



Status and plans of StoRM

Magnoni Luca

luca.magnoni@cnafr.infn.it

**WLCG Collaboration Workshop
(Tier0/Tier1/Tier2)**

January 25, 2007

Outline

- StoRM status
- About “SRM open issues”
- Some more general issues
- StoRM plan

New “green boxes” in StoRM (1/2)

- RequestToken, SpaceToken and userDescription are now case sensitive.
- **srmAbortRequest** and **srmAbortFiles** are done.
 - Further tests are needed.
- **srmExtendLifetime** almost done.
 - Further tests are needed.
- TRetentionPolicyInfo now is mandatory in **srmSpaceReservation**.
- Other minor fixes done.
 - Fixed the request status when *PtP* request is aborted.

New “green boxes” in StoRM (2/2)

- **srmLs :**
 - Now works on the top directory (issue #10)
 - without specify the "\" char in the SURL.
 - returns SRM_FILE_BUSY in case of PrepareToPut on the file.
 - recursive case supported (issue #19)
 - limit the number of entry.
 - returns *fileLifeTimeleft*
 - also in the recursive case.

StoRM lacks in MoU methods

- **srmCopy** in pull mode is not available yet.
 - Next step.
- **srmChangeSpaceForFile**
 - Effort depends on the space reservation type (static or dynamic)
- **srmStatusOfChangeSpaceForFilesRequest**
 - First implementation will be synchronous.

Non MoU methods in StoRM

(1/2)

- **srmUpdateSpace, srmPurgeFromSpace, srmExtendFileLifeTimeInSpace**
 - Related to Storage Area management.
- **srmStatusOfUpdateSpaceRequest**
 - StoRM will provides **srmUpdateSpace** method as synchronous.
- **srmStatusOfLsRequest**
 - StoRM provides **srmLs** method as synchronous.

Non MoU methods in StoRM

(2/2)

- **srmSuspendRequest, srmResumeRequest**
 - We don't have a plan for these methods.
Which is the priority?
- **srmCheckPermission, srmSetPermission, srmGetPermission**
 - Management of permissions through these methods requires more investigation on the security model (*See slides "Open issues"*).
- **LsRecursive, RmdirRecursive**
 - Already available. (issue #19, #20)

Comments on SRM Open Issues

(1/3)

- **srmExtendFileLifeTime (issue #9):**
 - Currently there are too much implicit behaviors. Most SRM users will ignore them and get unexpected results.
 - PtG allows to specify pinLifeTime at request level, but with **srmExtendFileLifeTime** you can modify pinLifeTime to a set of SURL. Incoherence?
 - Is plausible insert PinLT at SURL level within PtG request?
 - Should be better to have two different method for extend File (SURL) and Pin (TURL) lifetime to avoid confusion (in the next srm version?)

Comments on SRM Open Issues

(2/3)

- srmLs (issue #15, #18)
 - Information as fileStorageType, retentionPolicyInfo and fileLocality should be heavy to retrieve.
 - But not much more heavy than the LifeTimeLeft for each file we already provide.
 - LifeTimeLeft = lifetimeAssigned - time passed since the assignment
 - Which is the right value for the UserID in the UserPermission field? DN? uid?

Comments on SRM Open Issues

(3/3)

- srmAbortRequest (issue # 23)
 - **Do nothing** on completed / released / expired files.
- srmBringOnLine:
 - Everything works fine if it is synchronous?

More general issues : (1/3)

Permissions management

- Permissions are bound with LFN.
- At SRM level exists only SURL permissions.
 - The scope is on the single SE context.
 - Who is allowed to modify permissions?
 - SURL Owner?
 - Introduce new *PermissionMode* for control (e.g.'C')?
- What about ACL synchronization between two (or more) different SRMs? (e.g. one LFN with 2 SURLs)
 - An external services makes the synchronization?
 - E.g.: LFC sends **setPermissions** requests to all SRM related with LFN?

More general issues : (2/3)

Storage Classes

- How to create a Storage Area?
 - Agreement between VO admin and all SRM sites admin to use the same *UserSpaceTokenDesc*.
 - The SRM v3 method '***srmDeclareStorageArea***' is either
 - emulated in SRM v2.2 by some kind of SRM configuration features, **or**
 - A ***srmReserveSpace*** request submitted by SE admin.
 - Who are SE admins?
 - How to recognize them?

More general issues : (3/3)

Dynamic Space Reservation

- How to specify a SA in **srmReserveSpace** ?
 - If by USTokenDesc, then how to specify an alias for dynamic space reservation?
 - Instead, if USTokenDesc. is used to specify an alias, then only the default SA could be available for dynamic space reservation.

Storage Area in StoRM

- What is available now
 - SA is identified through path
 - Only Replica-Online is supported
- And in the next version (by April)
 - SA will be independent from path
 - Different qualities of SA will be supported.
 - Only disk based storage system.

StoRM plans

- Glue Schema v1.3
 - We plan to make it be available by April.
 - What about the use of G.S.v1.3 by high-level tools?
- Storage Area independent from path
 - We plan to make it be available by April.
- Missing methods
 - MoU methods:
 - after completion of the “previous point”.
 - Non-MoU methods :
 - when all other issues will be ok.

Testing

- Functionality and Interoperability.
- Use case test.
- Stress test.
 - Any plan about this?
- StoRM relies on compatibility with *lcg-utils*, *GFAL* and *FTS*.
 - There are many requests from the users interested in StoRM to have some kind of unsupported alfa-version of *lcg-utils* and *GFAL* available.

Questions and Suggestion



StoRM security framework

1. The user wants to access a file.
2. StoRM verifies if the principal holds a valid proxy certificate and delegates the external policy decision point to validate the request.
3. StoRM then queries the Authorization Sources to verify if the user can perform the specified operation on the file.
4. StoRM queries the Mapping Service to obtain the local user account corresponding to the grid identity of the requester.
5. The file system wrapper enforces permissions by setting a new ACL on the physical file.
6. The user job can be executed into the worker node.
7. The application can perform a standard POSIX call to access the file from the storage system.

