

4. HEP-CG Workshop
13.12 - 14.12. 2007
Siegen



dCache

The Scalable Storage Element



Martin Radicke
Patrick Fuhrmann

GEFÖRDERT VOM



Bundesministerium
für Bildung
und Forschung



The new dCache 1.8

DESY



Features of 1.8



- ★ SRM 2.2 support
 - ◆ backwards compatible to SRM 1.1
- ★ embedded DB holding pool metadata
 - faster pool startups
- ★ files with/without tape-backup on the same pool
- ★ support of individual HSMs per poolgroup
- ★ HSMCleaner: providing hook on file delete
- ★ reduced number of PNFS-mounts
 - ◆ SRM- and GridFTP-Door still need mount for 'ls'
- ★ makes full use of Java 5 enhancements
 - ◆ cleaner code due to generics, annotations, etc.
 - ◆ benefit from new NIO- and concurrency-libraries

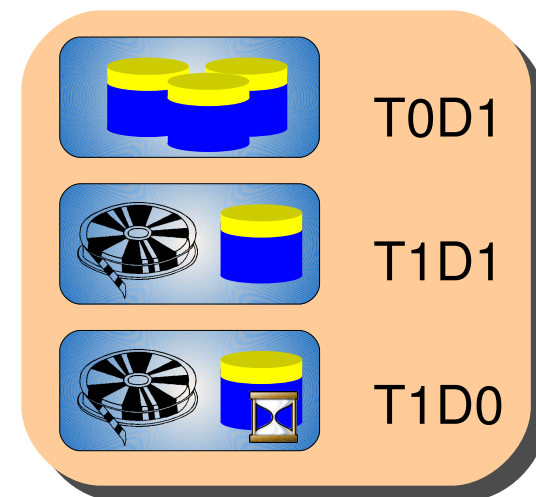


SRM 2.2



★ dCache supports

- CUSTODIAL (T1D_x)
- NON-CUSTODIAL (T0D1)
- static and dynamic space reservation
 - ◆ pool binding happens at file open, not at space creation
→ load balancing not sacrificed
- namespace operations, pre-stage
- implicit space reservation for GridFTP



★ central testing done by WLCG

- ◆ full testsuite running basic-, usecase and stress-tests against Cern, Tier-1s and few other endpoints
- ◆ also testing interoperability (3rd party transfers with FTS) across all SRM implementations



Wide Area Access



- ★ full GridFTP-2 support
 - always direct dataflow client <-> pool
 - built-in checksums
 - better performance (less load on the pool)
 - ◆ direct data transfer from file system cache to network buffer without copying to main memory
 - patches for GridFtp-Client-tools currently under evaluation by Globus
 - backwards compatible with GridFtp-1
 - heavily used in Scandinavia (NDGF)

DESY



Local Access Protocols



★ dCap-Client in latest ROOT version

■ VectorRead

- ◆ max. vector size: 1024
- ◆ client falls back to normal READ if v. size > 1024

■ ReadAhead-Buffer

- ◆ available, but disabled by ROOT due to internal caching

★ xrootd

■ VectorRead now supported

■ async server under development

- ◆ overlapping I/O-requests to reduce latency
- ◆ out-of-band-messages allow internal reordering



- ★ replacement of PNFS
(dCache's classical namespace provider)
- ★ motivation:
 - PNFS ist the **expected performance bottleneck** when dCache scales into the Petabyte-range with millions of file entries
- ★ some features:
 - ◆ not bound to the limitations of NFS 2/3
 - ◆ thin layer on top of any relational database
 - ◆ scales along the DB
- ★ part of dCache 1.8, but not enabled by default
 - ◆ we need further real-world experience





Current Status



- ★ we are in transition from dCache 1.7 to 1.8
- ★ most of the Tier-1s have upgraded or upgrading now (9 out of 11 use dCache)
 - mostly with SpaceManagement switched off
- ★ *Should the Tier-2s upgrade now?*
 - no rush, unless they really need SRM 2.2
 - ◆ experiments do full SRM 2.2-tests on Tier-1s in February (CCRC08)
 - ◆ only if everything goes right, the Tier-2s will be included
 - the feedback of the Tier-1s will stabilise code a lot
- ★ no Apt/Yum-repositories yet, but install by YAIM possible



Known Issues With 1.8



- ★ implicit space reservation for GsidCap not there
 - but is under development
- ★ space miscalculation in pools
 - problem not yet fully understood, hard to reproduce
 - porting SpaceMonitorWatch from 1.7 for more debug-output
- ★ SRM-SpaceManager and PinManager not as stable as expected, fixes expected next week

DESY



New Developments

DESY



The NFS 4.1 Protocol



- ★ upcoming industry standard for local area access
 - ◆ protocol spec is expected to finalise in Jan. 2008
 - ◆ industry consortium (incl. dCache) specifies the standard and does interoperability tests with all implementations
 - relatively slow progress, but high quality
- ★ features
 - ◆ **Clients will come with all major OSs for free**
(officially part of Linux kernel within next year)
 - ◆ 1st nfs-version which understands distributed data
 - ◆ **faster**: compound RPC calls
 - ◆ open/close-semantic to keep track of open files
 - ◆ 'Dead' client recovery (ping)
 - ◆ **GSS** authentication on FS-level
 - ◆ **ACLs**



NFS 4.1: Status of dCache



- ★ Java-based protocol-engine almost complete
 - comprises of MetaData- and Data-Server (aka Door and Mover)
 - pluggable GSS-auth under development
- ★ integration into dCache still needs a lot of work
 - model of the generic pool handling protocol-specific movers needs to be extended

DESY



Access Control Lists (ACLs)



★ compliant to NFS 4

- ◆ Unix Permissions < **NFS 4 ACLs** < Windows ACLs

★ Design Goals

- ◆ provide functionality which is compatible to other relevant LCG-middleware
- ◆ (uid, gid) or (DN,FQAN) mapped to virtuid and virtguid
- ◆ control any kind of namespace objects (dirs, files, VO spaces) as well as dCache admin-commands (future)

★ impl. status

- ◆ external component ready, currently integrated into PNFS and Chimera
- ◆ fallback to classical Unix Permissions where ACLs undefined (or to a default '*Allow all/Deny all*')
- ◆ intuitive tool for managing ACLs needs to be developed



ACLs: Information Flow



AR-handler queries the principal handler

It returns

- a virtual user ID (virtUid)
- a list of virtual group IDs (virtGid)

The AclHandler component is queried to retrieve an ACL for a given object.

Principal Handler

AclHandler

A request contains information on

- The **subject** that started the request.
- The **object** which is the target of the request.
- The **origin** of the request.

Access Request Handler

Access is
allowed/denied
or
'don't know'

ACLMapper

Task: evaluate an ACL.

Input: information on
subject, object, origin.

Output: two access masks

- *defMsk* (all defined access bits)
- *allowMsk* (allowed access bits)

AclMapper

AclMatcher

This component matches an access request (*action*) and the access masks. The reply is used to allow or deny access.

by courtesy of
David Melkumyan
Irina Kozlova

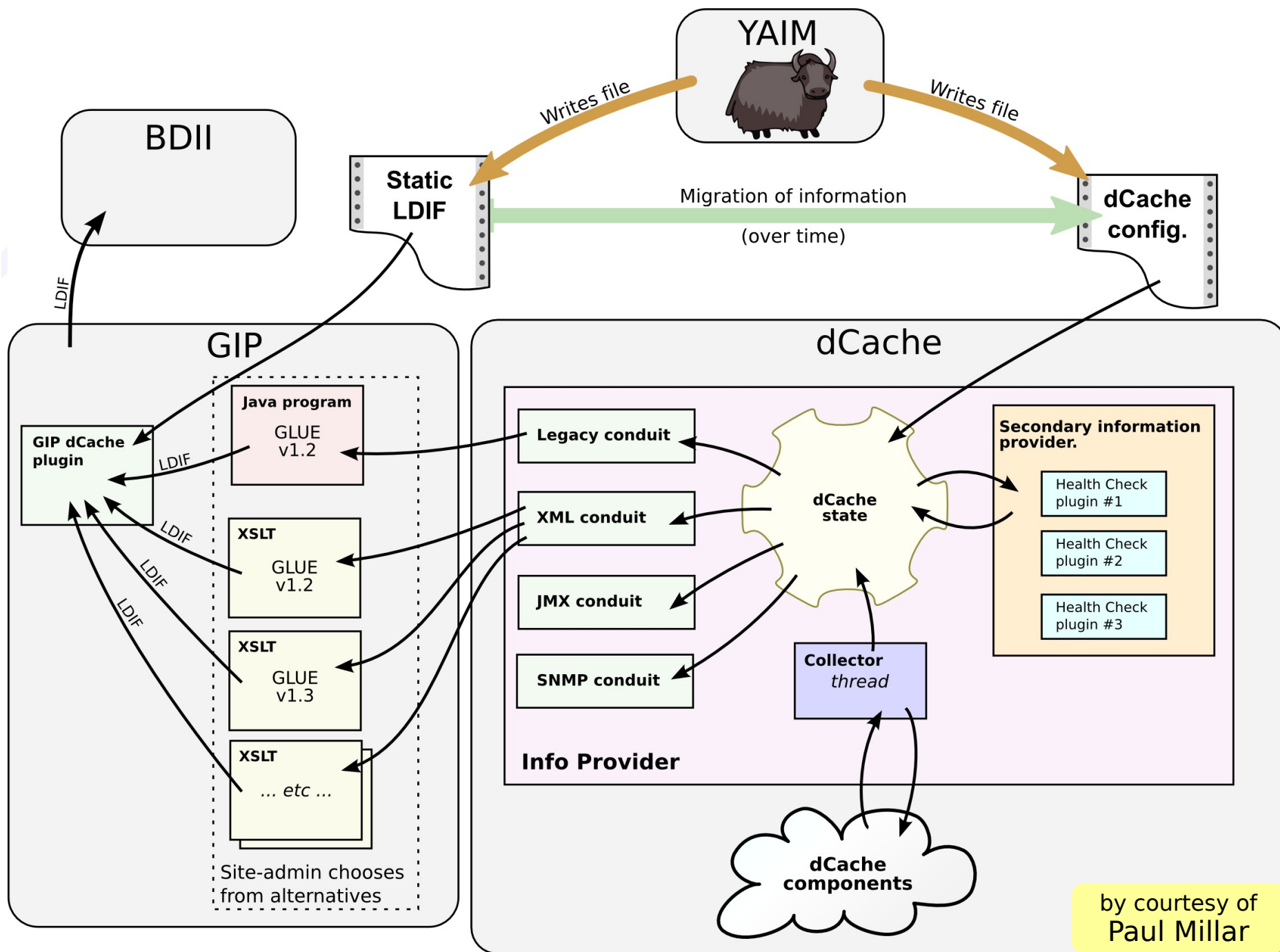


- ★ GlueSchema Version 1.3 agreed Sep. 07
- ★ InfoProvider currently refactored to support Glue Schema 1.2 AND 1.3
- ★ Design Goals
 - flexible architecture to support future types of information publishing and monitoring (e.g. dCache health status)
 - support Glue Schema 1.2 AND 1.3
- ★ impl. status: just started

DESY



InfoProvider: new architecture



by courtesy of
Paul Millar



Sustainability

DESY



dCache as a Service



- ★ develop/ reproduce and fix bugs
- ★ test (unit and functional)/package/release
- ★ document (wiki/book)
- ★ support on several levels
- ★ pass CERN certification process to get packages into gLite
- ★ phone/physical conferences (developers, WLCG and SRM, GDB, Tier-1-centers, D-Grid)
- ★ schools (e.g. part of GridKa Grid School)

DESIGN



Support Details



- ★ support@dcache.org, direct phone calls
- ★ GGUS now linked with dCache ticket system
- ★ as part of the
“*Physics at the Terascale*”-project:
 - coordinated by DESY
 - goal: build a national dCache support infrastructure
 - starting Januar 2008
 - instituts contributing FTEs: Aachen, München, Karlsruhe, DESY (hopefully soon)

DESY



The Distributed dCache



★ NDGF distributed Tier-1

- running since January 2007
- code contributions to meet specific requirements
- challenges
 - ◆ limited bandwidth (dCache spans over WAN)
 - ◆ high latency
 - ◆ frequent network failures
 - ◆ distinct administrative domains

★ *An interesting model for (federated) Tier-2s?*

DESY

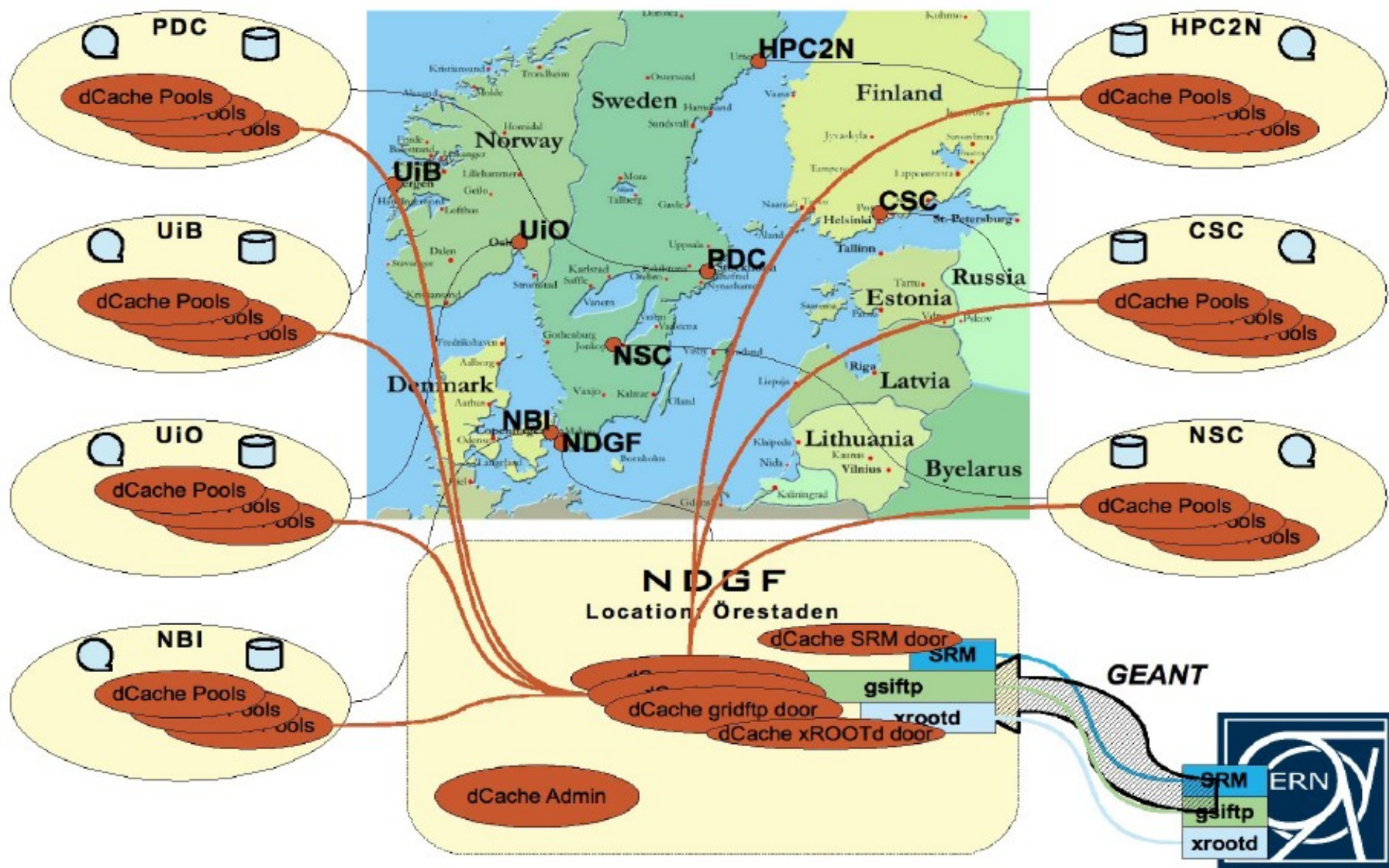


NDGF Tier-1 Storage: Outline



Radicke, Fuhrmann

4. HEP-CG Workshop
Siegen



copyright by
Michael Grønager, NDGF



dCache in Non-HEP communities?



- ★ dCache SE was presented to AstroGrid and C3Grid
 - but have their own ideas of storage, already developed solutions and different use cases
 - maybe interested in the “NDGF-approach” for the “D-Grid Sonderinvestitionsmittel”
- ★ weak points identified
 - Posix-like I/O is not Posix (actually a file system driver)
→ [solution: NFS 4.1](#)
 - HTTP(s) not really supported
 - security and metadata support might not be sufficient



Thanks for listening!



Contact:

www.dcache.org

Specific help for your installation: suport@dcache.org

User Forum:

user-forum@dcache.org

DESY