GridFTP 2 – the road to redirection

Andrey Kiryanov

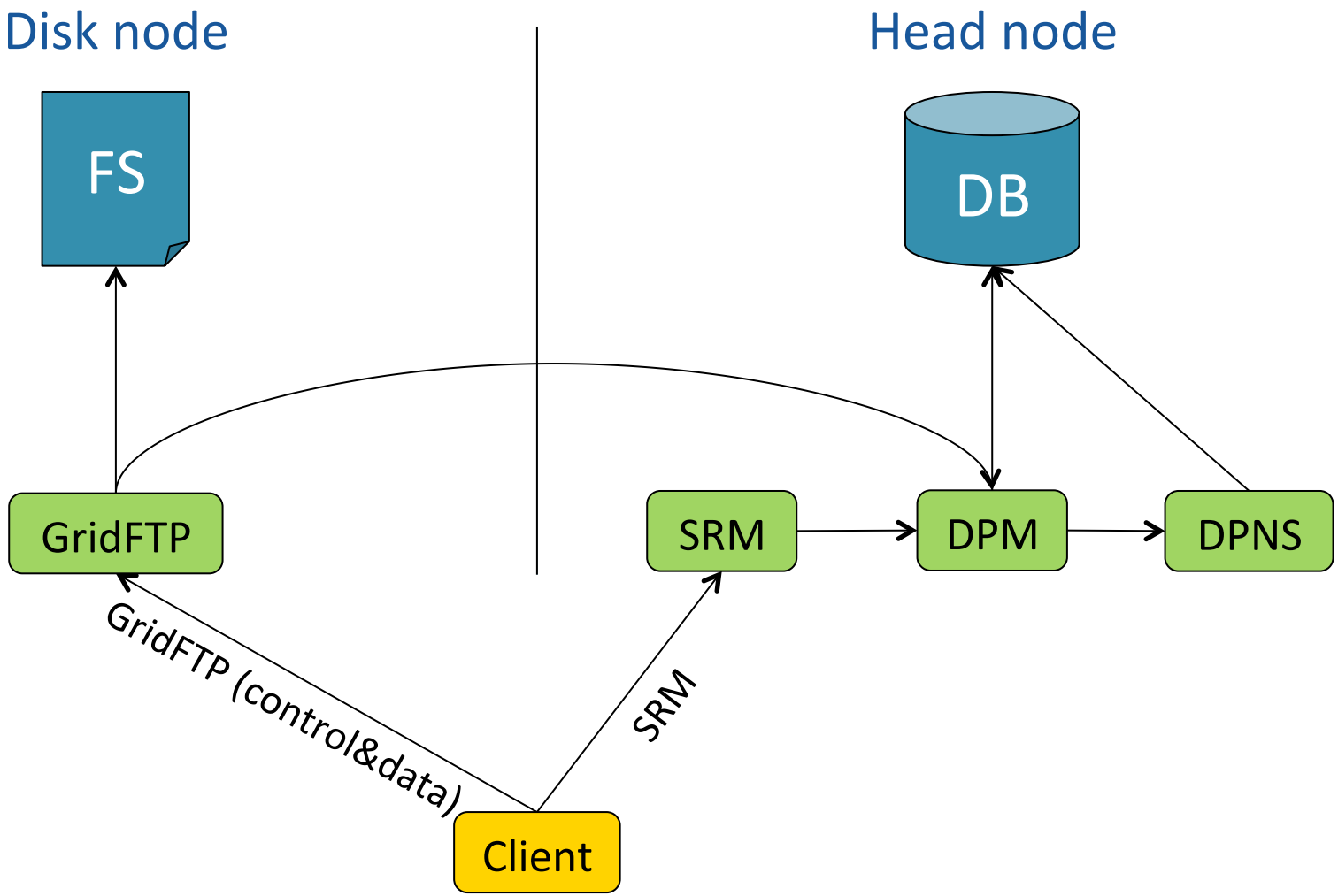
DPM Workshop, Naples

9/10/2014

GridFTP in DPM

- Globus GridFTP server supports pluggable backend modules: DSIs (Data Storage Interfaces)
  - No need to implement a full FTP server from scratch
  - Some things you cannot control
- DPM interaction is implemented with a DSI
- Clients have to contact disk nodes directly. For LFNs an advance SRM call is necessary to get a TURL.
- If a file is not available locally, RFIO is used for transparent staging (slow)

GridFTP in DPM



Weaknesses of the old stack

- A client needs SRM for LFN to TURL conversion, which is a time killer for normal file access
- Direct GridFTP connection to a head node is possible, but results in inefficient internal transfers (data is staged from disk node via RFIO)
- SRM is not interesting outside of HEP community
  - Other protocols like http/webdav and xroot have fancy built-in redirection capabilities that render SRM namespace conversion unnecessary

Can we improve it?

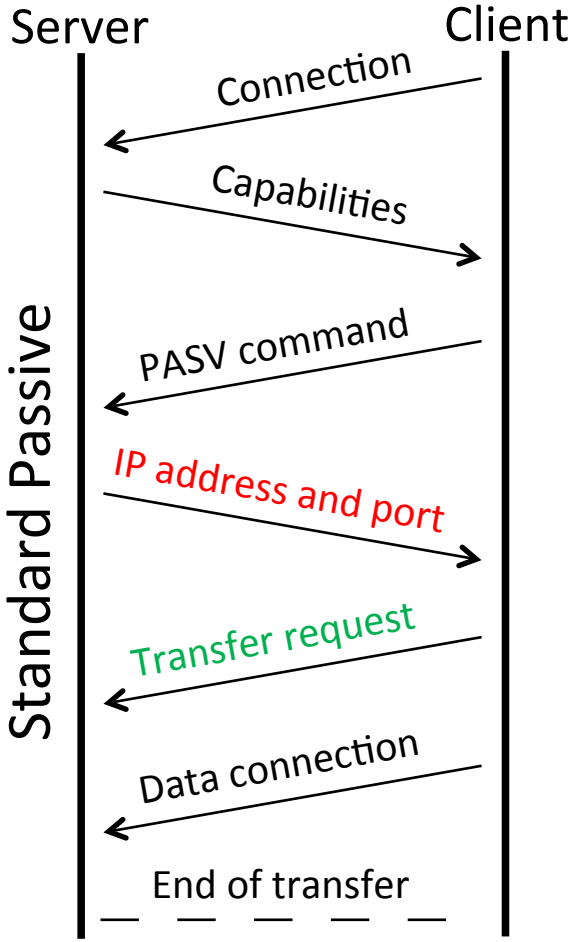
Globus GridFTP server natively supports clusterization and has three modes of operation:

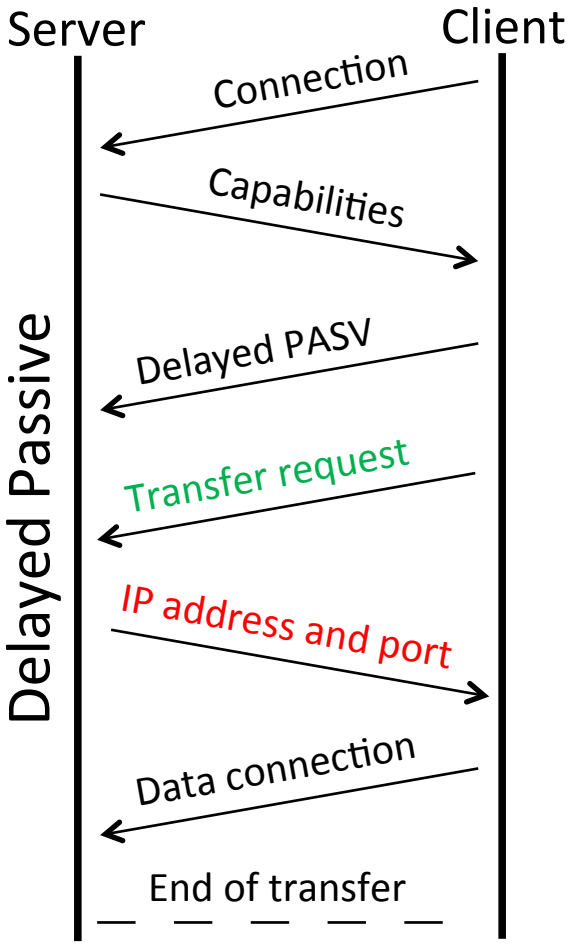
- Standalone – used everywhere so far
- Frontend node – accepts control connections, IPC with backend nodes
- Backend node – accepts or initiates data connections, IPC with frontend node, no control connections

Useful (Grid)FTP basics

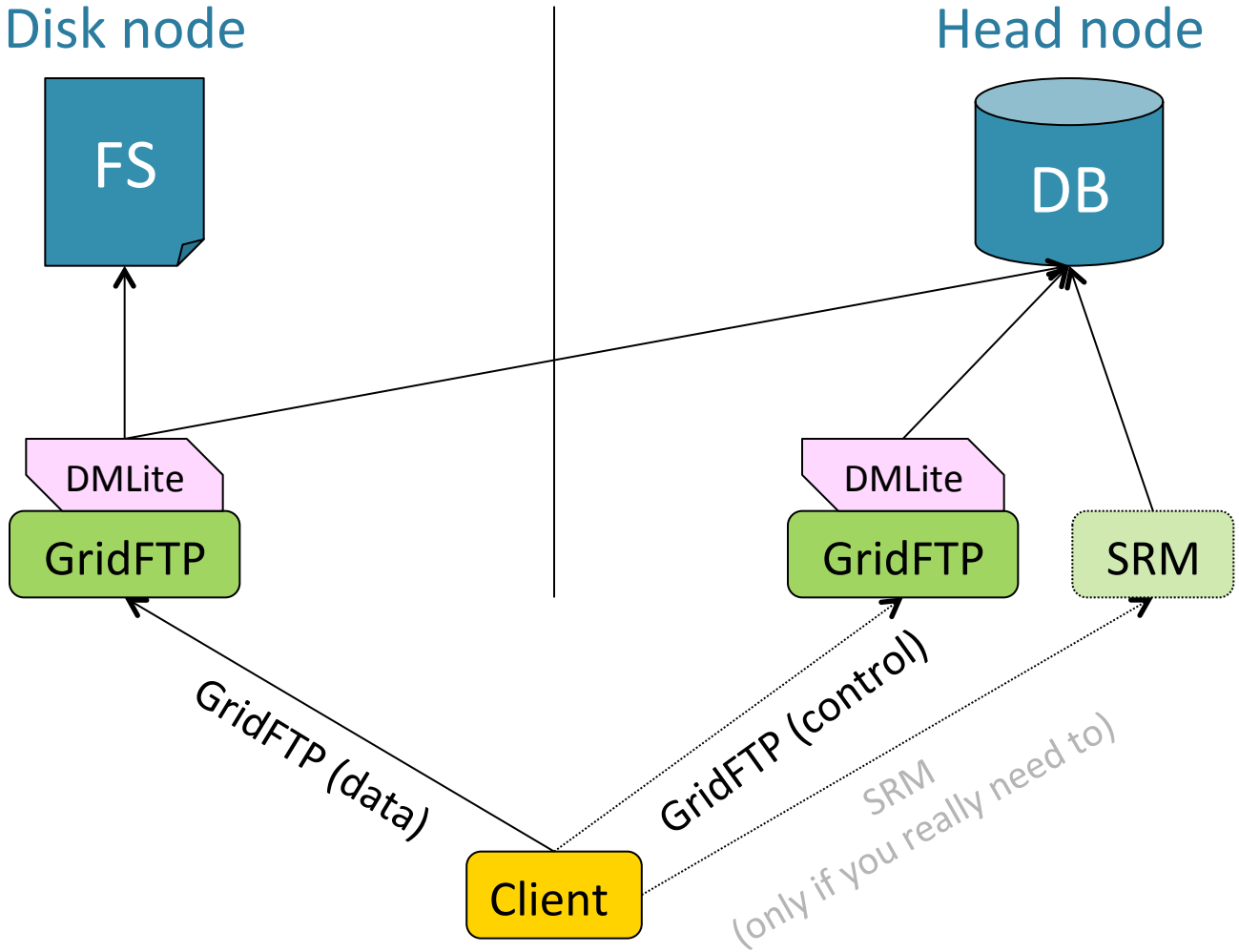
- GridFTP is not much different from good old FTP, but it has GSI auth. on control channel and a bunch of extensions
- A separate TCP connection(s) is used for data transfer, endpoint parameters are negotiated via control channel
- Two standard FTP data connection modes: Passive (initiated by client) and Active (initiated by server)
- GridFTP 2 adds a third data connection mode: Delayed Passive

What exactly is Delayed in Passive mode?





GridFTP in DMLite



Current deployment scenario

- That's what we have in production now
- Proper configuration of DMLite is essential
- Almost no visible changes
- Old DPM DSI is replaced with new DMLite DSI
  - A couple of bugs were immediately found and fixed
- Clients still need to contact SRM prior to disk nodes
- If you also use your head node as a disk node, it's the right time to stop doing it

Future deployment scenario

- Will be supported with Puppet, meanwhile a manual configuration is necessary
- Head node cannot be used as storage
- Clients always contact the head node, direct control connections to disk nodes are no longer supported
- Clients do not need to contact SRM, but if they do, a proper TURL with a head node is given
- Clients need to support Delayed Passive mode for optimal performance
  - Older clients will end up on a random disk node, a file transfer will be done with transparent RFIO staging (slow)

Configuration reference

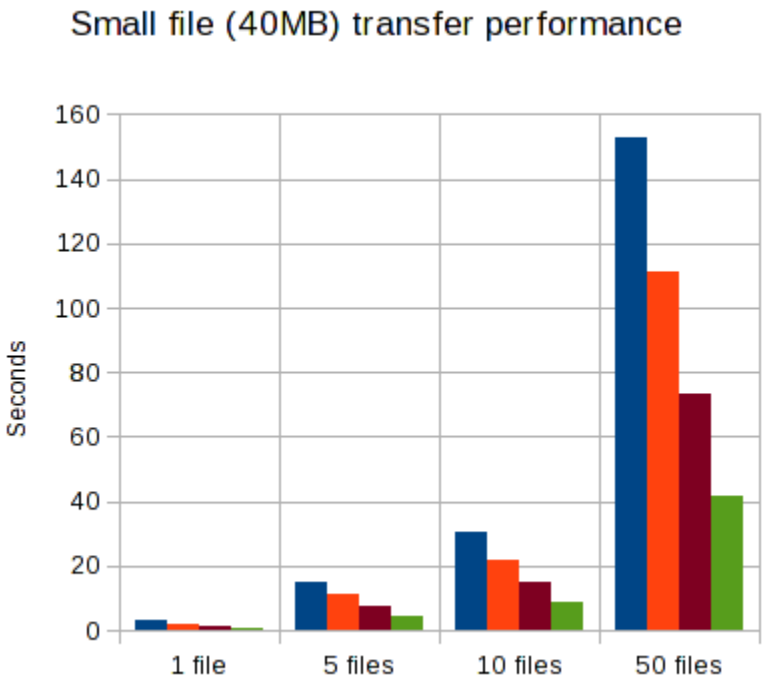
- Disk nodes: add **data_node 1** option to /etc/gridftp.conf and restart GridFTP
  - Don't be surprised that you won't be able to directly access these nodes with GridFTP clients anymore
- FTP head node: add **remote_nodes <list>** option to /etc/gridftp.conf and restart GridFTP
  - **<list>** is a comma-separated list of disk node FQDNs with ports (e.g. disk1.domain.org:2811,disk2.domain.org:2811)
- SRM head node: add **DPM FTPHEAD <head>** option to /etc/shift.conf and restart SRM
  - **<head>** is FQDN of FTP head node from the item above

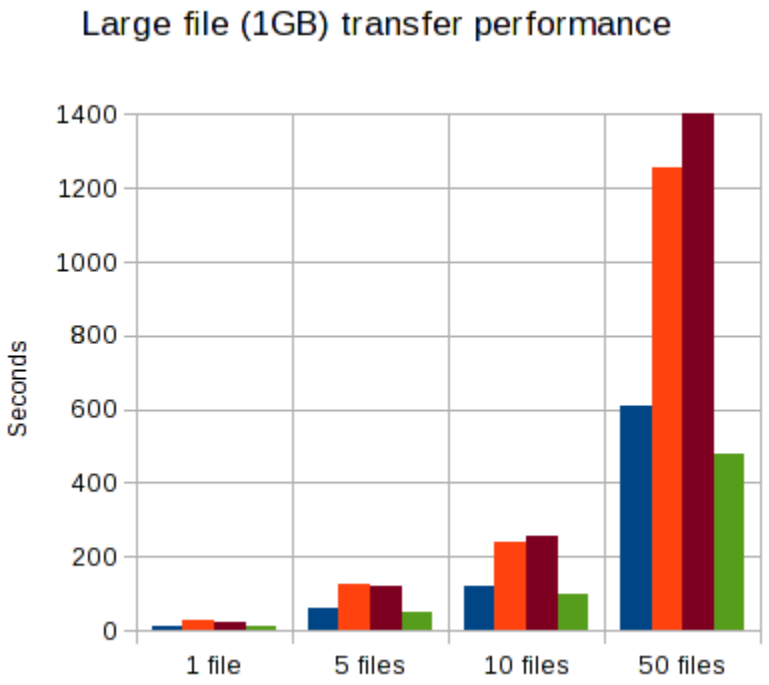
Is the juice worth the squeeze?

- Check RFIO logs on your disk nodes. If there are much more transfers than it was before then you have lots of clients that do not support Delayed Passive mode.
  - Fall-back transfers will happen *between the disk nodes*. Head node will not be involved.
  - You can easily turn redirection off by undoing changes from the slide above
- For small-file workloads (up to tens of megabytes per file) avoiding SRM might be profitable even if it causes RFIO fallback
- For large-file workloads (hundreds of megabytes per file or more) you should avoid RFIO at all costs

Performance study

Sequential transfer of files





- SRM + GridFTP (lcg-cp -b)
- GridFTP + int. RFIO (lcg-cp -b)
- GridFTP + int. RFIO (globus-url-copy -dp)
- GridFTP redir. (globus-url-copy -dp)

Current status

- FTS3 and other GFAL2-based tools support Delayed Passive mode with zero configuration. For **globus-url-copy** you should add **-dp** command-line argument, which will turn on auto-detection of Delayed Passive mode and will not break connection with other GridFTP servers.
  - For third-party transfers the initiator (FTS3) has to support Delayed Passive, not the other endpoint
- Clients that rely on Globus libraries can easily be modified to support Delayed Passive mode:
  - Use the globus_ftp_client_operationattr_set_delayed_pasv(), Luke!
- Please migrate your scripts from lcg-util to gfal2-util
- https://svnweb.cern.ch/trac/lcgdm/wiki/Dpm/GridFTP

Experiment readiness

- Unmodified experiment workflows will continue to work
- GridFTP-only transfers will show a large performance increase wrt current SRM model
- Older, unmodified clients (e.g. lcg-util) will get a performance degradation
  - lcg-util is officially obsolete and experiments have all agreed to adopt newer clients which support delayed passive (gfal2-util)
- Experiments which use space tokens will still have to pass through SRM otherwise ST usage accounting will break
- Redirection currently most interesting to sites who give substantial storage to "non-spacetoken" VOs
- We invite a couple of pioneer sites to activate redirection and work with the dev team to monitor the impact. We will provide a recipe for how to detect problems and how to revert quickly to the current configuration.

That's all

Thank you!