache/StaticEscape_001 # project StaticEscape on filesystem /mpathb [poolsDomain_\${host.name}_StaticEscape] [poolsDomain_\${host.name}_StaticEscape/pool] pool.name=StaticEscape_002 pool.tags=hostname=\${host.name} poolgroup=StaticEscape pool.path=/mpathb/dcache/StaticEscape_002 # project StaticEscape on filesystem /mpathd [poolsDomain_\${host.name}_StaticEscape] [poolsDomain_\${host.name}_StaticEscape/pool] pool.name=StaticEscape_004 pool.tags=hostname=\${host.name} poolgroup=StaticEscape pool.path=/mpathd/dcache/StaticEscape_004</pre>	<pre># project StaticEscape on filesystem /mpathe [poolsDomain_\${host.name}_StaticEscape] [poolsDomain_\${host.name}_StaticEscape/pool] pool.name=StaticEscape_005 pool.tags=hostname=\${host.name} poolgroup=StaticEscape pool.path=/mpathe/dcache/StaticEscape_005 # project StaticEscape on filesystem /mpathc [poolsDomain_\${host.name}_StaticEscape] [poolsDomain_\${host.name}_StaticEscape/pool] pool.name=StaticEscape_003 pool.tags=hostname=\${host.name} poolgroup=StaticEscape pool.path=/mpathc/dcache/StaticEscape_003 # project StaticEscape on filesystem /mpathf [poolsDomain_\${host.name}_StaticEscape] [poolsDomain_\${host.name}_StaticEscape/pool] pool.name=StaticEscape_006 pool.tags=hostname=\${host.name} poolgroup=StaticEscape pool.path=/mpathf/dcache/StaticEscape_006 # project staticPool on filesystem /mpathg [poolsDomain_\${host.name}_staticPool] [poolsDomain_\${host.name}_staticPool/pool] pool.name=staticPool_009 pool.tags=hostname=\${host.name} poolgroup=staticPool pool.path=/mpathg/dcache/staticPool_009 # project staticPool on filesystem /mpathh [poolsDomain_\${host.name}_staticPool] [poolsDomain_\${host.name}_staticPool/pool] pool.name=staticPool_010 pool.tags=hostname=\${host.name} poolgroup=staticPool pool.path=/mpathh/dcache/staticPool_010</pre>
---	--	--	---

dCache

Architecture VII - Layout Files, i.e. building domains

[centralDomain]

dcache.broker.scheme = core

[centralDomain/zookeeper]

[centralDomain/pnfsmanager]

[centralDomain/cleaner]

[centralDomain/poolmanager]

[centralDomain/spacemanager]

[centralDomain/pinmanager]

[centralDomain/billing]

[centralDomain/gplazma]

All Core cells are in the central Domain

dCache

Architecture VII - Layout Files, i.e. building domains

```
# Headnode dCache layout configuration
#
=====

[centralDomain]
dcache.broker.scheme = core
[centralDomain/zookeeper]
[centralDomain/pnfsmanager]
[centralDomain/cleaner]
[centralDomain/poolmanager]
[centralDomain/spacemanager]
[centralDomain/pinmanager]
[centralDomain/billing]
[centralDomain/gplazma]
[adminDoorDomain]
[adminDoorDomain/admin]
[informationDomain]
[informationDomain/httpd]
[informationDomain/topo]
[informationDomain/info]
[informationDomain/statistics]
[informationDomain/frontend]
frontend.authn.basic=true
frontend.authn.protocol=http
frontend.authz.anonymous-
operations=READONLY
frontend.srr.public=true
[doorsDomain]
[doorsDomain/webdav]
webdav.authn.protocol = https
webdav.net.port = 443
[doorsDomain/xrootd]
xrootd.plugins=gplazma:gsi
[doorsDomain/ftp]
ftp.authn.protocol=gsi
[srmDomain]
[srmDomain/srm]
srm.net.port = 8446
[srmmanagerDomain]
[srmmanagerDomain/srmmanager]
[srmmanagerDomain/transfermanagers]
```

```
# Disknode dCache layout configuration
# =====

# disknodes needs to know how to contact central node
dcache.zookeeper.connection = t2-dpm-dome.na.infn.it:2181

[doorsDomain-${host.name}]
[doorsDomain-${host.name}/webdav]
webdav.authn.basic = true
webdav.authn.protocol = https
webdav.loginbroker.port = 443

[doorsDomain-${host.name}/xrootd]
xrootd.plugins=gplazma:gsi

[doorsDomain-${host.name}/ftp]
ftp.authn.protocol=gsi

# project staticPool on filesystem /data/t2-disk01-static
[poolsDomain_${host.name}_staticPool]
[poolsDomain_${host.name}_staticPool/pool]
pool.name=staticPool_008
pool.tags=hostname=${host.name} poolgroup=staticPool
pool.path=/data/t2-disk01-static/dcache/staticPool_008
```

```
# Disknode dCache layout configuration
# =====

# disknodes needs to know how to contact central node
dcache.zookeeper.connection = t2-dpm-
dome.na.infn.it:2181

[doorsDomain-${host.name}]
[doorsDomain-${host.name}/webdav]
webdav.authn.basic = true
webdav.authn.protocol = https
webdav.loginbroker.port = 443

[doorsDomain-${host.name}/xrootd]
xrootd.plugins=gplazma:gsi

[doorsDomain-${host.name}/ftp]
ftp.authn.protocol=gsi

# project StaticEscape on filesystem /mpatha
[poolsDomain_${host.name}_StaticEscape]
[poolsDomain_${host.name}_StaticEscape/pool]
pool.name=StaticEscape_001
pool.tags=hostname=${host.name}
poolgroup=StaticEscape
pool.path=/mpatha/dcache/StaticEscape_001

# project StaticEscape on filesystem /mpathb
[poolsDomain_${host.name}_StaticEscape]
[poolsDomain_${host.name}_StaticEscape/pool]
pool.name=StaticEscape_002
pool.tags=hostname=${host.name}
poolgroup=StaticEscape
pool.path=/mpathb/dcache/StaticEscape_002

# project StaticEscape on filesystem /mpathd
[poolsDomain_${host.name}_StaticEscape]
[poolsDomain_${host.name}_StaticEscape/pool]
pool.name=StaticEscape_004
pool.tags=hostname=${host.name}
poolgroup=StaticEscape
pool.path=/mpathd/dcache/StaticEscape_004
```

```
# project StaticEscape on filesystem /mpathe
[poolsDomain_${host.name}_StaticEscape]
[poolsDomain_${host.name}_StaticEscape/pool]
pool.name=StaticEscape_005
pool.tags=hostname=${host.name} poolgroup=StaticEscape

pool.path=/mpathe/dcache/StaticEscape_005
# project StaticEscape on filesystem /mpathc
[poolsDomain_${host.name}_StaticEscape]
[poolsDomain_${host.name}_StaticEscape/pool]
pool.name=StaticEscape_003
pool.tags=hostname=${host.name} poolgroup=StaticEscape

pool.path=/mpathc/dcache/StaticEscape_003
# project StaticEscape on filesystem /mpathf
[poolsDomain_${host.name}_StaticEscape]
[poolsDomain_${host.name}_StaticEscape/pool]
pool.name=StaticEscape_006
pool.tags=hostname=${host.name} poolgroup=StaticEscape
pool.path=/mpathf/dcache/StaticEscape_006

# project staticPool on filesystem /mpathg
[poolsDomain_${host.name}_staticPool]
[poolsDomain_${host.name}_staticPool/pool]
pool.name=staticPool_009
pool.tags=hostname=${host.name} poolgroup=staticPool
pool.path=/mpathg/dcache/staticPool_009

# project staticPool on filesystem /mpathh
[poolsDomain_${host.name}_staticPool]
[poolsDomain_${host.name}_staticPool/pool]
pool.name=staticPool_010
pool.tags=hostname=${host.name} poolgroup=staticPool
pool.path=/mpathh/dcache/staticPool_010
```


dCache

Architecture VII - Layout Files, i.e. building domains

```
# Disknode dCache layout configuration
# =====
```

```
# disknodes needs to know how to contact central node
dcache.zookeeper.connection = t2-dpm-dome.na.infn.it:2181
```

→ Zookeeper connection

```
[doorsDomain-${host.name}]
[doorsDomain-${host.name}/webdav]
webdav.authn.basic = true
webdav.authn.protocol = https
webdav.loginbroker.port = 443
```

→ Setting secure connection for WebDAV on port 443

```
[doorsDomain-${host.name}/xrootd]
xrootd.plugins=gplazma:gsi
```

```
[doorsDomain-${host.name}/ftp]
ftp.authn.protocol=gsi
```

→ Using GPLazma GSI for xrootd authentication and GSI + ftp to achieve Grid FTP authentication

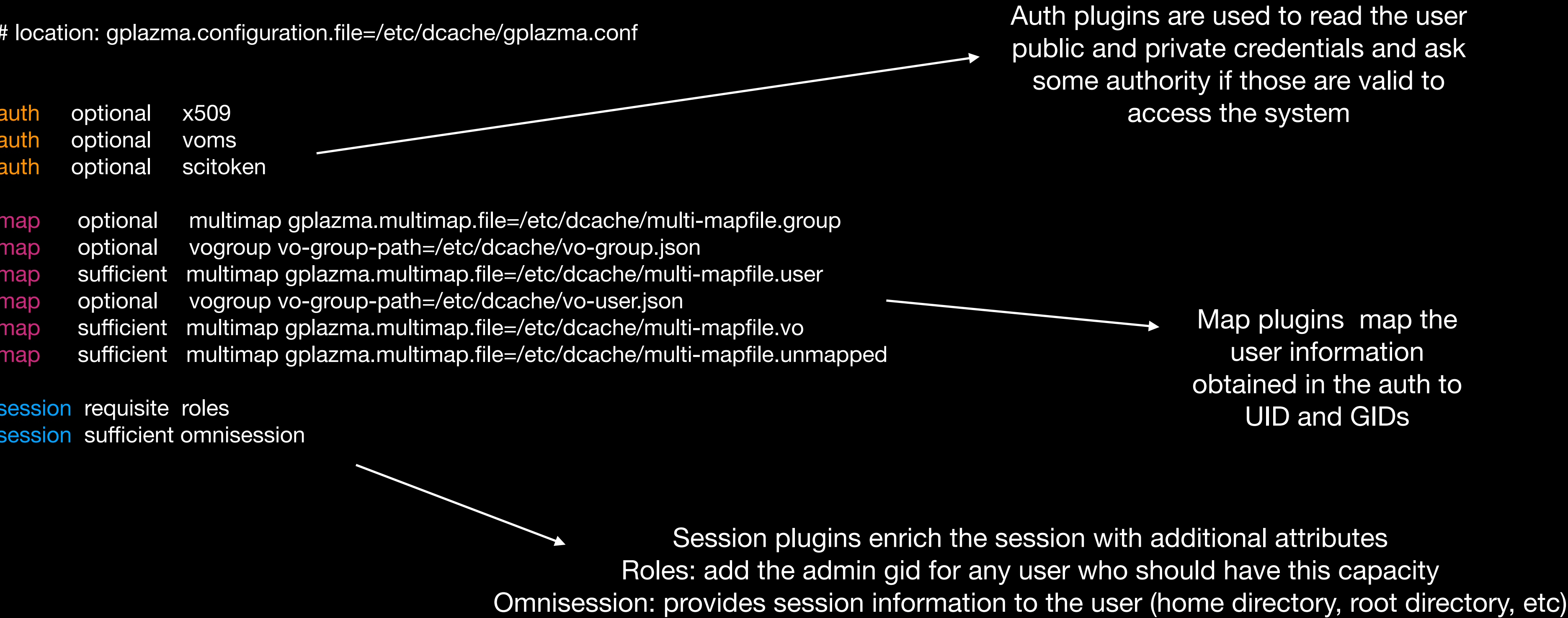
```
# project staticPool on filesystem /data/t2-disk01-static
[poolsDomain_${host.name}_staticPool]
[poolsDomain_${host.name}_staticPool/pool]
pool.name=staticPool_008
pool.tags=hostname=${host.name} poolgroup=staticPool
pool.path=/data/t2-disk01-static/dcache/staticPool_008
```

→ Pool Domains names must be unique
Pool Group is used for space reservation and authentication

dCache

Architecture VI - Authentication and Authorisation

The gPlazma interface (Grid-aware Pluggable AuthoriZation Management) allows to build complex authorisation logics through configurable plugins. Plugins are chained together in stacks.



dCache

Architecture VII - Authentication and Authorisation (The Stack)

Optional: the success or failure of this plug-in is only important if it is the only plug-in in the stack associated with this type

Sufficient: success of this plug-in is enough to satisfy the authentication requirements. Failure is not fatal

Required: Failure is fatal but only after all the other plugins in the stack have been invoked

Requisite: like required, but in this case control is directly returned to the door

location: gplazma.configuration.file=/etc/dcache/gplazma.conf

auth	optional	x509
auth	optional	voms
auth	optional	scitoken
map	optional	multimap gplazma.multimap.file=/etc/dcache/multi-mapfile.group
map	optional	vogroup vo-group-path=/etc/dcache/vo-group.json
map	sufficient	multimap gplazma.multimap.file=/etc/dcache/multi-mapfile.user
map	optional	vogroup vo-group-path=/etc/dcache/vo-user.json
map	sufficient	multimap gplazma.multimap.file=/etc/dcache/multi-mapfile.vo
map	sufficient	multimap gplazma.multimap.file=/etc/dcache/multi-mapfile.unmapped

session requisite roles
session sufficient omnisession

X509 extract the X.509 certificate chains from the credentials of a user (certificate stored in: /etc/grid-security/certificates)

Scitokens certification

Multimap dCache requires that authenticated credentials are mapped to posix style **username**, **uid** and **gid**. Records of the mappings are keeper in the gplazma.multimap.file. Multiple files can be configured to organise complex relations. Files are located in: /etc/dcache/

voms takes x509 certificates and tests them against trusted CAs. Verified certificates are passed along in the stack (voms are stored in: /etc/grid-security/vomsdir/)

Vogroup same as multi map but utilising json for scitoken

dCache

Architecture VIII - multi-map file

```
# multi-map file.group
# =====

# ATLAS mapping of client groups/roles to gid numbers
fqan:/atlas          gid:1000 group:writer
group:atlas_oidc     gid:1000, true group:writer
username:atlas_oidc
oidcgrp:/atlas       gid:1000
fqan:/atlas/Role=production  gid:1001
oidcgrp:/atlas/production  gid:1001
fqan:/atlas/usatlas/Role=production  gid:1002
oidcgrp:/atlas/usatlas/production  gid:1002

# BELLE mapping of client groups/roles to gid numbers
fqan:/belle          gid:2000 group:writer
group:belle_oidc     gid:2000,true group:writer username:belle_oidc
oidcgrp:/belle       gid:2000

# DTEAM mapping of client groups/roles to gid numbers
fqan:/dteam          gid:3000 group:writer
group:dteam_oidc     gid:3000,true group:writer username:dteam_oidc
oidcgrp:/dteam       gid:3000

# ESCAPE mapping of client groups/roles to gid numbers
fqan:/escape         gid:4000 group:writer
group:escape_oidc    gid:4000,true group:writer username:escape_oidc
oidcgrp:/escape      gid:4000

# OPS mapping of client groups/roles to gid numbers
fqan:/ops             gid:5000 group:writer
group:ops_oidc        gid:5000,true group:writer username:ops_oidc
oidcgrp:/ops          gid:5000

# WLCG mapping of client groups/roles to gid numbers
fqan:/wlcg            gid:6000 group:writer
group:wlcg_oidc       gid:6000,true group:writer username:wlcg_oidc
oidcgrp:/wlcg         gid:6000

# multi-map file.user
# =====

"dn:/C=IT/O=INFN/OU=Personal Certificate/L=Napoli/CN=Alessandra Doria" username:alessandra_doria uid:11000
"dn:/C=UK/O=eScience/OU=Manchester/L=HEP/CN=james collinson" username:james_collinson uid:11001
"dn:/C=UK/O=eScience/OU=Manchester/L=HEP/CN=robert barnsley" username:robert_barnsley uid:11002
"dn:/C=UK/O=eScience/OU=Manchester/L=HEP/CN=rohini joshi" username:rohini_joshi uid:11003
"dn:/DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=afkiaras/CN=817926/CN=Aristeidis Fkiaras" username:afkiaras uid:11004
"dn:/DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=arturos/CN=679537/CN=Arturos Sanchez Pineda" username:arturos uid:11005
"dn:/DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=avendrel/CN=849027/CN=Alba Vendrell Moya" username:avendrel uid:11006
"dn:/DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=ddmadmin/CN=531497/CN=Robot: ATLAS Data Management" username:ddmadmin uid:11007
"dn:/DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=ewp2c01/CN=531497/CN=Robot: ESCAPE WP2 CERN 01" username:ewp_c__ uid:11008
"dn:/DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=ewp2c01/CN=817926/CN=Robot: ESCAPE WP2 CERN 01" username:ewp_c____1 uid:11009
"dn:/DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=jhover/CN=653174/CN=John Raymond Hover" username:jhover uid:11010
"dn:/DC=org/DC=terena/DC=tcs/C=ES/O=Port dInformacio Cientifica/CN=Agustin Bruzzese bruzzese@pic.es" username:agustin_bruzzese_bruzzese_pic_es uid:11011
"dn:/DC=org/DC=terena/DC=tcs/C=IT/L=Frascati/O=Istituto Nazionale di Fisica Nucleare/OU=Cloud/CN=escape-monitoring.cloud.cnaf.infn.it"
username:escape_monitoring_cloud_cnaf_infn_it uid:11012
"dn:/DC=org/DC=terena/DC=tcs/C=IT/O=Istituto Nazionale di Fisica Nucleare - INFN/CN=Federica Agostini fagostin@infn.it" username:federica_agostini_fagostin_infn_it
uid:11013
"dn:/DC=org/DC=terena/DC=tcs/C=IT/O=Istituto Nazionale di Fisica Nucleare - INFN/CN=Robot - Andrea Ceccanti aceccant@infn.it"
username:robot___andrea_ceccanti_aceccant_infn_it uid:11014
"dn:/DC=org/DC=terena/DC=tcs/C=IT/O=Istituto Nazionale di Fisica Nucleare/CN=Alessandra Doria adoria@infn.it" username:alessandra_doria_adoria_infn_it uid:11015
"dn:/DC=org/DC=terena/DC=tcs/C=IT/O=Istituto Nazionale di Fisica Nucleare/CN=Bernardino Spisso spisso@infn.it" username:bernardino_spisso_spisso_infn_it uid:11016
"dn:/DC=org/DC=terena/DC=tcs/C=IT/O=Istituto Nazionale di Fisica Nucleare/CN=Davide Michelino dmichelino@infn.it" username:davide_michelino_dmichelino_infn_it
uid:11017
"dn:/DC=org/DC=terena/DC=tcs/C=IT/O=Istituto Nazionale di Fisica Nucleare/CN=Elisabetta Vilucchi evilucch@infn.it" username:elisabetta_vilucchi_evilucch_infn_it uid:11018
"dn:/O=GRID-FR/C=FR/O=CNRS/OU=LAPP/CN=Stephane Jezequel" username:stephane_jezequel uid:11019

# Omnisession plugin (omnisession) configuration
# =====
#
# where "group:writer" have privileges to the whole namespace should be
# sufficient (fine grained permissions comes from user/group/ACLs).

group:writer      root:/ home:/
```

- OpenID Connect (OIDC): built on top of OAuth 2.0
- @OP style suffix for oidcgrp fields

dCache

Architecture VIII - multi-map file

```
# multi-mapfile.vo
# =====
#
# example for DTEAM:
#fqan:/dteam      uid:1000 username:dteam
#username:dteam_oidc      uid:1001
```

generated from imported config file

```
# ATLAS mapping of username resolved by vogroup plugin to uid
username:atlas      uid:1000
#username:atlas_oidc      uid:1000
username:atlas_production      uid:1001
username:atlas_usatlas_production uid:1002
```

```
# BELLE mapping of username resolved by vogroup plugin to uid
username:belle      uid:2000
#username:belle_oidc      uid:2000
```

```
# DTEAM mapping of username resolved by vogroup plugin to uid
username:dteam      uid:3000
#username:dteam_oidc      uid:3000
```

```
# ESCAPE mapping of username resolved by vogroup plugin to uid
username:escape      uid:4000
#username:escape_oidc      uid:4000
```

```
# OPS mapping of username resolved by vogroup plugin to uid
username:ops      uid:5000
#username:ops_oidc      uid:5000
```

```
# WLCG mapping of username resolved by vogroup plugin to uid
username:wlcg      uid:6000
#username:wlcg_oidc      uid:6000
```

```
# multi-mapfile.unmapped
# =====
#
# example for DTEAM mapping for groups and roles with no explicit configuration
# (remove to grant access only to explicitly configured VO FQAN):
#fqan:/dteam      uid:1000 gid:1000,true username:dteam
```

generated from imported config file

```
# ATLAS default mapping for groups and roles with no explicit configuration
fqan:/atlas      uid:1000 gid:1000,true username:atlas
```

```
# BELLE default mapping for groups and roles with no explicit configuration
fqan:/belle      uid:2000 gid:2000,true username:belle
```

```
# DTEAM default mapping for groups and roles with no explicit configuration
fqan:/dteam      uid:3000 gid:3000,true username:dteam
```

```
# ESCAPE default mapping for groups and roles with no explicit configuration
fqan:/escape      uid:4000 gid:4000,true username:escape
```

```
# OPS default mapping for groups and roles with no explicit configuration
fqan:/ops      uid:5000 gid:5000,true username:ops
```

```
# WLCG default mapping for groups and roles with no explicit configuration
fqan:/wlcg      uid:6000 gid:6000,true username:wlcg
~
```

vo-group.json

```
[
  {
    "fqan": "/atlas",
    "mapped_gid": 1000
  },
  {
    "fqan": "/atlas/Role=production",
    "mapped_gid": 1001
  },
  {
    "fqan": "/atlas/usatlas/Role=production",
    "mapped_gid": 1002
  },
  {
    "fqan": "/belle",
    "mapped_gid": 2000
  },
  {
    "fqan": "/dteam",
    "mapped_gid": 3000
  },
  {
    "fqan": "/escape",
    "mapped_gid": 4000
  },
  {
    "fqan": "/ops",
    "mapped_gid": 5000
  },
  {
    "fqan": "/wlcg",
    "mapped_gid": 6000
  }
]
```

vo-user.json

```
[
  {
    "fqan": "/atlas",
    "mapped_gid": 1000,
    "mapped_undef": "atlas"
  },
  {
    "fqan": "/atlas/Role=production",
    "mapped_gid": 1001,
    "mapped_undef": "atlas_production"
  },
  {
    "fqan": "/atlas/usatlas/Role=production",
    "mapped_gid": 1002,
    "mapped_undef": "atlas_usatlas_production"
  },
  {
    "fqan": "/belle",
    "mapped_gid": 2000,
    "mapped_undef": "belle"
  },
  {
    "fqan": "/dteam",
    "mapped_gid": 3000,
    "mapped_undef": "dteam"
  },
  {
    "fqan": "/escape",
    "mapped_gid": 4000,
    "mapped_undef": "escape"
  },
  {
    "fqan": "/ops",
    "mapped_gid": 5000,
    "mapped_undef": "ops"
  },
  {
    "fqan": "/wlcg",
    "mapped_gid": 6000,
    "mapped_undef": "wlcg"
  }
]
```

dCache

Architecture VIII - multi-map file

```
# multi-map file.group
# =====

# ATLAS mapping of client groups/roles to gid numbers
fqan:/atlas          gid:1000 group:writer
group:atlas_oidc     gid:1000, true group:writer
username:atlas_oidc
oidcgrp:/atlas       gid:1000
fqan:/atlas/Role=production  gid:1001
oidcgrp:/atlas/production  gid:1001
fqan:/atlas/usatlas/Role=production  gid:1002
oidcgrp:/atlas/usatlas/production  gid:1002

# BELLE mapping of client groups/roles to gid numbers
fqan:/belle          gid:2000 group:writer
group:belle_oidc     gid:2000,true group:writer username:belle_oidc
oidcgrp:/belle       gid:2000

# DTEAM mapping of client groups/roles to gid numbers
fqan:/dteam          gid:3000 group:writer
group:dteam_oidc     gid:3000,true group:writer username:dteam_oidc
oidcgrp:/dteam       gid:3000

# ESCAPE mapping of client groups/roles to gid numbers
fqan:/escape         gid:4000 group:writer
group:escape_oidc    gid:4000,true group:writer username:escape_oidc
oidcgrp:/escape      gid:4000

# OPS mapping of client groups/roles to gid numbers
fqan:/ops            gid:5000 group:writer
group:ops_oidc       gid:5000,true group:writer username:ops_oidc
oidcgrp:/ops         gid:5000

# WLCG mapping of client groups/roles to gid numbers
fqan:/wlcg           gid:6000 group:writer
group:wlcg_oidc      gid:6000,true group:writer username:wlcg_oidc
oidcgrp:/wlcg        gid:6000
```

```
# multi-map file.user
# =====

"dn:/C=IT/O=INFN/OU=Personal Certificate/L=Napoli/CN=Alessandra Doria" username:alessandra_doria uid:11000
"dn:/C=UK/O=eScience/OU=Manchester/L=HEP/CN=james collinson" username:james_collinson uid:11001
"dn:/C=UK/O=eScience/OU=Manchester/L=HEP/CN=robert barnsley" username:robert_barnsley uid:11002
"dn:/C=UK/O=eScience/OU=Manchester/L=HEP/CN=rohini joshi" username:rohini_joshi uid:11003
"dn:/DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=afkiaras/CN=817926/CN=Aristeidis Fkiaras" username:afkiaras uid:11004
"dn:/DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=arturos/CN=679537/CN=Arturos Sanchez Pineda" username:arturos uid:11005
"dn:/DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=avendrel/CN=849027/CN=Alba Vendrell Moya" username:avendrel uid:11006
"dn:/DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=ddmadmin/CN=531497/CN=Robot: ATLAS Data Management" username:ddmadmin uid:11007
"dn:/DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=ewp2c01/CN=531497/CN=Robot: ESCAPE WP2 CERN 01" username:ewp_c__ uid:11008
"dn:/DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=ewp2c01/CN=817926/CN=Robot: ESCAPE WP2 CERN 01" username:ewp_c____1 uid:11009
"dn:/DC=ch/DC=cern/OU=Organic Units/OU=Users/CN=jhover/CN=653174/CN=John Raymond Hover" username:jhover uid:11010
"dn:/DC=org/DC=terena/DC=tcs/C=ES/O=Port dInformacio Cientifica/CN=Agustin Bruzzese bruzzese@pic.es" username:agustin_bruzzese_bruzzese_pic_es uid:11011
"dn:/DC=org/DC=terena/DC=tcs/C=IT/L=Frascati/O=Istituto Nazionale di Fisica Nucleare/OU=Cloud/CN=escape-monitoring.cloud.cnaf.infn.it"
username:escape_monitoring_cloud_cnaf_infn_it uid:11012
"dn:/DC=org/DC=terena/DC=tcs/C=IT/O=Istituto Nazionale di Fisica Nucleare - INFN/CN=Federica Agostini fagostin@infn.it" username:federica_agostini_fagostin_infn_it
uid:11013
"dn:/DC=org/DC=terena/DC=tcs/C=IT/O=Istituto Nazionale di Fisica Nucleare - INFN/CN=Robot - Andrea Ceccanti aceccant@infn.it"
username:robot___andrea_ceccanti_aceccant_infn_it uid:11014
"dn:/DC=org/DC=terena/DC=tcs/C=IT/O=Istituto Nazionale di Fisica Nucleare/CN=Alessandra Doria adoria@infn.it" username:alessandra_doria_adoria_infn_it uid:11015
"dn:/DC=org/DC=terena/DC=tcs/C=IT/O=Istituto Nazionale di Fisica Nucleare/CN=Bernardino Spisso spisso@infn.it" username:bernardino_spisso_spisso_infn_it uid:11016
"dn:/DC=org/DC=terena/DC=tcs/C=IT/O=Istituto Nazionale di Fisica Nucleare/CN=Davide Michelino dmichelino@infn.it" username:davide_michelino_dmichelino_infn_it
uid:11017
"dn:/DC=org/DC=terena/DC=tcs/C=IT/O=Istituto Nazionale di Fisica Nucleare/CN=Elisabetta Vilucchi evilucch@infn.it" username:elisabetta_vilucchi_evilucch_infn_it uid:11018
"dn:/O=GRID-FR/C=FR/O=CNRS/OU=LAPP/CN=Stephane Jezequel" username:stephane_jezequel uid:11019

# Omnisession plugin (omnisession) configuration
# =====
#
# where "group:writer" have privileges to the whole namespace should be
# sufficient (fine grained permissions comes from user/group/ACLs).

group:writer      root:/ home/
```

- OpenID Connect (OIDC): built on top of OAuth 2.0
- @OP style suffix for oidcgrp fields

dCache

Architecture VIII - multi-map file

```
# multi-map file.group
# =====

# ATLAS mapping of client groups/roles to gid numbers
fqan:/atlas                gid:1000  group:writer
group:atlas_oidc           gid:1000, true group:writer
username:atlas_oidc
oidcgrp:/atlas              gid:1000
fqan:/atlas/Role=production gid:1001
oidcgrp:/atlas/production   gid:1001
fqan:/atlas/usatlas/Role=production gid:1002
oidcgrp:/atlas/usatlas/production gid:1002
```

```
# multi-map file.user
# =====

"dn:/C=IT/O=INFN/OU=Personal Certificate/L=Napoli/CN=Alessandra Doria"
username:alessandra_doria uid:11000

# Omnisection plugin (omnisection) configuration
# =====
#
# where "group:writer" have privileges to the whole namespace should be
# sufficient (fine grained permissions comes from user/group/ACLs).

group:writer      root:/ home:/
```

Predicates:

- **Dn**: matches user's Distinguished Name, obtained by authenticating via X509;
- **Group**: matches user's group;
- **fqan**: matches user's VOMS FQAN;
- **Oidc**: matches an OpenID-Connect subject claim from a specific OP;
- **Oidcgrp**: matches an OpenID-Connect groups claim value;
- **Username**: matches the username of a user

User principals may match multiple line in the configuration. Attributes override each other, first win.

Fully Qualified Authorities, Groups, Username and odcgrp are associated to gid

dCache

Architecture IX - PoolManager Service

The heart of a dCache System is the poolmanager. When a user performs an action on a file - reading or writing - a transfer request is sent to the dCache system. The poolmanager then decides how to handle this request

PSU (Pool Selection Unit): responsible for finding the set of **pools** which can be used for a transfer request.

- We can adjust the transfer conditions by telling the PSU which pools are permitted for which type of file requests.
The PSU generates a list of allowed storage pools for each request, which are then dynamically used by the PoolManager.

→ **Link**: consists of a set of **unit** groups and a list of pools. Each link contains one or more **unit groups** (conditions). Each group contains several units and the unit group is matched if at least one unit within the group is matched. All group units have to be matched by the transfer request to initiate a transfer.

- | | |
|--------------------------|--|
| • Network (-net) | IP address and a net mask. It is satisfied if the request is coming from an IP address with the subnet given by the address/netmask pair |
| • Protocol (-protocol) | Name of the protocol and version number. It is satisfied if the request is coming from the correct protocol |
| • Storage Class (-store) | Storage Class. It is satisfied if the request file has this storage class |
| • Cache Class (-dcache) | Cache Class. It is satisfied if the cache class of the requested file agrees with it |

Type of Transfer: four possible attributes (-readpref, -writepref, -p2ppref, -cachepref). A value of zero disables this type of transfer, any positive assert preference. A negative value for -p2ppref sets it equal to -readpref.

Pool Groups: (pgroup) pools can be grouped together in **pool groups**. If the -dynamic parameter is used, pools are added dynamically on the basis of a pool tag

Link Groups: (linkGroup) are used by the SRM SpaceManager to make reservations against space. A **link group** is a collection of **links** and each link pointing to the associated **pool groups**. Each link group knows about the size of its available space (SUM of pools). Has five boolean properties (replicaAllowed, outputAllowed, custodiaAllowed, onlineAllowed, nearlineAllowed)

dCache

Architecture IX - PoolManager Service

```
# PoolManager configuration in /var/lib/dcache/config/poolmanager.conf
# =====
#
```

```
psu create unit -store *@*
psu create unit -net 0.0.0.0/0.0.0.0
psu create unit -net ::/0
psu create unit -protocol */*
```

Units

```
psu create ugroup any-protocol
psu addto ugroup any-protocol */*
```

unit groups

```
psu create ugroup any-store
psu addto ugroup any-store *@*
```

```
psu create ugroup world-net
psu addto ugroup world-net 0.0.0.0/0.0.0.0
psu addto ugroup world-net ::/0
```

```
psu create pgroup spacemanager_poolGroup_StaticEscape -dynamic -tags=poolgroup=StaticEscape
psu create pgroup spacemanager_poolGroup_Inf-volatile -dynamic -tags=poolgroup=Inf-volatile
psu create pgroup spacemanager_poolGroup_roma1-volatile -dynamic -tags=poolgroup=roma1-volatile
psu create pgroup spacemanager_poolGroup_staticPool -dynamic -tags=poolgroup=staticPool
```

pool groups

```
psu create link default-link any-protocol any-store world-net
psu set link default-link -readpref=10 -writepref=10 -cachepref=10 -p2ppref=-1
```

```
psu create link spacemanager_link_StaticEscape any-protocol any-store world-net
psu set link spacemanager_link_StaticEscape -readpref=10 -writepref=10 -cachepref=0 -p2ppref=-1
psu addto link spacemanager_link_StaticEscape spacemanager_poolGroup_StaticEscape
```

```
psu create link spacemanager_link_Inf-volatile any-protocol any-store world-net
psu set link spacemanager_link_Inf-volatile -readpref=10 -writepref=10 -cachepref=0 -p2ppref=-1
psu addto link spacemanager_link_Inf-volatile spacemanager_poolGroup_Inf-volatile
```

Links

```
psu create link spacemanager_link_roma1-volatile any-protocol any-store world-net
psu set link spacemanager_link_roma1-volatile -readpref=10 -writepref=10 -cachepref=0 -p2ppref=-1
psu addto link spacemanager_link_roma1-volatile spacemanager_poolGroup_roma1-volatile
```

```
psu create link spacemanager_link_staticPool any-protocol any-store world-net
psu set link spacemanager_link_staticPool -readpref=10 -writepref=10 -cachepref=0 -p2ppref=-1
psu addto link spacemanager_link_staticPool spacemanager_poolGroup_staticPool
```

```
psu create linkGroup spacemanager_linkGroup_StaticEscape
psu set linkGroup custodialAllowed spacemanager_linkGroup_StaticEscape true
psu set linkGroup replicaAllowed spacemanager_linkGroup_StaticEscape true
psu set linkGroup nearlineAllowed spacemanager_linkGroup_StaticEscape true
psu set linkGroup outputAllowed spacemanager_linkGroup_StaticEscape true
psu set linkGroup onlineAllowed spacemanager_linkGroup_StaticEscape true
psu addto linkGroup spacemanager_linkGroup_StaticEscape spacemanager_link_StaticEscape
```

Link Groups

```
psu create linkGroup spacemanager_linkGroup_Inf-volatile
psu set linkGroup custodialAllowed spacemanager_linkGroup_Inf-volatile true
psu set linkGroup replicaAllowed spacemanager_linkGroup_Inf-volatile true
psu set linkGroup nearlineAllowed spacemanager_linkGroup_Inf-volatile true
psu set linkGroup outputAllowed spacemanager_linkGroup_Inf-volatile true
psu set linkGroup onlineAllowed spacemanager_linkGroup_Inf-volatile true
psu addto linkGroup spacemanager_linkGroup_Inf-volatile spacemanager_link_Inf-volatile
```

```
psu create linkGroup spacemanager_linkGroup_roma1-volatile
psu set linkGroup custodialAllowed spacemanager_linkGroup_roma1-volatile true
psu set linkGroup replicaAllowed spacemanager_linkGroup_roma1-volatile true
psu set linkGroup nearlineAllowed spacemanager_linkGroup_roma1-volatile true
psu set linkGroup outputAllowed spacemanager_linkGroup_roma1-volatile true
psu set linkGroup onlineAllowed spacemanager_linkGroup_roma1-volatile true
psu addto linkGroup spacemanager_linkGroup_roma1-volatile spacemanager_link_roma1-volatile
```

```
psu create linkGroup spacemanager_linkGroup_staticPool
psu set linkGroup custodialAllowed spacemanager_linkGroup_staticPool true
psu set linkGroup replicaAllowed spacemanager_linkGroup_staticPool true
psu set linkGroup nearlineAllowed spacemanager_linkGroup_staticPool true
psu set linkGroup outputAllowed spacemanager_linkGroup_staticPool true
psu set linkGroup onlineAllowed spacemanager_linkGroup_staticPool true
psu addto linkGroup spacemanager_linkGroup_staticPool spacemanager_link_staticPool
```

dCache

Architecture X - Space Reservation

Space reservation guarantees the the request amount of storage space type is made available by the storage system.

- Retention Policy: describes the quality of the storage service
 - OUTPUT: output files are allowed
 - REPLICA: lower quality, only one copy is stored
 - CUSTODIAL: higher quality, storage on TAPE
- Access Latency: describes the data availability
 - NEARLINE: data is allowed to migrate to permanent media. Retrieving data may result in delays due to the transfer from permanent media
 - ONLINE: data is readily available allowing for faster access (guaranties a copy on disk)

```
psu create pgroup spacemanager_poolGroup_StaticEscape -dynamic -tags=poolgroup=StaticEscape
```

```
psu create link spacemanager_link_StaticEscape any-protocol any-store world-net
psu set link spacemanager_link_StaticEscape -readpref=10 -writepref=10 -cachepref=0 -p2ppref=-1
psu addto link spacemanager_link_StaticEscape spacemanager_poolGroup_StaticEscape
```

```
psu create linkGroup spacemanager_linkGroup_StaticEscape
psu set linkGroup custodialAllowed spacemanager_linkGroup_StaticEscape true
psu set linkGroup replicaAllowed spacemanager_linkGroup_StaticEscape true
psu set linkGroup nearlineAllowed spacemanager_linkGroup_StaticEscape true
psu set linkGroup outputAllowed spacemanager_linkGroup_StaticEscape true
psu set linkGroup onlineAllowed spacemanager_linkGroup_StaticEscape true
psu addto linkGroup spacemanager_linkGroup_StaticEscape spacemanager_link_StaticEscape
```

perform space reservation

```
ssh -p 22224 -l admin localhost
```

```
\c SrmSpaceManager
```

```
reserve space -owner=/atlas/Role=production -desc=atlas-static-escape
               -lg=spacemanager_linkGroup_staticEscape 5TB
```

```
# SpaceManagerLinkGroupAuthorizationFile
# =====
# location: spacemanager.authz.link-group-file-
# name=/etc/dcache/LinkGroupAuthorization.conf
#
# example
#LinkGroup spacemanager_linkGroup
#*/Role=*
```

```
spacemanager_linkGroup_StaticEscape
/atlas/Role=*
/escape/Role=*
```

```
spacemanager_linkGroup_Inf-volatile
/atlas/Role=*
```

```
spacemanager_linkGroup_roma1-volatile
/atlas/Role=*
```

```
spacemanager_linkGroup_staticPool
/escape/Role=*
/belle/Role=*
/atlas/Role=*
```

Coming back to authorisation

We are allowing users with a specific role to make reservations on given linkgroups, and this will have dedicate storage pools

dCache

Wrapping Up Authorisation and Space and Pool Reservation

1. We have created in the layout file at least a Pool with a given pool tag
2. We have created the linkGroup in the poolmanager.conf with the associated pool tag
3. We have associated gids to a given VO in the multi-map file
4. We have given to the VO space reservation permissions in the SpaceManagerLinkGroupAuthorizationFile and reserved some space through the admin interface

```
[poolsDomain_${host.name}_StaticEscape]
[poolsDomain_${host.name}_StaticEscape/pool]
pool.name=StaticEscape_006
pool.tags=hostname=${host.name} poolgroup=StaticEscape
pool.path=/mpathf/dcache/StaticEscape_006
```

```
psu create pgroup spacemanager_poolGroup_StaticEscape -dynamic -tags=poolgroup=StaticEscape

psu create link spacemanager_link_StaticEscape any-protocol any-store world-net
psu set link spacemanager_link_StaticEscape -readpref=10 -writepref=10 -cachepref=0 -p2ppref=-1
psu addto link spacemanager_link_StaticEscape spacemanager_poolGroup_StaticEscape

psu create linkGroup spacemanager_linkGroup_StaticEscape
psu set linkGroup custodialAllowed spacemanager_linkGroup_StaticEscape true
psu set linkGroup replicaAllowed spacemanager_linkGroup_StaticEscape true
psu set linkGroup nearlineAllowed spacemanager_linkGroup_StaticEscape true
psu set linkGroup outputAllowed spacemanager_linkGroup_StaticEscape true
psu set linkGroup onlineAllowed spacemanager_linkGroup_StaticEscape true
psu addto linkGroup spacemanager_linkGroup_StaticEscape spacemanager_link_StaticEscape
```

fqan:/atlas	gid:1000	group:writer
group:atlas_oidc	gid:1000,	true group:writer username:atlas_oidc
oidcgrp:/atlas	gid:1000	
fqan:/atlas/Role=production	gid:1001	
oidcgrp:/atlas/production	gid:1001	
fqan:/atlas/usatlas/Role=production	gid:1002	
oidcgrp:/atlas/usatlas/production	gid:1002	

```
spacemanager_linkGroup_StaticEscape
/atlas/Role=*
/escape/Role=*
```


dCache

Architecture XI - Configuration

```
# dCache configuration in /etc/dcache/dcache.conf
# =====
#
# If you have a big pool or many pools per domain, you probably want more
# memory in the java process than the default 512m. 1024m ought to be enough
# for a single 10TiB pool, but you probably need 2048m for a 20TiB pool. As a rule
# of thumb, use 512m + 512m for every 10 TiB of pool (rounded to a nice value).

# Tape write pools are usually not big and you can use 1024m. A large file system
# cache is more useful on such pools.dcache.java.memory.heap = 4096m

# The default is 512m, but in particular with xrootd this isn't quite enough.
#
# There is no reason to scale this by pool size - how much is required depends more on
# access patterns. For Alice tape write pools this MUST be at least 2048m.
dcache.java.memory.direct = 2048m

# dCache services configuration
dcache.layout = layout-`${host.fqdn}`

# database configuration
#dcache.db.host = localhost
#dcache.db.user = dcache
#dcache.db.password =

# Diskonly storage
# https://dcache.org/old/manuals/Book-7.2/config-SRM.shtml#utilization-of-space-reservations-for-data-storage
dcache.default-retention-policy=REPLICA
dcache.default-access-latency=ONLINE

# enable ACL support
pnfsmanager.enable.acl = true

# Explicit port configuration for LAN/WAN access with GridFTP and HTTP protocol
dcache.net.wan.port.min = 20000
dcache.net.wan.port.max = 25000
# LAN port range for internal pool to pool communication
dcache.net.lan.port.min = 33115
dcache.net.lan.port.max = 33145
# use same ports for all protocols (GridFTP, WebDAV, xroot)
pool.mover.xrootd.port.min = ${dcache.net.wan.port.min}
pool.mover.xrootd.port.max = ${dcache.net.wan.port.max}
```

```
# Allow Let's encrypt certificates that doesn't provide CRLs
#dcache.authn.hostcert.verify=true
#dcache.authn.crl-mode = IF_VALID

# BDII Glue Info Provider
# Besides these two basic configuration option it's necessary to update also
# /etc/dcache/info-provider.xml and install bdii packages for details see
# https://dcache.org/old/manuals/Book-7.2/config-info-provider.shtml
info-provider.site-unique-id=GOCDB_SITE_NAME
info-provider.se-unique-id=dcache.example.com
```

```
# EGI StAR (APEL Storage Accounting)
#star.gid-mapping = 2000=/atlas, 1010=/dteam, 1020=/wlcg
```

```
# By default, dCache auto initializes pool directories. To avoid
# accidentally initializing a pool in a mount point, we configure all
# pools to wait for the pool file system to be mounted.
pool.wait-for-files = ${pool.path}/data
```

```
# All pools are tagged by the FQDN of the host. dCache uses this to avoid
# replicating files to pools on the same host.
pool.tags = hostname=${host.fqdn}
```

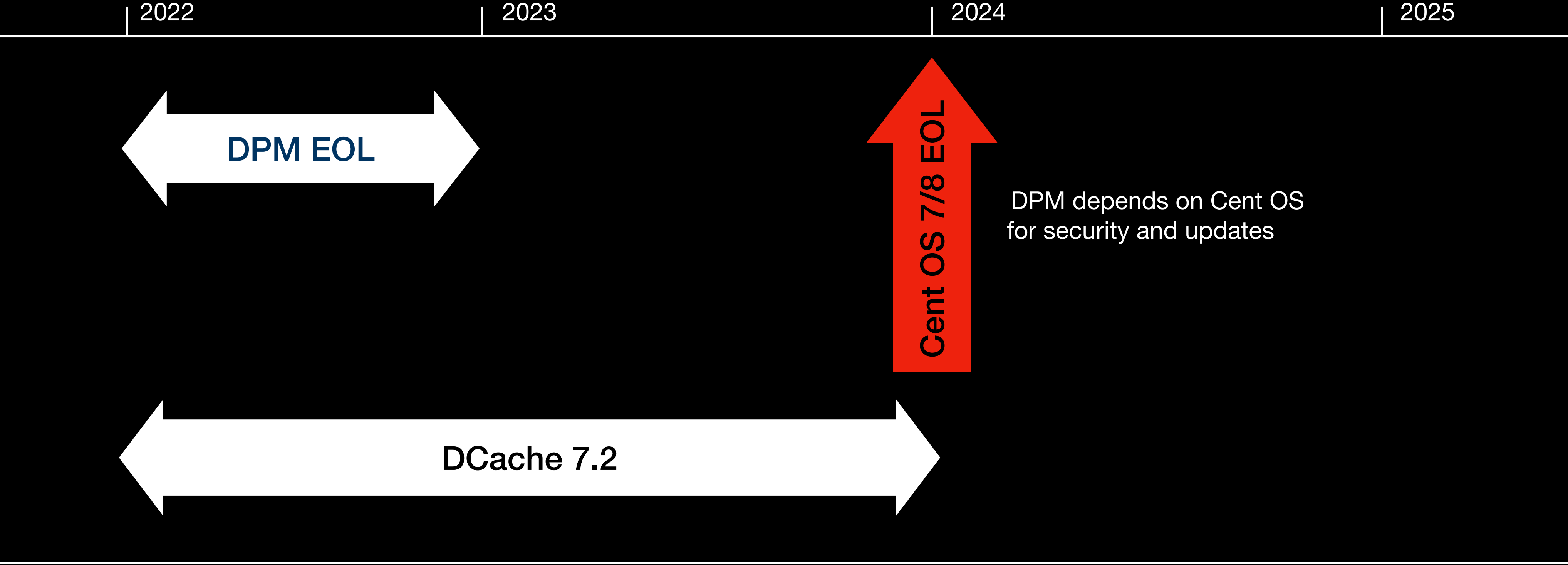
```
# To be able to use xroot-tpc with EOS it is necessary to add "unix" (now enabled by default)
#pool.mover.xrootd.tpc-authn-plugins=gsi,unix
#pool.mover.xrootd.plugins=gsi,unix
```

```
# pool startup optimization
# https://indico.desy.de/event/25462/contributions/57176/attachments/36943/46186/dcache-project-whatsnew.pdf
pool.limits.scan-threads=8
pool.plugins.meta.db!je.checkpointer.wakeupInterval = 60 s
pool.plugins.meta.db!je.checkpointer.bytesInterval = 0
pool.plugins.meta.db!je.cleaner.wakeupInterval= 0 s
pool.plugins.meta.db!je.log.fileCacheSize=1024
```

```
# Enable writing with xroot protocol (be aware that xroot protocol doesn't
# protect established connection from packet content changes on the wire
# unless you enable integrity validation or TLS)
xrootd.authz.write-paths = /
```


OS and DPM Migration

Countdown to EOL



Installing dCache

Link to the Official Guide: [guide](#)
Link to the Migration Guide: [guide](#)

Tested in: Cent OS 7, 8, Rocky 9

1. Install most recent dCache 7.2 release on all machines

```
yum install -y https://dcache.org/old/downloads/1.9/repo/7.2/dcache-7.2.22-1.noarch.rpm
```

2. Install OpenJDK or set it through alternatives to 11

```
yum install java-11-openjdk-headless httpd-tools
```

```
alternatives --set java $(alternatives --display java | grep 'family java-11-openjdk' | cut -d' ' -f1)
```

3. Configure direct SSH access to the admin shell

```
ssh-keygen -C admin@localhost -t rsa -N "" -f id_rsa
```

```
cat /root/.ssh/id_rsa.pub > /etc/dcache/admin/authorized_keys2
```

4. Install PostgreSQL 14 (Cent OS 7), PostgreSQL 13 (CentOS8, 9)

Cent OS 7

```
yum install -y https://download.postgresql.org/pub/repos/yum/reporpms/EL-7-x86\_64/pgdg-redhat-repo-latest.noarch.rpm
```

```
yum install -y postgresql14-server
```

```
# enable PostgreSQL for dCache local password-less access
```

```
/usr/pgsql-14/bin/postgresql-14-setup initdb
```

```
cat > /var/lib/pgsql/14/data/pg_hba.conf <<EOF
```

```
# enable connections to postgresql database from remote machines in case you use dedicated dbnode
```

```
#perl -p -i -e "s/^#.*listen_addresses *= *'localhost'/listen_addresses = '*'/" /var/lib/pgsql/14/data/postgresql.conf
```

```
systemctl enable postgresql-14
```

```
systemctl start postgresql-14
```

```
# create PostgreSQL databases used by dCache
```

```
createuser -U postgres --no-superuser --no-createrole --createdb --pwprompt --no-password dcache
```

```
createdb -U dcache chimera
```

```
createdb -U dcache spacemanager
```

```
createdb -U dcache pinmanager
```

```
createdb -U dcache srm
```

Installing dCache

Link to the Official Guide: [guide](#)
Link to the Migration Guide: [guide](#)

Tested in: Cent OS 7, 8, Rocky 9

4. Install PostgreSQL 14 (Cent OS 7), PostgreSQL 13 (CentOS8, 9) Cent OS 8 / 9

```
dnf module list postgresql
```

```
dnf module enable postgresql:13
```

```
dnf install postgresql-server
```

```
# enable PostgreSQL for dCache local password-less access
```

```
/usr/bin/postgresql-setup --initdb
```

```
cat > /var/lib/pgsql/data/pg_hba.conf <<EOF
```

```
# enable connections to postgresql database from remote machines in case you use dedicated dbnode
```

```
#perl -p -i -e "s/^#.*listen_addresses *= *'localhost'/listen_addresses = '*'/" /var/lib/pgsql/14/data/postgresql.conf
```

```
systemctl enable postgresql-13
```

```
systemctl start postgresql-13
```

```
# create PostgreSQL databases used by dCache
```

```
createuser -U postgres --no-superuser --no-createrole --createdb --pwprompt --no-password dcache
```

```
createdb -U dcache chimera
```

```
createdb -U dcache spacemanager
```

```
createdb -U dcache pinmanager
```

```
createdb -U dcache srm
```

5. Set WebDAV permission to use default port (443 Port)

```
mkdir /etc/systemd/system/dcache@.service.d
```

```
cat > /etc/systemd/system/dcache@.service.d/capabilities.conf <<EOF
```

```
[Service]
```

```
AmbientCapabilities=CAP_NET_BIND_SERVICE
```

```
EOF
```


Installing dCache

Link to the Official Guide: [guide](#)
Link to the Migration Guide: [guide](#)

Tested in: Cent OS 7, 8, Rocky 9

6. Create your dCache configuration by performing the DPM migration or by manually creating the following files

1. *.layout files in /etc/dcache/layout/*.layout (one for each node)
2. dcache.conf (one for each node)
3. gplazma.conf
4. Multi-map file.* (this could be even a single file depending on your needs)
5. omnisession.conf
6. vo-group.json, vo-user.json
7. LinkGroupAuthorization.conf
8. poolmanager.conf (/var/lib/dcache/config/)
9. Setting your space reservations (see <https://www.dcache.org/manuals/Book-6.2/config-SRM.shtml>)

7. Host TLS certificates

chown dcache /etc/grid-security/hostcert.pem /etc/grid-security/hostkey.pem

8. Initialise dCache database structures and start dcache service

start dCache services
systemctl daemon-reload
systemctl start dcache.target

9. Update CA certificates and CRL update

yum install fetch-crl
fetch-crl

10. Verify dCache is working and configuration is correct

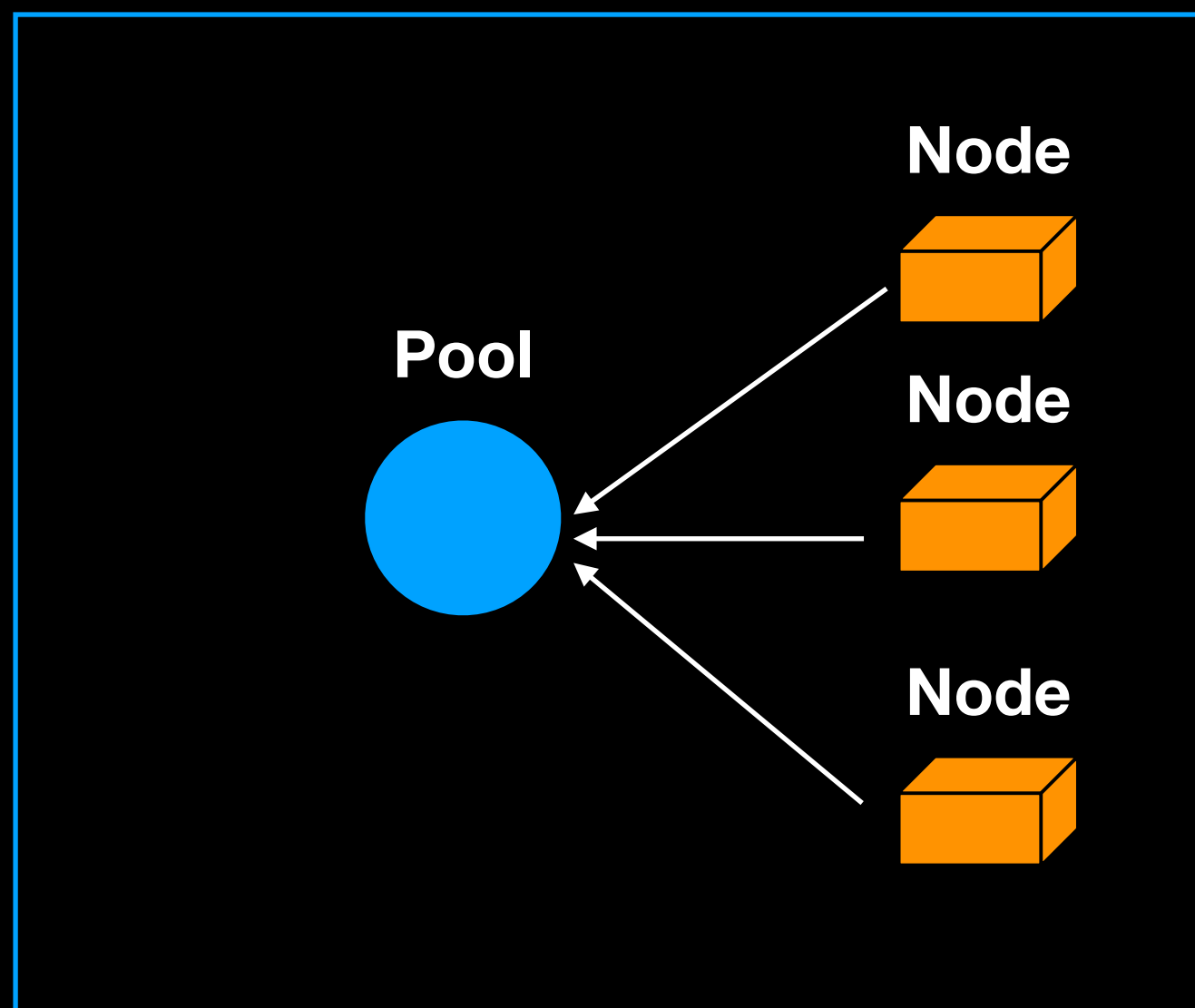
1. Check logs: **tail -fn 100 /var/log/dcache/**
2. Check configured link groups and space available for reservation: **ssh -p 22224 -l admin localhost '\c SrmSpaceManager; ls link groups'**
3. Check configured space reservations: **ssh -p 22224 -l admin localhost '\c SrmSpaceManager; ls spaces -e'**

Migrating from DPM

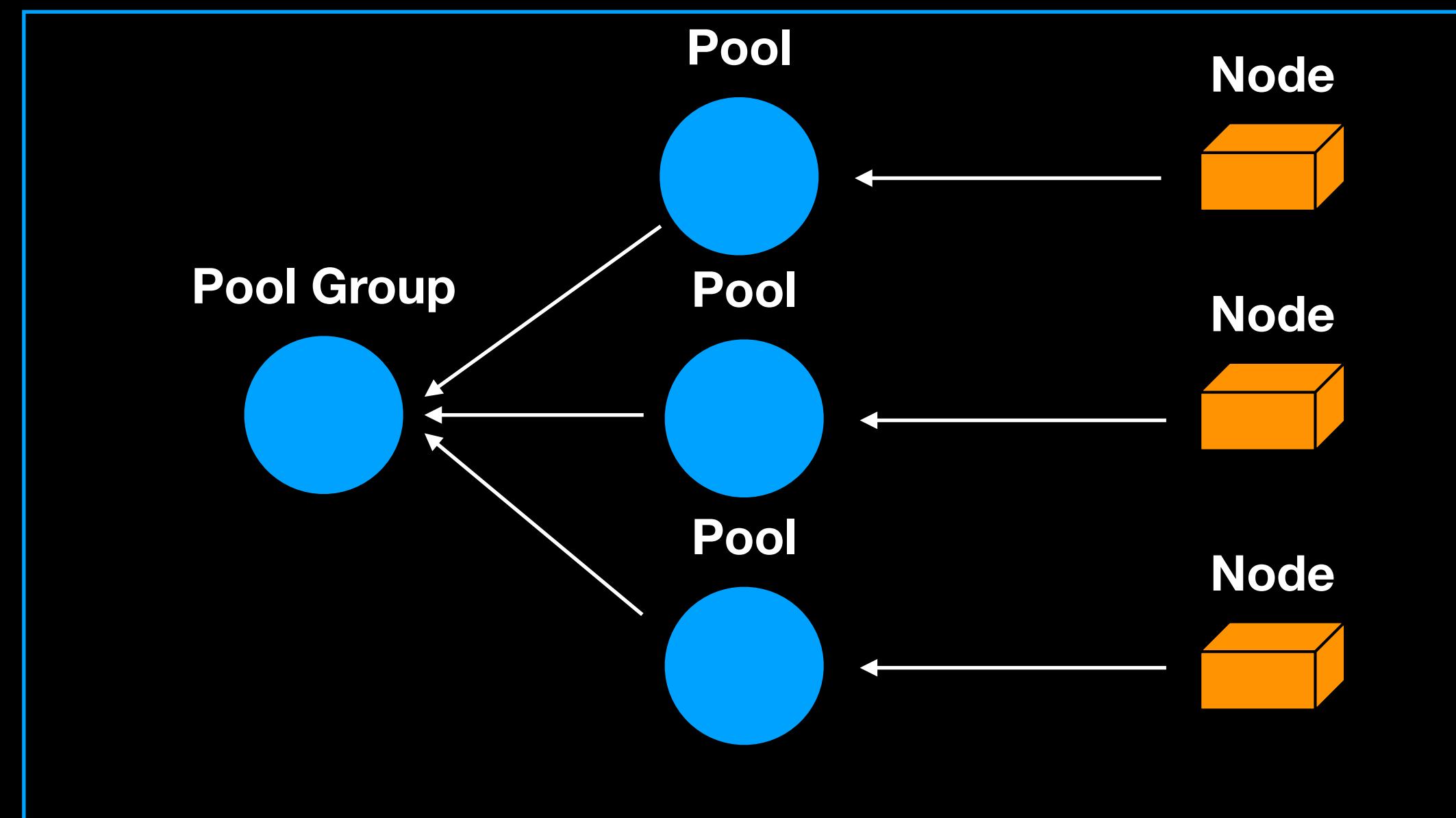
General Overview of the migration process

1. Fix DPM database inconsistencies
2. Generate dCache configuration on the basis of the DPM configuration and dump DPM namespace
3. Distribute generated configuration and authentication files and import namespace
4. Finalising the migration and checking dCache status

DPM



dCache



Migrating from DPM

Taking point 6 and let migrate.py do it for you. Perform steps 1 - 5 of the Installation guide before starting

1. Make sure that DPM is in consistent state and fix potential issues:

1. Fix lost and dark data (assuming ssh password-less access to the disk nodes):

```
dmlite-shell --log-level=INFO --log-file=/tmp/dpm-lost-and-dark.log -e 'dbck lost-and-dark-show script' > /tmp/dpm-lost-and-dark.sh  
sh /tmp/dpm-lost-and-dark.sh
```

2. Fix internal database inconsistencies:

```
dmlite-shell --log-level=DEBUG --log-size=104857600 --log-file=/tmp/dpm-dbck.log -e 'dbck dpm-dbck update'
```

3. Correct pool for space token (sometimes there is a file / space token mismatch that must be fixed):

```
dmlite-shell --log-level=DEBUG --log-size=104857600 --log-file=/tmp/dpm-dbck.pool-file.log -e 'dbck pool-file update nthreads=8'
```

4. Calculating missing checksum (direct database query, set threads to the number of filesystems):

```
mysql cns_db -h dpmdb.example.com -u dpm -p -e "SELECT r.host, r.fs, COUNT(*) AS count, SUM(m.filesize) AS size FROM Cns_file_metadata m INNER JOIN  
Cns_file_replica r USING(fileid) WHERE m.filemode & 32768 = 32768 AND r.status = '-' AND m.status = '-' AND m.csumtype != 'AD' GROUP BY r.host,  
r.fs ORDER BY r.host, r.fs"
```

5. Further recommendations: delete all the references to inactive nodes before starting calculation of the missing checksum (speedup)

2. Dump DPM namespace and Configuration

1. If you are on Cent OS, install the following packages:

```
yum install -y python3 python36-dateutil python36-pycurl python36-m2crypto python36-mysql python36-paramiko python36-ldap3 python36-rpm python36-lxml  
yum install -y python36-psycpg2
```

2. Grab the migrate.py script:

CentOS7 with installed dmlite-shell package

```
cp /usr/lib/python2.7/site-packages/dmliteshell/migrate.py .
```

CentOS8 with installed dmlite-shell package

```
cp /usr/lib/python3.6/site-packages/dmliteshell/migrate.py .
```

3. Stop and disable DPM services to avoid any updates:

```
systemctl stop httpd rfiod srmv2.2 dpnsdaemon dpm dpm-gsftp xrootd@dpmredir
```

```
systemctl disable httpd rfiod srmv2.2 dpnsdaemon dpm dpm-gsftp xrootd@dpmredir
```

Migrating from DPM

Taking point 6 and let migrate.py do it for you

2. Dump DPM namespace and Configuration

4. Export DPM namespace and configuration

```
python3 migrate.py --log-level=DEBUG --log-file=dpm-migrate.log --dpm-export --dpm-dbhost=dpmdb.fqdn --dpm-dbuser=dpmdb_user --dpm-dbpasswd=dpmdb_secret
```

3. Import namespace data in dCache and distribute generated dCache configuration files

1. Distribute the generated files as per step 6 of the installation guide, then perform step 7 and 8 (TLS and dCache service start)

2. Space configuration and reservation

import dCache link group configuration by running (this script is writing in the poolmanager.conf)

```
cat admin-cli.psu | grep -v ^# | ssh -p 22224 -l admin localhost
```

```
ssh -p 22224 -l admin localhost '\c PoolManager; psu dump setup' (to check results)
```

once you have configured the linkgroups, check if space is available for reservation

```
ssh -p 22224 -l admin localhost '\c SrmSpaceManager; ls link groups'
```

perform space reservations

```
cat admin-cli.reserve | grep -v ^# | ssh -p 22224 -l admin localhost
```

list space reservations

```
ssh -p 22224 -l admin localhost '\c SrmSpaceManager; ls spaces -e'
```

3. Namespace import

stop dCache services on headnote

```
systemctl stop dcache.target
```

```
yum install -y python3 python3-psycpg2
```

populate dCache namespace with data from namespace.csv and config.csv

```
python3 migrate.py --log-level=DEBUG --log-file=migrate-dcache-import.log --dcache-import
```

4. Move data files from DPM to dCache directories on each disk node

Successfull namespace import provide data-*.example.com.csv files with informations about DPM source directory and dCache destination directory for all data files registered in the dCache namespace. There will be one file for each disk node, copy it alongside the migrate.py script and then execute

```
python migrate.py --log-level=INFO --log-file=migrate-dcache-link.log --link --link-file=data-dpmdisk1.example.com.csv
```


Migrating from DPM

Finishing Touches

4. Host TLS certificates

```
chown dcache /etc/grid-security/hostcert.pem /etc/grid-security/hostkey.pem
```

5. Initialise dCache database structures and start dcache service

```
# start dCache services
```

```
systemctl daemon-reload
```

```
systemctl start dcache.target
```

6. Update CA certificates and CRL update

```
yum install fetch-crl
```

```
fetch-crl
```

7. Verify dCache is working and configuration is correct

1. Check logs: `tail -fn 100 /var/log/dcache/`

2. Check configured link groups and space available for reservation: `ssh -p 22224 -l admin localhost '\c SrmSpaceManager; ls link groups'`

3. Check configured space reservations: `ssh -p 22224 -l admin localhost '\c SrmSpaceManager; ls spaces -e'`

8. Remove DPM services, old DPM data directories and database

```
yum remove "*dmlite*" "*dpm*"
```

```
cat config.csv | grep ^filesystem | sed 's/.*,\([^,]*\),\([^,]*\),[[:digit:]]\+,[[:digit:]]\+/\1 \2/' > dpm-hostfs.dat
```

```
cat dpm-hostfs.dat | while read H P; do echo "ssh $H ls $P | grep -v '^ *$' | grep -v dcache | sed 's#^#ssh $H rm -rf $P/#'; done > dpm-hostfs-discover.sh
```

```
sh dpm-hostfs-discover.sh > dpm-hostfs-cleanup.sh
```

```
sh dpm-hostfs-cleanup.sh (this file contains all the bash commands to remove DPM each file from the storage)
```

```
mysql -u root -p
```

```
DROP DATABASE cns_db
```

```
DROP DATABASE dpm_db
```


Migrating our Testbed

Testbed Architecture

- OLD DPM storage architecture:
 - DPM Head (Naples)
 - Disk Nodes:
 - 2 in Naples (~2TB, ~20TB) - ONLINE
 - 1 in Rome (~1TB) - ONLINE
 - 1 Frascati (~20TB) - OFFLINE (to test DPM migration behaviour)

The system was part of the DIOS (Escape EU Project) test infrastructure. It supported VO based authentication (ATLAS, ESCAPE, BELLE, DTEAM).

System was upgraded to DPM version 1.15.2

Migrating our Testbed

About migration time

- Migration is fast, on our system 4.6 TB of data were transferred in 30 minutes;
- DPM files which have been lost due to disconnected hard drive, are removed sequentially after checksum which took a long time. The current solution is to remove from DPM namespace the corrupted entries before starting the migration.
- Bug in the migrate.py script at line 1439 which has been notified to Petr (EPEL package update coming soon)

The script was previously like this:

```
if len(self._config['spacetoken']):  
    raise Exception("Importing dCache namespace without any spacetoken is not currently supported")
```

And it has been modified to this:

```
if len(self._config['spacetoken']) == 0:  
    raise Exception("Importing dCache namespace without any spacetoken is not currently supported")
```

Thank you for your Attention

Keep In Touch

- micheledelliveneri@gmail.com
- michele.delliveneri@na.infn.it
- michele.delliveneri@unina.it

