**SoS, Disk space management (910)**

|  |  |
| --- | --- |
| Unique ID of the requirement | #910 |
| Title of requirement | Disk space management |
| Reference | <https://rt.egi.eu/guest/Ticket/Display.html?id=910> |
| Priority | 4 |
| Submitter | Life Sciences Grid Community (LSGC VRC - <http://www.egi.eu/collaboration/LSGC.html>) |
| Status at TCB | In Clarification |
| RT ticket belonging to this requirement | Disk space management: <https://rt.egi.eu/guest/Ticket/Display.html?id=910> |
| Description of the requirement | Simplify the work of grid application developers and user by preventing storages to become full and return failures related to this to users and their jobs. How this is achieved is up to the technology provider to decide. Example possibilities:   * introducing storage quotas on storages (per user and/or per job), supporting users and jobs with tools to watch their quotas * monitors that check SE status and alarm site admins to free up space by removing or moving files elsewhere when the storage starts to fill up. Providing tools for the admins to perform these tasks * monitors that can automatically remove or move files from an SE to another SE in case of fill up   Currently a short-term solution was offered to the LSGC VRC, this may be used by the technology providers to develop a long-term solution:   * Nagios probe for sending alarms if SE is filling up. See: [#2766](https://rt.egi.eu/guest/Ticket/Display.html?id=2766) * VO Services team provided manual for using LFCBrowse to migrate all the needed files in case SE is filling up.REF: [https://wiki.egi.eu/wiki/VO\_Services/Services\_and\_Tools\_Portfolio#SE\_intervention:\_LFCBrowseSE](https://wiki.egi.eu/wiki/VO_Services/Services_and_Tools_Portfolio) |
| Related tickets in the EGI Helpdesk | Proposed solution only based on Nagios alarms:  <https://ggus.eu/ws/ticket_info.php?ticket=74741> |
| Goals and objectives | Preserve the files and protect the SE from filling up the disk space. |
| Impact | Lost files. |
| Affected services | Services of the LSGC VRC, i.e. sites supporting the biomed, vlemed and lsgrid VOs.  List of SE services supporting the VO (from Top-BDII at CERN):  **SEs:** DPM, StoRM, dCache |

**Name:** Disk Space Management

**Assessed Requirements and execute summary**

The request is manifold and the use case not clearly defined. My interpretation is that the customer is describing three different issues.

1. An error return should be provided if a storage element is full and the currently transferred data can’t be completely received.
2. The customer would like to have a general idea how much space is left on a storage element to schedule transfers.
3. The customer would like storage elements to transfer data to other storage element when space is running short to make space for newly incoming data.

**Executive Summary**

1. *An error return should be provided if a storage element is full and the currently transferred data can’t be completely received*. All EMI storage elements return error messages in case incoming data can’t be stored. That might however happen in the middle of a transfer. This is normal behavior for storage. The application has to be prepared that a transfer failed for different reasons. “Disk Full” is one of those conditions. If a storage element doesn’t report such an error and data goes lost, the customer is asked to submit a bug report against the corresponding storage element provider. If the customer would like to avoid the situation of a full disk, EMI storage elements provide an SRM interface, which allows allocating storage prior to the transfer. (Consult the SRM specification or SE documentation for details) The corresponding calls a identical for all storage elements implementing the SRM interface.
2. *The customer would like to have a general idea how much space is left on a storage element to schedule transfers.* This can be achieved by two different mechanisms. The already mentioned SRM interface allows querying the condition of spaces on the storage element. That can be configured arbitrarily fine-grained. Another possibility is to utilize the GLUE(2) interface for storage. Here spaces are reported as well. In addition, for some storage elements (e.g. DPM) nagios probes are available, reporting ‘disk full’. This however is difficult to be integrated into application frameworks. GLUE and SRM are the recommended mechanisms to handle filling up storage elements.
3. *The customer would like storage elements to transfer data to other storage element when space is running short to make space for newly incoming data*. Although theoretically possible it could be hard to achieve, as the storage element doesn’t know which data to migrate where. However, EMI provides all building blocks allowing customer to build such a system, depending on their particular requirements. Either with SRM or with the GLUE information system, the customer can detect how much space is already used. Based on file system name space queries (including file meta data information) the customer can select files, which are no longer needed locally and which are save to be transferred somewhere else. Subsequently the customer needs to find available space on other storage systems. Finally the framework can submit transfer requests to the EMI FTS service to get the actual transfer done.

**Efforts assessment**

a) and b) can only be achieved by the customer and the efforts depend on which solution is selected. For c) efforts are significant and again depend on how sophisticated the final system should be. For WLCG, this is entirely in under control of the experiments.

**Milestones and timelines**

N.A.

**Resources**

N.A.

**Risks**

N.A.

**Constraints**

N.A.

**Assumptions**

N.A.