

Summary of recommendations/proposal regarding SRM v2.2

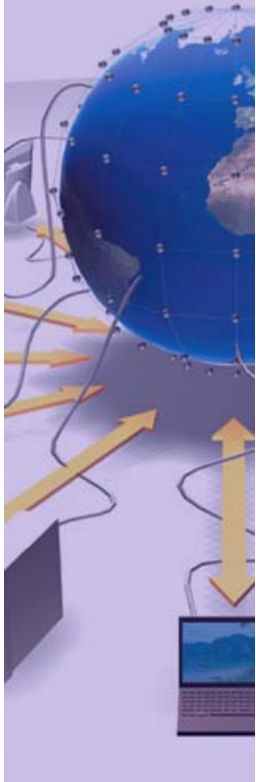
Flavia Donno

CERN

WLCG Collaboration Workshop (T0/T1/T2)

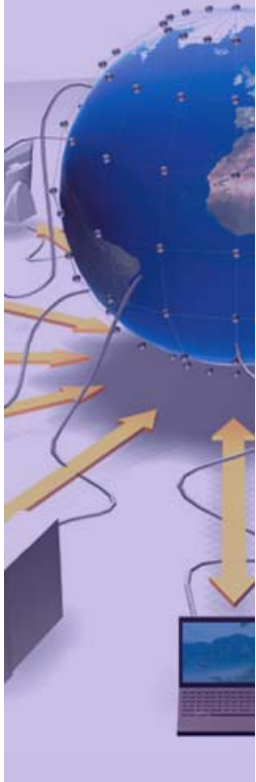
CERN, 21-25 April 2008

- Current issues
- Storage Baseline Versions for CCRC08 in May
- Report from Storage Solution Working Group

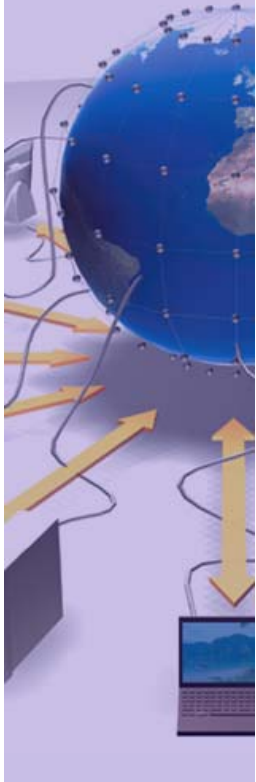


- **FTS error: Failed DESTINATION error during PREPARATION phase: [FILE_EXISTS]**
- **Problem observed by CMS (GGUS/Savannah tickets open by CMS, ATLAS, LHCb, ALICE)**
- **It is due to original file creation srmPrepareToPut request failing and subsequent srmAbort/srmRm requests failing as well. Next retry fails because the file was not properly cleaned up.**
- **Specific phoneconf with developers. It was agreed:**
 - **dCache will optimize internally the handling of abort and rm operations. S2 specific tests will be created to test the optimization.**
 - **All implementations will return SRM_FILE_BUSY at file level if a second srmPrepareToPut with overwrite flag enabled is issued before an srmPutDone is issued on the first request for the same SURL.**
 - **When FTS is requested to perform a put operation with the overwrite flag enabled, if srmPrepareToPut fails, FTS will remove the SURL and then retry the operation. All this has to be carefully tested to make sure no more problems are introduced.**

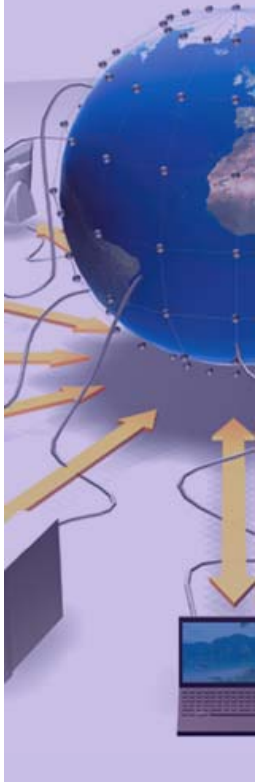
- CMS request to “ChangeSpaceForFiles” from T1D1 to T1D0
 - Allowed transition but not supported through SRM at the moment.
 - **dCache has produced documentation (assisting PIC). Administrative support for this transition is available in dCache 1.8.0-15**
 - **Other implementation provide support through an administrative interface**



- ATLAS reports poor performance for file deletion operations for all implementations but one (StoRM). The goal is to be able to delete file at a rate of 1Hz centrally via SRM
 - **dCache has provided an improved handling of the srmRm request in 1.8.0-14.**
 - **S. De Witt – CASTOR: “I am currently working on a more performing version of srmRm to support bulk deletion. This may be available in time for the May CCRC”**



- ATLAS reports problems staging files from tape with both CASTOR and dCache.
 - **dCache 1.8.0-14 has been installed at FZK and first test performed. Please check the presentation from Birger.**
 - **The problem reported is under investigation also by CASTOR developers.**



- **CASTOR** : SRM v 1.3-20, backend $\geq 2.1.6$
 - Only the ATLAS instance at CERN will be at 2.1.7
 - Removed a couple of possible database deadlock scenarios
 - srmLs return structure now conforms to that of other implementations
 - Various minor database fixes
 - Fix for leaking sockets when srmCopy attempted
 - Added correct user mappings when executing PutDone
 - Various corrections to log information level
- Better bulk deletion (?)

dCache : 1.8.0-14 highly recommended since it improves bulk deletion + staging.
1.8.0-15 available on Wednesday recommended for CCRC08 in May.

Beside a list of bugs:

<http://trac.dcache.org/trac.cgi/wiki/manuals/CodeChangeLogs>

there are some major issues solved :

- a) T1D1 can be set to T1D0 if the file is NOT in a token space.
(This is not done through the SRM)
- b) There is a new PinManager available.(improved stability)
- c) Space Tokens can be specified in a directory, to become the token used for writing if no other information is provided.
- d) We provide to the HSM script :
 - i) Directory/file
 - ii) Space Token
 - iii) Space Token Description (new)

DPM : $\geq 1.6.7-4$ is being certified and will probably be the release available for CCRC08 in May.

Issues fixed for 1.6.7-4:

- *Pool free space correctly updated after filesystem drain and removal*
- *SRM2.2 srmMkdir will now create directories that are group writable (if the default ACL of the parent gives that permission)*

Known issues or points:

- *No srmCopy is available*
- *Only round robin selection of filesystems within a pool*
- *No transfer stream limit per node*

StoRM : Release \geq 1.3.20 available since the end of March 2008

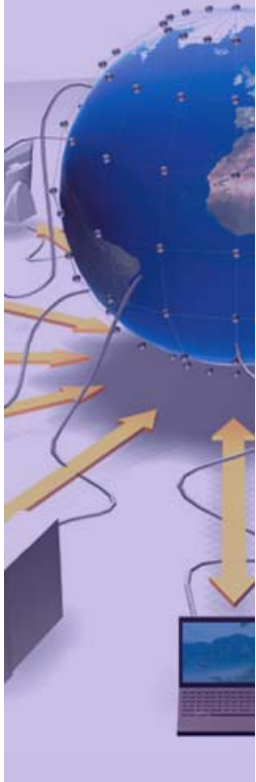
The list of fixes/features can be found here:

http://storm.forge.cnaf.infn.it/documentation/storm_release_plan

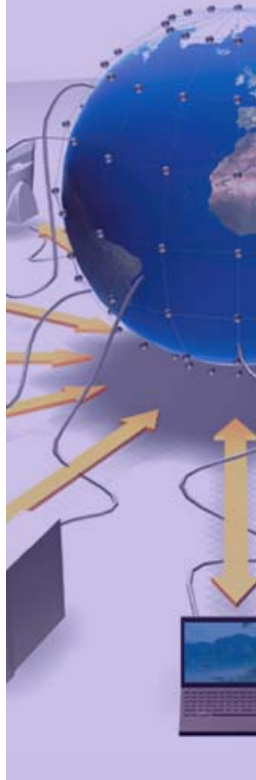
- Upgraded information providers, both static and dynamic
- Fix on the file size returned by srmPrepareToGet/srmStatusOfPtG for file > 2GB.
- New ROOT protocol support.
- New "default ACLs" feature as required by the CMS use case. It allows to define a default ACL for Storage Area that StoRM will enforce on each new file created.
- New srmGetSpaceMetaData bound with quota information.
- Improved support for Tape1Disk1 Storage Class with GPFS 3.2 and TSM

- A first proposal for an Addendum to the WLCG SRM v2.2 Usage Agreement was circulated on Wednesday 9 April 2008
- **Priorities** on the features to be implemented agreed:
 - Protecting spaces from (mis-)usage by generic users
 - Selecting spaces on Get operations
 - Proper implementation of GetSpaceMetaData
 - Selecting tape sets
- The goal is to finalize the document with the experiments possibly in the week 21-25 April and pass the proposal to the WLCG MB for approval.
- Then the implementation planning phase will start.

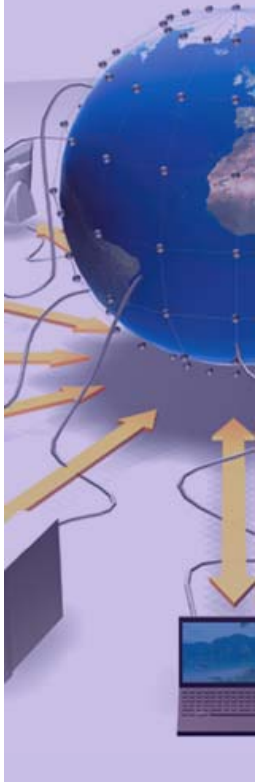
- Only technical agreement. No commitment.
 - **Feasibility has to be discussed at WLCG MB.**
- Discussed by the storage developers only
 - No developers of SRM clients have been really involved at this point.
- No WSDL modifications will be introduced at this time



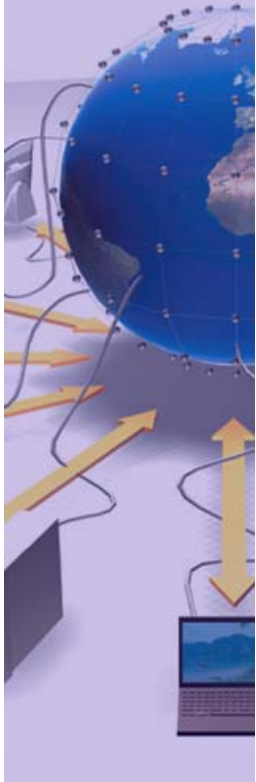
- An SRM space is a logical view of an **online** physical space allocation that is reserved for read/write operations on files
- Space owner not used in WLCG
- There must not be any expectation that the space occupied by a removed or purged file is released immediately (Rm asynchronous).
- No concept of a DEFAULT space
 - Implementation specific only
 - It might differ from implementation to implementation
 - To be configured in agreement with the experiments.



- Clear definition of files and copies
- No retention policy info is now attached to a file
- No primary copies of files
- A file can have several copies in several spaces
- TURLs might not be unique



- srmRm should be asynchronous
- srmLs only requires privileges on the namespace
- srmPurgeFromSpace only removes online copies of a file
- srmBringOnline can also operate in CUSTODIAL-NEARLINE spaces
- srmRelease: unspecified behavior in case of PERMANENT files



- Expressed in terms of DN or VOMS/FQAN
- Only primary FQAN used for match
- ACLs definition based on extension of NFSv4.1
- Initial requestor of space or service admin have all privileges
- ACLs apply to space tokens and define a set of possible operations: Release, Update, Read, Write, Stage, Purge, Query, Modify-ACL
- Positive and Negative Access Control Entries (ACE) are allowed. Negative ACEs deny access.

- Tokens passed on Get/BringOnline requests are honored. A copy of the file associated to the SURL specified must be retrieved in the space specified by the space token if the user making the request has sufficient privileges on the specified space token and namespace entry.
- The copy served to the user must be the one for which the user has read permission on the correspondent space.
- Only `srmPurgeFromSpace` needs to be provided at the moment
 - `No need for srmChangeSpaceForFiles`

- srmLs must return the file locality that reflects the situation of all copies of a file
 - If there is a copy in T1D0 on tape only and a copy in T0D1 on disk, the correspondent file locality must be `NEARLINE_AND_ONLINE`
- Space tokens/descriptions and paths are already passed.
- Only `srmPurgeFromSpace` needs to be provided at the moment
 - No need for `srmChangeSpaceForFiles`

