

dCache Workshop DESY  
18.- 19.01.2007  
Hamburg



# Xrootd/dCache

# Implementation



Martin Radicke



# File transfer methods in 1.7.0



- ◆ wide-area transfer (stream-based)
  - GridFTP (GSI authentication)
  - HTTP
- ◆ local-area transfer (random access)
  - dCap (dCache native protocol, GSI auth. available)
  - **xrootd**



# What is xrootd?



- ◆ well-defined protocol, specification freely available
- ◆ client/server suite using the xrootd protocol
  - distributed daemon serving disk data, developed by SLAC
  - client (integrated in ROOT, POSIX wrapper), developed by INFN Padova
- ◆ major design goals
  - fault tolerance (adding or removing servers, failover)
  - performance (TCP connection multiplexing, load balancing)
  - smart client supports server by understanding redirects and doing several retries in case of server failures



# Xrootd/dCache features



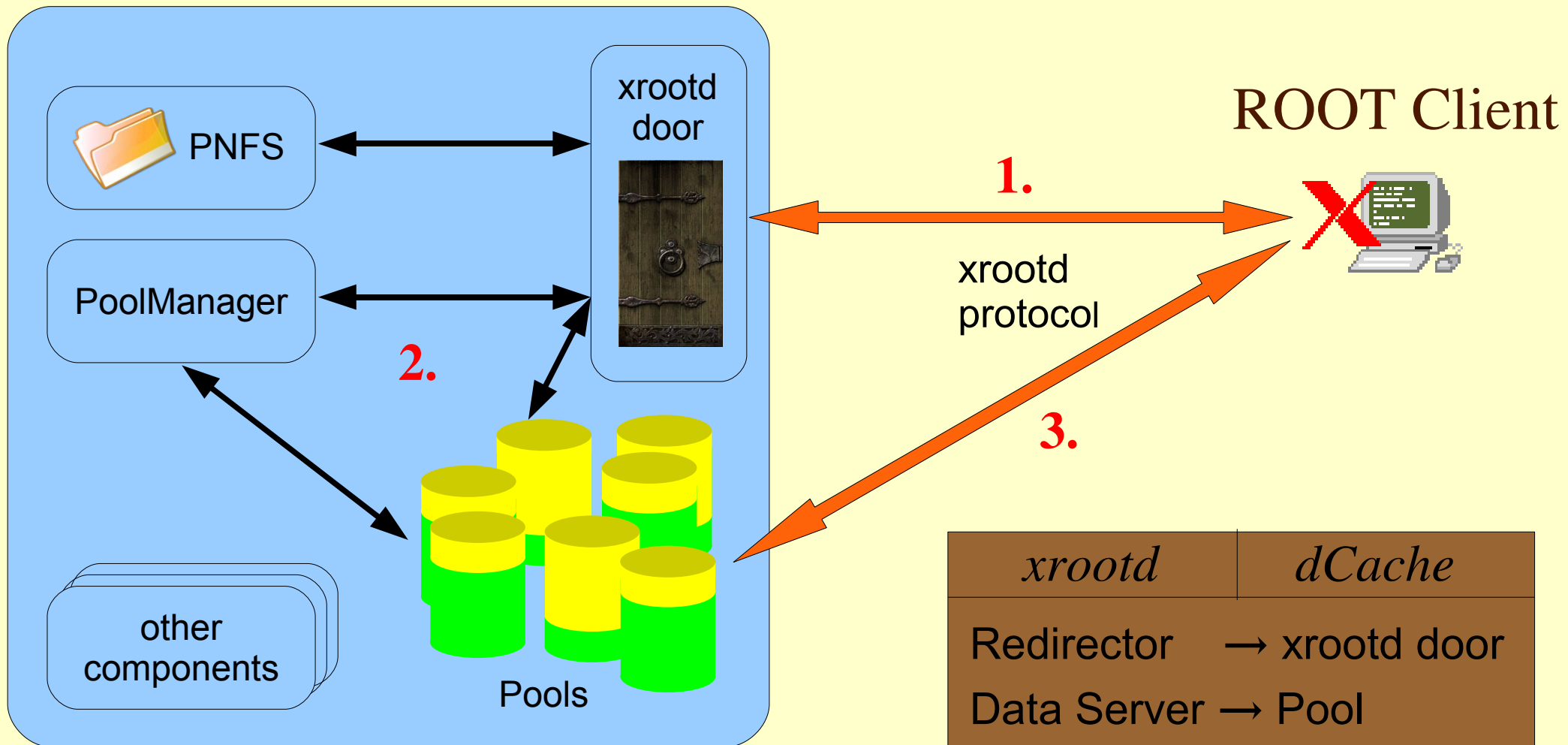
- ◆ dCache SE is a fully functional xrootd-server
  - native implementation of all required protocol methods (xrootd door)
  - from dCache point of view same treatment as other protocols
- ◆ transparency on the client side
  - no code or configuration changes necessary
- ◆ works with main clients
  - ROOT toolkit `TXNetFile::Open("root://dCacheServer:1094/pnfs/pathToFile", "r")`
  - xrscp (basic CLI)
- ◆ xrootd redirection scheme maps to dCache's internal load balancing mechanism, based on load and space of pools



# Architectural overview



## dCache SE





- ◆ remote policy: Token-based authorization (ALICE)
  - encrypted token attached to xrootd file open request
  - created by external service (e.g. file catalogue)
  - has limited lifetime
  - carries DN of user, permissions (r/w) for a set of files
  - xrootd/dCache decrypts token and applies permissions
  - more authorization methods pluggable
- ◆ local dCache SE policy
  - xrootd access can be restricted to read-only (for each door)
- ◆ authentication: yet to come



- ◆ multiple xrootd doors

- client iterates over server list to find an available door

```
TXNetFile::Open("root://door1,door2,door3/pnfs/pathToFile", "r")
```

- ◆ applying different access pattern

- one xrootd door set read-only

-> allowing public access

- another xrootd door set to read-write, but require authorization

-> centrally controlled write access (file catalogue with ACLs)



- ◆ LHC ALICE experiment
  - analysis applications heavily based on ROOT/PROOF
  - xrootd in use for data management, additional need for interfacing LCG/gLite services (SRM, FTS)
- ◆ evaluation of xrootd/dCache finished successfully
  - GSI Darmstadt, CERN, GridPP
- ◆ close contact to ALICE and fast development cycles
- ◆ xrootd/dCache about to go into production as an ALICE SE during PDC07





# Conclusion



- ♦ dCache SE got enhanced by the xrootd access protocol
- ♦ acts as an xrootd-server while making full use of dCache core functionalities (mainly pool selection and namespace handling)
- ♦ first security mechanism added (Token authorization)
- ♦ about to go into production (ALICE service challenge)



- ◆ authentication based on GSI under discussion
  - mapping of DN to local user, rights management
  - reduces the risk of stealing the authorization token to a minimum
- ◆ as user community grows, more protocol features are implemented

If you are interested in a special feature,  
contact the developers!