

## Questions to the SRM server developers:

### 1- Should the storage servers allow clients to specify the DesiredTotalRequestTime and return the RemainingRequestTime ?

- This would allow for a more optimized polling loop for asynchronous operations since the client can stop making status request to the server if the request time has already expired.

About this topic see also questions to SMR clients developers.

### 2- What should be the behavior of the SRM server when overloaded?

There is the need to communicate to the client that the server is overloaded by too many requests and cannot fulfill the request in that moment:

- In case of synchronous requests, in which case do we want the server to signal to the client that it cannot answer the request because it is too busy? How do we define 'too busy'?

It was proposed to return to the client the code SRM\_FILE\_BUSY at request level.

- In case of asynchronous requests, in which cases do we want the server to reject the request because it is too busy? How to define the server 'too busy'? Is it the same definition of 'too busy' than for synchronous requests? Furthermore, should the algorithm used for polling on asynchronous request be changed if the server is busy and if so, how ?

About this topic see also questions to SMR clients developers.

### 3- How to communicate the reason why the SRM server is busy?

Define a standard *explanation* to return in case of SRM\_FILE\_BUSY to give more details?

How to express the degree of occupation of the server? It has been proposed to use the srmPing *otherinfo* output parameter. Which quantities could be used to quantify the degree of occupation? the load of the machine hosting the front-end? Other..?

-- FlaviaDonno - 10 Feb 2009

---

This topic: LCG > CCRC08SSWGQuestionsDevelopers

Topic revision: r4 - 2009-02-11 - ElisaLanciotti



Copyright &© 2008-2020 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

Ideas, requests, problems regarding TWiki? Send feedback