



dCache

*LCG Storage Element and
enhanced use cases*

*Patrick Fuhrmann, DESY
for the dCache Team*

dCache is a joint effort between the Deutsches
Elektronen Synchrotron (DESY)
and the Fermi National Laboratory (FNAL)



The Team

Jon Bakken, FNAL

Rob Kennedy, FNAL

Alex Kulyavtsev, FNAL

Timur Perelmutov, FNAL

Don Petravick, FNAL

Vladimir Podstavkov, FNAL

Michael Ernst, DESY

Patrick Fuhrmann, DESY

Martin Gasthuber, DESY

Tigran Mkrtchyan, DESY

Mathias de Riese, DESY

Sven Sternberger, DESY

Acknowledgments

CERN : Jean-Philipp Baud, Maarten Litmaath, Andreas Unterkircher









Karlsruhe (gridKa) : Doris Ressimann

BNL : Scott O'Hare, Offer Rind

Vanderbilt : Matthew T. Calef








-  *Single 'rooted' file system name space tree*
-  *Data may be distributed among a huge amount of disk servers.*
-  *Supports multiple internal and external copies of a single file*
-  *Automatic load balancing by cost metric and inter pool transfers.*
-  *Data removed only if space is needed*
-  *Distributed Access Points (Doors)*
-  *Using standard 'ssh' protocol for administration interface.*
-  *First version of graphical interface available for administration*









-  *Fine grained configuration of pool attraction scheme*
-  *Pool to pool transfers on config. or data access hot spot detection*
-  *CRC checksum calculation and comparison (partially implemented yet)*
-  *Pluggable door / mover pairs*
-  *Automatic HSM migration and restore*
-  *Convenient HSM connectivity (done for enstore,osm,tsm, prelim. for Hpss)*









-  *DESY dCap lib incorporates with CERN's GFAAL library*
-  *gsiFtp support*
-  *SRM version ~ 1 (1.7) in production*
 - Improved (persistent) version in beta testing*
 - see Timur Perelmutovs, FNAL talk*
-  *limited GRIS functionality (using workaround)*





-  Controls number of copies for each dCache dataset
-  Makes sure $n < \text{copies} < m$
-  Adjusts replica count on pool failures
-  Adjusts replica count on scheduled pool maintenance

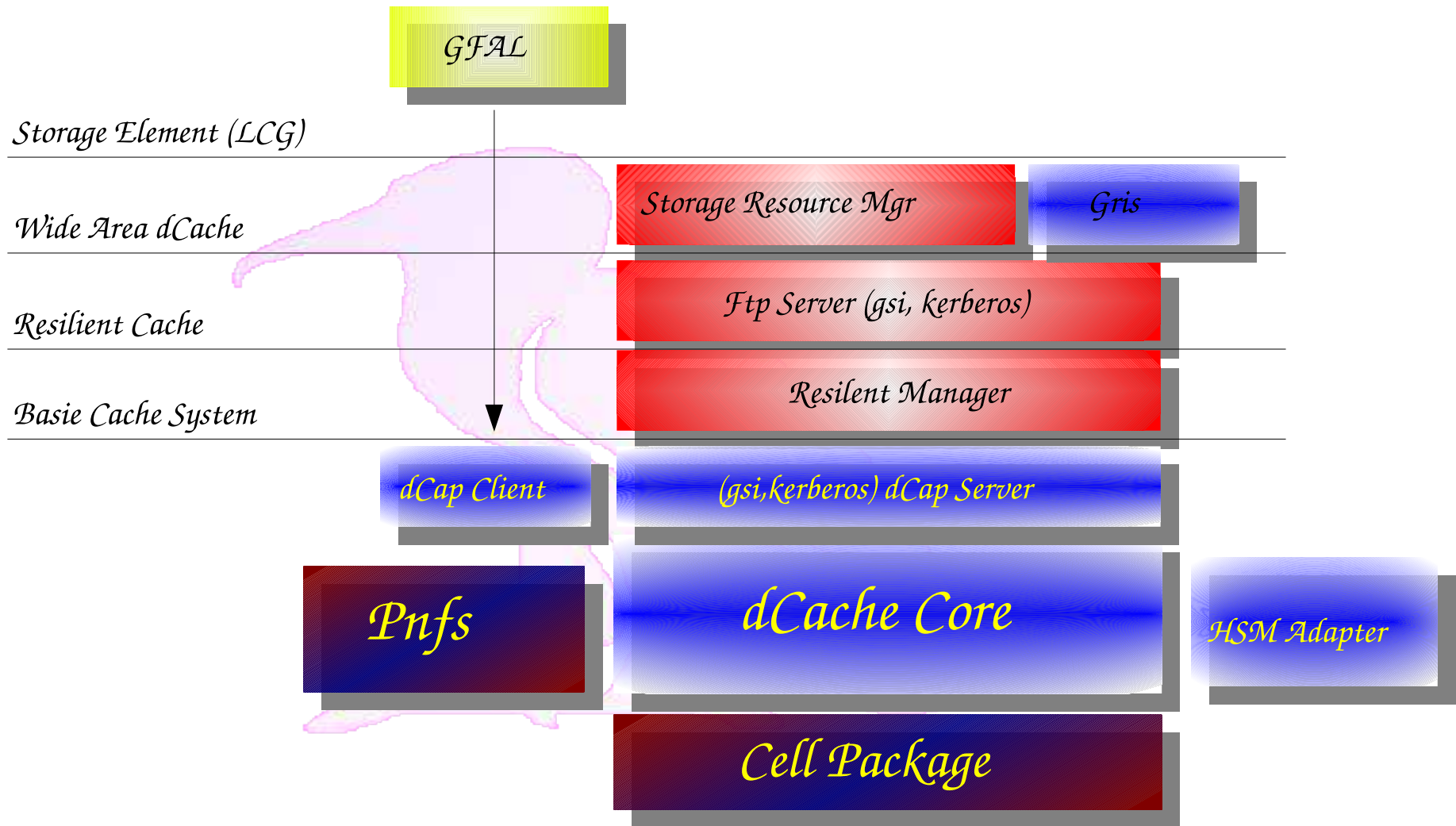
Not yet in official distribution, but in production



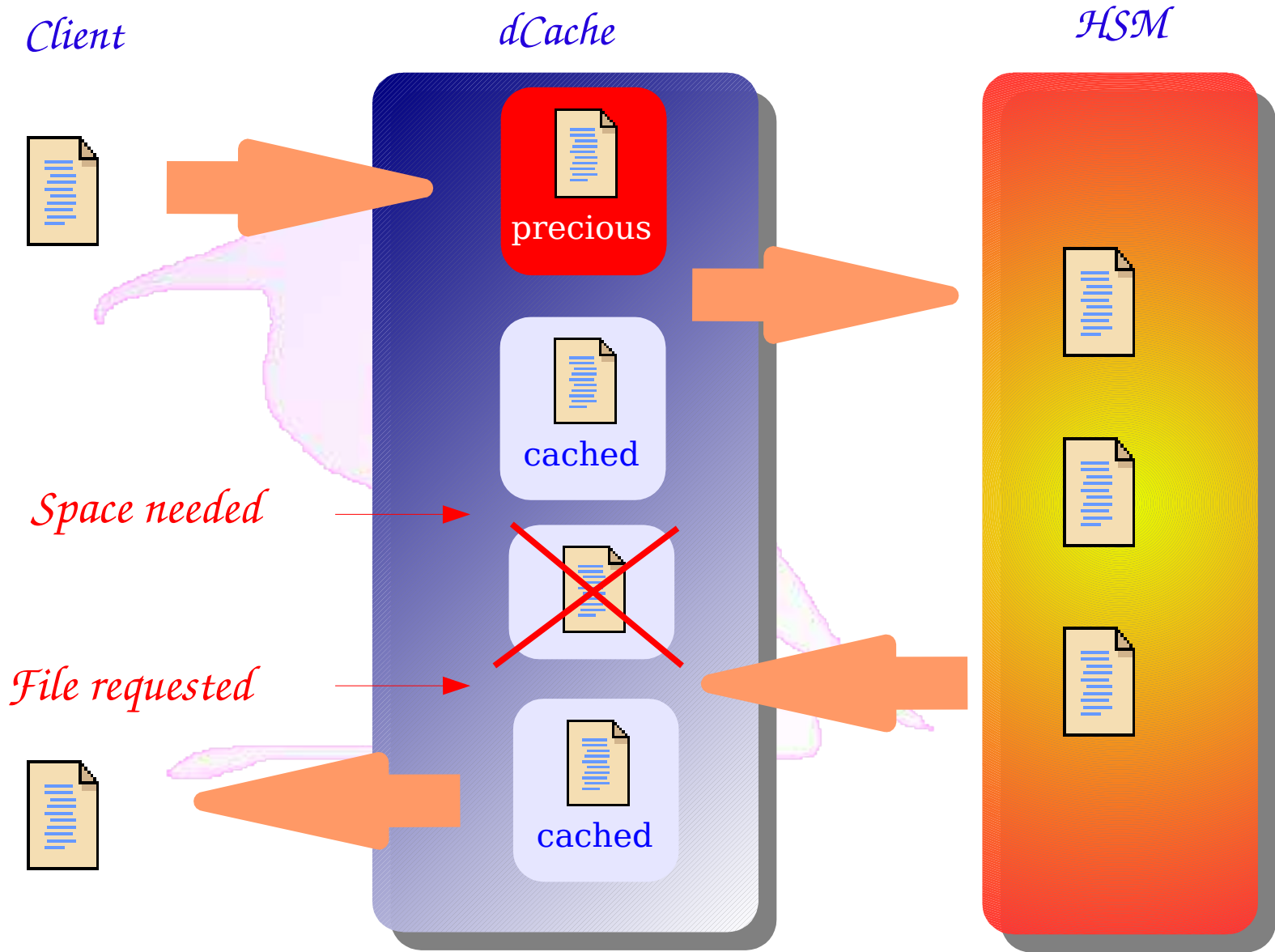


-  *implements I/O and name space operations including 'readdir'*
-  *available as standard shared object and preload library*
`ls -l dcap://dcachedoor.desy.de/user/patrick`
-  *positive tested for Linux, Solaris, Irix (partially for XP)*
-  *automatic reconnect on server door and pool failures*
-  *supports read ahead buffering and deferred write*
-  *supports ssl, kerberos and gsi security mechanisms*
-  *Thread safe*
-  *Interfaced by ROOT ®*











Precious data is separately collected per storage class

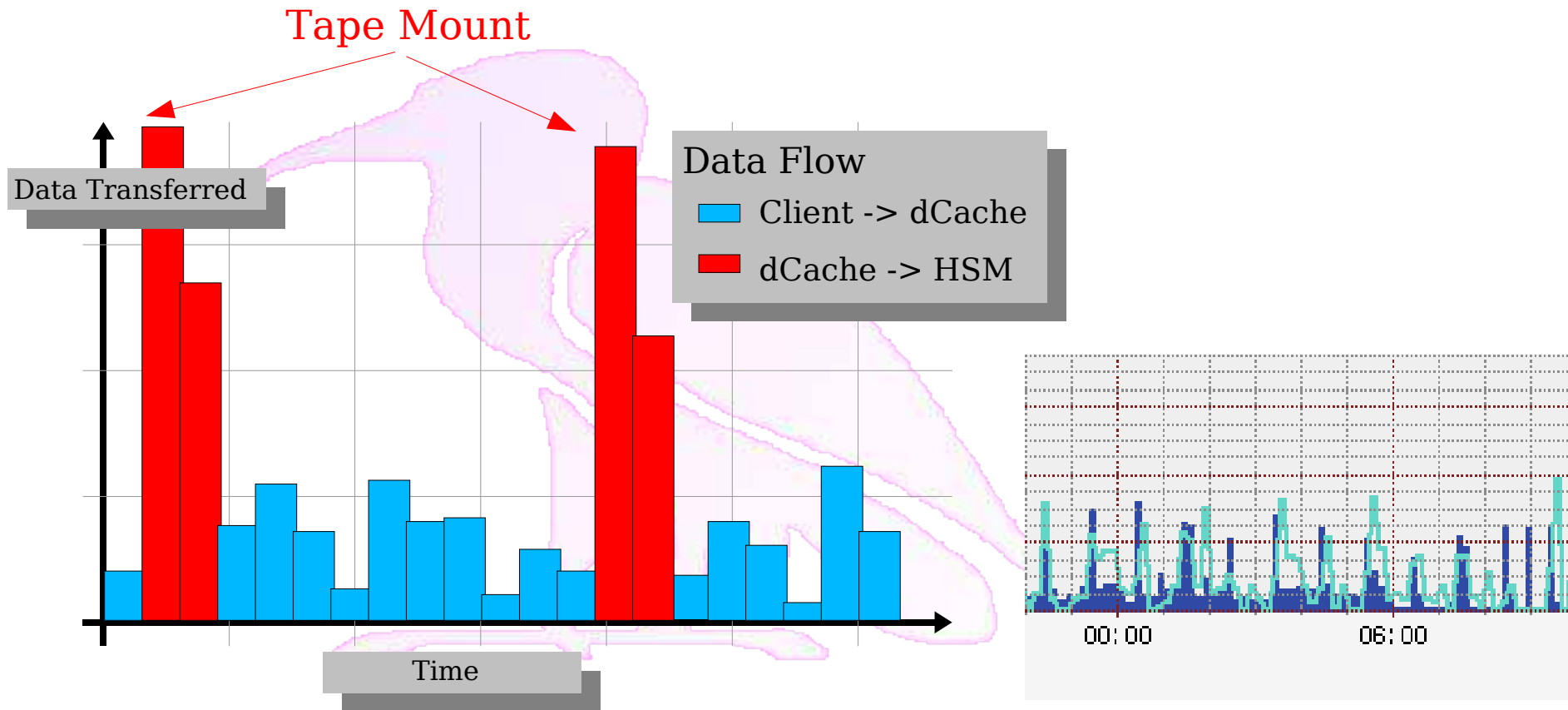
Each 'storage class queue' has individual parameters, steering the HSM flush operation.

- Maximum time, a file is allowed to be 'precious' per 'storage class'.*
- Maximum number of precious bytes per 'storage class'*
- Maximum number of precious files per 'storage class'*

Maximum number of simultaneous 'HSM flush' operations can be configured

Multiple HSMs instances and HSM classes are supported simultaneously





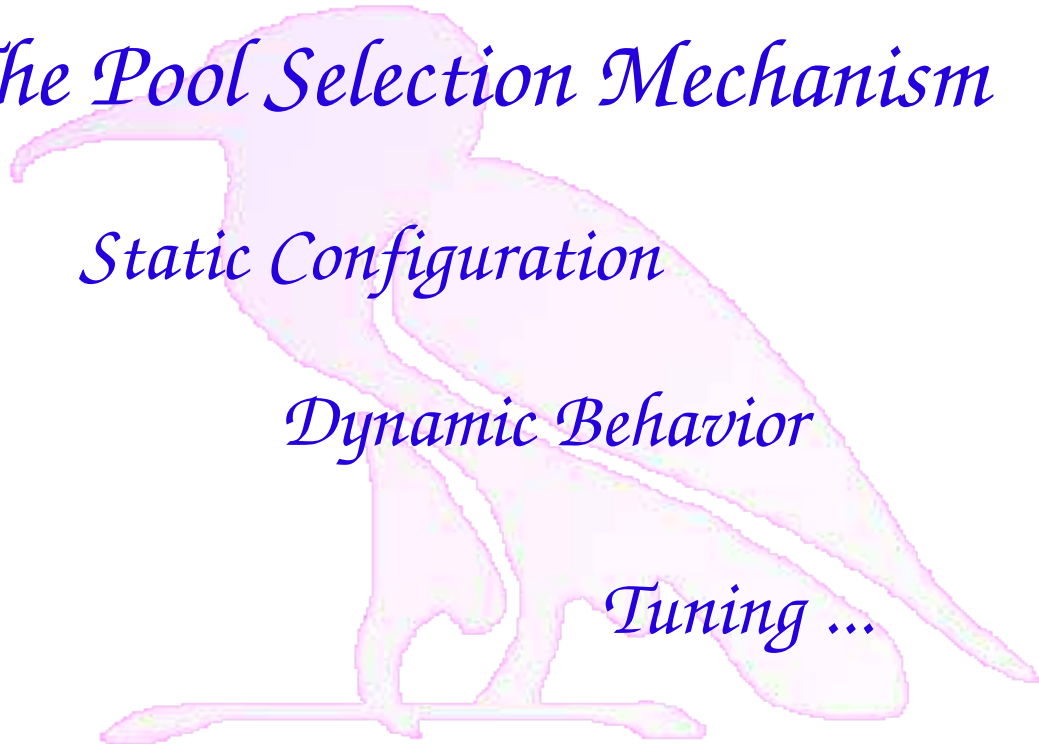


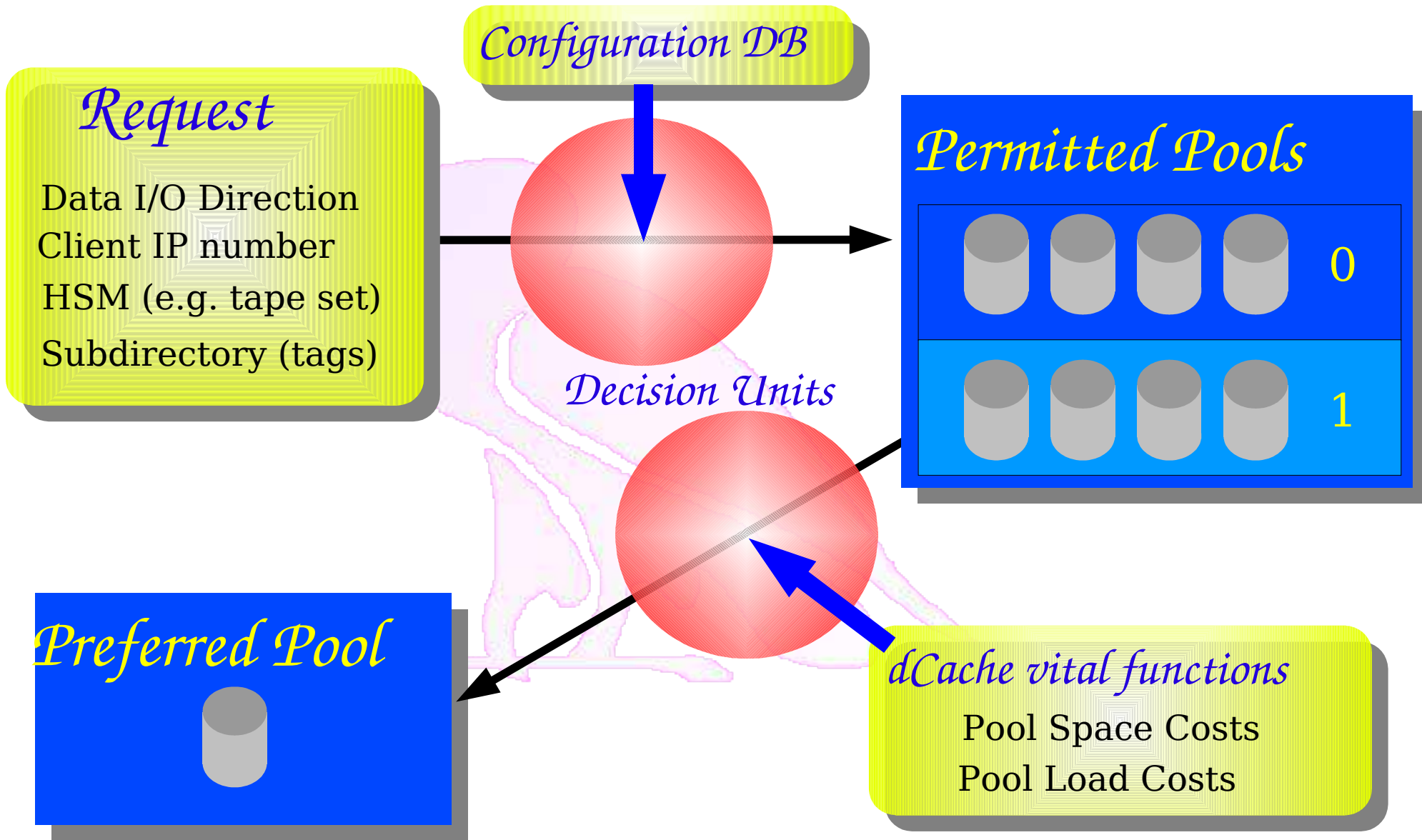
The Pool Selection Mechanism

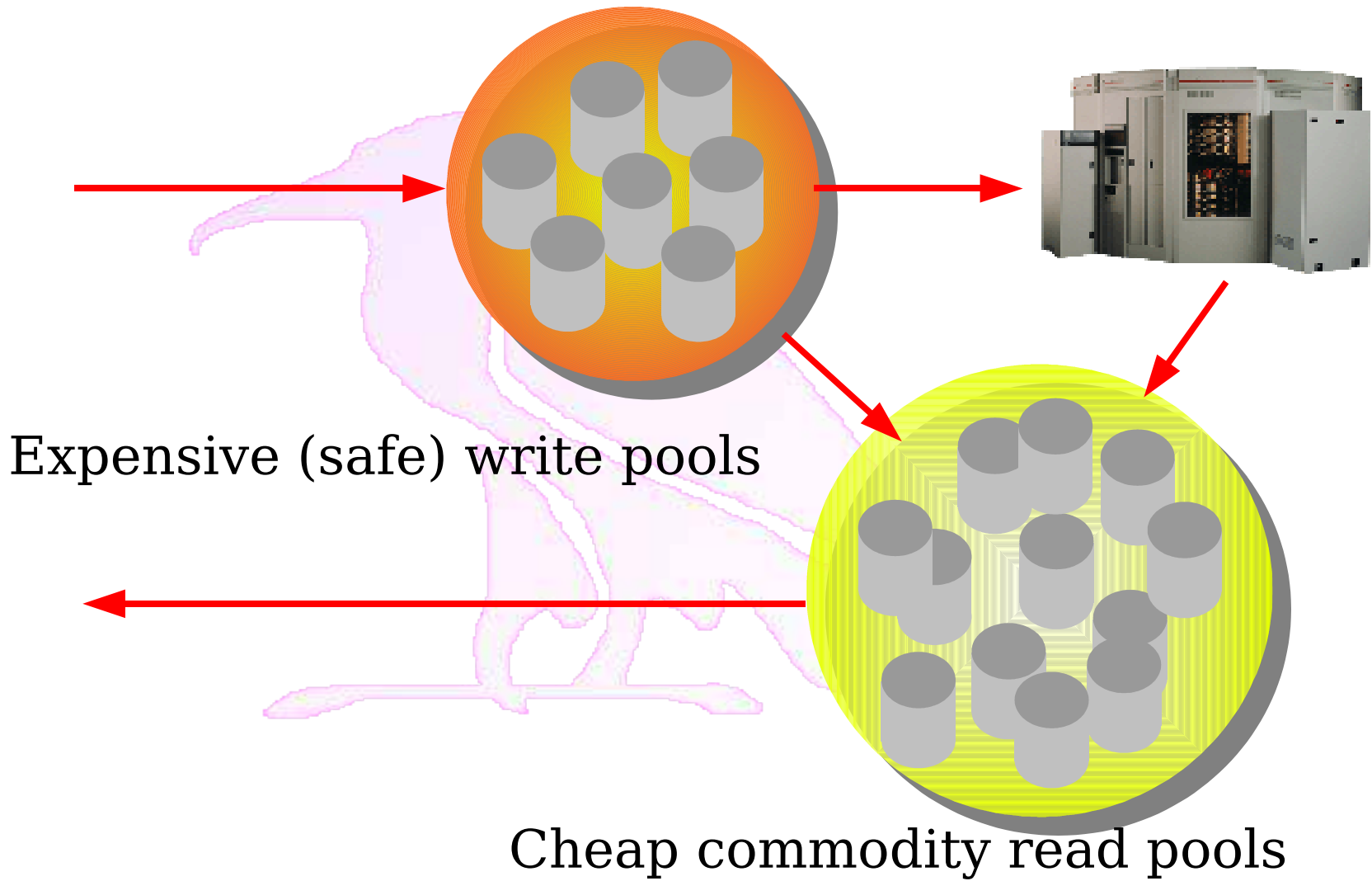
Static Configuration

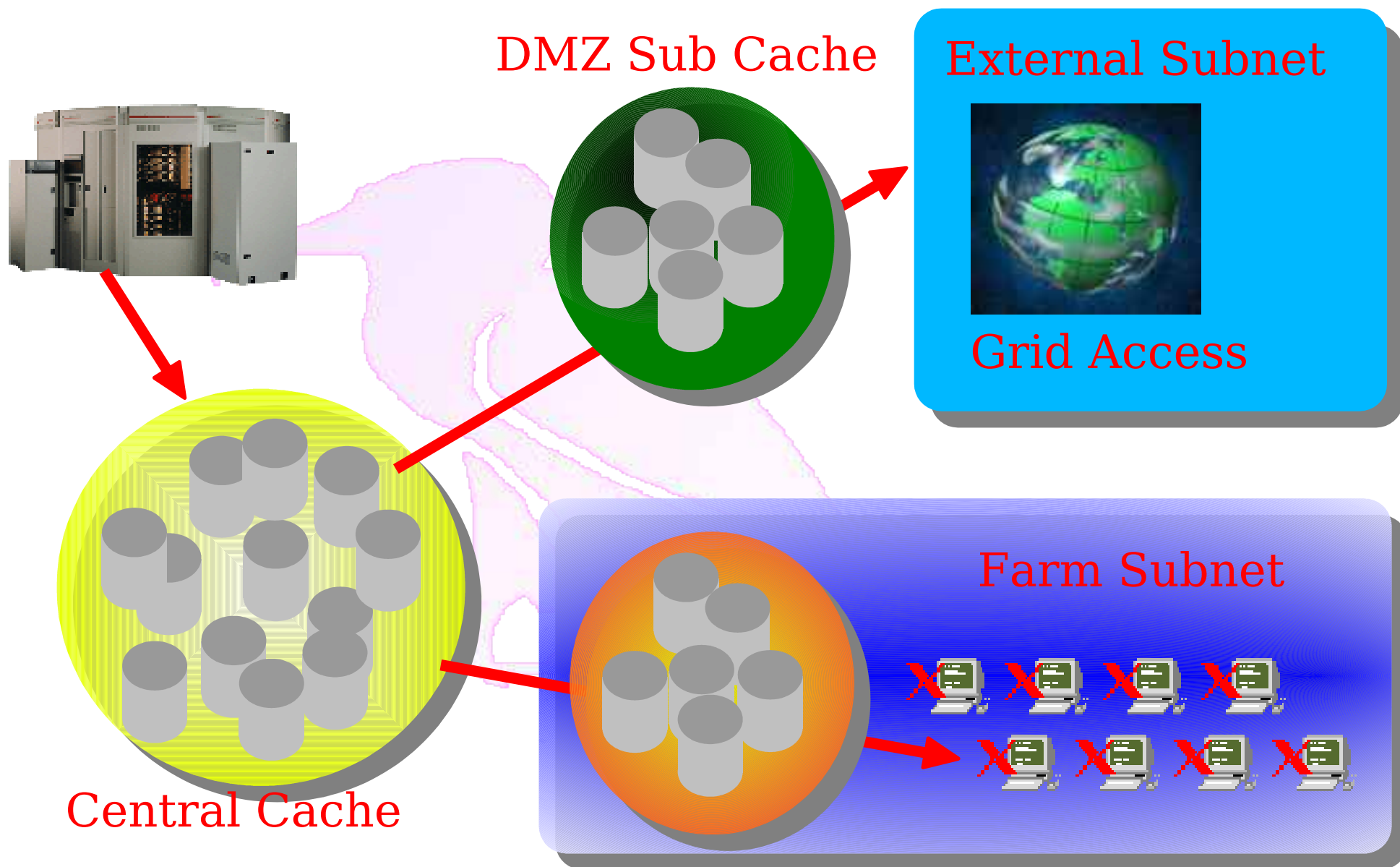
Dynamic Behavior

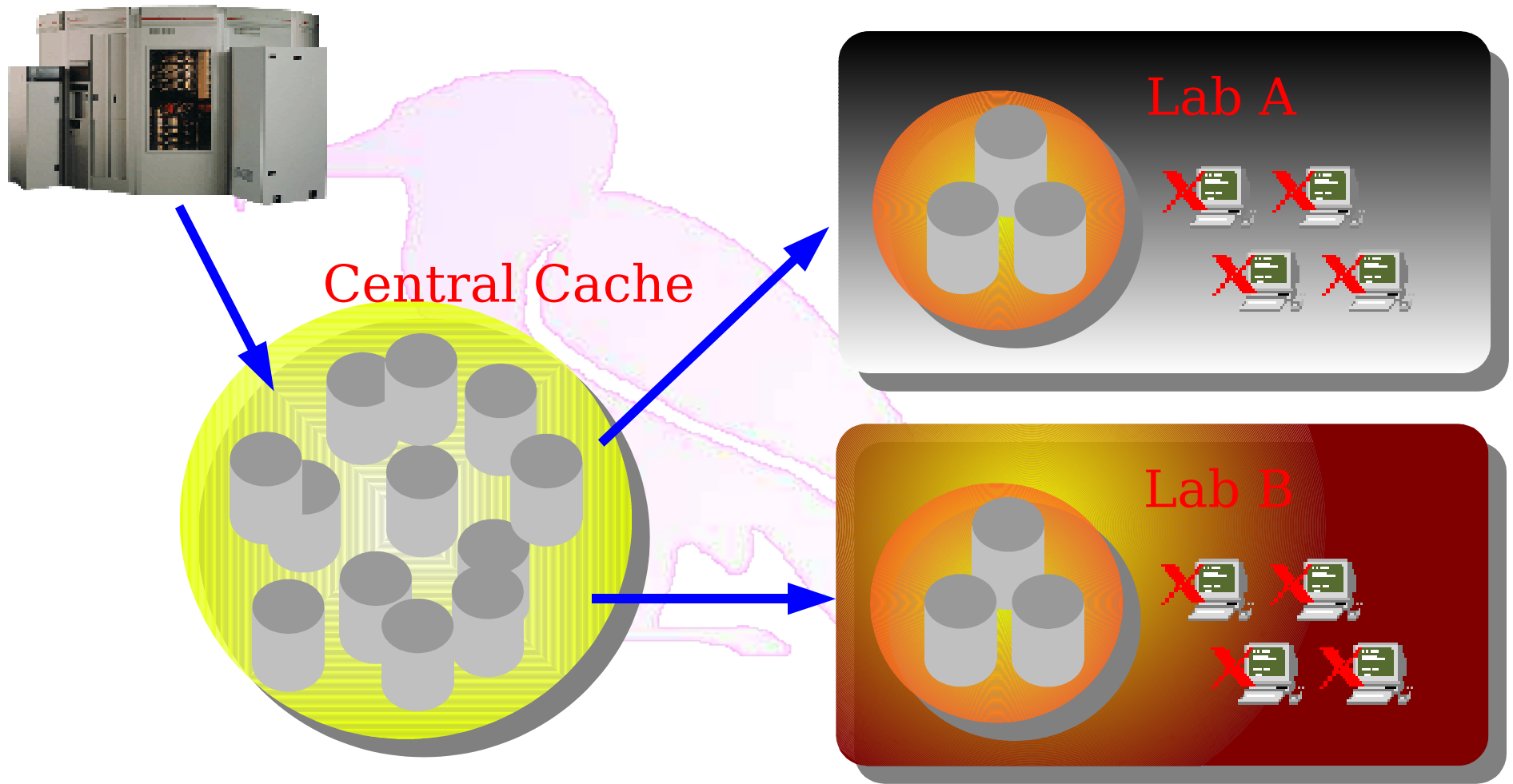
Tuning ...

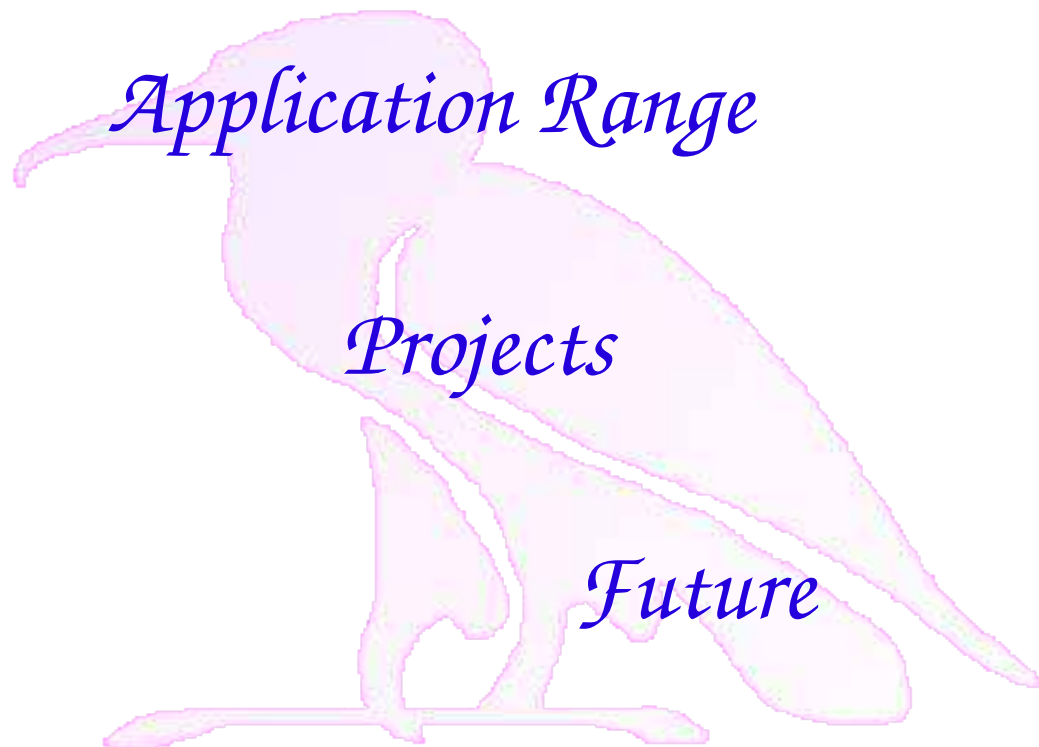














μ - Cache

≤ 10 TBytes
 ≤ 3 pool nodes
ZERO Service
no HSM

macro - Cache

> 150 TBytes
 > 80 pool nodes
full HSM support





μ - Cache

How to achieve a 'zero service' micro Cache System ?
Possible partners and funding from D-Grid initiative

macro - Cache

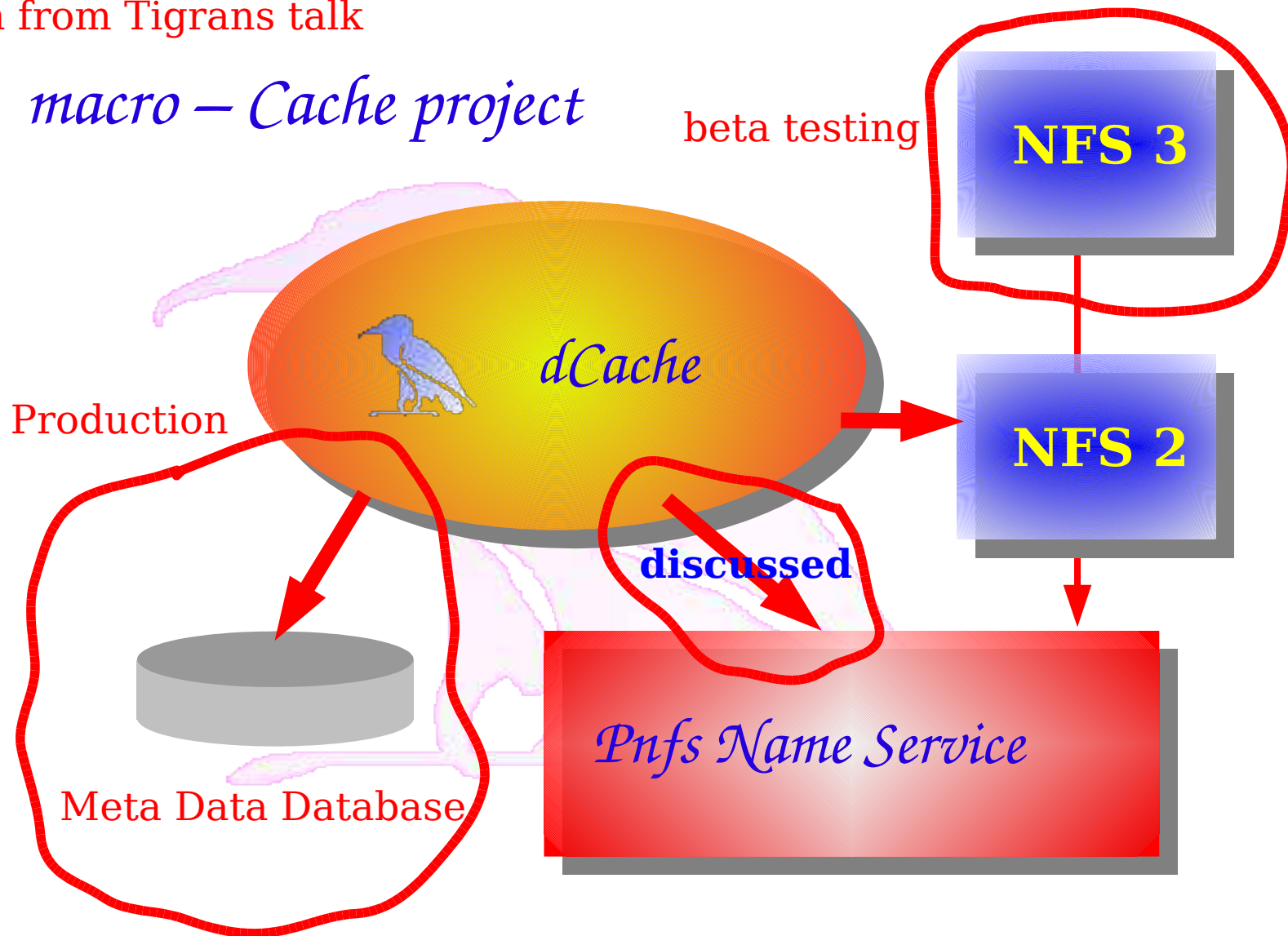
Find 'non scalable components' in dCache.
First candidate : name service (pnfs)





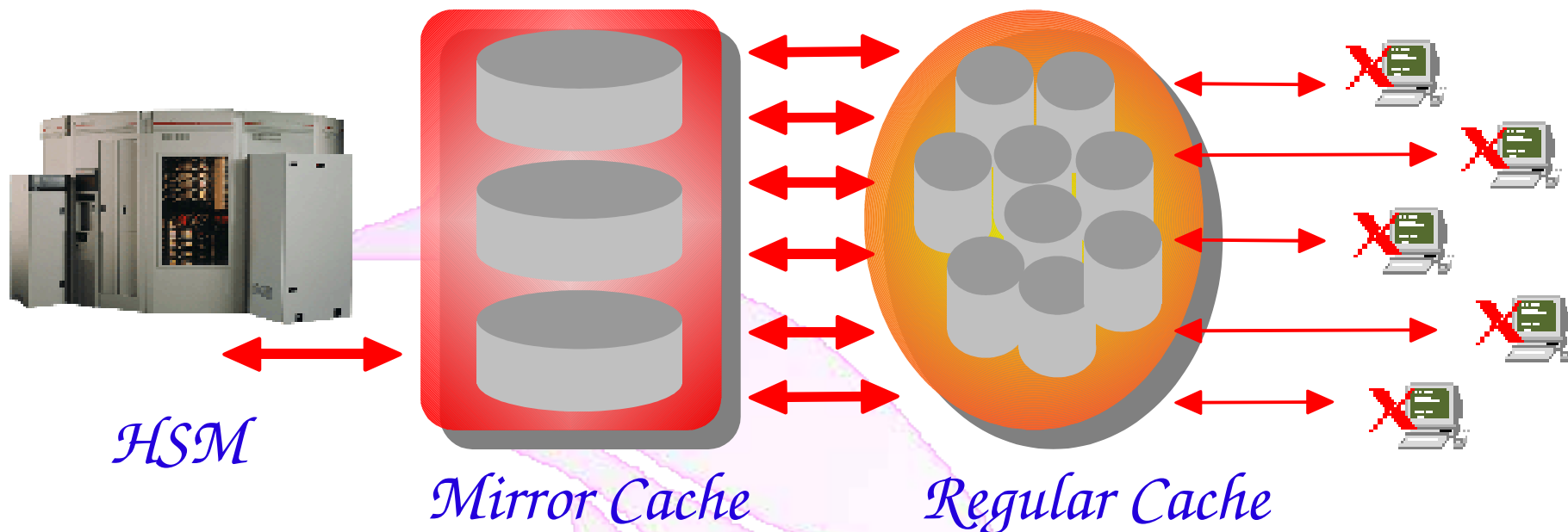
Stolen from Tigrans talk

macro – Cache project





Project managed by Martin Gasthuber, DESY



nearly all *HSM* data on *Mirror Cache*

Mirror Cache has highest possible data density (lowest dollars/Tbyte)

Controlled number of high speed streams between
Mirror Cache and *Regular Cache*

Mirror Cache behaves like HSM (except for mount/dismount delays)

Mirror Cache disks switched OFF if not accessed

HSM to *Mirror Cache* transfers only after disk replacement





Documentation





Packages available at www.dcache.org

Support requests to support@dcache.org

*LCG support and evaluation by
Ian Birds Deployment Group*

users group mailing list prepared



dCache

End of official presentation