Motivation

- In order to interact with DPM a client needs SRM to convert LFN to TURL.
- Going directly to DPM head node with GridFTP is possible, but results in inefficient internal transfers (data is staged from disk nodes via RFIO).
- SRM is not interesting outside of HEP community.
- Other frontends like HTTP have fancy built-in redirection capabilities. Why can't we have nice things with GridFTP?
How GridFTP works

- Basically, it's a standard FTP with GSI on control channel and a bunch of extensions. Control connection starts in plaintext mode.
- Data transfer uses a separate connection(s), endpoint parameters are negotiated via control channel.
- Two FTP data connection modes: Passive and Active.
- GridFTP v2 adds striped extensions and third data connection mode:Delayed Passive.
Active mode

- In Active mode server initiates data connection to a client.
- PORT command contains IP address and port on which client waits for connection.
- Server connects after the transfer request.
Passive mode

- In Passive mode client initiates data connection to a server.
- Server responds to PASV command with IP address and port on which it waits for a client connection.
- Client connects after the transfer request.
- At the moment of PASV response server knows nothing about the file or transfer direction.
Delayed Passive mode

- Works almost like normal Passive, except that PASV response is delayed until the actual transfer command.
- Client connects after the transfer request, when server knows the file name and transfer direction.
- Delayed Passive is supported by standard Globus library and can easily be implemented in all clients that use it (globus-url-copy, gfal2, etc). It does work for third party transfers as well.
Globus GridFTP server

• Three modes of operation
  • Standalone
  • Frontend node – accepts control connections, IPC with backend nodes.
  • Backend node – accepts or initiates data connections, IPC with frontend node, does not accept control connections. From API point of view it looks exactly as a standalone mode.

• Frontend/backend mechanism was initially designed for striped operations (SPOR, SPAS).
GridFTP with DMLite

- DMLite plugin for Globus GridFTP server is built as a DSI (Data Storage Interface) library.
- It uses DMLite Adapter as a fallback if a file is not available locally.
  - Client connection to a standalone node without redirection.
  - Non-delayed Passive client connection to a redirecting head node (uberftp). Backend node is randomly selected in this case.
GridFTP with DMLite

Client → Control connection → Head node

Disk node 1 → Globus IPC
Disk node 2
Disk node n

RFIO fallback

Data connection
Fine points

• Disk nodes listen to IPC connections on the same port that is normally used for control connections (2811 or whatever is configured). Disk nodes do not need to know their head node.

• Head node needs a list of disk nodes to be specified in configuration file, but it's possible to make this list dynamic.

• Globus IPC is not documented, but not very complicated either. Basically one needs to simply pass transfer information from DSI callback function to selected disk node and pass back transfer events.

• There's a defect in Globus DSI API which makes it impossible to tell whether you're reading or writing a file in Delayed Passive mode.

• It's possible to use more than one disk node for the same transfer in striped mode (SPOR, SPAS) if we will ever store files in chunks on different nodes.
Standalone GridFTP with SRM

Client

LFN

TURL

SRM

Disk node 1

Disk node 2

Disk node n

GridFTP with TURL

Data connection
Redirecting GridFTP without SRM

Diagram:
- Client
- GridFTP with LFN
- Head node
- Disk node 1
- Disk node 2
- Disk node n
- Data connection
Compatibility

- Deployment of redirecting GridFTP is transparent to clients except that it's no longer possible to connect directly to the disk nodes.
- Luckily SRM TURL usually has two host names:
  
  \[ gsiftp://<hostname>://<hostname>://<path> \]
- Client connects to the disk node based on the first host name. Second one remains a part of URL and is silently ignored by current DPM GridFTP implementation.
- Slight modification of the TURL allows transparent deployment of redirecting GridFTP with SRM:
  
  \[ gsiftp://<headname>://<diskname>://<path> \]
Thank you!