

On SRM usage

Monitoring and usage statistics
of the SRM service at CERN and at T1s

Giuseppe Lo Presti (CERN/IT)



- Motivation
- Current service deployment
- Usage statistics at CERN and at RAL
- Preliminary conclusions

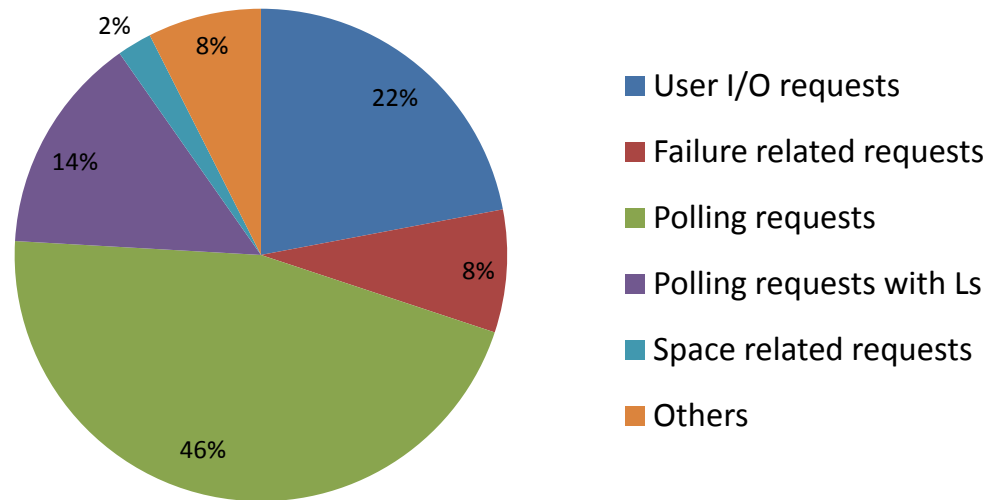
- SRM v2.2 is in production since ~1 year
 - First production experience: CCRC'08
 - Many lessons learnt, smoothly running in the past months
- **The objective is to measure current SRM utilization patterns**
 - To spot abuses/overload, to predict available headroom
 - We'll try to analyze the workload the SRM service is sustaining
 - Breakdown by request methods and by clients
- Input for the upcoming SRM workshop @ DESY, May 18th, 2009

- At CERN
 - 5 endpoints, one per LHC VOs + general public instance
 - 3 nodes each
- At RAL
 - 5 endpoints; looking at ATLAS and CMS only
 - 2 nodes each, except ATLAS (4 nodes)
- At CNAF
 - General endpoint + CMS dedicated endpoint
 - 3 nodes each
- Statistics gathered on a **single node** of each endpoint, **for a 2-month time interval**
 - From March 1st to April 30th

- To ease the breakdown analysis, the following categories have been defined:
- User I/O requests
 - srmPrepareTo, srmCopy, srmBringOnline, ...
- Failure related requests
 - srmAbortRequest, srmAbortFiles, srmReleaseFiles
- Polling/query requests
 - srmPing, srmStatusOf, srmLs
- Space related requests
 - srmGetSpaceTokens, srmReserveSpace, ...
- Others
 - **12** more methods (the specs include **39** methods)

srm-cms
@CERN

1.2 reqs/sec

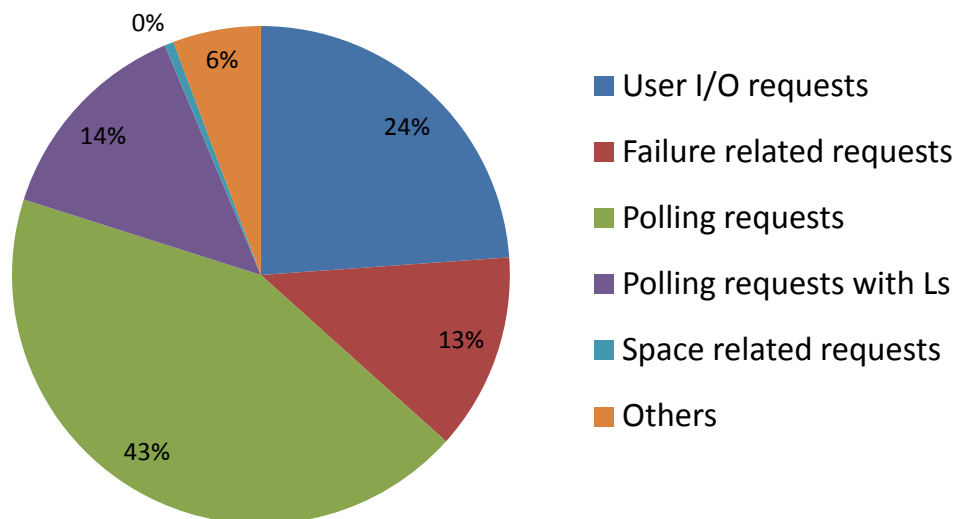


- Observations

- Fair ratio of polling vs. I/O (prepare) requests
 - but note the amount of srmLs, also used for polling
- **Failure/success ratio not taken into account**
 - These are all the incoming requests
 - “Failure related” requests are normally issued to clean up after a failure has occurred at either ends

**srm-public
@CERN**

0.4 reqs/sec

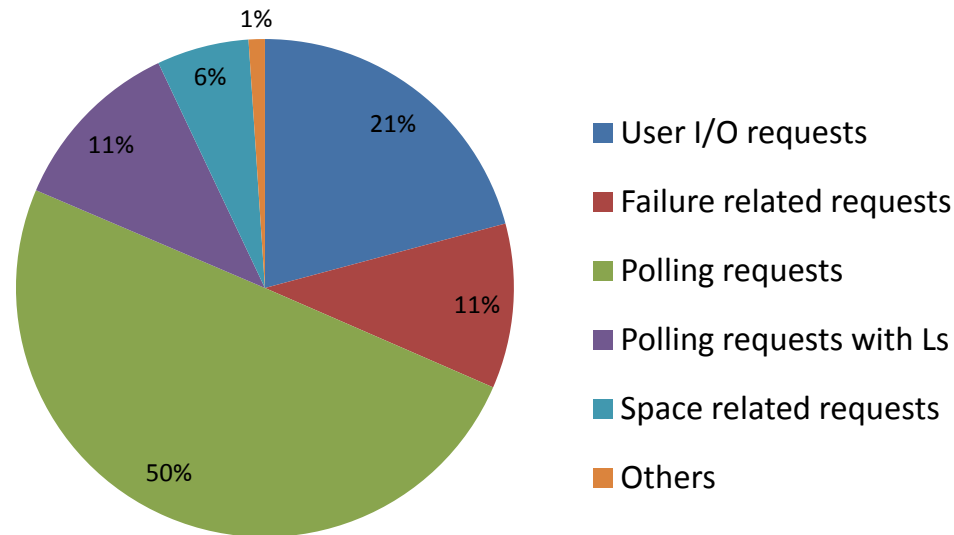


- **Observations**

- The “others” category for srm-public includes the whole set of existing SRM methods
 - whereas only a fraction of them is effectively used elsewhere
 - srm-public serves the DTEAM VO, and many SRM tests (e.g. S2) run as DTEAM...

srm-cms
@RAL

0.5 reqs/sec



- Observations

- At a Tier1 the ratio polling/prepare requests is slightly worse
- And the number of “other” requests is negligible
 - Only **14** SRM methods used, out of the **39** in the specs

- No detailed data yet...
- But main SRM client @ CERN is **FTS** by far
 - 80-90% of the total load, depending on the endpoint
- Clients at T1 sites typically just follow

- A clear evidence from this exercise is the different behavior depending on the VO
 - ATLAS ran at **8** requests/s, **5 times more** than LHCb or CMS, whereas ALICE ran at **2 orders of magnitude less**
 - The ATLAS average file size played a role here
 - **To be still checked whether over the observation period all VOs ran at any constant load**
 - **STEP'09 will hopefully provide a baseline**
 - The load at T1s is of the same order of the load at the T0
 - T1 storage activity is much more “Grid-oriented”, thus it mostly goes through SRM

- Results are very preliminary
 - More data are being gathered from **FTS logs** and from **DPM logs at T2s**
 - Will be interesting to compare these results with other sites running dCache and/or StoRM
 - And dCache have already shown interest to implement the same set of metrics
 - This will be input for discussion at DESY
- Plans
 - Clearly define relevant metrics
 - E.g. failure/success rate, #requests/real transfer, ...
 - Incorporate this process as a permanent automated monitoring activity