

# **The Storage Resource Manager Interface Specification**

## **Version 2.0**

Contributors:

JLAB: Ian Bird, Bryan Hess, Andy Kowalski

Fermi: Don Petravick, Rich Wellner

LBNL: Junmin Gu, Ekow Otoo, Alex Romosan, Alex Sim, Arie Shoshani

WP2-EDG: Wolfgang Hoscheck, Peter Kunszt, Heinz Stockinger,  
Kurt Stockinger, Brian Tierney

WP5-EDG: Jean-Philippe Baud

THIS IS A WORK IN PROGRESS DRAFT

It reflects decisions discussed in

<http://sdm.lbl.gov/srm/documents/joint.docs/ SRM.v2.0.joint.func.design.doc>

### **Introduction**

This document is the interface specification for grid-aware Storage Resource Managers (SRMs). It is based on the design document published in GGF4 entitled “SRM Joint Functional Design, Summary of Recommendations” (can be downloaded from [1]). This document represents the second version, SRM v2.0, of the joint SRM specification. The first version, SRM v1.0, is the basis for the current SRM implementation efforts at Jlab, and Fermi. The document describing SRM v1.0 can be downloaded from [2]. LBNL’s implementation of SRMs is based on an earlier version of the API, described in [3, 4]. A general document on concepts of SRMs and how they fit the grid environment was published recently, and can be downloaded from [5]. For people unfamiliar with SRM concepts and/or unfamiliar with the design issues we recommend reading the above documents. It is essential to read the companion design document in [1] to understand terms and reasons for the methods defined in this document.

The interface methods defined in SRM v2.0 are closely related to the methods used in SRM v1.0, but are more consistent with the view of the participants of the joint design, that included people from the European Data Grid Project (EDG) and the DOE labs: Jlab, Fermilab, and LBNL. The most notable differences are: 1) in addition to `srmGet` and `srmPut`, an `srmCopy` method was added to coordinate the storage allocation and pinning for third party transfers; 2) the files stored in SRMs can have three types associated with them: “volatile”, “durable”, and “permanent” (see discussion in [1] for meanings and reasons); and 3) pinning of a file is no longer an explicit method, but rather is implied by the requesting methods `srmGet` and `srmPut`, and `srmCopy`. The correspondence between the methods of SRM v1.0 and SRM v2.0 is shown in table 1.

| Method (v.2)         | Method (v.1)                                         |
|----------------------|------------------------------------------------------|
| srmGet               | Get, Pin                                             |
| srmRelease           | UnPin                                                |
| srmPut               | Put                                                  |
| srmPutDone           | setFileStatus                                        |
| srmCopy              |                                                      |
| srmTeminateRequest   |                                                      |
| srmAbortFile         |                                                      |
| srmChangeFileStatus  | MkPermanent, setFileStatus                           |
| srmSuspendRequest    |                                                      |
| srmResumeRequest     |                                                      |
| srmGetRequestStatus  | GetRequestStatus,<br>getEstGetTime,<br>getEstPutTime |
| SrmGetFilesStatus    |                                                      |
| srmGetRequestSummary | getEstGetTime,<br>getEstPutTime                      |
| srmGetFileMetaData   | getFileMetaData                                      |
| srmGetProtocols      | getProtocols                                         |
| srmAdvisoryDelete    | AdvisoryDelete                                       |
| srmRenewLifetime     |                                                      |

Table 1: correspondence between methods in SRM v1.0 and v2.0

### Description of methods

The methods described below use several terms as parameters, notably: LFN, SURL, TURL, and stFN. These concepts are discussed in [1], but because of their fundamental use here, we included them in an appendix in this document.

In order to have the methods described in a protocol independent fashion, we use a pseudo notation. This will permit these methods to be described in various languages such as XML-WSDL, CORBA IDL, C-API, etc. We are attempting to agree on a single language and protocol that anybody who is developing an SRM API to his or her implementation and systems will adhere to. But, each SRM implementation should be free to support any other languages or protocols as they see fit.

We decided to use a notation as close as possible to the notation used in SRM v1.0. But a few additions were necessary to make the specification precise in this version. The notation we use is as follows:

1. Each method has input and output parameters. We use a table to represent that in columns labeled “in” and “out” as is done in SRM v1.0. This is shown in Table 2.
2. A tuple notation is  $\langle \text{par}_1, \text{par}_2, \dots, \text{par}_n \rangle$ , such as  $\langle \text{LFN}, \text{size}, \text{SURL} \rangle$ . However, in order to include comments, in most cases we use a table notation too. This is done for several tuples shown in table 3 through table 8.
3. A set of parameter is denoted as  $\{\text{parameter}\}$ , such as  $\{\text{fileSpecification}\}$  which represent the file specification of a set of files.
4. An ordered set of parameters is denoted as  $(\text{parameter})$ , such as the ordered set of protocols provided by the user (protocol).
5. An optional parameter is denoted as  $[\text{parameter}]$ , such as  $[(\text{protocol})]$ . For tuples of parameters in tables, we have an additional column labeled “null ok?”

Table 2 contains all the methods supported. We used the terms “requestStatus” and “fileStatus” to represent information returned about each. Some functions return both, and some return either one or the other. For example, `srmGetRequestSummary` returns only “requestStatus” which includes query estimation as well.

| Method (v.2)                       | Arguments (in)                                                                                                  | Return (out)                   |
|------------------------------------|-----------------------------------------------------------------------------------------------------------------|--------------------------------|
| srmGet                             | userID, [storageUserID], [timeout]<br>{fileSpecification}, [filesType],<br>[RequestIDDescription], [(protocol)] | requestStatus,<br>{fileStatus} |
| srmRelease                         | userID, requestID, SURL                                                                                         | fileStatus                     |
| srmPut                             | userID, [storageUserID],<br>{fileSpecification}, [filesType],<br>RequestIDDescription, [(protocol)]             | requestStatus,<br>{fileStatus} |
| srmPutDone                         | userID, requestID, SURL                                                                                         | fileStatus                     |
| srmCopy                            | userID, [storageUserID],<br>{fileSpecification}, [filesType],<br>RequestIDDescription, [(protocol)]             | requestStatus,<br>{fileStatus} |
| srmTeminateRequest                 | userID, requestID                                                                                               | requestStatus                  |
| srmAbortFile                       | userID, requestID, SURL                                                                                         | fileStatus                     |
| srmChangeFileStatus                | userID, requestID, {SURL },<br>filesType                                                                        | {fileStatus}                   |
| srmSuspendRequest                  | userID, requestID                                                                                               | requestStatus                  |
| srmResumeRequest                   | userID, requestID                                                                                               | requestStatus                  |
| srmGetRequestStatus                | userID, requestID                                                                                               | requestStatus,<br>{fileStatus} |
| srmGetFilesStatus                  | userID, requestID, {SURL }                                                                                      | {fileStatus}                   |
| srmGetRequestSummary               | userID, requestID                                                                                               | requestStatus,                 |
| srmGetFilesMetaData                | userID, {SURL }                                                                                                 | {fileMetaData}                 |
| srmRequestEstimateTime<br>(remove) | userID, requestID                                                                                               | requestStatus                  |
| srmGetProtocols                    | userID                                                                                                          | (protocol)                     |
| srmAdvisoryDelete                  | userID, requestID, SURL                                                                                         | fileStatus                     |
| SrmGetRequestID                    | userID, requestIDDescription                                                                                    | {<requestID, date>}            |
| SrmRenewLifetime                   | userID, requestID, SURL                                                                                         | fileStatus                     |

Table 2: methods and their “input” and “output” arguments

| Key  | Value                             | fileSpecification    |                      |                       | comments                                                                |
|------|-----------------------------------|----------------------|----------------------|-----------------------|-------------------------------------------------------------------------|
|      |                                   | Null OK?<br>(srmGet) | Null OK?<br>(srmPut) | Null OK?<br>(srmCopy) |                                                                         |
| SURL | The site specific file identifier | No                   | Does not apply       | No                    |                                                                         |
| LFN  | Globally unique logical file name | Yes                  | Does not apply       | Yes                   | If not prvided, SFN can be used                                         |
| Size | File size in bytes                | No                   | No                   | No                    | In advanced version – Yes. If not provided, a default is assumed by SRM |
| stFN | Storage file name                 | Does not apply       | Yes                  | Yes                   | If provided, can be used<br>By srmPut or srmCopy                        |

Table 3: parameters for fileSpecification

| RequestStatus        |                                                                                                                                                                                                |          |
|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|
| Key                  | Value                                                                                                                                                                                          | Null OK? |
| requestID            | string, the unique requestID                                                                                                                                                                   | No       |
| requestType          | The method that formed the original request: srmGet, srmPut, srmCopy.                                                                                                                          | No       |
| filesTypeRequested   | The type of files requested with the original request: volatile, durable, permanent. Optional in request.                                                                                      | Yes      |
| filesTypeAssigned    | The type of files assigned by SRM to all files after the original request was made: volatile, durable, permanent.                                                                              | No       |
| requestIDDescription | As provided with the request. Optional in request.                                                                                                                                             | Yes      |
| protocolsRequested   | List of protocols provided with the original request. Optional in request.                                                                                                                     | Yes      |
| state                | Status of the request as a whole: pending, active, done, failed.                                                                                                                               | No       |
| errorMessage         | An error message in case of a “failed state”. Error message are defined in the table errorMessage and consist of an integer code and an explanation string.                                    | Yes      |
| submitTime           | Date and time when the request was submitted.                                                                                                                                                  | No       |
| startTime            | Date and time when the SRM first started work on a file for this request.                                                                                                                      | Yes      |
| finishTime           | Date and time when the last action for this request was completed.                                                                                                                             | Yes      |
| estTimeToStart       | Estimated delta time (in seconds) until the request will become active.                                                                                                                        | Yes      |
| estTimeToEnd         | Best effort time estimate for the request. If request started it provides time to bring all remaining files to SRM’s cache. If not started, it provides total time estimation for the request. | Yes      |
| retryDeltaTime       | The client must re-check status or perform another call to SRM in this many seconds or the SRM may assume that the client has died and cancel the request.                                     | No       |

Table 4: parameters for requestStatus

| fileStatus        |                                                                                                                                                             |                           |
|-------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------|
| Key               | Value                                                                                                                                                       | Null OK? (optional)       |
| SURL              | The site specific file identifier.                                                                                                                          | No                        |
| LFN               | Globally unique logical file name.                                                                                                                          | No                        |
| size              | File size in bytes.                                                                                                                                         | No                        |
| stFN              | Storage file name.                                                                                                                                          | Yes                       |
| state             | pending, ready, beingTransferred, released, failed.                                                                                                         | No                        |
| errorMessage      | An error message in case of a “failed state”. Error message are defined in the table errorMessage and consist of an integer code and an explanation string. | Yes                       |
| TURL              | Transfer URL.                                                                                                                                               | Yes, until state is Ready |
| owner             | The creator of the file.                                                                                                                                    | Yes                       |
| group             | The creating group for the file.                                                                                                                            | Yes                       |
| permissionMode    | Unix-style permissions of the file (e.g. 0644)                                                                                                              | Yes                       |
| checksumType      | May be null. Example: CRC32                                                                                                                                 | Yes                       |
| checksumValue     | May be null.                                                                                                                                                | Yes                       |
| isCached          | Is file in cache?                                                                                                                                           | No                        |
| isPinned          | Is file pinned for this client. Boolean or null.                                                                                                            | No                        |
| fileType          | volatile, durable, permanent.                                                                                                                               | No                        |
| remainingLifetime | Time remaining till lifetime expires                                                                                                                        | Yes                       |
| originalFileName  | Original filename. May not be meaningful to the HRM                                                                                                         | Yes                       |
| QueueOrder        | Expected order of service in the request                                                                                                                    | Yes                       |

Table 5: parameters for fileStatus

| fileMetaData     |                                                     |                     |
|------------------|-----------------------------------------------------|---------------------|
| Key              | Value                                               | Null OK? (optional) |
| SURL             | The site specific file identifier.                  | No                  |
| LFN              | Globally unique logical file name.                  | No                  |
| size             | File size in bytes.                                 | No                  |
| owner            | The creator of the file.                            | Yes                 |
| group            | The creating group for the file.                    | Yes                 |
| permissionMode   | Unix-style permissions of the file (e.g. 0644)      | Yes                 |
| checksumType     | May be null. Example: CRC32                         | Yes                 |
| checksumValue    | May be null.                                        | Yes                 |
| isCached         | Is file in cache?                                   | No                  |
| isPinned         | Is file pinned for this client. Boolean or null.    | No                  |
| fileType         | volatile, durable, permanent.                       | No                  |
| originalFileName | Original filename. May not be meaningful to the HRM | Yes                 |

Table 6: parameters for fileMetaData

| StorageUserID |                                                                                          |          |
|---------------|------------------------------------------------------------------------------------------|----------|
| Key           | Value                                                                                    | Comments |
| securityType  | Takes the values: “globus”, “kerberos”, “ssl”, “clear”, etc.                             |          |
| ID            | String, can be Globus user proxy, Kerberos token, SSL-encrypted user login+password, etc |          |

Table 7: parameters for userID

| Other parameters used in methods |                                                                                                            |          |
|----------------------------------|------------------------------------------------------------------------------------------------------------|----------|
| Key                              | Value                                                                                                      | Comments |
| fileType                         | Takes the values: “volatile”, “durable”, “permanent”                                                       |          |
| requestID                        | String. This is assigned by SRM. Need to be used to refer to request in srmStatus, srmRelease, etc.        |          |
| protocol                         | String.                                                                                                    |          |
| requestIDDescription             | String. May be provided by client to find out requestIDs assign by SRM by using the method srmGetRequestID |          |

Table 8: other parameters and their explanation

LFN- is a logical file name that is globally unique for a given dataset. Thus, the dataset name is usually the first part of the LFN. It is the choice of the dataset designer how to assign these names. If the files are organized in directories then the directory names are part of the LFN. The dataset name and directory names are separated by “/”. An example of an LFN is: “CERN-dataset-7/run17/part1/file-123”.

SFN- is a file name assigned by a site to a file. Normally, the site file name will consist of a “machine:port/directory/LFN”, but the site can choose to use another string instead of the LFN. An example of an SFN is: “sleepy.lbl.gov:4000/tmp/foo-3000”. In this example we used the simple file name “foo-3000” to simplify the example.

SURL – is a “site URL” which consists of “protocol://SFN”. The protocol for communicating with an SRM is simply “srm”. An example of an SURL for a file managed by SRM is: “srm://sleepy.lbl.gov:4000/tmp/foo-3000”.

TFN – is the “transfer” file name of the actual physical location of a file that needs to be transferred. It has a format similar to an SFN.

TURL – is the “transfer URL” that an SRM returns to a client for the client to “get” or “put” a file in that location. It consists of “protocol://TFN”, where the protocol must be a specific transfer protocol selected by SRM from the list of protocols provided by the client (see recommendation 9). An example is: “gridftp://sleepy.lbl.gov:4000/tmp/foo-3000”.

StFN – is the “storage” file name that a client may request SRM to use when it stores the file. This is useful for telling SRMs where to archive a file. Normally, SRMs that archive files, such as an HRM, may choose to honor that request. But, the SRM may choose to give it another name, and return that to the client.

## References

1. Arie Shoshani, et al., “SRM Joint Functional Design, Summary of Recommendations”, presented in Global Grid Forum 4, Toronto, Canada, Feb. 2002.  
Can be downloaded from:  
<http://sdm.lbl.gov/srm/documents/joint.docs/SRM.v2.0.joint.func.design.doc>
2. Ian Bird, et al., “Common Storage Resource Manager Operations”. Can be downloaded from: <http://sdm.lbl.gov/srm/documents/joint.docs/SRM.v2.0.doc>
3. Arie Shoshani, Alex Sim, Junmin Gu, “Description of SRM interface and C++ bindings of SRM IDLs”. Can be downloaded from:  
<http://sdm.lbl.gov/srm/documents/srm-idl-interface-description-092401.pdf>

4. Arie Shoshani, Alex Sim, Junmin Gu, “SRM Design Considerations”. Can be downloaded from: <http://sdm.lbl.gov/srm/documents/srm-design-considerations-092601.pdf>
5. Storage Resource Managers: Middleware Components for Grid Storage, Arie Shoshani, Alex Sim, Junmin Gu, Nineteenth IEEE Symposium on Mass Storage Systems, 2002 (MSS '02). Can be downloaded from: <http://gizmo.lbl.gov/~arie/papers/srm.mss02.pdf>

## **Appendix**

### **File names and URLs used in methods**

LFN- is a logical file name that is globally unique for a given dataset. Thus, the dataset name is usually the first part of the LFN. It is the choice of the dataset designer how to assign these names. If the files are organized in directories then the directory names are part of the LFN. The dataset name and directory names are separated by “/”. An example of an LFN is: “CERN-dataset-7/run17/part1/file-123”.

SFN- is a file name assigned by a site to a file. Normally, the site file name will consist of a “machine:port/directory/LFN”, but the site can choose to use another string instead of the LFN. An example of an SFN is: “sleepy.lbl.gov:4000/tmp/foo-3000”. In this example we used the simple file name “foo-3000” to simplify the example.

SURL – is a “site URL” which consists of “protocol://SFN”. The protocol for communicating with an SRM is simply “srm”. An example of an SURL for a file managed by SRM is: “srm://sleepy.lbl.gov:4000/tmp/foo-3000”.

TFN – is the “transfer” file name of the actual physical location of a file that needs to be transferred. It has a format similar to an SFN.

TURL – is the “transfer URL” that an SRM returns to a client for the client to “get” or “put” a file in that location. It consists of “protocol://TFN”, where the protocol must be a specific transfer protocol selected by SRM from the list of protocols provided by the client (see recommendation 9). If the physical storage location matches the one provided by the SURL, then only the protocol is replaced in the TURL. For the above SURL example, and assuming the protocol is “gridftp”, the TURL will be: “gridftp://sleepy.lbl.gov:4000/tmp/foo-3000”. However, the physical file location can be anywhere at that site, giving the freedom for the site manager to change the physical locations of files without having to change the SURL or update the replica catalog. If for the above example the physical location of the file is on another machine (e.g. “dm.lbl.gov”, another path (e.g. “/home /level1”), and even another file-name (e.g. “abc-3000”) then the TURL will be: “gridftp://dm.lbl.gov:4000/home /level1/abc-3000”.

StFN – is the “storage” file name that a client may request SRM to use when it stores the file. This is useful for telling SRMs where to archive a file. Normally, SRMs that

archive files, such as an HRM, may choose to honor that request. But, the SRM may choose to give it another name, and return that to the client.