

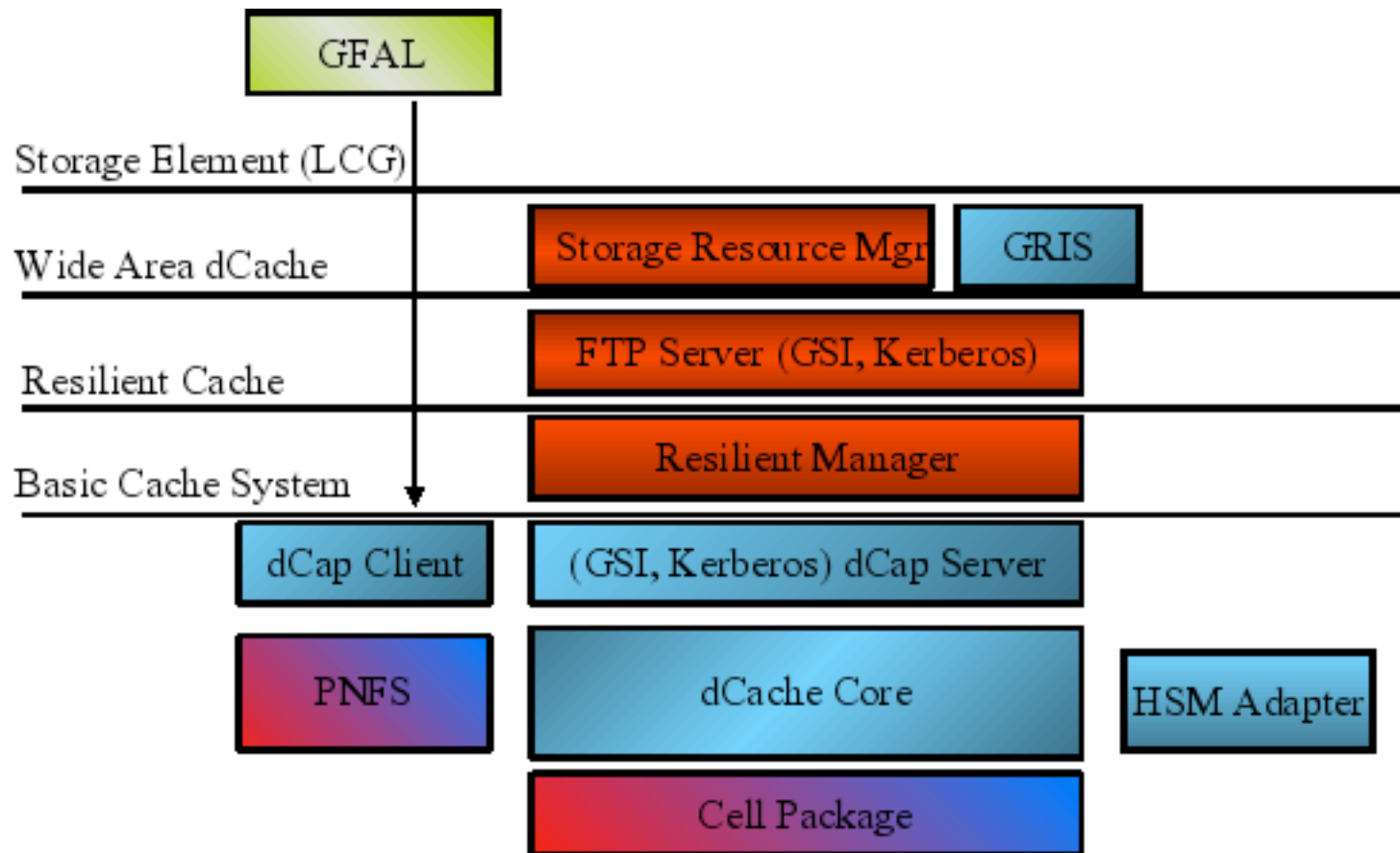
dCache Storage Systems in a Global Network

Matt Crawford
Fermilab

Why create dCache?

- Federate file systems to hide/recover from individual disk or host failures.
- Integrate HSM (tape or future alternative).
- Schedule and distribute load to scale the service and protect it from overload.

dCache structure



(concept by P. Fuhrmann)

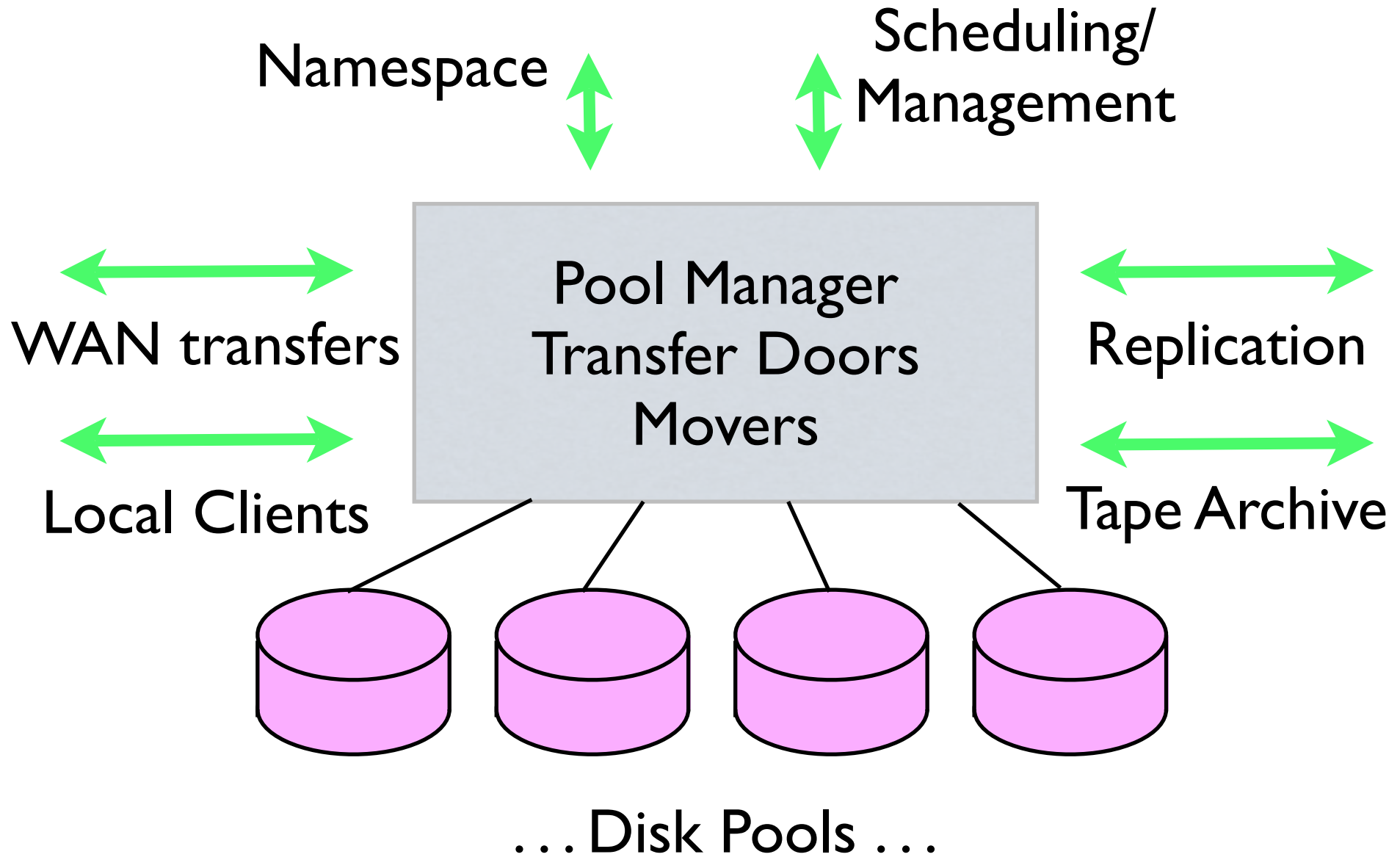
dCache core functions

- Combines hundreds of disk pools into a single huge name/file space.
- Supports multiple internal and external copies of a single file system entry.
- Reserves space, pins files in use, limits loads.
- Automatically replicates hot files for load spreading; migrates precious files to permanent storage.
- Allows multiple access & transfer protocols.
- Authn/Authz through Kerberos & GSI.

dCache extra functions

- Resilient dCache (at Tier 2, for example)
 - Ensure that N copies of each file exist in different pools, where $min \leq N \leq max$, even if nodes go down.
- HSM support
 - Group files onto tapes according to rules.
 - Retrieve files from tape as needed, without human intervention.

A storage node



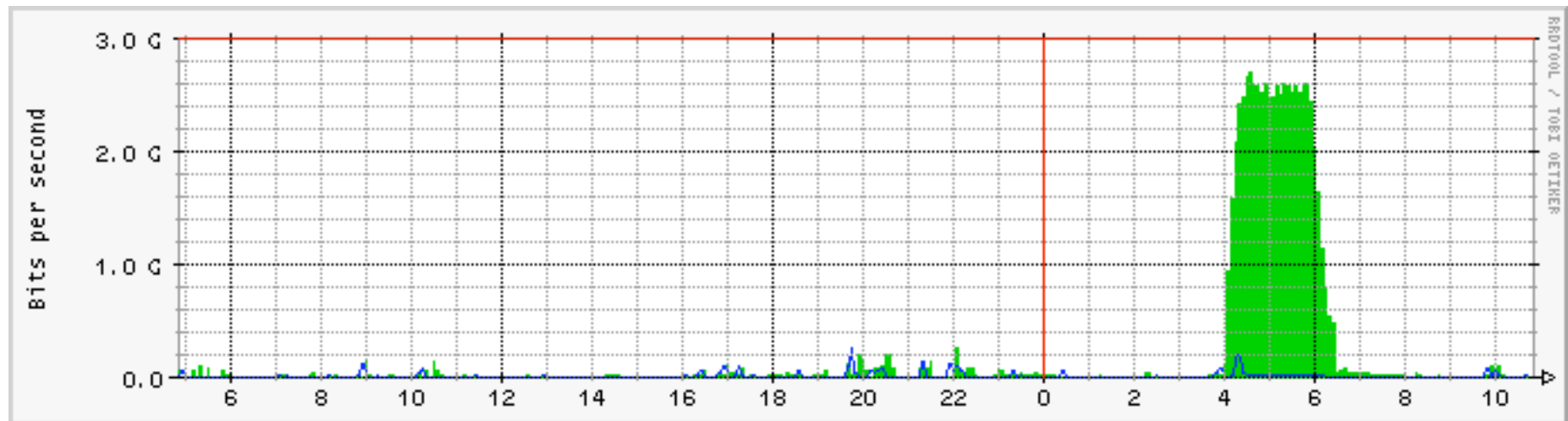
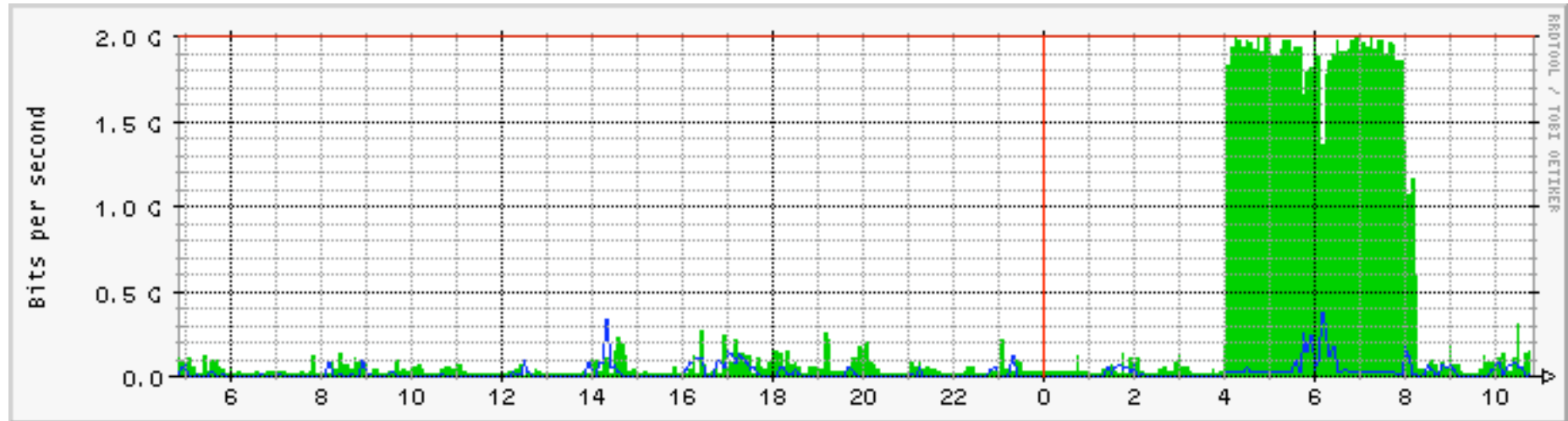
Typical node (CMS TI)

- 4 pools @ 2TB each, 95% filled
- 2 x 1 GE, bonded (peak 1.99+ Gb/s)
- Transfer limits:
 - 20 WAN
 - 8 to/from tape
 - 80 p2p migration or replication
 - 2400 local (POSIX) clients
- (730 nodes, 1261 pools, 551 TB in cluster)

SRM layer

- Standardized protocol – multiple implementations – hides storage system.
- Storage Resource Manager functions
 - Security checks; uses delegated credentials
 - Prepares transfers; negotiates transport
 - dcap, rfio, ftp, gsiftp, http
 - Retries failed transfers

This morning's traffic



<https://www-dcn.fnal.gov/~netadmin/b/cms/bondview.cgi>