



NORDUGRID-TECH-7

7/7/2006

PROTOCOLS, UNIFORM RESOURCE LOCATORS (URL) AND EXTENSIONS SUPPORTED IN ARC

Description

A.Konstantinov*

Contents

- 1 Protocols** **2**

- 2 Extensions** **2**
 - 2.1 Options 2
 - 2.2 Quasi-URLs/Meta-URLs 3

1 Protocols

URLs

All components of NorduGrid's ARC [?] should support following data transfer protocols and corresponding URLs:

- *ftp* - ordinary FTP without user and password fields.
- *gsiftp* - GridFTP [?], enhanced FTP protocol with security, encryption, etc. developed by The Globus Alliance [?].
- *http* - ordinary HTTP with PUT and GET methods using multiple streams.
- *httpg* - HTTP wrapped with Globus' GSI.
- *https* - HTTP wrapped with SSL.
- *file* - local file.

2 Extensions

2.1 Options

In addition to standard URL fields GM supports *options*. Options are of kind *name* or *name=value* and are inserted into URL after host and port and are separated by ';'.

protocol://host[:port][;option[:option[...]]]/path

Following options are supported:

threads=# specify number of parallel transfers to use (currently works for GridFTP, HTTP, HTTPS and HTTPG, default is one),

blocksize=# specify size of chunks/blocks/buffers used in transactions (currently works for GridFTP, HTTP, HTTPS and HTTPG, default is protocol dependent),

cache=yes/no whether file is allowed to be cached on local side (for transfers to local destinations, default is *yes*),

secure=yes/no whether data should be transferred using encryption (currently works for GridFTP only, default is *no* unless specified differently in configuration),

readonly=yes/no if file after transfer to local location can be stored in a way which prevents user, job, etc. from modifying it (use *readonly=no* if You plan to modify file),

checksum=cksum/md5 specifies type of checksum to be computed and provided to the indexing server,

exec=yes/no means file should be treated as executable (currently only used internally).

Example: *gsiftp://grid.domain.org:2811;threads=10;secure=yes/dir/input_12378.dat*

preserve=yes/no specify if file must be uploaded to this destination even if job processing failed (default is *no*)

pattern=pattern defines file matching pattern - currently works for file listing requests sent to *se://* endpoint (see below)

guid=yes/no make software use GUIDs instead of LFNs while communicating to indexing services - meaningful for *rls://* only (see below)

overwrite=yes/no make software try to overwrite existing file(s), i.e. before writing to destination, tools will try to remove any information/content associated with specified URL

protocol=gsi/gssapi to distinguish between two kinds of *httpg*. *gssapi* stands for implementation using only GSSAPI functions to pack and unpack data packets and *gsi* uses additional wrapping of data packets as implemented in Globus IO module (default is *gsi*)

2.2 Quasi-URLs/Meta-URLs

There are also few quasi-URLs supported. Those are URLs which are resolved to one of ordinary protocols for actual transfer of data.

- *se* - to contact "Smart" Storage Element (SSE) being developed at NorduGrid project is also supported. It's syntax is *se://host[:port][;options]/path[?file_id]*. For more information see document "The NorduGrid "Smart" Storage Element".
- *srm* - means Storage Resource Management (SRM) service [?]. Methods defined in version 1.1 of protocol are used. It's syntax is *se://host[:port][;options]/[service_path?SFN=]file_id*. Default *service_path* is *srm/managerv1*.

Also following quasi-URLs of meta-data (indexing) servers are supported:

- *rc* - Globus Replica Catalog: *rc://[location[/location[...]]][@host[:port]/distinguished_name]/lfn*
- *rls* - Globus Replica Location Service: *rls://[url[/url[...]]@]host[:port]/lfn*
- *fireman* - Fireman indexing service which is part of gLite initiative [?]: *fireman://[url[/url[...]]@]host[:port]/service_path?lfn*

Here

lfn Logical File Name (LFN), that should be used for registering/querying in the Indexing Service,

distinguished_name DN of collection in LDAP server used for registering/querying,

location name of location (usually this is host[:port] of SE), each *location* can have *options* appended to it in a way *location;options*. If only *options* part is preset, those options are treated as specified for every location.

url full or partial URL corresponding to physical location of file.

service_path endpoint path of Web service.