

Hi, all,

Enclosed is the second revision of the methods document, SRM.v2.1.methods.rev1.doc. This revision has some important changes made as a result of comments received from several people. Many thanks to those who provided comments. Special thanks to Jean-Philippe who provided many constructive comments. I discuss below all the changes that were made. Typos were fixed, too. I think we addressed all the comments we got.

Arie.

1) Following the suggestions provided, all the methods now use a uniform TReturnStatus that includes a “statusCode” and an “explanation”. The statusCode is has the same format for all methods, which is an integer made of two part: 3 digits for the method, and 2 digits for the error code. For example, the method *srmReserveSpace* has 101 assigned to the method, and 01, 02, etc assigned for the error codes, as follows:

```
#define      SPACE_RESERVED                10101
#define      SPACE_PARTIALLY_RESERVED     10102
#define      SPACE_RESERVE_FAILED         10103
```

A section at the end of the document contains all the codes that we needed. Others could be added. If some SRM wants to use additional error codes private to it, it can be assigned some error codes block.

2) Ttime Duration is now in seconds

3) File size in now in bytes

4) *srmLs* and *srmGetFileMetaData* are now combined into one. We now have *srmLs* only. Permission and creation date were added to the return information TmetaDataPathDetail. The permission refers to the caller’s permission for this path/file.

5) There is no reference to “current directory”. It is assumed that SRM keeps track of it, and it is returned as a result of *srmPwd*. *SrmCd* assumes current directory unless otherwise specified.

6) *srmRmdir* can be specified as “recursive”. It refers to a single top directory. It cannot be applied to an array of directories. This was not changed.

7) *srmLs* has now 3 options: zero directory levels (only file in the directory shown), one level directory (similar to `ls -l`), and recursive (all the way to the bottom leaves of the directory).

8) An explanation was added for *userRequestDescription*. This was extensively discussed in the V2.0 *func.spec* document, but an explanation was added for completeness.

9) We left the reference to “permission list” unchanged, since we do not want to enforce the use of ACLs. Each SRM can implement its permission list as it chooses.

10) The term “world” for permission was changed to “*”

11) A new parameter was added to srmCopy, called removeSourceFiles. When it is on, it tell the SRM to remove the files in the source location, so it has the effect of “move” between sites (recall that this is different from srmMv, which is only used between spaces in the same SRM site, since it is a directory function).

12) A new function was added, srmRemoveFiles. This function can be called to force file removals from a remote site by a client. It is similar to the srmRm for local files, but it uses an SURL instead of a directory path. This function is also useful to support the removeSourceFiles parameter in srmCopy for “pull mode”. After the file transfer is successful, the targetSRM (which pulls the file) can issue a srmRemoveFiles if the removeSourceFiles parameter in srmCopy is set. This is not necessary in the srmCopy “push mode”, since the file is local to the “pushing” SRM.