

dCache

Patrick Fuhrmann

Tigran Mkrtchyan

presented by
Peter van der Reest

Hepix Fall 2003
TRIUMF, Vancouver

Overview

dCache basics

Goals

Modes of operation

Attraction Model

Dataset Location Management

dCache Components (Overview)

Native dCache access method (dCap)

The GRID Storage Resource Manager (srm)

The Goal of the Srm Initiative

The Storage System Abstraction

An SRM initiated transfer example

Major SRM Topics

dCache : Goals

Unique access methods to highly distributed data repository

Hiding possible media transfers from clients (Tape <-> Disk)

Providing common access protocols (Kerberos, Grid FTP)

Fault tolerant regarding storage nodes (failover of disk storage)

Robot related

Reducing Robot activities

Optimizing access to high capacity (slow) tape systems.

dCache : Modes of Operation

Used as an **HSM frontend**, the dCache provides standard caching mechanisms to optimize tape accesses :

Transfer speed adaption

Tunable deferred HSM stores (space , time)

Automatic staging

Continuous garbage collection (no tresholds)

Fetch ahead (from Hsm) [*in preparation*]

dCache : Modes of Operation

- dCache Pools **without HSM backend** can hold :

Precious datasets

*Files are **never** automatically removed.
System can run out of disk space.*

Volatile datasets

*Unused files are automatically removed.
System won't run out of disk space.*

- The dCache can be operated in hybrid mode, running HSM and NON - HSM pools.

dCache : Modes of Operation

No HSM backend but central ReplicaManager

Stores precious files on cheap, non RAID disks

Ensures min, max replica count per file

Automatically adjusts min, max count on pool failures

dCache : The Attraction Model

File (resp. store/retrieve requests) are attracted by pools, based on :

Statically configured parameters, e.g. :

Client host IP or subnet numbers

HSM groups

Subdirectory trees

Dynamically taken parameters from live system, e.g. :

Pool CPU cost (number of active movers)

Pool Space costs (space left, age of datasets)

dCache : Dataset Location Management

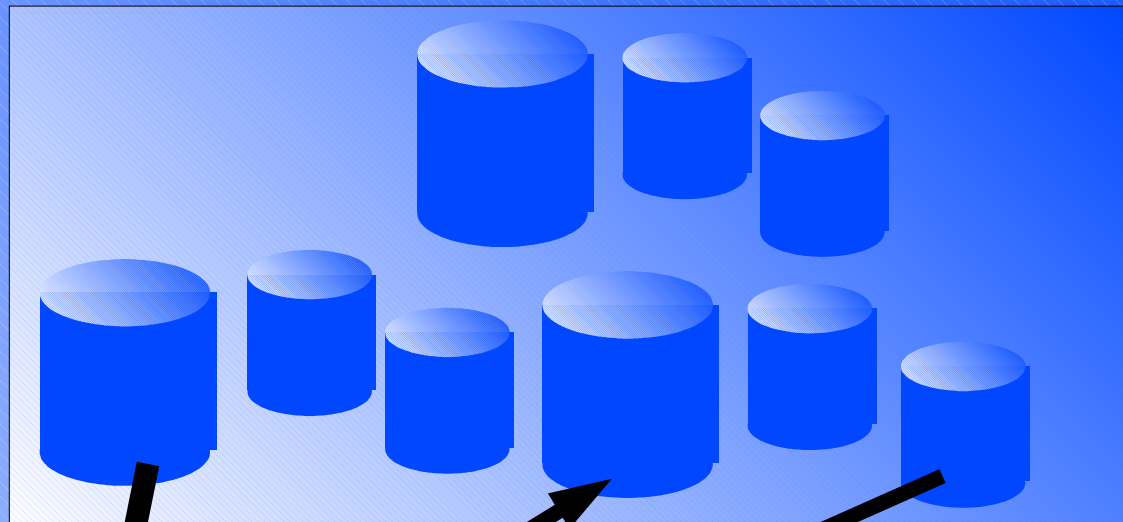
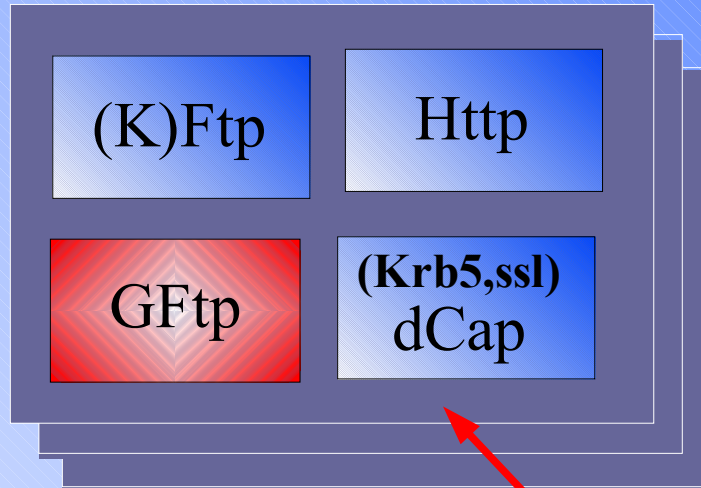
On increasing load of pool nodes, the cache creates dataset duplicates on moderately used nodes to smoothen hot spots.

Decreasing load marks dataset duplicates for removal in case space is running short.

Datasets can be defined 'sticky', independetly of its status, CACHED, DUPLICATED or ON TAPE ONLY.

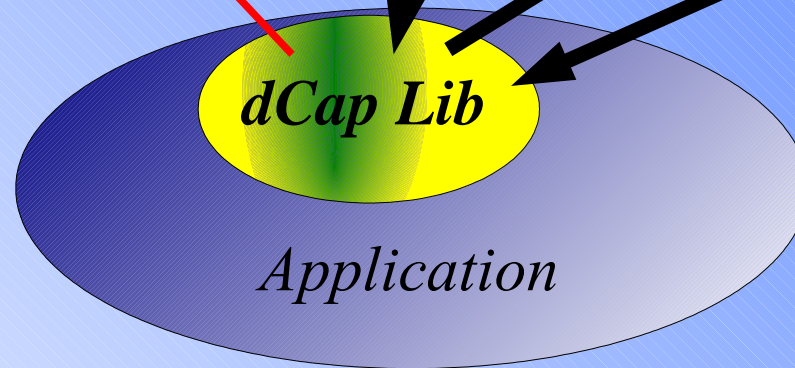
dCache Access Scheme

I/O Door Nodes



Control Line

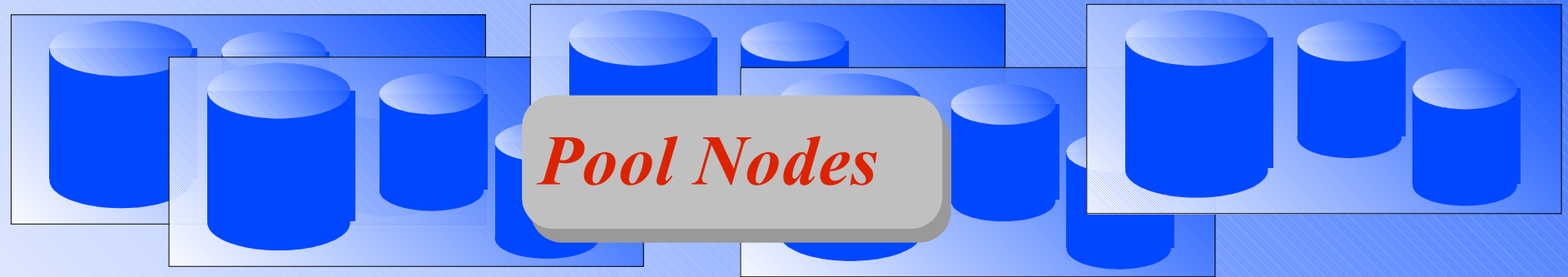
Data Lines



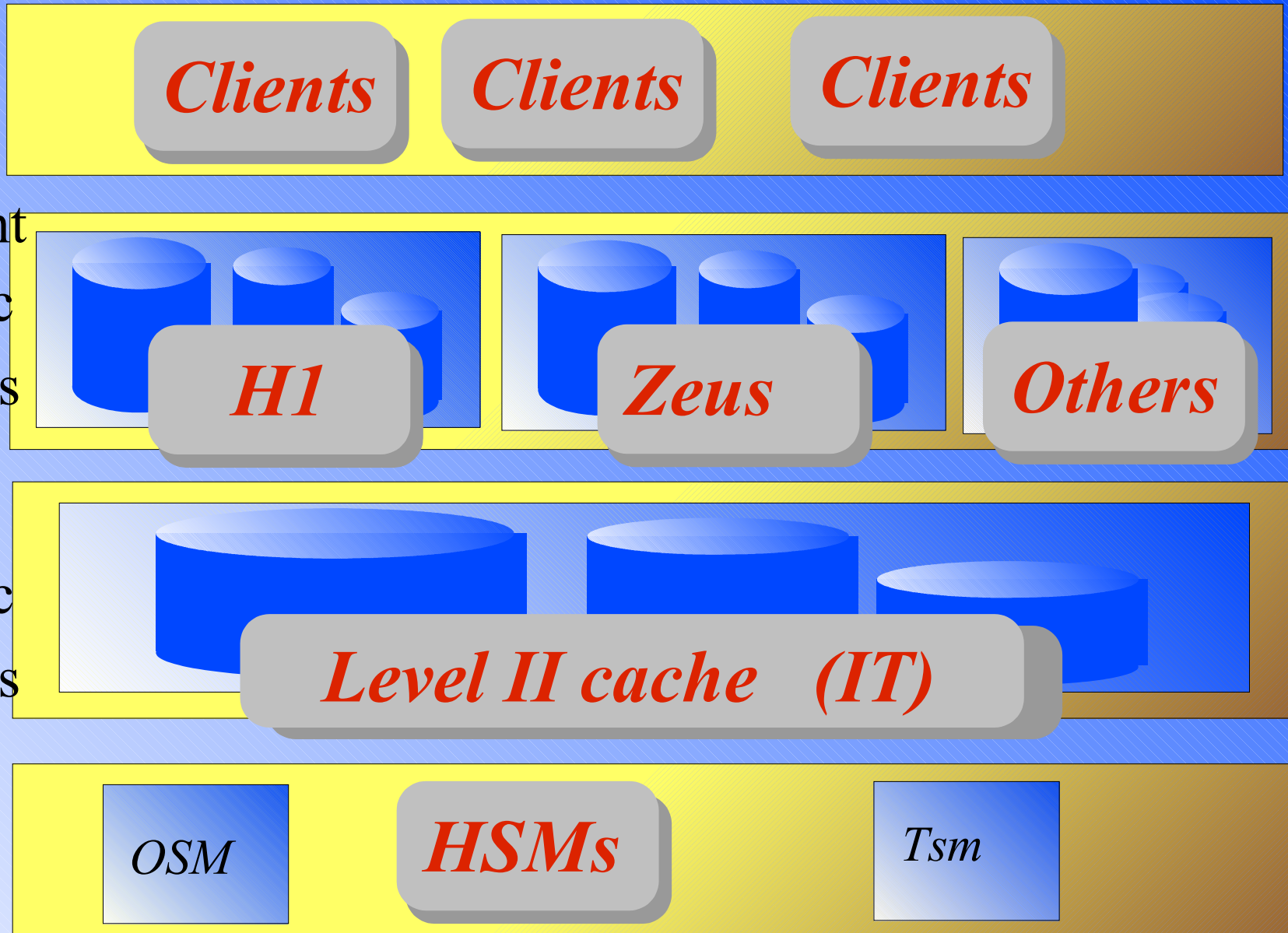
dCache Components

I/O Door Nodes

Admin Doors



d2Cache : stacked dCache



dCache : Native Access Method (dCap)

Beside supporting Ftp, Gftp, KeberosFtp and Http, dCache defines a native access protocol (dCap), allowing posix like file operations.

dCache provides a dCap c-language implementation

As shared object or preload library

For linux, solaris, irix OS and windows XP.

*Supporting automatic reconnect on network
or server problems*

Providing security tunnels for Kerberos and ssl.

Interfacing ROOT

dCache : DESY setup and statistics

450 TBytes in HSM storage (stk and adic robotics)

30 TBytes first level cache (owned by experiments)

20 TBytes second level cache (owned by IT)

8 TBytes write cache (owned by IT)

organized in 30 + 25 + 6 hosts

currently 4 files / second \sim 1.6 GBytes/sec sustained

50,000 – 160,000 files / day or 2 – 20 Tbytes / day

Moderate Fermi dCache users

D0 (Tevatron) -> dccp, SAM

Minos (Neutrino) -> GFtp from Soudan mine

MiniBoone (Neutrino) -> GFtp and dCache

Auger (Cosmic Rays) GFtp from Argentina and France

Grid Condor Project (High Throughput Computing)

NeST -> SRM -dCache -> Enstore

Grid KA (Karlsruhe) in preparation

Heavy Fermi dCache users

CMS – US Groups

Installations at Fermi , San Diego, Italy and CERN

CDF (Tevatron) Experiment

100 TBytes of disk space (w Storage backend, enstore)

Approaching 500 Tbytes of disk space end of next Year

20 TBytes of scratch pool space (w/o Storage backend)

Measured up to 50 Tbytes transferred per day

*The **S**Storage **R**Resource **M**anager Initiative*

In order to make site local storage resources like disk space, tertiary storage space and large quantities of HEP information globally available, e.g. in the GRID context, an initiative has been setup by JLAB, FermiLab, LBNL and CERN, defining some kind of abstract storage system, covering methods for :

Storing and retrieving datasets

Obtaining status information about datasets

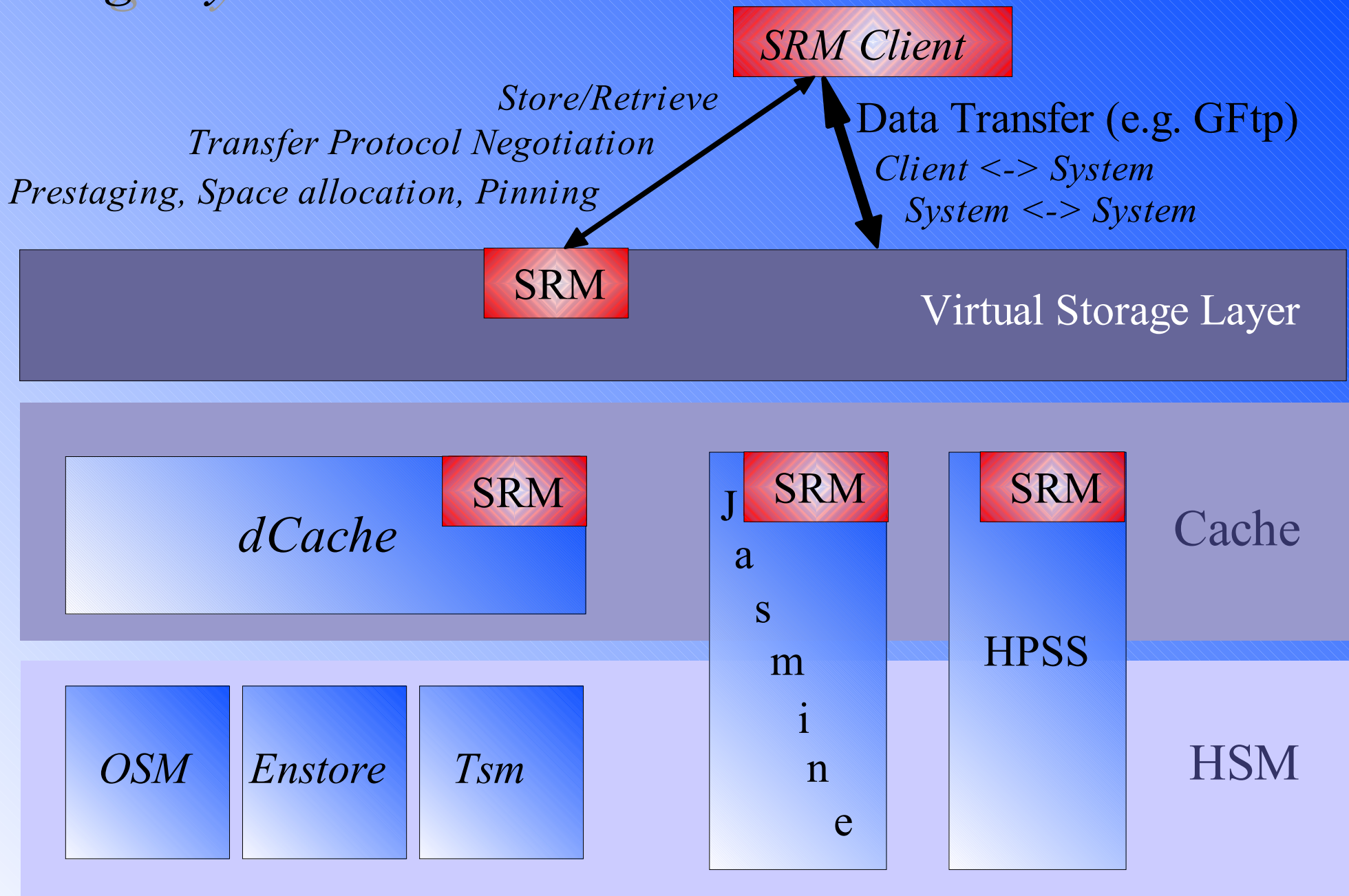
Pinning datasets (guarantee of availability)

Negotiating data transfer protocols

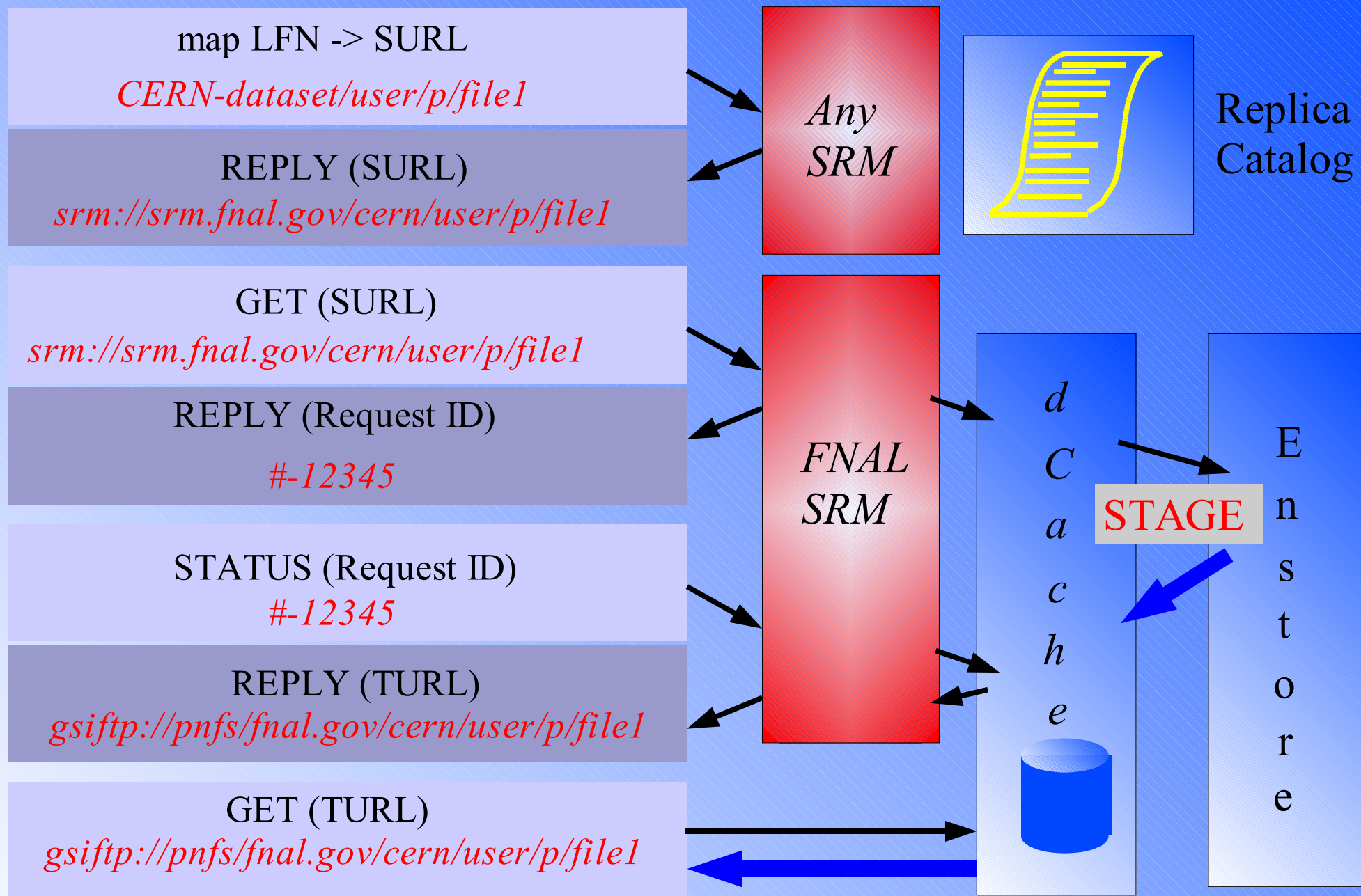
Defining dataset lifetimes

HEP sites, intending to access remote storage resources, are assumed to implement the SRM protocol into their local storage system. For the dCache, FermiLab took over this task.

Storage System Abstraction



SRM Initiated Transfer



SRM Topics (Srm 2.1)

Storing and retrieving datasets.

Transfer Protocol Negotiation.

(includes direct I/O methods rfio, dCap globus-xio)

Obtaining dataset status information.

Pinning datasets (make DS available)

File Space Allocation / Reservation

Dataset / space lifetime definitions (*volatile, durable, perm.*)

Srm to Srm third party transfers.

Directory support (*mkdir / rmdir*)

Security (*srm will support gsi over http*)

For Details

dCache

www.dCache.org

SRM

http://sdm.lbl.gov/srm-wg/